Risk-aware black-box portfolio construction using Bayesian optimization with adaptive weighted Lagrangian estimator

Zinuo You
University of Bristol
School of Computer Science
Bristol, UK
zinuo.you@bristol.ac.uk

John Cartlidge
University of Bristol
Financial Engineering Lab, SEMT
Bristol, UK
john.cartlidge@bristol.ac.uk

Karen Elliott
University of Birmingham
Business School
Birmingham, UK
k.elliott@bham.ac.uk

Menghan Ge
Stratiphy Limited
London, UK
menghan.ge@stratiphy.io

Daniel Gold
Stratiphy Limited
London, UK
daniel@stratiphy.io

Abstract

Existing portfolio management approaches are often black-box models due to safety and commercial issues in the industry. However, their performance can vary considerably whenever market conditions or internal trading strategies change. Furthermore, evaluating these non-transparent systems is expensive, where certain budgets limit observations of the systems. Therefore, optimizing performance while controlling the potential risk of these financial systems has become a critical challenge. This work presents a novel Bayesian optimization framework to optimize black-box portfolio management models under limited observations. In conventional Bayesian optimization settings, the objective function is to maximize the expectation of performance metrics. However, simply maximizing performance expectations leads to erratic optimization trajectories, which exacerbate risk accumulation in portfolio management. Meanwhile, this can lead to misalignment between the target distribution and the actual distribution of the black-box model. To mitigate this problem, we propose an adaptive weight Lagrangian estimator considering dual objective, which incorporates maximizing model performance and minimizing variance of model observations. Extensive experiments demonstrate the superiority of our approach over five backtest settings with three black-box stock portfolio management models. Ablation studies further verify the effectiveness of the proposed estimator.

CCS Concepts

• **Computing methodologies** → **Search with partial observations**; **Continuous space search**; **Heuristic function construction**; • **Information systems** → **Expert systems**; • **Applied computing** → **Forecasting**.

Keywords

Bayesian optimization, Bayesian neural network, tree-structured Parzen estimator, importance sampling, portfolio optimization, financial investing, stock trading

1 Introduction

Stock portfolio management has been an open issue in academia and industry. This refers to selecting optimal stocks and determining optimal weights to maximize portfolio returns or minimize portfolio risks. Traditional methods of portfolio management, such as the Markowitz approach, prioritize diversification of assets to mitigate risk by balancing asset correlations [8, 63]. However, these methods have some restrictive prior assumptions, such as normally distributed returns and stationary markets, contradicting the evolving nature of market conditions [26, 63]. Accordingly, research efforts have shifted toward flexible methods to capture the complex dynamics in stock markets [25, 28]. The deep learning models have achieved remarkable performance in portfolio management, for example: [38] treats the stock selection as a linear programming problem with a multi-layer perceptron as the solver; [1] proposes an ensemble framework to select optimal stocks via different neural networks with genetic algorithm as hyperparameter optimizer; [7] adopts recurrent neural networks to capture temporal dependencies in asset price movements for online stock portfolio management; [31] applies attention mechanisms to learn dependencies of stock time series features, which adaptively assign weights of assets in portfolios; and MDGNN [44] utilizes graph neural networks to model inter-dependencies between stocks to capture evolving market dynamics explicitly.

However, in real-world industrial applications, portfolio management often relies on non-transparent, pre-defined trading models (i.e., proprietary black-box systems). These models are generally expensive to evaluate due to either computational costs or safety issues. Conversely, most academic works, especially deep learning methods, can flexibly define trading models and evaluation metrics. Furthermore, deep learning methods struggle in such settings, as gradient information and large training datasets are rarely available. Consequently, Bayesian optimization has become a powerful framework for optimizing these black-box systems. Unlike traditional methods (e.g., Black-Litterman Model [47] and particle swarm optimization [65]), Bayesian optimization can efficiently navigate exploration (reducing uncertainty) and exploitation (maximizing returns) processes in non-convex objectives and noisy high-dimensional inputs, making it ideal for high-stakes financial applications. It consists of a surrogate model and an acquisition function. The former approximates the unknown objective function by iteratively learning from observations, and the latter guides future observations by balancing exploration and exploitation. In terms of the surrogate model, Gaussian processes (GPs) are a popular choice due to their closed-form posterior distributions and rigorous uncertainty quantification. GPs define predictions via a mean function and covariance kernel (e.g., Matérn kernel, or radial basis function kernel) [51], however computational complexity limits scalability in high-dimensional portfolios. In contrast, the Tree-structured Parzen Estimators (TPEs) [43, 64] partition observations into good and bad groups with kernel density estimators [43]. TPE avoids explicit modeling of covariance structures, enabling faster performance in high-dimensional spaces [64]. Compared to GPs and TPEs, Bayesian neural networks (BNNs) [52, 53] as surrogate models can dynamically adapt non-stationary and complex patterns, by treating network weights as approximated probability distributions. Besides surrogate models, Bayesian optimization relies critically on acquisition functions. These acquisition functions navigate the search of optimal parameter configuration, where popular acquisition functions include Expected Improvement (EI) [3, 51], Probability of Improvement (PI) [51, 57], Upper Confidence Bound (UCB) [21, 55], and Knowledge Gradients (KG) [60, 61].

Conventional Bayesian optimization can lead to two problems. First, by maximizing expected values in a noisy high-dimensional space, Bayesian optimization can produce erratic optimization trajectories. This instability is particularly detrimental in risk-sensitive applications like financial portfolio management and safety-critical systems. To mitigate this issue, several studies [6, 18, 40] have incorporated risk-sensitive proxies into acquisition functions, such as value-at-risk (VaR) [6, 39] and conditional value-at-risk (CVaR) [15, 36], which penalize worst-case outcomes and therefore prioritize robustness over raw performance. Furthermore, RAHBO [32] dynamically models noise distributions to optimize expected values and minimize observation variance jointly. Similarly, LogEI [3], uses a logarithmic reformulation of acquisition functions that prevent vanishing acquisition values in high dimensions or under constraints. However, all these risk-sensitive approaches require careful tuning of confidence parameters (e.g., the alpha factor in CVaR) or noise model parameters (the alpha factor and beta factor in RAHBO), and inappropriate parameter selection can render optimization overly conservative or dangerously reckless. Second, there exists a mismatch between the output distribution of black-box models and the desired distribution characterized by specific performance criteria. Thus, optimizing directly toward the output distribution may not yield the desired outcomes when certain properties of the target distribution are pre-specified. Importance sampling is a variance reduction technique to estimate the target probability distribution with re-weighted samples from the actual output distribution. Consequently, importance sampling is commonly adopted to alleviate this challenge, for example in off-policy training in reinforcement learning and simulation methods.

To address these challenges, we propose a novel adaptive Bayesian optimization framework for stock portfolios. First, our method extends previous risk-aware Bayesian optimization methods [3, 12, 32], such that we jointly maximize the expectation of observations and minimize observational variance. This enables stock portfolio optimization while maintaining stable trajectories. Second, we leverage importance sampling to approximate the desired stock portfolios defined by certain metric-based intervals. The contribution of this work is three-fold:

- We propose a novel Bayesian optimization framework that optimizes portfolio returns (expectation maximization) and risk stability (variance minimization) through a bi-objective function. (See Section 4.3.)
- We present an adaptive importance sampling strategy to effectively guide the surrogate model to approximate target metric-constrained stock portfolios, which is reached by utilizing stock portfolios of non-interested regions that are easy to obtain. (See Section 4.2.)
- We conduct extensive experiments for three black-box trading models on five real-world stock portfolio optimization settings. Compared with single-objective approaches, our method achieves an average improvement of the Sharpe ratio by 0.32 and an average reduction in portfolio variance by 0.034 under 500 observations of stock portfolios. (See Section 5.)

2 Related Work

2.1 Stock Portfolio Management

Stock portfolio management has evolved from Markowitz's modern portfolio theory [34], which balances risk and return by a mean-variance strategy with normality and stationarity assumed. Subsequent studies integrated stochastic control [4, 14], multi-period optimization [41], and risk measures, such as VaR and CVaR [22, 46] adapting to dynamic market changes and mitigate tail risks. Meanwhile, genetic algorithms [35] and approaches based on reinforcement learning [2, 58] have been employed to handle non-convex portfolio objectives and adaptive decision-making for stock selections. Additionally, various deep learning models [9, 62] have been applied to modern portfolio management. For instance, [50] proposes an ensemble framework composed of a multi-scale convolutional neural network online stock portfolio management; DBRNN [49] introduces a meta-heuristic algorithm with returns predictions of stock portfolios; [11] utilizes graph neural networks and incorporates prior knowledge to model non-stationary market patterns, which improves risk-return trade-off in portfolio management; and [24] presents a transformer-based stock portfolio management framework, predicting and adaptively assigning asset weights in future time steps. Besides these approaches, Bayesian Optimization [19, 33] has been adopted in various financial applications, such as hyperparameter tuning for trading strategies [30] and optimizing portfolio weights under uncertainty [16].

2.2 Bayesian Optimization

Bayesian Optimization is a widely adopted framework in evaluating expensive black-box systems, particularly when gradient information is unavailable or observations are constrained [48]. Former studies [27, 37] establish probabilistic models as surrogates to approximate the unknown distribution, enabling uncertainty-aware observation points for optimizations. [5, 20] later formalize GPs as the surrogate model for Bayesian optimization, arising from their flexibility in modeling smooth functions and providing analytic uncertainty estimations. Alternative surrogate models such as random forests [10] and TPEs [43] have been applied in dealing with high-dimensional or discrete spaces. Moreover, Bayesian neural networks have been adopted as surrogate models, for instance, the network weights are treated as the approximated distribution of the black-box models [23, 29]. Regarding acquisition functions, for instance, EI-based methods [3, 51], and UCB-based approaches [21, 55] balance exploration-exploitation trade-offs. Besides, entropy search maximizes information gain to select future observations [13, 59], which commonly prioritizes explorations over exploitations.

3 Preliminary

3.1 Bayesian Optimization

Bayesian optimization [48] is a sample-efficient strategy for optimizing expensive-to-evaluate black-box models under a certain objective. Consider an unknown model represented by a function

$$f : \mathcal{X} \to \mathbb{R}, \tag{1}$$

defined over a bounded domain. The goal is to solve the following maximization problem,

$$\mathbf{x}^* = \arg\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}). \tag{2}$$

Surrogate Model. In the Bayesian optimization framework, $f(\mathbf{x})$ is treated as a random function with a prior distribution $p(f)$. Given a series of observations $\mathcal{D}_n = \{(\mathbf{x}_i, f(\mathbf{x}_i))\}_{i=1}^n$, the posterior distribution over f is derived as,

$$p(f|\mathcal{D}_n) = \frac{p(\mathcal{D}_n|f)p(f)}{p(\mathcal{D}_n)}. \tag{3}$$

For an input $\mathbf{x}$, the predictive distribution is obtained by marginalizing over f,

$$p(f(\mathbf{x})|\mathcal{D}_n) = \int p(f(\mathbf{x})|f)p(f|\mathcal{D}_n)\, df. \tag{4}$$

Acquisition Function. An acquisition function balances exploration (high uncertainty) and exploitation (low predicted $f(\mathbf{x})$). It computes the expected utility of sampling at $\mathbf{x}$,

$$\alpha(\mathbf{x}; \mathcal{D}_n) = \mathbb{E}_{f(\mathbf{x})\sim p(f(\mathbf{x})|\mathcal{D}_n)}[\beta(f(\mathbf{x}))], \tag{5}$$

Here, $\beta(\cdot)$ encodes the utility (e.g., improvement, confidence bounds). The next observation is chosen as,

$$\mathbf{x}_{n+1} = \arg\max_{\mathbf{x}\in\mathcal{X}} \alpha(\mathbf{x}; \mathcal{D}_n). \tag{6}$$

3.2 Importance Sampling

Suppose we wish to compute the expectation of a function $f(\mathbf{x})$ under a target distribution $g(\mathbf{x})$,

$$\mathbb{E}_g[f(\mathbf{x})] = \int f(\mathbf{x})g(\mathbf{x})\, d\mathbf{x}. \tag{7}$$

Directly sampling from $g(\mathbf{x})$ might be challenging. Instead, we sample from an alternative proposal distribution $q_\theta(\mathbf{x})$, which is easier to sample from, provided that $q_\theta(\mathbf{x}) > 0,\ g(\mathbf{x})f(\mathbf{x}) \neq 0$. This allows us to rewrite the expectation as,

$$\begin{aligned}\mathbb{E}_g[f(\mathbf{x})] &= \int f(\mathbf{x})g(\mathbf{x})\, d\mathbf{x}\\ &= \int f(\mathbf{x})\frac{g(\mathbf{x})}{q_\theta(\mathbf{x})}q_\theta(\mathbf{x})\, d\mathbf{x}\\ &= \mathbb{E}_q\Big[f(\mathbf{x})\frac{g(\mathbf{x})}{q_\theta(\mathbf{x})}\Big].\end{aligned} \tag{8}$$

Here, the factor $\frac{g(\mathbf{x})}{q_\theta(\mathbf{x})}$ is known as the importance weight. It corrects for the discrepancy between the target distribution $g(\mathbf{x})$ and the proposal distribution $q_\theta(\mathbf{x})$, allowing the expectation under g to be expressed as a weighted average over samples drawn from g [42, 56].

3.3 Problem Formulation

In this work, consider a portfolio management model as f, which accepts a parameter vector $\mathbf{x} = \{x_1, x_2, \ldots, x_m\}$ and $m \in \mathbb{Z}^+$ is the number of model parameters. The primary objectives are to maximize the model performance $f(\mathbf{x})$ while simultaneously minimizing a risk proxy $\mathcal{R}(f(\mathbf{x}))$. Then, this problem can be expressed as

$$\begin{aligned}&\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}),\\ &\min_{\mathbf{x}\in\mathcal{X}} \mathcal{R}(f(\mathbf{x})).\end{aligned} \tag{9}$$

where $\mathcal{R}(\cdot)$ denotes a risk measure applied to the distribution of model performance induced by $\mathbf{x}$. The model evaluation cost is subject to an implicit constraint on the number of model evaluations η during optimization.

4 Methodology

4.1 Surrogate Model

To effectively optimize the portfolio management model f under limited evaluations, we employ probabilistic surrogate models that approximate f. These surrogates enable scalable evaluation and guide the optimization of Eq. (9) through principled exploration-exploitation trade-offs. We present three surrogate models (GP, TPE, and BNN) compatible with acquisition functions (EI, PI, UCB) that are available in our framework.

4.1.1 *Gaussian Process (GP).*

GP [5, 20] is a non-parametric Bayesian model that places a posterior distribution over functions. Given observations $\mathcal{D}_n = \{(\mathbf{x}_i, f(\mathbf{x}_i))\}_{i=1}^n$, the GP models f as,

$$f(\mathbf{x}) \sim \mathcal{N}\left(\mu(\mathbf{x}),, k(\mathbf{x}, \mathbf{x}')\right), \tag{10}$$

where, $\mu(\mathbf{x})$ is the mean function (often zero without prior knowledge), and $k(\mathbf{x}, \mathbf{x}')$ is a kernel function encoding input similarity ($\mathbf{x}'$ denotes previous inputs). The posterior uncertainty estimates from the GP covariance structure naturally balance exploration and exploitation when used with acquisition functions like EI, PI, and UCB. While GPs excel at sample-efficient optimization in low-dimensional spaces, their complexity $O(m^3)$ scaling limits high-dimensional practicality.

4.1.2 *Tree-structured Parzen Estimator (TPE).*

TPE [43] is a sequential model-based optimization method that models two adaptive densities, which are defined by,

$$\gamma(\mathbf{x}) = P\left(\mathbf{x}|f(\mathbf{x}) < f^*\right), \quad \Omega(\mathbf{x}) = P\left(\mathbf{x}|f(\mathbf{x}) \geq f^*\right), \tag{11}$$

where f^* is a threshold separating "promising" observations (typically the top quantile of evaluated $f(\mathbf{x})$). TPE fits $\gamma(\mathbf{x})$ and $\Omega(\mathbf{x})$ using kernel density estimation, avoiding explicit functional assumptions. Through iteratively refining these distributions and sampling candidates from $\frac{\gamma(\mathbf{x})}{\Omega(\mathbf{x})}$, TPE efficiently handles high-dimensional

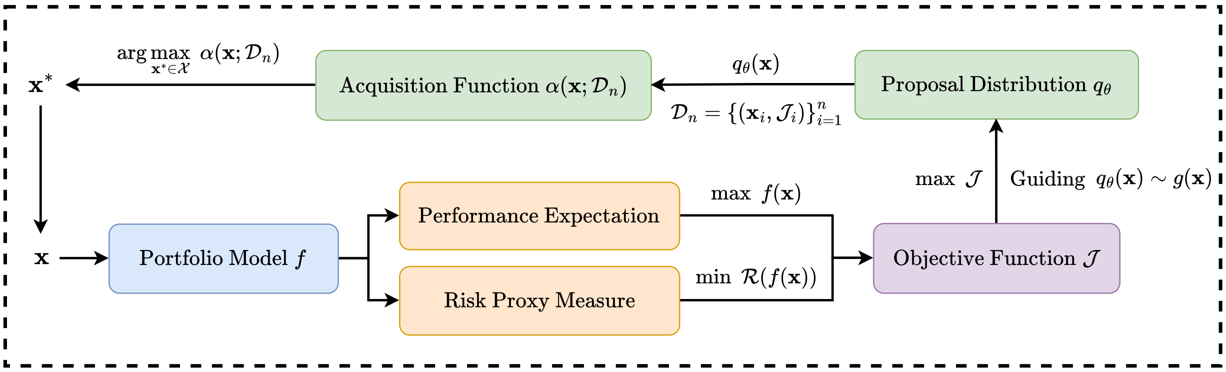


Figure 1: The overall framework for optimizing unknown portfolio management model f. The optimization follows four steps: 1) pass a candidate parameter vector x to the model f; 2) formulate the objective $\mathcal{J}$ based on the maximization of performance expectation and minimization of risk proxy measures; 3) optimizing the proposal distribution q_θ with $\mathcal{J}$; 4) Last, sample the next candidate parameter vector x* by maximizing the acquisition function $\alpha(\mathrm{x}; \mathcal{D}_n)$.

mixed search spaces, making it particularly suitable for portfolio optimization problems.

4.1.3 Bayesian Neural Network (BNN). BNN [23, 29] treats network weights $\mathbf{w}$ as random variables with a learned posterior distribution, thereby naturally capturing model uncertainty. According to the Bayes' theorem, the posterior distribution of the weights $\mathbf{w}$ given the observed data $\mathcal{D}_n$ is expressed as,

$$P(\mathbf{w}|\mathcal{D}_n) = \frac{P(\mathcal{D}_n|\mathbf{w})P(\mathbf{w})}{P(\mathcal{D}_n)}, \tag{12}$$

where $P(\mathbf{w})$ is the prior distribution (e.g., Gaussian), $P(\mathcal{D}_n|\mathbf{w})$ is the likelihood of the data given the weights, and $P(\mathcal{D})$ is the marginal likelihood or evidence. Since the exact posterior $P(\mathbf{w}|\mathcal{D}_n)$ is generally intractable, one can approximate it with variational inference, optimizing a parameterized distribution $q(\mathbf{w}|\theta)$ by maximizing evidence lower bound,

$$\mathcal{L}(\theta) = \mathbb{E}_{q(\mathbf{w}|\theta)}\left[\ln P(\mathcal{D}_n|\mathbf{w})\right] - \mathbb{D}_{\mathrm{KL}}\Big(q(\mathbf{w}|\theta)||P(\mathbf{w})\Big). \tag{13}$$

Here, the second term represents the Kullback–Leibler divergence between the approximate posterior and the prior. Maximizing $\mathcal{L}(\theta)$ yields the optimal variational parameters, effectively quantifying the uncertainty of the surrogate model.

Once trained, BNN provides a predictive distribution for an input $\mathbf{x}$,

$$P(f(\mathbf{x})|\mathbf{x}, \mathcal{D}_n) = \int P(f(\mathbf{x})|\mathbf{x}, \mathbf{w})P(\mathbf{w}|\mathcal{D}_n)\, d\mathbf{w}. \tag{14}$$

In practice, this integral is approximated using Monte Carlo sampling. Specifically, by drawing T samples $\{\mathbf{w}_t\}_{t=1}^T$ from the variational distribution $q(\mathbf{w}|\theta)$, we can obtain,

$$P(f(\mathbf{x})|\mathbf{x}, \mathcal{D}_n) \approx \frac{1}{T}\sum_{t=1}^{T} P(f(\mathbf{x})|\mathbf{x}, \mathbf{w}_t). \tag{15}$$

Typically, the likelihood $P(f(\mathbf{x})|\mathbf{x}, \mathbf{w}_t)$ is assumed to be Gaussian,

$$P(f(\mathbf{x})|\mathbf{x}, \mathbf{w}_t) = \mathcal{N}\Big(f(\mathbf{x})|F_{\mathbf{w}_t}(\mathbf{x}), \sigma_0^2\Big), \tag{16}$$

where $F_{\mathbf{w}_t}(\mathbf{x})$ is the network output for the weight sample and σ_0^2 denotes variance of intrinsic noise from f. Consequently, the predictive distribution becomes a mixture of Gaussians,

$$P(f(\mathbf{x})|\mathbf{x}, \mathcal{D}n) \approx \frac{1}{T}\sum_{t=1}^{T} \mathcal{N}\Big(f(\mathbf{x})|F_{\mathbf{w}_t}(\mathbf{x}), \sigma_0^2\Big). \tag{17}$$

From this mixture, the predictive mean and variance can be computed as,

$$\mu(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} F_{\mathbf{w}t}(\mathbf{x}), \tag{18}$$

$$\sigma^2(\mathbf{x}) = \sigma_0^2 + \frac{1}{T}\sum_{t=1}^{T}\Big(F_{\mathbf{w}_t}(\mathbf{x}) - \mu(\mathbf{x})\Big)^2, \tag{19}$$

where $\mathbf{w}_t \sim q(\mathbf{w}|\theta)$. In practice, we can have $P(f(\mathbf{x})|\mathbf{x}, \mathbf{w}_t) = \mathcal{N}(f(\mathbf{x})|F_{\mathbf{w}}(\mathbf{x}), \sigma_0^2)$, such that $\mu_t = F_{\mathbf{w}_t}(\mathbf{x})$. Therefore, Eq. (17) can be re-written as,

$$P(f(\mathbf{x})|\mathbf{x}, \mathcal{D}_\mathbf{n}) \approx \frac{1}{T}\sum_{t=1}^{T} \mathcal{N}(f(\mathbf{x})|\mu_t, \sigma_0^2). \tag{20}$$

Then, the predictive mean and variance are given by,

$$\mu(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T}\mu_t,\ \ \sigma^2(\mathbf{x}) = \sigma_0^2 + \frac{1}{T}\sum_{t=1}^{T}(\mu_t - \mu(\mathbf{x}))^2. \tag{21}$$

This captures aleatoric (σ_0^2) and epistemic (variance across $F_{\mathbf{w}_t}$) uncertainty, crucial for guiding robust optimization.

4.2 Acquisition Function

Acquisition functions strategically guide parameter search by leveraging the surrogate model's predictions to balance exploration (sampling uncertain regions) and exploitation (focusing on known promising areas). This balance is critical for optimizing portfolio management models with expensive evaluations, where optimal sampling directly impacts convergence speed and solution quality.

4.2.1 Expected Improvement (EI). EI [3, 51] selects the next evaluation point by maximizing the expected improvement over the current best observation f^+. For a candidate $\mathbf{x}$ the improvement is defined as,

$$\mathbb{I}(\mathbf{x}) = \max\left(f(\mathbf{x}) - f^+, 0\right). \tag{22}$$

When the surrogate model is a GP with predictions modeled as a Gaussian distribution $\mathcal{N}(\mu(\mathbf{x}), \sigma^2(\mathbf{x}))$, the EI has a closed-form solution,

$$\alpha(\mathbf{x}) = \mathbb{E}[\mathbb{I}(\mathbf{x})] = \begin{cases} (\mu - f^+ - \zeta)\Phi(Z) + \sigma\phi(Z), & \sigma > 0 \\ 0, & \text{otherwise} \end{cases} \tag{23}$$

Here, $\Phi(Z)$ denotes the cumulative distribution of Gaussian distribution, $\phi(Z)$ denotes the probability function of Gaussian distribution, $Z = \frac{\mu(\mathbf{x}) - f^+ - \zeta}{\sigma(\mathbf{x})}$, $\zeta > 0$ is a hyper-parameter that encourages exploration by effectively lowering the threshold for improvement. For surrogates based on TPE, EI is defined through density ratio maximization,

$$\alpha(\mathbf{x}) = \max \frac{\gamma(\mathbf{x})}{\Omega(\mathbf{x})}, \tag{24}$$

where, γ and Ω represent promising/non-promising observation densities. This non-parametric approach adapts well in high dimensional spaces.

4.2.2 Upper Confidence Bound (UCB). UCB [21, 55] balances exploitation and exploration by combining the predicted mean and uncertainty. The UCB is defined as

$$\alpha(\mathbf{x}) = \mu(\mathbf{x}) + k_t\sigma(\mathbf{x}). \tag{25}$$

Here, k_t controls exploration intensity. Common schedules include $k_t = \sqrt{v \ln t}$ for theoretical guarantees (with $v \in [0.2, 2]$ or constant values for empirical tuning. Unlike the improvement-focused metric of EI-based methods, UCB can directly penalize uncertainty over the optimization process by gradually changing the optimization step t.

4.2.3 Probability of Improvement (PI). PI [51, 57] measures the likelihood of a candidate $\mathbf{x}$ exceeding the current best f^+. It is defined as

$$\alpha(\mathbf{x}) = P(f(\mathbf{x}) > f^+ + \zeta) = \Phi(\frac{\mu(\mathbf{x}) - f^+ - \zeta}{\sigma(\mathbf{x})}). \tag{26}$$

While computationally light, PI solely concentrates on improvement probability rather than magnitude, which is prone to over-exploitation in low-uncertainty regions, premature convergence without proper ζ tuning, and noise in f^+ estimation. In practice, PI requires careful annealing of ζ to maintain exploration, making it less robust than approaches based on EI or UCB for complex portfolio landscapes.

4.3 Adaptive Weighted Lagrangian Estimator

The portfolio optimization problem in Eq. (9) presents a dual challenge, maximizing the performance of the portfolio management model while minimizing the associated risks. Directly optimizing these two objectives typically requires multi-stage procedures, which are undesirable and impractical for real-time financial decision-making due to their computational complexity and lack of unified optimization criteria. Moreover, Bayesian optimization frameworks are most efficient when applied to a single objective. To reconcile these issues, we present an adaptive Lagrangian estimator that balances these competing objectives.

Accordingly, we reformulate Eq. (9) into a constrained optimization problem via Lagrangian multiplier [17],

$$\max \mathcal{J}(\mathbf{x}, \lambda) = \max \left[f(\mathbf{x}) - \lambda\Big(\mathcal{R}(f(\mathbf{x})) - c\Big)\right], \tag{27}$$

where λ is an adaptive regularization parameter and c is a risk tolerance. This formulation explicitly enforces the risk constraint $\mathcal{R}(f(\mathbf{x})) \leq c$ while maximizing expected performance. In this sense, λ enables dynamic response to market regime changes, increasing during volatile periods to prioritize risk reduction, and decreasing in stable markets to enhance returns.

Furthermore, recognizing the divergence between target distribution $g(\mathbf{x})$ and proposal distribution $q_\theta(\mathbf{x})$ (the predictive distribution), we incorporate importance sampling for considering distributional shifts,

$$\max \mathcal{J}(\mathbf{x}, \lambda) = \max \left[\mathbb{E}_{q_\theta}(f(\mathbf{x})) - \lambda\sigma^2_{q_\theta}\left(f(\mathbf{x})\frac{g(\mathbf{x})}{q_\theta(\mathbf{x})}\right)\right], \tag{28}$$

where the expectation is taken with respect to the target distribution g, and the second variance term is computed under the proposal distribution q. In this sense, Eq. (29) enables a principled trade-off between achieving high performance and controlling risk under distributional shifts. In implementation, the importance weight $\frac{g(\mathbf{x})}{q(\mathbf{x})}$ can vary considerably due to market fluctuations or updates in the portfolio management model. To mitigate this, we apply a clipping mechanism on the importance weight to restrict the weight within a specified range,

$$\max \left[\mathbb{E}_{q_\theta}(f(\mathbf{x})) - \lambda\sigma^2_{q_\theta}\left(f(\mathbf{x}) \cdot \mathbf{clip}\left(\frac{g(\mathbf{x})}{q_\theta(\mathbf{x})}, 1-\epsilon, 1+\epsilon\right)\right)\right]. \tag{29}$$

Here, ϵ controls bias-variance trade-off (normally $\epsilon \in [0.1, 0.35]$), where small ϵ reduces variance but increases approximation bias and large ϵ preserves unbiasedness at risk of weight explosion. Furthermore, λ at optimization step t is given by the following schedule,

$$\lambda_t = \frac{1 - \cos\left(\min\left(\frac{t}{500}\pi, \pi\right)\right)}{2}. \tag{30}$$

5 Empirical Analysis

5.1 Dataset Information

In this work, the proposed framework is evaluated with three black-box portfolio management models (denoted as M1, M2, and M3) and five backtest settings (S1, S2, S3, S4, and S5) composing fifteen scenarios. The black-box models and backtest systems are unavailable to the public due to commercial and safety issues. The code of the proposed optimization framework is available at **HYPERREF HERE**.

5.2 Experimental Setting

In this work, the observation budget $\eta = 500$. If the surrogate model is BNN, then the sample for generating predictive distribution $T = 1600$. The number of layers ranges $[2, 3]$, the activation function is set as PReLU, the optimizer is set to AdamW, the number of training epochs is set to 1800, the learning rate is set to 10^{-3}, the intrinsic noise $\sigma_0^2 = 0.1$, and the prior distribution $P(\mathbf{w}) \sim \mathcal{N}(0, 1)$.

Table 1: Optimization results of the proposed adaptive Lagrangian estimator with different surrogate-acquisition configurations (η = 500). Here, M1 - M3 denote three black-box portfolio management models and S1 - S5 denote five backtest settings. The maximum model performance of the optimization is denoted as $f_{\max}$, and the model performance's variance of the optimization $\sigma^2_{q_\theta}(f(\mathbf{x}))$ represents the stability of the optimization during optimization. Best $f_{\max}$ is in bold and best $\sigma^2_{q_\theta}(f)$ is double underlined.

Method	M1-S1		M1-S2		M1-S3		M1-S4		M1-S5	
	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$
GP-EI	2.998	1.278	1.904	0.407	1.028	0.180	0.719	0.078	0.504	0.037
TPE-EI	**3.091**	0.613	1.678	0.111	**1.035**	0.073	**0.754**	0.049	0.532	0.017
BNN-EI	2.999	1.917	**2.001**	1.106	0.989	0.453	0.689	0.217	**0.590**	0.175
GP-UCB	2.752	1.382	1.656	0.497	0.965	0.191	0.545	0.102	0.470	0.072
BNN-UCB	2.646	1.955	1.872	1.119	0.998	0.697	0.568	0.222	0.569	0.201
GP-PI	2.866	0.690	1.665	0.223	1.001	0.116	0.699	0.059	0.500	0.026
BNN-PI	2.665	1.341	1.547	0.533	0.981	0.209	0.711	0.110	0.483	0.096

Method	M2-S1		M2-S2		M2-S3		M2-S4		M2-S5	
	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$
GP-EI	2.976	0.900	1.672	0.277	**1.222**	0.171	**0.860**	0.085	0.604	0.038
TPE-EI	**3.405**	0.492	**2.006**	0.170	1.049	0.111	0.787	0.049	0.593	0.021
BNN-EI	3.145	1.667	1.849	1.070	1.093	0.468	0.776	0.221	**0.671**	0.178
GP-UCB	2.730	1.004	1.424	0.367	1.159	0.182	0.686	0.109	0.570	0.073
BNN-UCB	2.792	1.705	1.920	1.083	1.102	0.712	0.655	0.226	0.650	0.204
GP-PI	2.844	0.512	1.433	0.193	1.195	0.107	0.840	0.066	0.600	0.027
BNN-PI	2.811	1.091	1.595	0.497	1.085	0.224	0.798	0.114	0.564	0.099

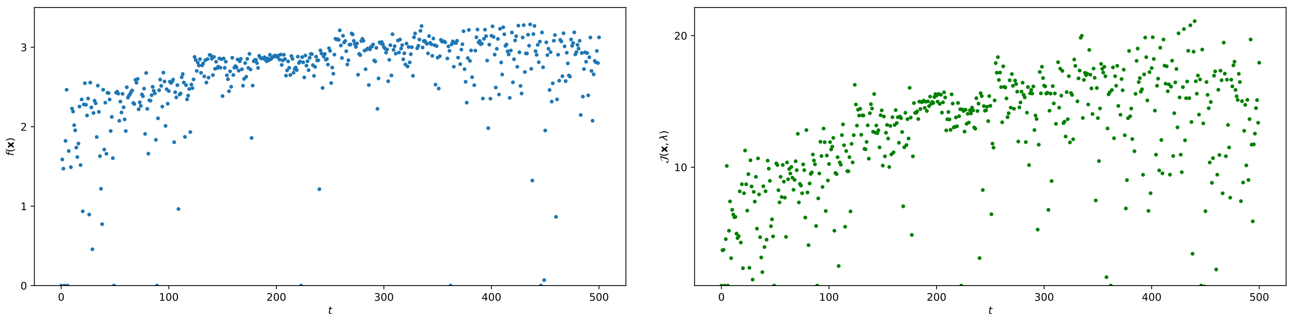
Method	M3-S1		M3-S2		M3-S3		M3-S4		M3-S5	
	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$	$f_{\max}(\mathbf{x})$	$\sigma^2_{q_\theta}(f(\mathbf{x}))$
GP-EI	**3.480**	1.327	1.596	0.220	1.040	0.176	0.783	0.052	0.635	0.041
TPE-EI	3.288	0.309	**1.673**	0.084	**1.047**	0.071	**0.821**	0.033	**0.670**	0.019
BNN-EI	3.479	1.991	1.660	0.597	1.001	0.443	0.751	0.145	0.543	0.194
GP-UCB	3.192	1.435	1.386	0.268	0.977	0.187	0.594	0.068	0.592	0.080
BNN-UCB	3.070	2.031	1.565	0.604	1.010	0.682	0.619	0.148	0.517	0.223
GP-PI	3.323	0.717	1.394	0.120	1.013	0.113	0.761	0.039	0.630	0.029
BNN-PI	3.095	1.393	1.295	0.288	0.993	0.204	0.775	0.073	0.508	0.106

For discrete values in $\mathbf{x} = \{x_1, x_2, \ldots, x_m\}$, the predicted candidate is rounded to integers.

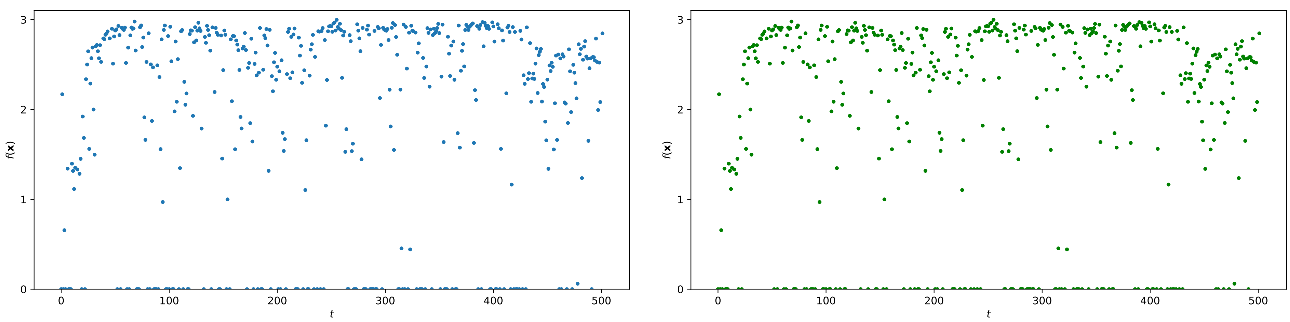
6 Performance Evaluation

From Table. 1, we can observe that the TPE-EI generally outperforms other methods across three black-box portfolio management models on five backtest settings. Concretely, TPE with EI achieves the smallest $\sigma^2_{q_\theta}(f(\mathbf{x}))$ over all optimization scenarios, though it does not maintain the highest $f_{\max}(\mathbf{x})$. Conversely, GP with EI possesses larger $f_{\max}(\mathbf{x})$ than GP with either UCB or PI. Meanwhile, GP-PI performs more mildly during optimization compared to GP-UCB and GP-EI. Besides the above two non-parametric surrogate models, BNN-based methods have the largest $\sigma^2_{q_\theta}(f(\mathbf{x}))$ over all optimization scenarios, while occasionally exceeding other methods in terms of $f_{\max}(\mathbf{x})$. For instance, BNN-EI outperforms other models in M1-S2, M1-S5, M2-S5, and M3-S2. Recall that BNN approximates the unknown portfolio management model by training a neural network with observations from the model. Therefore, this can lead to a severe overfitting issue when the observations are limited, which further results in high $\sigma^2_{q_\theta}(f(\mathbf{x}))$ during optimization. Additionally, when paired with PI, BNN performs more stable than with EI and UCB though in some scenarios BNN-PI underperforms BNN-EI and BNN-UCB regarding $f_{\max}(\mathbf{x})$.

These results indicate that when guided by the proposed adaptive weighted Lagrangian estimator, the exploration and exploitation during optimization are appropriately balanced. Except for these



(a) Optimization trajectories of TPE-EI with proposed estimator $\mathcal{J}(\mathbf{x}, \lambda)$, where $f_{\max}(\mathbf{x}) = 3.288$ and $\sigma^2_{q_\theta}(f(\mathbf{x})) = 0.309$. The optimized model performance $f(\mathbf{x})$ gradually increases as optimization step t grows, with the smoothest optimization regions achieving around $t = 200$. Although the objective scores follow similar patterns as $f(\mathbf{x})$ before $t = 300$, the objective scores tend to be more distributive as t keeps increasing.



(b) Optimization trajectories of TPE-EI with conventional estimator $f(\mathbf{x})$, where $f_{\max}(\mathbf{x}) = 2.980$ and $\sigma^2_{q_\theta}(f(\mathbf{x})) = 1.367$. Unlike the above trajectories guided by $\mathcal{J}(\mathbf{x}, \lambda)$, directly optimizing $f(\mathbf{x})$ results in more volatile and spread trajectories. Furthermore, the points around $f(\mathbf{x}) = 0$ are more than that guided by $\mathcal{J}(\mathbf{x}, \lambda)$.

Figure 2: Optimization trajectories of TPE-EI, with x-axis showing 500 optimization steps. Blue dots represent portfolio model performance $f(\mathbf{x})$; green dots represent the corresponding objective score for each optimization step.

observations, we can draw one conclusion on the portfolio management models and backtest settings. Generally, the maximum model performance and instability are reduced in descending order of M3, M2, M1, and S1, S2, S3, S4, S5.

7 Ablation Study

In this section, we compare the conventional objective of Bayesian optimization ($f(\mathbf{x})$) and the proposed objective ($\mathcal{J}(\mathbf{x}, \lambda)$) with TPE-EI on M3-S1.

On the one hand, from Fig. 2, we can observe that under 500 observations, TPE-EI with the proposed estimator consistently outperforms TPE-EI with the conventional estimator in terms of $f_{\max}(\mathbf{x})$ and $\sigma^2_{q_\theta}(f(\mathbf{x}))$. Furthermore, the trajectories of TPE-EI with $\mathcal{J}(\mathbf{x}, \lambda)$ exhibit a more straightforward pattern, of which $f(\mathbf{x})$ is progressively increasing and forming relatively tight regions. Notably, the objective scores diverge after a middle stage of optimization (around $t = 300$), this suggests that the regularization intensity begins to significantly affect the proposal distribution q_θ to prioritize reducing the variance term in the objective. Consequently, this reiterates the importance of the adaptive scheduling of λ in the objective, effectively balancing portfolio management models' performance optimization and risk control. On the other hand, when guided by traditional objective, directly maximizing $f(\mathbf{x})$, the trajectories demonstrate a different pattern. The performance of the portfolio management model varies considerably during optimization, where points of $f(\mathbf{x}) = 0$ frequently occur (blue spikes and green spikes over the horizontal axis). Therefore, solely concentrating on the performance of portfolio management models for model optimization is neither feasible nor desirable.

Besides optimization performance, we provide comparisons between adopting $\mathcal{J}(\mathbf{x}, \lambda)$ as the objective and $f(\mathbf{x})$ as the objective in Tab. 2 and Tab. 3. From the results, the average time taken per optimization step with the proposed estimator is slightly higher than that with the conventional estimator. Concretely, the largest difference is 1.48 seconds, observed in GP-UCB for M3-S1, and the smallest difference is 0.50 seconds, occurring in BNN-PI for M3-S5. Moreover, BNN methods are consistently slower than other methods by about 19.89 seconds on average across all scenarios (M3-S1 to M3-S5).

Table 2: Average time (in seconds) taken for each optimization step using $\mathcal{J}(\mathbf{x}, \lambda)$ as objective including portfolio management model response time.

Method	M3-S1	M3-S2	M3-S3	M3-S4	M3-S5
GP-EI	46.47	47.88	50.02	61.23	74.45
TPE-EI	46.22	48.13	49.77	60.76	72.81
BNN-EI	65.01	70.13	69.69	79.16	92.72
GP-UCB	46.47	47.88	50.02	61.23	74.45
BNN-UCB	64.53	72.35	70.11	78.54	94.61
GP-PI	47.10	47.08	49.62	62.04	73.74
BNN-PI	65.47	73.20	70.91	79.08	93.75

Table 3: Average time (in seconds) taken for each optimization step using $f(\mathbf{x})$ as objective including portfolio management model response time.

Method	M3-S1	M3-S2	M3-S3	M3-S4	M3-S5
GP-EI	45.12	46.49	48.71	60.62	73.70
TPE-EI	45.42	47.61	49.02	59.35	72.22
BNN-EI	64.19	69.55	68.5	77.81	91.64
GP-UCB	44.99	46.33	48.93	60.41	73.49
BNN-UCB	63.57	70.90	69.17	77.16	93.85
GP-PI	46.48	45.66	48.69	60.72	72.82
BNN-PI	64.25	71.99	69.62	77.96	93.25

8 Discussion

According to previous experimental results, we can draw three conclusions. In general, the proposed adaptive Lagrangian estimator consistently outperforms the conventional preference (solely maximizing model performance) for optimizing black-box portfolio management models.

First, risk control has always been a critical issue in many modern financial applications, which sometimes is even more crucial than maintaining high performance. Especially, when evaluation budgets are limited or portfolio management models are operating in dynamic environments (e.g., real-world markets and changing backtest settings). Second, principled trade-offs between multiple requirements are essential in optimizing portfolio management models. Concretely, as the optimization continues the customers in the industry often prefer models with more stable performance with acceptable performance. Therefore, adaptive scheduling in the optimization framework from explorative sampling to exploitive sampling is an appropriate approach. Last, we can observe that BNNs as surrogate models are not ideal in optimizing black-box portfolio management models with limited observations. This arises from the inherent nature of neural networks, which requires sufficient effective samples to learn and approximate latent distributions [45, 54].

In addition, BNNs take almost 20 seconds more than other surrogate models and possess the most volatile optimization trajectories, though they have larger maximum model performance in some scenarios. Conversely, TPEs and GPs are more practical in this case, which effectively balances exploration and exploitation given constrained observation budgets. Considering the scalability of parameters, compatibility of discrete or categorial parameters, and performance-risk trade-off, TPEs are more balanced choices as surrogate models for portfolio management model optimization.

9 Conclusions

In this work, we present a novel objective for optimizing black-box portfolio management models with Bayesian optimization, namely the adaptive weighted Lagrangian estimator. Compared to most application scenarios where maximized model performance is the only objective, risk-sensitive and time-sensitive financial applications require optimizing model performance while controlling the volatility of model performance.

From the experimental results, we make three conclusions. First, the proposed adaptive weighted Lagrangian estimator-guided optimization maintains similar maximum model performance to the performance-first objective and achieves lower variance in observations of model performance. Second, among the Bayesian optimization methods on three black-box portfolio management models over five backtest settings, TPE with EI generally outperforms other popular methods, such as maximum model performance and variance of model observations. Last, TPE with EI surpasses other surrogate-acquisition pairs on optimization efficiency, realizing faster optimization given the same observation budget.

Although achieving these promising results, this work has two limitations. First, the weight clipping relies on tuning the hyperparameter ϵ, which can result in drastic weights or severe bias if not carefully balanced. Second, the magnitude of the regularization term λ_t requires appropriate tuning when the two terms in the proposed estimator do not have similar scales. In addition, TPE-based Bayesian optimization utilizes two density kernels to model the target distribution implicitly, future work will consider integrating uncertainty estimates into the density models.

Acknowledgements

This work was supported by UK Research and Innovation (UKRI) Innovate UK grant number 10094067: Stratlib.AI - A trusted machine learning platform for asset and credit managers.

References

[1] Chaher Alzaman. 2024. Deep learning in stock portfolio selection and predictions. Expert Systems with Applications 237:B (2024), 121404. doi:10.1016/j.eswa.2023.121404

[2] Chaher Alzaman. 2025. Optimizing portfolio selection through stock ranking and matching: A reinforcement learning approach. Expert Systems with Applications 269 (2025), 126430. doi:10.1016/j.eswa.2025.126430

[3] Sebastian Ament, Samuel Daulton, David Eriksson, Maximilian Balandat, and Eytan Bakshy. 2023. Unexpected improvements to expected improvement for Bayesian optimization. Advances in Neural Information Processing Systems 37 (2023), 20577–20612.

[4] Diana Barro, Giorgio Consigli, and Vivek Varun. 2022. A stochastic programming model for dynamic portfolio management with financial derivatives. Journal of Banking & Finance 140 (2022), 106445. doi:10.1016/j.jbankfin.2022.106445

[5] Adam D Bull. 2011. Convergence rates of efficient global optimization algorithms. Journal of Machine Learning Research 12, 88 (2011), 2879–2904.

[6] Sait Cakmak, Raul Astudillo Marban, Peter Frazier, and Enlu Zhou. 2020. Bayesian optimization of risk measures. Advances in Neural Information Processing Systems 33 (2020), 20130–20141.
[7] Xinwei Cao, Adam Francis, Xujin Pu, Zenan Zhang, Vasilios Katsikis, Predrag Stanimirovic, Ivona Brajevic, and Shuai Li. 2023. A novel recurrent neural network based online portfolio analysis for high frequency trading. Expert Systems with Applications 233 (2023), 120934. doi:10.1016/j.eswa.2023.120934
[8] Apichat Chaweewanchon and Rujira Chaysiri. 2022. Markowitz mean-variance portfolio optimization with predictive stock selection using machine learning. International Journal of Financial Studies 10, 3 (2022), 64. doi:10.3390/ijfs10030064
[9] Qishuo Cheng, Le Yang, Jiajian Zheng, Miao Tian, and Duan Xin. 2024. Optimizing portfolio management and risk assessment in digital assets using deep learning for predictive analysis. In International Conference on Artificial Intelligence, Internet and Digital Economy (ICAID) (Atlantis Highlights in Intelligent Systems, Vol. 11). Springer Nature, Dordrecht, Netherlands, 30. doi:10.2991/978-94-6463-490-7_5
[10] Siva Krishna Dasari, Abbas Cheddad, and Petter Andersson. 2019. Random forest surrogate models to support design space exploration in aerospace use-case. In Artificial Intelligence Applications and Innovations (AIAI) (IFIP Advances in Information and Communication Technology, Vol. 559). Springer, Cham, 532–544. doi:10.1007/978-3-030-19823-7_45
[11] Ömer Ekmekcioğlu and Mustafa Ç Pınar. 2023. Graph neural networks for deep portfolio optimization. Neural Computing and Applications 35, 28 (2023), 20663–20674. doi:10.1007/s00521-023-08862-w
[12] Yingjie Fei, Zhuoran Yang, Yudong Chen, Zhaoran Wang, and Qiaomin Xie. 2020. Risk-sensitive reinforcement learning: Near-optimal risk-sample tradeoff in regret. Advances in Neural Information Processing Systems 33 (2020), 22384–22395.
[13] Daniel Hernández-Lobato, Jose Hernandez-Lobato, Amar Shah, and Ryan Adams. 2016. Predictive entropy search for multi-objective bayesian optimization. In International Conference on Machine Learning. PMLR, 1492–1501.
[14] Florian Herzog, Gabriel Dondi, and Hans P Geering. 2007. Stochastic model predictive control and portfolio optimization. International Journal of Theoretical and Applied Finance 10, 02 (2007), 203–233. doi:10.1142/S0219024907004196
[15] Takuya Hiraoka, Takahisa Imagawa, Tatsuya Mori, Takashi Onishi, and Yoshimasa Tsuruoka. 2019. Learning robust options by conditional value at risk optimization. Advances in Neural Information Processing Systems 33 (2019), 11 pages.
[16] Matthew Hoffman, Eric Brochu, Nando De Freitas, et al. 2011. Portfolio Allocation for Bayesian Optimization. In Conference on Uncertainty in Artificial Intelligence (UAI). 327–336.
[17] Kazufumi Ito and Karl Kunisch. 2008. Lagrange multiplier approach to variational problems and applications. SIAM.
[18] Shogo Iwazaki, Yu Inatsu, and Ichiro Takeuchi. 2021. Mean-variance analysis in Bayesian optimization under uncertainty. In International Conference on Artificial Intelligence and Statistics. PMLR, Cambridge, MA, 973–981.
[19] Syed Hasan Jafar. 2022. Financial applications of gaussian processes and bayesian optimization. In Bayesian Reasoning and Gaussian Processes for Machine Learning Applications. Chapman and Hall/CRC, 111–122.
[20] Donald R Jones, Matthias Schonlau, and William J Welch. 1998. Efficient global optimization of expensive black-box functions. Journal of Global Optimization 13 (1998), 455–492. doi:10.1023/A:1008306431147
[21] Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. 2012. On Bayesian upper confidence bounds for bandit problems. In Artificial Intelligence and Statistics, Vol. 22. PMLR, 592–600.
[22] Michal Kaut, Hercules Vladimirou, Stein W Wallace, and Stavros A Zenios. 2007. Stability analysis of portfolio management with conditional value-at-risk. Quantitative Finance 7, 4 (2007), 397–409. doi:10.1080/14697680701483222
[23] Baptiste Kerleguer, Claire Cannamela, and Josselin Garnier. 2024. A Bayesian neural network approach to multi-fidelity surrogate modeling. International Journal for Uncertainty Quantification 14, 1 (2024), 43–60. doi:10.1615/Int.J.UncertaintyQuantification.2023044584
[24] Damian Kisiel and Denise Gorse. 2022. Portfolio transformer for attention-based asset allocation. In International Conference on Artificial Intelligence and Soft Computing. Springer, Cham, 61–71. doi:10.1007/978-3-031-23492-7_6
[25] Siddarth Krishnamoorthy, Satvik Mehul Mashkaria, and Aditya Grover. 2023. Diffusion models for black-box optimization. In International Conference on Machine Learning. PMLR, 17842–17857.
[26] Ihsan Kulali. 2016. Portfolio optimization analysis with Markowitz quadratic mean-variance model. European Journal of Business and Management 8, 7 (2016), 73–79.
[27] Harold J Kushner. 1964. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. J. Basic Eng. 86, 1 (1964), 97–106. doi:10.1115/1.3653121
[28] Yang Li, Yu Shen, Wentao Zhang, Yuanwei Chen, Huaijun Jiang, Mingchao Liu, Jiawei Jiang, Jinyang Gao, Wentao Wu, Zhi Yang, et al. 2021. OpenBox: A generalized black-box optimization service. In ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 3209–3219. doi:10.1145/3447548.3467061
[29] Yee-Fun Lim, Chee Koon Ng, US Vaitesswar, and Kedar Hippalgaonkar. 2021. Extrapolative Bayesian optimization with Gaussian process and neural network ensemble surrogate models. Advanced Intelligent Systems 3, 11 (2021), 2100101. doi:10.1002/aisy.202100101
[30] Peng Liu. 2023. Optimizing Trading Strategies with Bayesian optimization. In Quantitative Trading Strategies Using Python: Technical Analysis, Statistical Testing, and Machine Learning. Springer, Cham, 257–301. doi:10.1007/978-1-4842-9675-2_9
[31] Tian Ma, Wanwan Wang, and Yu Chen. 2023. Attention is all you need: An interpretable transformer-based asset allocation approach. International Review of Financial Analysis 90 (2023), 102876. doi:10.1016/j.irfa.2023.102876
[32] Anastasia Makarova, Ilnura Usmanova, Ilija Bogunovic, and Andreas Krause. 2021. Risk-averse heteroscedastic bayesian optimization. Advances in Neural Information Processing Systems 34 (2021), 17235–17245.
[33] Godeliva Petrina Marisu and Chi Seng Pun. 2023. Bayesian estimation and optimization for learning sequential regularized portfolios. SIAM Journal on Financial Mathematics 14, 1 (2023), 127–157. doi:10.1137/21M1427176
[34] Harry Markowitz. 1952. Portfolio selection. Journal of Finance 7, 1 (1952), 77–91. doi:10.2307/2975974
[35] Konstantinos Metaxiotis and Konstantinos Liagkouras. 2012. Multiobjective evolutionary algorithms for portfolio management: A comprehensive literature review. Expert systems with applications 39, 14 (2012), 11685–11698. doi:10.1016/j.eswa.2012.04.053
[36] Zakaria Mhammedi, Benjamin Guedj, and Robert C Williamson. 2020. Pac-bayesian bound for the conditional value at risk. Advances in Neural Information Processing Systems 33 (2020), 17919–17930.
[37] Jonas Močkus. 1975. On Bayesian methods for seeking the extremum. In Optimization Techniques IFIP Technical Conference Novosibirsk, July 1–7, 1974 (Lecture Notes in Computer Science, Vol. 27). Springer, Berlin, Heidelberg, 400–404. doi:10.1007/3-540-07165-2_55
[38] Alireza Nazemi, Behzad Abbasi, and Farahnaz Omidi. 2015. Solving portfolio selection models with uncertain returns using an artificial neural network scheme. Applied Intelligence 42 (2015), 609–621. doi:10.1007/s10489-014-0616-z
[39] Quoc Phong Nguyen, Zhongxiang Dai, Bryan Kian Hsiang Low, and Patrick Jaillet. 2021. Optimizing conditional value-at-risk of black-box functions. Advances in Neural Information Processing Systems 34 (2021), 4170–4180.
[40] Quoc Phong Nguyen, Zhongxiang Dai, Bryan Kian Hsiang Low, and Patrick Jaillet. 2021. Value-at-risk optimization with Gaussian processes. In International Conference on Machine Learning. PMLR, 8063–8072.
[41] Razvan Oprisor and Roy Kwon. 2021. Multi-period portfolio optimization with investor views under regime switching. Journal of Risk and Financial Management 14(1), 3 (2021), 31 pages. doi:10.3390/jrfm14010003
[42] Art Owen and Yi Zhou. 2000. Safe and effective importance sampling. J. Amer. Statist. Assoc. 95, 449 (2000), 135–143.
[43] Yoshihiko Ozaki, Yuki Tanigaki, Shuhei Watanabe, Masahiro Nomura, and Masaki Onishi. 2022. Multiobjective tree-structured Parzen estimator. Journal of Artificial Intelligence Research 73 (2022), 1209–1250. doi:10.1613/jair.1.13188
[44] Hao Qian, Hongting Zhou, Qian Zhao, Hao Chen, Hongxiang Yao, Jingwei Wang, Ziqi Liu, Fei Yu, Zhiqiang Zhang, and Jun Zhou. 2024. MDGNN: Multi-Relational Dynamic Graph Neural Network for Comprehensive and Dynamic Stock Investment Prediction. In AAAI Conference on Artificial Intelligence, Vol. 38(13). 14642–14650. doi:10.1609/aaai.v38i13.29381
[45] Wojciech Samek, Grégoire Montavon, Sebastian Lapuschkin, Christopher J Anders, and Klaus-Robert Müller. 2021. Explaining deep neural networks and beyond: A review of methods and applications. Proc. IEEE 109, 3 (2021), 247–278.
[46] Sergey Sarykalin, Gaia Serraino, and Stan Uryasev. 2008. Value-at-risk vs. conditional value-at-risk in risk management and optimization. In State-of-the-art decision-making tools in the information-intensive age. INFORMS, Chapter 13, 270–294. doi:10.1287/educ.1080.0052
[47] Stephen Satchell and Alan Scowcroft. 2007. A demystification of the Black-Litterman model: Managing quantitative and traditional portfolio construction. In Forecasting expected returns in the financial markets. Elsevier, 39–53.
[48] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. 2015. Taking the human out of the loop: A review of Bayesian optimization. Proc. IEEE 104, 1 (2015), 148–175. doi:10.1109/JPROC.2015.2494218
[49] Meeta Sharma and Hardayal Singh Shekhawat. 2022. Portfolio optimization and return prediction by integrating modified deep belief network and recurrent neural network. Knowledge-Based Systems 250 (2022), 109024. doi:10.1016/j.knosys.2022.109024
[50] Si Shi, Jianjun Li, Guohui Li, and Peng Pan. 2019. A multi-scale temporal feature aggregation convolutional neural network for portfolio management. In ACM International Conference on Information and Knowledge Management (CIKM). 1613–1622. doi:10.1145/3357384.3357961
[51] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. 2012. Practical bayesian optimization of machine learning algorithms. Advances in Neural Information Processing Systems 25 (2012), 9 pages.

[52] Jasper Snoek, Oren Rippel, Kevin Swersky, Ryan Kiros, Nadathur Satish, Narayanan Sundaram, Mostofa Patwary, Mr Prabhat, and Ryan Adams. 2015. Scalable bayesian optimization using deep neural networks. In International Conference on Machine Learning. PMLR, 2171–2180.

[53] Jost Tobias Springenberg, Aaron Klein, Stefan Falkner, and Frank Hutter. 2016. Bayesian optimization with robust Bayesian neural networks. Advances in Neural Information Processing Systems 29 (2016), 9 pages.

[54] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S Emer. 2017. Efficient processing of deep neural networks: A tutorial and survey. Proc. IEEE 105, 12 (2017), 2295–2329.

[55] Shion Takeno, Yu Inatsu, and Masayuki Karasuyama. 2023. Randomized Gaussian process upper confidence bound with tighter Bayesian regret bounds. In International Conference on Machine Learning. PMLR, 33490–33515.

[56] Surya T Tokdar and Robert E Kass. 2010. Importance sampling: a review. Wiley Interdisciplinary Reviews: Computational Statistics 2, 1 (2010), 54–60.

[57] Xilu Wang, Yaochu Jin, Sebastian Schmitt, and Markus Olhofer. 2023. Recent advances in Bayesian optimization. Comput. Surveys 55, 13s (2023), 1–36. doi:10.1145/3582078

[58] Zhicheng Wang, Biwei Huang, Shikui Tu, Kun Zhang, and Lei Xu. 2021. DeepTrader: A deep reinforcement learning approach for risk-return balanced portfolio management with market conditions embedding. In AAAI Conference on Artificial Intelligence, Vol. 35(1). 643–650. doi:10.1609/aaai.v35i1.16144

[59] Zi Wang and Stefanie Jegelka. 2017. Max-value entropy search for efficient Bayesian optimization. In International Conference on Machine Learning. PMLR, 3627–3635.

[60] Jian Wu and Peter Frazier. 2016. The parallel knowledge gradient method for batch Bayesian optimization. Advances in Neural Information Processing Systems 29 (2016), 9 pages.

[61] Jian Wu, Matthias Poloczek, Andrew G Wilson, and Peter Frazier. 2017. Bayesian optimization with gradients. Advances in Neural Information Processing Systems 30 (2017), 12 pages.

[62] Hyungbin Yun, Minhyeok Lee, Yeong Seon Kang, and Junhee Seok. 2020. Portfolio management via two-stage deep learning with a joint cost. Expert Systems with Applications 143 (2020), 113041. doi:10.1016/j.eswa.2019.113041

[63] Yuanyuan Zhang, Xiang Li, and Sini Guo. 2018. Portfolio selection problems with Markowitz's mean–variance framework: a review of literature. Fuzzy Optimization and Decision Making 17 (2018), 125–158. doi:10.1007/s10700-017-9266-z

[64] Meng Zhao and Jinlong Li. 2018. Tuning the hyper-parameters of CMA-ES with tree-structured Parzen estimators. In International Conference on Advanced Computational Intelligence (ICACI). IEEE, 613–618. doi:10.1109/ICACI.2018.8377530

[65] Hanhong Zhu, Yi Wang, Kesheng Wang, and Yun Chen. 2011. Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem. Expert Systems with Applications 38, 8 (2011), 10161–10169.