Online Portfolio Selection with ML Predictions

Ziliang Zhang
School of Computer Science
The University of Sydney
Camperdown NSW 2050, Australia
zzha0461@uni.sydney.edu.au

Tianming Zhao
School of Computer Science
The University of Sydney
Camperdown NSW 2050, Australia
tzha2101@uni.sydney.edu.au

Albert Y. Zomaya
School of Computer Science
The University of Sydney
Camperdown NSW 2050, Australia
albert.zomaya@sydney.edu.au

Abstract

Online portfolio selection seeks to determine a sequence of allocations to maximize capital growth. Classical universal strategies asymptotically match the best constant-rebalanced portfolio but ignore potential forecasts, whereas heuristic methods often collapse when belief fails. We formalize this tension in a learning-augmented setting in which an investor observes (possibly erroneous) predictions prior to each decision moment, and we introduce the *Rebalanced Arithmetic Mean portfolio with predictions (RAM)*. Under arbitrary return sequences, we prove that RAM captures at least a constant fraction of the hindsight-optimal wealth when forecasts are perfect while still exceeding the geometric mean of the sequence even when the predictions are adversarial. Comprehensive experiments on large-scale equity data strengthen our theory, spanning both synthetic prediction streams and production-grade machine-learning models. RAM advantages over universal-portfolio variants equipped with side information across various regimes. These results demonstrate that modest predictive power can be reliably converted into tangible gains without sacrificing worst-case guarantees.

1 Introduction

Online portfolio selection, the sequential allocation of capital across multiple assets (e.g., stocks), sits at the crossroads of mathematical finance, statistical learning, and online algorithmic design. In every trading period, an investor must commit to a nonnegative, unit-sum weight vector before observing the next price relatives[1], relying on two imperfect guides: extracted patterns from historical returns and predictive signals on future returns that may be noisy. The puzzle is therefore two-fold: (i) distill from the past statistical regularities that remain informative, and (ii) fuse them with fallible forecasts in a way that preserves worst-case guarantees. Resolving this tension is challenging because we can neither guarantee the persistence of past regularities nor trust the accuracy of future forecasts blindly.

Literature review. The growth-optimal portfolio paradigm traces back to Kelly's [1] information-theoretic formulation of proportional betting, which showed that compounding wealth at the exponential rate is achievable when returns are i.i.d. Capital growth theory [2] later extended this principle to multi-asset markets, providing a rigorous benchmark for long-horizon investment under more

[1]Throughout, "price relative" and "return" are used interchangeably.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

realistic dynamics. Contemporary research has bifurcated into two main streams. The first is to find universal algorithms that provide provable guarantees against adversarial return sequences without committing to a return model; the second relies on heuristic assumptions or data-driven forecasting powered by machine-learning models, trading theoretical certainty for empirical performance. These complementary approaches motivate our study, we therefore begin with a overview of each.

Universal algorithms, such as Universal Portfolio [3] and Exponential Gradient Update [4], make no market assumptions about price dynamics yet remain provably competitive with the best constant-rebalanced portfolio (BCRP) selected in hindsight. After n rounds, their cumulative wealth lags BCRP by at most a polynomial factor, yielding sublinear regret in logarithmic growth. However, to hedge against adversarial sequences, universal algorithms deliberately diffuse capital across assets, forfeiting the windfalls that arise when strong predictive structure is present. A more frugal member of the provably safe camp is the uniform buy-and-hold strategy [5]. Its final wealth is bounded at the arithmetic mean of single-asset returns, but the absence of rebalancing makes it even more conservative than the universal-portfolio family, and empirical evidence shows it often lags behind whenever return dispersion is substantial. Further algorithmic families are reviewed comprehensively in the survey by Li and Hoi [6]. However, despite theoretical guarantees against offline benchmarks, these traditional algorithms share a cautious bias that may overlook significant predictive gains.

On the other hand, heuristic methods ranging from mean reversion [7; 8] to pattern matching approaches [9] rely on market assumptions to inform algorithmic decisions based on signaled patterns. Furthermore, there is growing interest in employing machine-learning (ML) models to forecast future returns, heavily inclining the portfolio to the highest predicted asset in attempt to yield optimal return. Morris et al. [10] introduce an ensemble framework that combines data mining techniques with long short-term memory networks for forecasting stock and Bitcoin prices. Singh and Srivastava [11] deploy a deep neural network for stock price forecasting, demonstrating enhanced performance over recurrent neural networks. Zhang et al. [12] integrate a generative adversarial network with a long short-term memory module, achieving promising results in real-world closing-price prediction. Feng et al. [13] propose an advanced machine learning approach to stock movement prediction, employing adversarial training to enhance the generalization capability of neural networks. These prediction-based strategies can exploit favorable market conditions and achieve superior returns if the data follows the underlying model. However, when the distribution of future returns deviates from the model, these approaches may make poor predictions and suffer unbounded losses compared to more conservative methods. In addition, ML models may suffer from overfitting if trained on insufficient or unrepresentative features. As wealth compounds exponentially, even a brief sequence of poor forecasts can rapidly erode capital, leading to catastrophic losses within a short time horizon.

Navigating preservation versus prediction-driven aggressiveness brings us to a fundamental question:

How to exploit forecasts yet preserve worst-case guarantees in portfolio selection?

Our answer is to weave ML forecasts into the fabric of classically robust schemes. The design is guided by the emerging paradigm of algorithms with predictions [14], where an online procedure receives a potentially faulty glimpse of the future and must convert that hint into improved average-case performance while keeping any inflation of worst-case regret provably minimal. In this framework, a baseline algorithm is augmented by some predictive signals that balances follow-the-prediction against hedge-for-adversity, yielding an error-sensitive competitive ratio that smoothly interpolates between perfect-forecast optimality and classical worst-case guarantees under adversarial prediction.

The learning-augmented paradigm has been instantiated across a wide spectrum of classic problems. Lykouris and Vassilvitskii [15] demonstrate how a single tunable parameter yields an error sensitive competitive ratio for the paging problem, outperforming LRU [16] whenever the predictor is informative while retaining its worst-case bound. Kumar et al. [17] adapt the same recipe to both the ski-rental dilemma and non-clairvoyant job scheduling. Bai and Coester [18] present a sorting algorithm that harnesses potentially erroneous predictions to enhance computational efficiency. Learning-augmented techniques have also appeared in financal settings, including Pareto-optimal threshold-based algorithms for online conversion problems [19] and tight consistency-robustness bounds for One-Max-Search [20]. For a more comprehensive overview, we refer to the survey by Mitzenmacher and Vassilvitskii [21] and the online repository at [14]. Online portfolio selection shares the same sequential-decision procedure: each round demands an irrevocable allocation before the next return vector is revealed. By importing the learning-augmented toolkit from algorithms with predictions into this framework, we aim to fuse the upside of ML forecasts with the downside pro-

tection of conservative strategies, yielding wealth trajectories that improves smoothly with accurate predication while retaining a measurable offline benchmark when predictive quality degrades.

To date, very few studies have embedded modern ML forecasts directly into the update rules of classical online portfolio algorithms. The main precursor is the side-information variant of Cover's universal portfolio [22]. At every round, the investor receives a discrete signal that labels the current market regime (i.e., bull or bear) and maintains a separate universal strategy for each regime, with the goal of matching the state-dependent BCRP (BSCRP). Borodin et al. [7] instead assume a mean-reversion pattern in next-period prices, but offer no error-sensitive guarantees linking performance to the quality of its forecasts. More recent work has refined the side-information idea: Bhatt et al. [23] extend universal portfolios to continuous side information via a probabilistic partitioning that achieves first-order asymptotic optimality against BSCRP, while Yang et al. [24] employ an expert-aggregation framework to construct state-dependent expert ensembles that achieve the same asymptotic benchmark.

However, these constructions inherits several limitations. (i) Both BCRP and BSCRP are modest benchmarks, which can lag the true offline optimum that reallocates into the single best-performing asset each day by an exponential factor. (ii) Universal algorithms converge to BCRP and/or BSCRP only at a polynomial rate in wealth, so the lower bound becomes practically vacuous once the exponential gap to the global optimum is accounted for. (iii) The regret bound on growth rate with side information grows linearly with the number of states, so a richer information alphabet paradoxically weakens the guarantee. (iv) Focusing solely on asymptotic growth obscures performance over the finite horizon which matters most in practice. (v) Most critically, existing results are prediction-agnostic. They provide no quantitative characterization of how performance scales with prediction error, leaving unanswered how the algorithm fares when the side information is perfectly informative, entirely misleading, or somwehere in between.

These limitations motivate our approach, which fuses ML forecasts with adversarial safeguards and provides error-sensitive guarantees that explicitly couple portfolio performance to prediction quality. We now formalize the problem and introduce the necessary notation.

Problem statement. We study online portfolio selection over n trading periods and m[2] assets. Let

$$\mathbf{x}^n \;=\; \big(\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(n)\big), \qquad \mathbf{x}(i) \;=\; \big(x_1(i), x_2(i), \ldots, x_m(i)\big) \in \mathbb{R}_+^m \tag{1}$$

where $x_j(i)$ is the price relative of asset j from period $i-1$ to i. A portfolio strategy defines

$$\mathbf{b}^n \;=\; \big(\mathbf{b}(1), \mathbf{b}(2), \ldots, \mathbf{b}(n)\big), \qquad \mathbf{b}(i) \;=\; \big(b_1(i), \ldots, b_m(i)\big) \in \Delta^{m-1}$$

with $b_j(i) \geq 0$ and $\sum_{j=1}^m b_j(i) = 1$ (self-financed, no margin/shorting). The wealth after n periods is

$$S_n\big(\mathbf{b}^n, \mathbf{x}^n\big) \;=\; \prod_{i=1}^n \big(\mathbf{b}(i) \cdot \mathbf{x}(i)\big) \;=\; \prod_{i=1}^n \sum_{j=1}^m b_j(i)\, x_j(i)$$

We set the initial capital to $S_0 = 1$; thus, S_n equals both the terminal wealth and total growth factor. At the start of each period i, the investor receives a ranking forecast from a black-box ML oracle. Let $[m] \coloneqq \{1, \ldots, m\}$. The oracle outputs a permutation $\sigma(i)$ of $[m]$, where $\sigma(i)_k$ denotes the index of the asset ranked k-th (highest) by predicted one-step return. For analysis, we define $\mathbf{y}(i)$ by reindexing the realized returns according to the predicted ranking:

$$\mathbf{y}(i) \;\coloneqq \big(y_1(i), \ldots, y_m(i)\big), \qquad y_k(i) \coloneqq x_{\sigma(i)_k}(i) \tag{2}$$

Thus $y_1(i) \geq y_2(i) \geq \cdots \geq y_m(i)$ reflects the oracle's predicted order (highest to lowest). Importantly, our algorithm and guarantees depend only on the ranking $\sigma(i)$; we do not assume access to, nor require, predicted magnitudes of next-period returns. The vector $\mathbf{y}(i)$ is introduced solely as analysis notation to track the realization under the predicted order.

We make no assumptions about how this vector (ranking) is generated, and it may come from any ML model, e.g.: a neural network [10] that predicts one-step-ahead returns and then ranks them; a gradient-boosted tree model [25] that outputs the ranking directly; or any heuristic ranking rules. For analysis, we also define the clairvoyant order statistics $x_{(1)}(i) \geq \cdots \geq x_{(m)}(i)$ as the entries of $\mathbf{x}(i)$ sorted in

[2]We fix the number of assets m at the outset and treat it as a constant.

non-increasing order, and write $\mathbf{x}^{\downarrow}(i) \coloneqq (x_{(1)}(i), \ldots, x_{(m)}(i))$ and $\mathbf{x}^{\uparrow}(i) \coloneqq (x_{(m)}(i), \ldots, x_{(1)}(i))$. Note the distinction: $x_j(i)$ is the return of asset j (original index), whereas $x_{(j)}(i)$ is the j-th largest return at time i (ranked value). Ties, when present, are broken by a fixed deterministic rule; following analysis are invariant to this choice. We apply the same shorthand to the current weights $\mathbf{b}^{\downarrow}(i)$:

$$\mathbf{b}^{\downarrow}(i) = \big(b_{(1)}(i), b_{(2)}(i), \cdots, b_{(m)}(i)\big) \tag{3}$$

Our goal is to design an online portfolio algorithm with prediction that translates the oracle's noisy forecasts $\mathbf{y}^n$ into higher wealth whenever they are informative, and preserves a provable worst-case guarantee matching the best prediction-agnostic baseline when the forecasts are arbitrarily inaccurate. We adopt the geometric mean of all returns referring the *value line index*, as such a baseline. Formally,

$$S_n^{\text{GM}} = \left[\prod_{j=1}^{m}\prod_{i=1}^{n} x_j(i)\right]^{1/m} \tag{4}$$

The best-known competitive [26] prediction-agnostic strategy is the Universal Portfolio [3], which only guarantees wealth no less than S_n^{GM} [3, Prop. 5]. Matching this floor therefore secures state-of-the-art worst-case performance while leaving headroom for upside when the oracle proves informative.

Our contribution. We propose a learning-augmented portfolio with updating rules guided by ML forecasts, the Rebalanced Arithmetic Mean (RAM). For the prediction error[3] $\eta_n \in (0, 1]$ induced by any predictions $\mathbf{y}^n = (\mathbf{y}(1), \mathbf{y}(2), \cdots, \mathbf{y}(n))$, the resulting wealth of RAM after n periods satisfies:

$$S_n^{\text{RAM}} = \prod_{i=1}^{n} \mathbf{b}^{\downarrow}(i) \cdot \mathbf{y}(i) \geq \max\left\{ S_n^{\text{GM}} = \left[\prod_{j=1}^{m}\prod_{i=1}^{n} x_j(i)\right]^{1/m},\ \eta_n \prod_{i=1}^{n} \mathbf{b}^{\downarrow}(i) \cdot \mathbf{x}^{\downarrow}(i) \right\}$$

with weight updating rules $\mathbf{b}^{\downarrow}(i)$ followed by rank-matching strategy[4] using predicted permutations:

$$\mathbf{b}^{\downarrow}(i) = \big(b_{(1)}(i), \cdots, b_{(m)}(i)\big), \quad b_k(i) = \frac{b_{(j)}(i-1)y_j(i-1)}{\sum_{j=1}^{m} b_{(j)}(i-1)y_j(i-1)}, \quad b_k(1) = \frac{1}{m}, \forall i, j, k \tag{5}$$

When the predictions are maximally informative $\eta_n = 1$ and $\mathbf{y}(i) = \mathbf{x}^{\downarrow}(i), \forall i$, RAM achieves wealth at least a constant fraction of the hindsight-optimal "all-in-best-asset" benchmark:

$$S^{\text{RAM}} \geq \frac{1}{m} S_n^{\text{OPT}} := \frac{1}{m} \prod_{i=1}^{n} \max_j x_j(i)$$

Conversely, when forecasts are actively hostile (adversarial), we prove that RAM dominates the value line S_n^{GM}. The results are independent of market assumptions other than positive return factors.

We substantiate our theory with real-world equity experiments in progressively richer settings:

1. Prediction-free benchmark: We show that a randomized, prediction-free variant of RAM outperforms the Universal Portfolio in expectation.
2. Controlled-noise study: Using synthesized predictions under controlled accuracy thresholds, RAM degrades gracefully under heavy noise and raises smoothly with improved predictions.
3. Real-world deployment: We deploy RAM with real-world ML model on recent market data, achieving significant gains over existing baselines, underscoring its practical utility.

2 Portfolio with predictions

Blindly following the forecasts. Consider two assets x_1 and x_2, where one either doubles or halves while the other does the opposite. Let the forecast be wrong with independent probability $\epsilon \in (0, 1)$. An investor who completely follows the forecast accumulates wealth $S_n = 2^{(1-\epsilon)n}(0.5)^{\epsilon n} = 2^{(1-2\epsilon)n}$. When $\epsilon < 0.5$, the investor still gains, but at an exponential gap of $2^{2\epsilon n}$ against the optimum $S_n^* = 2^n$. Conversely, when $\epsilon > 0.5$, the wealth decays exponentially to zero. Therefore, even a small error rate produces unbounded competitive loss and expose the strategy to total ruin.

[3] Formally defined in Section 2.1 (Eq. 6)

[4] Detailed in Section 2.1; see Algorithm 1 for pseudocode implementation.

Partially trusting the forecast. A natural repair is to invest a fraction $\lambda \in [0,1]$ in the predicted winner and keep the remaining in a safe baseline such as the uniform portfolio $(0.5, 0.5)$. Lemma A.1 (Appendix) proves that, when the forecast is wrong with independent probability $\epsilon \in (0,1)$, the expected wealth is $\mathbb{E}[S_n] = 1.25^n(1+0.6\lambda(1-2\epsilon))^n$. Consequently, the expected per-period log-growth is $W_n(\lambda,\epsilon) = \log 1.25 + (1-\epsilon)\log(1+0.6\lambda) + \epsilon\log(1-0.6\lambda)$. For every fixed $\lambda \geq 1/3$, there always exists a $\epsilon^*(\lambda) < 0.5$ forcing $W_n(\lambda,\epsilon^*) = 0$. If an adversary pushes the error rate even slightly above this threshold, $W_n(\lambda,\epsilon) < 0$ and wealth decays exponentially to zero. Meanwhile the competitive ratio against the optimal hindsight 2^n still diverges exponentially whenever $\lambda > 0$ and $\epsilon > 0$. Thus, fractional betting ameliorates but does not eliminate worst-case fragility.

Our intuition is to decouple magnitudes from directional signs through a two-stage architecture:

1. *Base generator.* At each period, a prediction-agnostic strategy updates a weight vector using only realized return history, preserving the base algorithm's worst-case guarantees.
2. *Permutation layer.* The ML predicted ranking (the ordering of upcoming returns) then re-labels these weights, concentrating larger masses on assets it considers promising.

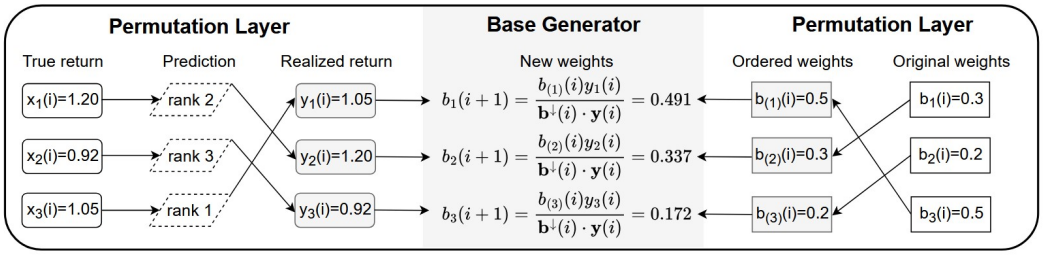
This design inherits the robustness of the base generator yet exploits predictive insights whenever it is present. The crux is to pair a provably safe generator with a permutation rule that minimally degrades its guarantee while delivering gains under accurate predictions. The remainder of the paper therefore concentrates on this permutation-based portfolio class.

2.1 Rebalanced arithmetic mean

We propose the Rebalanced Arithmetic Mean (RAM) portfolio which overlays an oracle-supplied ranking of next-period gross returns onto the buy-and-hold baseline. Let the baseline assign weights $\{b_j(i)\}_{j=1}^m$ at period i according to equation 5, where the pipeline follows by matching higher weights to higher predicted returns according to their ranks. Concurrently, an oracle outputs a permutation $\sigma(i)$ intended to rank the forthcoming returns $\mathbf{x}(i)$; applying it produces the oracle-ordered vector $\mathbf{y}(i)$ as defined in equation 2. Conceptually, RAM is still a buy-and-hold policy, yet executed in a dynamically permuted coordinate system determined by permutations $\{\sigma(i)\}_{i=1}^n$.

RAM operates by greedy matching: it pairs the largest weight $b_{(1)}(i)$ with the asset which the oracle predicts to deliver the highest return, the second-largest weight with the second-highest predicted return, and so on. The portfolio held at period i therefore earns the inner product $\mathbf{b}^{\downarrow}(i) \cdot \mathbf{y}(i)$. Importantly, we highlight the key relation of $\mathbf{b}^{\text{new}} \cdot \mathbf{x}(i) = \mathbf{b}^{\downarrow}(i) \cdot \mathbf{y}(i)$ as in Algorithm 1. A visual example of RAM's update process is shown in Figure 1.

Figure 1: RAM's rank-matching architecture: larger weights paired with higher predicted ranks.



To quantify the oracle's accuracy, we compare its ordering with the clairvoyant optimal ordering of the upcoming returns $\mathbf{x}^{\downarrow}(i)$. We define the per-period error rate $\eta(i)$ and total error rate[5] η_n:

$$\eta(i) := \frac{\mathbf{b}^{\downarrow}(i) \cdot \mathbf{x}^{\downarrow}(i)}{\mathbf{b}^{\downarrow}(i) \cdot \mathbf{y}(i)} = \frac{\sum_{j=1}^m b_{(j)}(i) x_{(j)}(i)}{\sum_{j=1}^m b_{(j)}(i) y_j(i)}, \qquad \eta_n := \prod_{i=1}^n \frac{1}{\eta(i)} \tag{6}$$

[5]The daily error rate $\eta(i)$ is defined so that it decreases as predictions improve and increases as they degrade. The aggregated quantity η_n behaves oppositely; although we phrase it as an overall error rate for brevity, it should be interpreted as a performance metric, where larger values indicate better performance.

Algorithm 1 RAM: rebalanced arithmetic mean with predictions

Require: return matrix $\{\mathbf{x}(i)\}_{i=1}^{n} \subset \mathbb{R}_{+}^{m}$, oracle permutation $\sigma(i) \leftarrow \textsc{ForecastPerm}\big(\mathbf{x}(i)\big)$

initialize weights $\mathbf{b}(1) \leftarrow (\frac{1}{m}, \ldots, \frac{1}{m})$, wealth $S(0) \leftarrow 1$
for $i \leftarrow 1$ **to** n **do**
  Receive a ranking prediction $\sigma(i)$ from oracle.
  $\mathbf{b}^{\downarrow} \leftarrow \mathrm{sort}^{\downarrow}\big(\mathbf{b}(i)\big)$
  $\mathbf{b}^{\text{new}} \leftarrow \mathbf{0}$
  for $j \leftarrow 1$ **to** m **do** ▷ reverse pairing
    $\mathbf{b}^{\text{new}}_{\sigma(i)_j} \leftarrow \mathbf{b}^{\downarrow}_{j}$
  end for
  Receive return $\mathbf{x}(i)$.
  $R(i) \leftarrow \mathbf{b}^{\text{new}} \cdot \mathbf{x}(i)$
  $S(i) \leftarrow S(i-1) \times R(i)$
  $\mathbf{b}(i+1) \leftarrow \big(\mathbf{b}^{\text{new}} \odot \mathbf{x}(i)\big)\big/R(i)$
end for
return $S(n)$

The cumulative penalty η_n captures the aggregate loss in wealth relative to the clairvoyant optimal matcher. The final aggregated wealth of RAM can therefore be expressed in exchangeable forms:

$$S_n^{\text{RAM}} = \prod_{i=1}^{n} \mathbf{b}^{\downarrow}(i) \cdot \mathbf{y}(i) = \prod_{i=1}^{n} \frac{\mathbf{b}^{\downarrow}(i) \cdot \mathbf{x}^{\downarrow}(i)}{\eta(i)} = \eta_n \prod_{i=1}^{n} \sum_{j=1}^{m} b_{(j)}(i) x_{(j)}(i) \tag{7}$$

This multiplicative decomposition cleanly separates the contribution of the underlying buy-and-hold strategy from the oracle-induced error factor. When the oracle's ranking is accurate, $\eta_n = 1$ and RAM achieves the clairvoyant optimum. Even under maximally adversarial rankings, $\eta_n \in (0, 1]$, RAM's wealth cannot drop below the value line index (the lower envelope of the arithmetic mean).

The reason RAM works. RAM is built on two simple intuitions. The first design goal is the immediate optimality under perfect forecast. Universal portfolios with side information [22; 4] only promise to approach the optimum in the limit, which means that they can lag for thousands of rounds and still be theoretically "optimal". RAM avoids that delay: a perfect oracle yields the exact clairvoyant growth for each and every period. The second goal is to prove disciplined loss guarantee under adversarial regime. As inspired by the arithmetic-geometric mean (AM-GM) inequality, we note a key invariance: a uniform buy-and-hold portfolio is unaffected by any fixed permutation of asset coordinates. Once the ordering is frozen, the AM-GM lower envelope applies to the entire return sequence, independent of the chosen coordinate system. Consequently, re-indexing the next-period returns is essentially cost-free, and the resulting wealth trajectory remains above the value line. Theorem 2.1 formalizes our claim with tight lower bounds under different levels of prediction quality.

Theorem 2.1. *For arbitrary market sequence $\mathbf{x}^n$ as specified in equation 1 and predictions $\mathbf{y}^n = (\mathbf{y}(1), \mathbf{y}(2), \cdots, \mathbf{y}(n))$ where each $\mathbf{y}(i)$ follows by equation 2, we have claims for Algorithm 1:*

1. *For arbitrary predictions (worst-case guarantee):*

$$S_n^{RAM} \geq \left[\prod_{j=1}^{m} \prod_{i=1}^{n} x_j(i)\right]^{1/m}, \quad \forall \eta_n \in (0, 1] \tag{8}$$

2. *Under perfect predictions (best-case guarantee):*

$$S_n^{RAM} \geq \frac{1}{m} S_n^{OPT} = \frac{1}{m} \prod_{i=1}^{n} \max_j x_j(i), \qquad \textit{for } \eta_n = 1 \tag{9}$$

Proof. We first prove 9. When predictions are perfect, $\mathbf{y}(i) = \mathbf{x}^{\downarrow}(i)\ \forall i$ and $\eta_n = 1$:

$$S_n^{\text{RAM}} = \prod_{i=1}^{n} \mathbf{b}^{\downarrow}(i) \cdot \mathbf{x}^{\downarrow}(i) = \prod_{i=1}^{n} \sum_{j=1}^{m} b_{(j)}(i) x_{(j)}(i)$$

By the monotonicity of $\mathbf{b}^{\downarrow}(i)$ and $\mathbf{x}^{\downarrow}(i)$:

$$b_{(1)}(i) \geq \cdots \geq b_{(m)}(i) \text{ and } x_{(1)}(i) \geq \cdots \geq x_{(m)}(i) \implies b_{(1)}(i)x_{(1)}(i) \geq \cdots \geq b_{(m)}(i)x_{(m)}(i)$$

Therefore, the index carrying larger weight at period i must also carry larger wealth share after i:

$$b_k(i+1) = \frac{b_{(j)}(i)x_{(j)}(i)}{\sum_{j=1}^{m} b_{(j)}(i)x_{(j)}(i)} \implies b_{(j)}(i+1) = \frac{b_{(j)}(i)x_{(j)}(i)}{\sum_{j=1}^{m} b_{(j)}(i)x_{(j)}(i)}$$

Apply $b_{(j)}(n+1)$ and expand the right-hand side recursively down to period 1 yields:

$$b_{(j)}(n+1) = \frac{1}{m} \frac{\prod_{i=1}^{n} x_{(j)}(i)}{\prod_{i=1}^{n} \sum_{j=1}^{m} b_{(j)}(i)x_{(j)}(i)} \implies \sum_{j=1}^{m} b_{(j)}(n+1) =$$

$$\frac{1}{m} \frac{\sum_{j=1}^{m} \prod_{i=1}^{n} x_{(j)}(i)}{\prod_{i=1}^{n} \sum_{j=1}^{m} b_{(j)}(i)x_{(j)}(i)} = 1 \implies S_n^{\text{RAM}} = \frac{1}{m} \sum_{j=1}^{m} \prod_{i=1}^{n} x_{(j)}(i) \geq \frac{1}{m} \prod_{i=1}^{n} x_{(1)}(i) = \frac{1}{m} S_n^{\text{OPT}}$$

proving 9. Next, we prove 8. Recall under adversarial predictions $S_n^{\text{RAM}} = \prod_{i=1}^{n} \mathbf{b}^{\downarrow}(i)\mathbf{x}^{\uparrow}(i)$ and:

$$b_k(i+1) = \frac{b_{(j)}(i)x_{(m-j+1)}(i)}{\sum_{j=1}^{m} b_{(j)}(i)x_{(m-j+1)}(i)} \implies b_k(i+1) = \frac{b_k(i)x_{\overline{\sigma}(i)_k}(i)}{\sum_{j=1}^{m} b_k(i)x_{\overline{\sigma}(i)_k}(i)}$$

where we have re-indexed the assets relative to the weight vector such that $\overline{\sigma}(i)_k$ denotes the adversarially permuted pairing on weight $b_k(i)$, so the algebra is carried out in a moving coordinate system attached to the weights, not in the fixed asset labeling. The specific indices $\overline{\sigma}(i)_k$ are irrelevant to the wealth analysis, since the only property we need is that every return value $x_{\overline{\sigma}(i)_k}(i)$ is paired with exactly one weight coordinate. Apply $b_k(n+1)$ and expand the right-hand side yields:

$$b_k(n+1) = \frac{1}{m} \frac{\prod_{i=1}^{n} x_{\overline{\sigma}(i)_k}(i)}{\prod_{i=1}^{n} \mathbf{b}^{\downarrow}(i)\mathbf{x}^{\uparrow}(i)} \implies S_n^{\text{RAM}} = \frac{1}{m} \sum_{k=1}^{m} \prod_{i=1}^{n} x_{\overline{\sigma}(i)_k}(i) \geq \left[\prod_{k=1}^{m} \prod_{i=1}^{n} x_k(i) \right]^{1/m}$$

where the right-hand side follows by the arithmetic-geometric mean inequality. □

Relative to existing universal strategies, RAM occupies a distinctive midpoint. Under a perfect oracle, RAM extracts a constant $1/m$ fraction of the hindsight optimum, whereas the side-information universal portfolio family converges only to the modest best state-constant rebalanced portfolio and does so asymptotically. At the opposite end of the information spectrum, both RAM and universal portfolios enjoy the same distribution-free safe net, the value line index, ensuring no catastrophic under-performance. The gap between these extremes is where RAM's permutation layer becomes pivotal. Because weights are reassigned wholesale according to the forecast ordering, RAM reacts sharply on informative ranking, but with higher variance when the signal is noisy. Recent advances in machine learning have begun to yield predictors with demonstrably non-trivial forecasting power. RAM can harness these models immediately while its AM-GM floor caps downside risk. In addition, the algorithm runs in $O(m \log m)$ per round, making it readily deployable on large-scale portfolios where the exponential costs of richer universal mixtures are prohibitive.

Table 1: Stock combinations on NYSE.

NYSE(O)	**NYSE(N)**
1: Iro & Kin	7: Cok & Ahp & Gm & Ge & Sc
2: Com & Kin	8: Dow & Mmm & Ame & Mer & Jnj
3: Com & Mei	9: Ame & Kim & Alc & Ahp & Mor & Cok & Pan & Gm
4: Luk & Kin & Arc	10: Mmm & Dow & Mer & Ge & Ibm & Ing & Hp & Jnj
5: Shr & Ge & Kin	11: Ge & Mer & Gm & Ahp & Jnj & Pan & Ame & Ing & For & Dup & Dow
6: Gul & Esp & Pil	12: Alc & Cok & Hp & Ibm & Kim & Mmm & Mor & Sc & Gm & Ahp & Pan

3 Empirical study

We begin by assessing RAM on the canonical New York Stock Exchange (NYSE) benchmark, using i.i.d. random rankings to model a fully oblivious and uninformative oracle. Our primary dataset is the original NYSE(O) collection [27], which contains 36 stocks spanning 22 years (1962–1984) over 5,651 trading days. To capture a broader range of market volatility and ensure more recent coverage, we also consider the extended NYSE(N) dataset [28], encompassing 21 assets from 1962 to 2006 (11,178 trading days). To avoid duplication, we preprocess NYSE(N) by retaining only the period from 1985 to 2006 (5,526 trading days), thus removing any overlap with NYSE(O). Each data point represents the ratio of an asset's price on a given trading day to its price on the previous day. Table 1 details the various combinations of stocks: Combinations 1–3 have been widely adopted in seminal works such as [3; 4], Combinations 4–6 follow those introduced by Yang et al. [29], and the remaining combinations were chosen for additional diversity and expanded market dimension.

To study RAM's behavior under controllable signal quality, we draw rank-predictions from a one-parameter family $\{D_p\}_{p\in[0,1]}$ which interpolates linearly between adversarial and optimal orderings:

$$\mathcal{D}_p \;=\; \begin{cases} (1-2p)\,\delta_{\sigma^{\uparrow}} \;+\; 2p\,U, & 0 \le p \le \frac{1}{2}, \\ 2(1-p)\,U \;+\; (2p-1)\,\delta_{\sigma^{\downarrow}}, & \frac{1}{2} < p \le 1. \end{cases} \tag{10}$$

where $\sigma^{\uparrow}$ and $\sigma^{\downarrow}$ are the clairvoyant ascending and descending orderings of the upcoming returns, U is the uniform distribution over all $m!$ permutations, and δ_σ denotes a Dirac mass at permutation σ. Consequently, $p = 0$ produces the worst-case forecast (deterministically ascending), $p = \frac{1}{2}$ gives full randomness, and $p = 1$ provides the best-case forecast (deterministically descending). Moving p away from the middle point linearly transfers probability mass from the uniform component to the appropriate Dirac measure, facilitating a transparent and smoothly tunable notion of forecast quality. For each fixed p we draw one permutation per period: with probability equal to the Dirac weight we output the deterministic order; otherwise we apply a Fisher-Yates shuffle. The identically sampled permutation is fed to every applicable strategy under comparison, ensuring that performance differences arise solely from algorithmic design rather than additional stochasticity.

For each accuracy threshold p, we run 1,000 independent Monte Carlo simulations to obtain stable performance estimates. Table 2 summarizes the mean, standard deviation, and median, thereby reducing sensitivity to distributional skewness. For combinations 1 to 6, we ablate our method against the Universal Portfolio (UP) [3] and its side-information variant [22], owning to their strong theoretical guarantees. Because the exact UP incurs exponential computational cost in the number of assets, we also benchmark the Exponential Gradient (EG) [4] universal portfolio and its side-information extension on combination 7 to 12, which scale more favorably to large-scale portfolios. To ensure a fair comparison with side-information portfolios, we recast each forecast as a categorical signal: the asset assigned rank 1 defines the current state, yielding m possible states. Under perfect forecasts, the resulting best state-constant rebalanced portfolio (BSCRP) matches the hindsight optimum. Conversely, the ranking that is adversarial to RAM still offers SI-portfolios exploitable structure, since every state inevitably identifies at least one low-return asset. Constructing a ranking that is simultaneously worst-case for both methods is impractical, we therefore adopt this mapping, recognizing that it is conservative for RAM. All experiments compute under 6h on one standard CPU. Source code is available at `https://github.com/mroymd/OPML`.

Our controlled Monte Carlo study reveals a clear, monotone relationship between forecast quality and portfolio performance. When the oracle is completely uninformative (RAM-R), our algorithm still outperforms the vanilla UP in expectation, albeit with higher variance; its median wealth remains at least twice that of the geometric-mean benchmark. On the other hand, UPSI exhibits lower dispersion and reliably tracks UP, but its expected return trails RAM-R. Introducing even a modest informational edge at 53% forecast accuracy dramatically shifts the landscape: RAM now dominates both EGSI and standard EG in both mean and median wealth. The advantage widens with portfolio dimension: the regret for EGSI grows exponentially in the number

Figure 2: Comb. 1: RAM vs. EGSI over $p \in [0.5, 0.6]$. Similar trends hold for other comb. and $p \in (0.6, 1]$.

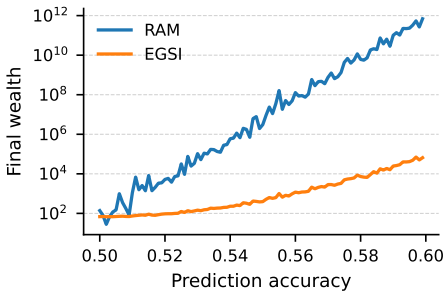


Table 2: Synthetic-forecast ablation on the NYSE dataset. Subscripts denote the forecast-accuracy parameter p. Reported statistics comprise the mean (μ), standard deviation (σ), and median ($\overline{\mu}$). Results average over 1,000 Monte Carlo trials, each drawing predictions according to equation 10.

Comb.	**GM**	**UP**		**RAM-R** $p = 50\%$			**UPSI** $p = 50\%$	
			μ	σ	$\overline{\mu}$	μ	σ	$\overline{\mu}$
1	6.06	39.97	57.78	215.75	13.28	52.33	36.81	40.57
2	14.65	80.53	136.39	551.33	32.15	97.32	64.84	79.14
3	34.52	74.07	101.03	185.44	53.84	84.88	22.54	77.26
4	6.69	32.18	45.66	107.83	17.81	40.61	10.68	37.13
5	5.96	24.40	35.49	96.26	15.14	30.74	8.58	27.95
6	19.35	47.11	56.97	73.23	36.53	55.94	9.84	53.23
Comb.	**GM**	**EG**		**RAM** $p = 53\%$			**EGSI** $p = 53\%$	
			μ	σ	$\overline{\mu}$	μ	σ	$\overline{\mu}$
7	12.88	21.74	1.5E+03	1.9E+03	9.3E+02	22.94	0.43	22.89
8	22.85	33.37	1.0E+03	9.3E+02	7.2E+02	34.80	0.44	34.77
9	17.61	30.18	4.5E+03	6.3E+03	2.6E+03	31.18	0.29	31.15
10	18.43	31.50	4.6E+03	6.5E+03	2.6E+03	32.82	0.36	32.78
11	16.59	27.81	6.1E+03	8.2E+03	3.5E+03	28.61	0.17	28.60
12	14.67	27.86	1.5E+04	2.3E+04	7.3E+03	28.82	0.22	28.81
Comb.	**RAM** $p{=}100\%$	**EGSI** $p{=}100\%$		**RAM** $p = 60\%$			**EGSI** $p = 60\%$	
			μ	σ	$\overline{\mu}$	μ	σ	$\overline{\mu}$
7	1.1E+41	4.9E+07	6.3E+08	8.6E+08	3.4E+08	36.25	2.35	36.19
8	1.5E+35	6.4E+05	5.8E+07	7.0E+07	3.8E+07	47.76	2.05	47.70
9	1.3E+49	6.7E+04	2.8E+10	4.6E+10	1.3E+10	39.59	1.15	39.52
10	3.4E+48	1.4E+05	2.0E+10	3.4E+10	1.0E+10	42.47	1.43	42.34
11	2.5E+53	2.9E+03	1.5E+11	2.3E+11	7.3E+10	33.18	0.59	33.14
12	2.3E+59	1.8E+04	2.5E+12	5.0E+12	1.1E+12	35.10	0.83	35.08

of states, whereas RAM's guarantees are dimension-independent, yielding superior performance in high-asset markets. At 60% accuracy, the gap becomes exponential. Under perfect forecasts where BSCRP coincides with the hindsight optimum: EGSI still converges only asymptotically, whereas RAM immediately captures the optimal growth. These results show that RAM exploits accurate predictive signals more effectively than universal-portfolio baselines; Figure 2 illustrates the resulting monotone gap. We verified the sub-50% accuracy regime but omit the plots for brevity: the trajectories closely match our theoretical bound, with RAM's wealth smoothly converging to the value line without ever crossing below yet retaining domination in expectation. Collectively, these results demonstrate that RAM amplifies even modest predictive edges while provably guarding against catastrophic drawdowns.

We further evaluate RAM on a live, production-grade setting that couples real market data with an industrial-strength learning-to-rank engine. We use the nightly-refreshed S&P 500 historical panel [30] available on Kaggle, containing 501 constituents from 2010 to 2024. The index is liquid, broad-based, and widely adopted in empirical-finance research, ensuring reproducibility and practical relevance. To stress-test robustness, we analyze two disjoint windows: (i) COVID-19 crash from July 2019 to May 2020, capturing extreme volatility; (ii) Recent market from Dec 2022 to Dec 2024, reflecting contemporary trading conditions. We construct three baskets as in Table 3: (i) α-Popular: High-capitalization names favored by retail brokers, mirroring real-world deployability; (ii) β-Diversified: Two tickers from each of five GICS sectors for cross-sector hedging; (iii) γ-Large-scale: A broad slice of the index to probe scalability. Full stock names are in Table 5.

We employ LightGBM LambdaMART [25] to forecast the ranks of next-day returns. The model is retrained each trading day using a 250-day sliding window, featuring contemporaneous and three lagged returns per asset. A decaying factor with $\theta = 0.995^{\text{age}}$ prioritizes recent observations while discarding stale information. A 60-day hold-out slice inside the same window provides early-stopping signals, eliminating look-ahead bias. This protocol guarantees strict online deployment where only data available at decision time enter either training or validation. Hyper-parameters and code are

Table 3: Basket specifications for the S&P 500 experiments. Regime A spans July 2019 to May 2020; regime B covers December 2022 to December 2024.

Market Regime	**Comb.** α	**Comb.** β	**Comb.** γ
A: High volatility	Popularity-weighted	Diversified sector	Random draw
B: Recent window	5 stocks	10 stocks	30 stocks

supplied in the supplementary material while we keep the exposition concise to focus on RAM's prediction-integrated yet risk-free contribution.

Table 4: Empirical results on S&P 500 with production-grade machine-learning forecasts. *Best* denotes the clairvoyant benchmark of the single top-performing stock. *ML* allocates all capital each round to the asset with the highest predicted return (rank-1) from the model.

Market Regime	**Comp.**	**Best**	**GM**	**RAM**	**EGSI**	**ML**
A (Covid-crash)	α	1.733	1.206	1.239	1.230	1.305
	β	1.358	0.953	1.003	0.982	1.678
	γ	1.733	0.885	0.915	0.924	0.532
B (Recent-window)	α	8.293	2.470	2.512	2.629	1.676
	β	3.153	1.300	1.377	1.373	0.572
	γ	8.293	1.401	1.493	1.501	1.391

Across both market regimes, RAM consistently outperforms EGSI whenever the forecast surpass the value-line benchmark and remains resilient even when prediction collapses. In the turbulent COVID-19 window (regime A): RAM systematically exploits informative forecasts, capturing an additional 10.9% and 2.6% of excessive predictive edge over EGSI in baskets α and β respectively; when forecasts turn catastrophic in basket γ, RAM stays within 1% of EGSI while still eclipsing the value line by 3.3%. In the recency-focused window (regime B), the gap widens: although every forecast underperforms the value line, RAM trails EGSI by only 4.5% in basket α and 0.6% in basket γ. Strikingly, under the worst-case basket β where the machine-learning signal lags the value line by 56%, RAM nevertheless beats EGSI by 2.9% and outstrips the value line by 5.9%. In aggregate, these findings provide compelling empirical evidence that RAM converts even modest predictive cues into tangible gains while preserving a universal-style safety net when those cues deteriorate. Notably, this robustness is achieved with an off-the-shelf rank predictor trained solely on historical returns within a narrow asset universe, which suggests that more sophisticated, finance-driven models would amplify RAM's advantage even further, as corroborated by our synthetic simulations.

4 Conclusion

This work advances learning-augmented portfolio selection by introducing the Rebalanced Arithmetic Mean portfolio with predictions (RAM), a principled framework that overlays machine-learning forecasts on a classical rebalancing rule. We prove that RAM captures a constant fraction of the hindsight-optimal wealth when forecasts are perfect and dominates the market's geometric mean baseline even under maximally adversarial signals. Extensive experiments on real-world equity data complements the theory, spanning both synthetic forecast simulations and real-world ML models: RAM harnesses informative predictive signs more effectively than side-information universal portfolios, delivering superior risk-adjusted returns across turbulent and benign regimes. These findings evidence the practical use of forecast without sacrificing worst-case safety

An important practical dimension left unexplored is transaction cost. Because RAM rebalances every round through a permutation layer, it incurs trading frictions analogous to those faced by universal portfolios. The attendant drag on portfolio wealth merits systematic investigation, for instance, through transaction-cost frameworks exemplified by the model in [31]. Future work is also encouraged to establish the Pareto-optimal consistency–robustness trade-off in online portfolio selection. This work represents one operating point via RAM; whether the frontier can be improved (and whether RAM is Pareto-optimal) remains open.

Acknowledgments

We thank the anonymous reviewers for their constructive feedback. Ziliang Zhang gratefully acknowledges the support of his wife, Wenyu Liu, and his parents, Chuanqing Zhang and Zhenying Huang.

References

[1] J. L. Kelly. A new interpretation of information rate. *Bell System Technical Journal*, 35(4): 917–926, 1956. doi: 10.1002/j.1538-7305.1956.tb03809.x.

[2] N. H. Hakansson and W. T. Ziemba. Capital growth theory. In *Handbooks in Operations Research and Management Science, Vol. 9: Finance*, chapter 3, pages 65–86. North-Holland, Amsterdam, 1995. doi: 10.1016/s0927-0507(05)80047-7.

[3] T. M. Cover. Universal portfolios. *Mathematical Finance*, 1(1):1–29, 1991. doi: 10.1111/j.1467-9965.1991.tb00002.x.

[4] D. P. Helmbold, R. E. Schapire, Y. Singer, and M. K. Warmuth. On-line portfolio selection using multiplicative updates. *Mathematical Finance*, 8(4):325–347, 1998. doi: 10.1111/1467-9965.00058.

[5] Burton G. Malkiel. *A Random Walk Down Wall Street: The Time-Tested Strategy for Successful Investing*. W. W. Norton & Company, New York, 13 edition, 1973. ISBN 0-393-06245-7.

[6] B. Li and S. C. H. Hoi. Online portfolio selection. *ACM Computing Surveys*, 46(3):1–36, 2014. doi: 10.1145/2512962.

[7] A. Borodin, R. El-Yaniv, and V. Gogan. Can we learn to beat the best stock. *Journal of Artificial Intelligence Research*, 21:579–594, 2004. doi: 10.1613/jair.1336.

[8] B. Li, P. Zhao, S. C. H. Hoi, and V. Gopalkrishnan. Pamr: Passive aggressive mean reversion strategy for portfolio selection. *Machine Learning*, 87(2):221–258, 2012. doi: 10.1007/s10994-012-5281-z.

[9] László Györfi, Gábor Lugosi, and Frederic Udina. Nonparametric kernel-based sequential investment strategies. *Mathematical Finance*, 16(2):337–357, 2006. doi: 10.1111/j.1467-9965.2006.00274.x.

[10] K. J. Morris, S. E. Egan, J. L. Linsangan, C. K. Leung, A. Cuzzocrea, and X. Calvin. Token-based adaptive time-series prediction by ensembling linear and non-linear estimators: A machine learning approach for predictive analytics on big stock data. In *Proceedings of the International Conference on Machine Learning and Applications (ICMLA)*, pages 1486–1491. IEEE, 2018. doi: 10.1109/icmla.2018.00242.

[11] R. Singh and S. Srivastava. Stock prediction using deep learning. *Multimedia Tools and Applications*, 76(18):18569–18584, 2016. doi: 10.1007/s11042-016-4159-7.

[12] K. Zhang, G. Zhong, J. Dong, S. Wang, and Y. Wang. Stock market prediction based on generative adversarial network. *Procedia Computer Science*, 147:400–406, 2019. doi: 10.1016/j.procs.2019.01.256.

[13] F. Feng, H. Chen, X. He, J. Ding, M. Sun, and T.-S. Chua. Enhancing stock movement prediction with adversarial training. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 5843–5849, 2019. doi: 10.24963/ijcai.2019/810.

[14] ALPS. ALPS: Algorithms with Predictions. `https://algorithms-with-predictions.github.io/`, 2024. Accessed May 10, 2025.

[15] T. Lykouris and S. Vassilvitskii. Competitive caching with machine learned advice. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 7, pages 5201–5213, Stockholm, Sweden, July 2018. International Machine Learning Society (IMLS). ISBN 978-151086796-3. Held from July 10–15, 2018, Code 141700.

[16] Daniel D. Sleator and Robert E. Tarjan. Amortized efficiency of list update and paging rules. *Communications of the ACM*, 28(2):202–208, 1985. doi: 10.1145/2786.2793.

[17] Ravi Kumar, Manish Purohit, and Zoya Svitkina. Improving online algorithms via ml predictions. In H. Larochelle, N. Cesa-Bianchi, H. Wallach, K. Grauman, S. Bengio, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, volume 2018-December, pages 9661–9670, Montreal, Canada, December 2018. Neural Information Processing Systems Foundation. Held from December 2–8, 2018, Code 147412.

[18] Xingjian Bai and Christian Coester. Sorting with predictions. In *Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS 2023)*, 2023. Full paper.

[19] Bo Sun, Russell Lee, Mohammad Hajiesmaili, Adam Wierman, and Danny H. K. Tsang. Pareto-optimal learning-augmented algorithms for online conversion problems. In *Advances in Neural Information Processing Systems*, volume 34, pages 10339–10350, 2021.

[20] Ziyad Benomar, Lorenzo Croissant, Vianney Perchet, and Spyros Angelopoulos. Pareto-optimality, smoothness, and stochasticity in learning-augmented one-max-search. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025. URL `https://openreview.net/forum?id=ZTOVlPC5Vf`. ICML 2025 (poster).

[21] Michael Mitzenmacher and Sergei Vassilvitskii. Algorithms with predictions. *Commun. ACM*, 65(7):33–35, June 2022. ISSN 0001-0782. doi: 10.1145/3528087.

[22] T. M. Cover and E. Ordentlich. Universal portfolios with side information. *IEEE Transactions on Information Theory*, 42(2):348–363, 1996. doi: 10.1109/18.485708.

[23] Alankrita Bhatt, J. Jon Ryu, and Young-Han Kim. On universal portfolios with continuous side information. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS 2023)*, volume 206 of *Proceedings of Machine Learning Research*, pages 4147–4163, Valencia, Spain, 2023. PMLR.

[24] Xingyu Yang, Xiaoteng Zheng, and Lina Zheng. Online portfolio selection strategy with side information based on learning with expert advice. *Asia-Pacific Journal of Operational Research*, 41(05):1–28, October 2024. doi: 10.1142/S0217595923500410.

[25] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*, pages 3146–3154, 2017.

[26] Erik Ordentlich and Thomas M. Cover. The cost of achieving the best portfolio in hindsight. *Mathematics of Operations Research*, 23(4):960–982, November 1998. doi: 10.1287/moor.23.4.960.

[27] L. Bin. Nyse (o) dataset. `http://www.mysmu.edu.sg/faculty/chhoi/olps/datasets/NYSE_O_Dataset.html`, 2024. Accessed January 14, 2025.

[28] L. Bin. Nyse (n) dataset. `http://www.mysmu.edu.sg/faculty/chhoi/olps/datasets/NYSE_N_2_Dataset.html`, 2024. Accessed January 14, 2025.

[29] X. Yang, J. He, J. Xian, H. Lin, and Y. Zhang. Aggregating expert advice strategy for online portfolio selection with side information. *Soft Computing*, 24(3):2067–2081, 2019. doi: 10.1007/s00500-019-04039-7.

[30] Kaggle contributor: Andrew MVD. S&P 500 Stocks (daily-updated) — kaggle dataset. `https://www.kaggle.com/datasets/andrewmvd/sp-500-stocks`, 2024. Dataset accessed 10 May 2025.

[31] A. Blum and A. Kalai. Universal portfolios with and without transaction costs. *Machine Learning*, 35(3):193–205, 1999. ISSN 0885-6125. doi: 10.1023/A:1007530728748.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims are rigorously grounded: Theorem 2.1 formalizes the complete theoretical guarantee, and the experiments empirically corroborate these claims.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The major limitation of this study is its omission of transaction-cost analysis. This choice aligns with much of the online-portfolio literature [3; 22] where the core algorithmic framework is first developed and subsequent work addresses trading frictions separately. We therefore treat transaction-cost modeling as a distinct research track and confine our discussion of it to the future-work (Conclusion) section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Our theoretical claim does not rely on ANY market assumptions beyond the mild requirement that each return factor is strictly positive (i.e., $1 stake cannot generate a loss greater than $1).

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: All sources of randomness are explicitly documented, and the experiments can be reproduced by fixing the global random seed to 42. The paper (and codes) provides every implementation detail required to replicate the primary results in full.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Both the NYSE and S&P 500 datasets are publicly available; moreover, we supply a one-click Colab notebook that fully replicates all reported experiments.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimentation section covers every key setup detail. Any peripheral settings omitted for space are documented in the code.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: For every metric we report the mean, standard deviation, and median derived from 1,000 Monte Carlo trials, thereby conveying the same dispersion information that conventional error bars provide.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All experiments compute under 6h on one standard CPU.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

Justification: This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: All data and model used are publicly available.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All third-party assets are fully credited and their licenses respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification:

    Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Technical Appendices and Supplementary Material

Lemma A.1 (Fractional betting)**.** *Let the forecast be wrong with independent probability $\epsilon \in (0,1)$. For two assets whose return factors in period i satisfy:*

$$(x_1(i), x_2(i)) = \begin{cases} (2, \frac{1}{2}), & \textit{forecast is correct} \\ (\frac{1}{2}, 2), & \textit{forecast is wrong} \end{cases}$$

For any constant $\lambda \in [0,1]$, let the portfolio be $\mathbf{b}(i) = \lambda \mathbf{e}_{\sigma(i)} + (1-\lambda)(\frac{1}{2}, \frac{1}{2})$ where $\mathbf{e}_{\sigma(i)}$ puts weight 1 on the predicted winner and 0 on the other. The following holds:

1. *Expected wealth.*
$$\mathbb{E}(S_n) = \left(\frac{5}{4}\right)^n \left(1 + \frac{3}{5}\lambda(1-2\epsilon)\right)^n$$
2. *Expected per-period log-growth.*
$$W_n(\lambda, \epsilon) := \frac{1}{n}\mathbb{E}\left[\log S_n\right] = \log\frac{5}{4} + (1-\epsilon)\log\left(1 + \frac{3}{5}\lambda\right) + \epsilon \log\left(1 - \frac{3}{5}\lambda\right)$$
3. *Zero-growth error rate. For every fixed $\lambda \geq \frac{1}{3}$, there exists a unique*
$$\epsilon^*(\lambda) = \frac{\log\frac{5}{4} + \log\left(1 + \frac{3}{5}\lambda\right)}{\log\left(1 + \frac{3}{5}\lambda\right) - \log\left(1 - \frac{3}{5}\lambda\right)} \in \left(0, \frac{1}{2}\right)$$
such that $W_n = 0$. For $\epsilon > \epsilon^*(\lambda)$, $W_n < 0$.
4. *Exponential competitive gap. Let $S_n^* = 2^n$ be the hindsight-optimum. Then*
$$\frac{S_n^*}{\mathbb{E}[S_n]} = \left(\frac{8}{5}\right)^n \left(1 + \frac{3}{5}\lambda(1-2\epsilon)\right)^{-n}$$
which grows like $e^{\Omega(n)}$ whenever $\lambda > 0$ and $\epsilon > 0$.

Proof. (1) The per-period growth is $R(i) := \mathbf{b}(i) \cdot \mathbf{x}(i) = \frac{5}{4} \pm \frac{3}{4}\lambda$. When predictions are correct $R^+(i) = \frac{5}{4} + \frac{3}{4}\lambda$, when predictions are wrong $R^-(i) = \frac{5}{4} - \frac{3}{4}\lambda$. Since $R(i)$ are i.i.d,

$$\mathbb{E}[S_n] = \mathbb{E}\left[\prod_{i=1}^n R(i)\right] = \prod_{i=1}^n \mathbb{E}\left[R(i)\right] = \left(\mathbb{E}[R(1)]\right)^n$$
$$= \left[(1-\epsilon)\left(\frac{5}{4} + \frac{3}{4}\lambda\right) + \epsilon\left(\frac{5}{4} - \frac{3}{4}\lambda\right)\right]^n = \left[\frac{5}{4} + \frac{3}{4}\lambda - \frac{3}{2}\lambda\epsilon\right]^n$$
$$= \left[\frac{5}{4} + \frac{3}{4}\lambda(1-2\epsilon)\right]^n = \left(\frac{5}{4}\right)^n \left(1 + \frac{3}{5}\lambda(1-2\epsilon)\right)^n$$

(2) Because $R(i)$ are i.i.d.,

$$W_n(\lambda, \epsilon) = \frac{1}{n}\mathbb{E}[\log S_n] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}[\log R(i)] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}[\log R(1)] = \mathbb{E}[\log R(1)]$$
$$= (1-\epsilon)\log\frac{5}{4}\left(1 + \frac{3}{5}\lambda\right) + \epsilon\log\frac{5}{4}\left(1 - \frac{3}{5}\lambda\right) = \log\frac{5}{4} + (1-\epsilon)\log\left(1 + \frac{3}{5}\lambda\right) + \epsilon\log\left(1 - \frac{3}{5}\lambda\right)$$

(3) Observing $W_n(\lambda, \epsilon)$ is strictly decreasing on ϵ, $\forall \lambda \in [0,1]$:

$$\frac{dW_n(\lambda, \epsilon)}{d\epsilon} = \log\left(1 - \frac{3}{5}\lambda\right) - \log\left(1 + \frac{3}{5}\lambda\right) < 0$$

At $\epsilon = 0$, $W_n(\lambda, 0) = \log\frac{5}{4} + \log\left(1 + \frac{3}{5}\lambda\right) > 0$; At $\epsilon = \frac{1}{2}$, $W_n(\lambda, \frac{1}{2}) = \log\frac{5}{4} + \frac{1}{2}\log\left(1 - (\frac{3}{5}\lambda)^2\right)$, which is negative if and only if $1 - (\frac{3}{5}\lambda)^2 < (\frac{4}{5})^2 \Leftrightarrow \lambda > \frac{1}{3}$. If $\lambda > \frac{1}{3}$, $W_n(\lambda, \epsilon)$ is positive at $\epsilon = 0$,

negative at $\epsilon = \frac{1}{2}$ and strictly decreasing on ϵ. With the intermediate value theorem, there exists exactly one root $\epsilon^* \in (0, \frac{1}{2})$ such that $W_n(\lambda, \epsilon^*) = 0$. Solving this yields:

$$\log \frac{5}{4} + \log\left(1 + \frac{3}{5}\lambda\right) - \epsilon^* \left[\log\left(1 + \frac{3}{5}\lambda\right) - \log\left(1 - \frac{3}{5}\lambda\right)\right] = 0$$

$$\epsilon^*(\lambda) = \frac{\log \frac{5}{4} + \log\left(1 + \frac{3}{5}\lambda\right)}{\log\left(1 + \frac{3}{5}\lambda\right) - \log\left(1 - \frac{3}{5}\lambda\right)} \in \left(0, \frac{1}{2}\right), \qquad \lambda \geq \frac{1}{3}$$

Because $W_n(\lambda, \epsilon)$ is strictly decreasing in ϵ and satisfies $W_n(\lambda, \epsilon^*(\lambda)) = 0$, it immediately follows that $W_n(\lambda, \epsilon) < 0, \forall \epsilon > \epsilon^*(\lambda)$.

(4)

$$\frac{S_n^*}{\mathbb{E}[S_n]} = \left(\frac{2}{\frac{5}{4} + \frac{3}{4}\lambda(1 - 2\epsilon)}\right)^n = \left(\frac{8}{5}\right)^n \left(1 + \frac{3}{5}\lambda(1 - 2\epsilon)\right)^{-n}$$

Since $1 + \frac{3}{5}\lambda(1 - 2\epsilon) < \frac{8}{5}$ whenever $\lambda > 0, \epsilon > 0$, the ratio grows exponentially in n.

□

Table 5: Ticker compositions of three baskets on S&P 500.

Basket	**Constituent tickers**
α	T, MSFT, NVDA, AMZN, V
β	CSCO, MSFT, WRB, RF, T, NFLX, SBUX, BBY, ABT, BAX
γ	CMCSA, AMP, HSIC, FSLR, FCX, DE, CE, VLO, BWA, PH, ANSS, AMZN, C, EXPE, FDX, TJX, WST, EMN, PGR, FAST, PODD, HST, ADM, NVDA, PAYX, BRO, MO, ESS, DTE, WEC