Risk-averse Heteroscedastic Bayesian Optimization

Anonymous Author(s)
Affiliation
Address
email

Abstract

Many black-box optimization tasks arising in high-stakes applications require risk-averse decisions. The standard Bayesian optimization (BO) paradigm, however, optimizes the expected value only. We generalize BO to *trade mean and input-dependent variance* of the objective, both of which we assume to be unknown a priori. In particular, we propose a novel risk-averse heteroscedastic Bayesian optimization algorithm (RAHBO) that aims to identify a solution with high return and low noise variance, while learning the noise distribution on the fly. To this end, we model both expectation and variance as (unknown) RKHS functions, and propose a novel risk-aware acquisition function. We bound the regret for our approach and provide a robust rule to report the final decision point for applications where only a single solution must be identified. We demonstrate the effectiveness of RAHBO on synthetic benchmark functions and hyperparameter tuning tasks.

1 Introduction

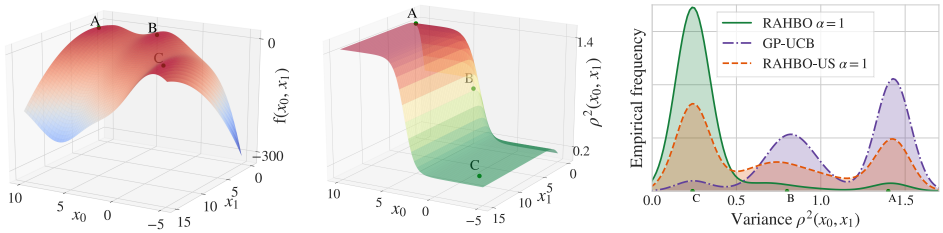
Black-box optimization tasks arise frequently in high-stakes applications such as drug and material discovery [21, 16, 28], genetics [15, 27], robotics [3, 10, 24], hyperparameter tuning of complex learning systems [20, 11, 34], to name a few. In many of these applications, there is often a trade-off between achieving high utility and minimizing risk. Moreover, uncertain and costly evaluations are an inherent part of black-box optimization tasks, and modern learning methods need to handle these aspects when balancing between the previous two objectives.

Bayesian optimization (BO) is a powerful framework for optimizing such costly black-box functions from noisy zeroth-order evaluations. Classical BO approaches are typically *risk-neutral* as they seek to optimize the expected function value only. In practice, however, two different solutions might attain similar expected function values, but one might produce significantly noisier realizations. This is of major importance when it comes to actual deployment of the found solutions. For example, when selecting hyperparameters of a machine learning algorithm, we might prefer configurations that lead to slightly higher test errors but at the same time lead to smaller variance.

In this paper, we generalize BO to trade off mean and input-dependent noise variance when sequentially querying points and outputting final solutions. We introduce a practical setting where *both* the black-box objective and input-dependent noise variance are *unknown* a priori, and the learner needs to estimate them on the fly. We propose a novel optimistic risk-averse algorithm – RAHBO – that makes sequential decisions by simultaneously balancing between *exploration* (learning about uncertain actions), *exploitation* (choosing actions that lead to high gains) and *risk* (avoiding unreliable actions). We bound the cumulative regret of RAHBO as well as the number of samples required to output a single near-optimal risk-averse solution. In our experiments, we demonstrate the risk-averse performance of our algorithm and show that standard BO methods can severely fail in applications where reliability of the reported solutions is of utmost importance.

Related work. Bayesian optimization (BO) [26] refers to approaches for optimizing a noisy black-box objective that is often expensive to evaluate. A great number of BO methods have been developed

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.



(a) Objective f (b) Variance ρ^2 (c) Histogram of variance

Figure 1: When there is a choice between identical optima with different noise level, risk-neutral BO tends to choose noisier inputs and RAHBO fixes this issue. (a) Unknown objective with 3 global maxima marked as (A, B, C). (b) Heteroscedastic noise variance over the same domain: the noise level at (A, B, C) varies according to the sigmoid function. (c) Empirical variance distribution at all points acquired during BO (over 9 experiments with different seeds). The standing out bumps correspond to the three global optima with different noise variance, i.e., smaller variance is better. RAHBO dominates in choosing the risk-averse optimum, consequently yielding lower risk-averse regret in Fig. 5a.

over the years, including a significant number of variants of popular algorithms such as GP-UCB [35], Expected Improvement [25], and Thompson Sampling [12]. While the focus of standard Bayesian optimization approaches is mainly on trading-off exploration vs. exploitation and optimizing for the expected performance, in this work, we additionally focus on the risk that is involved when working with noisy objectives, as illustrated in Figure 1.

The vast majority of previous BO works assume (sub-) Gaussian and *homoscedastic* noise (i.e., input independent and of some known fixed level). Both assumptions can be restrictive in practice. For example, as demonstrated in [14], the majority of hyperparameter tuning tasks exhibit heteroscedasticity. A few works relax the first assumption and consider, e.g., heavy-tailed noise models [13] and adversarially corrupted observations [6]. The second assumption is typically generalized via heteroscedastic GPs, allowing an explicit dependence of the noise distribution on the evaluation point [4, 8, 5, 19]. Similarly, in this work, we consider heteroscedastic GP models, but unlike the previous works, we specifically focus on the risk that is associated with querying and reporting noisy points.

Several works have recently considered robust and risk-averse aspects in Bayesian optimization. Their central focus is on designing robust strategies and protecting against the change/shift in uncontrollable covariates. They study various notions including worst-case robustness [7], distributional robustness [18, 29], robust mixed strategies [33] and other notions of risk-aversion [17, 9, 30], and while some of them report robust regret guarantees, their focus is primarily on the robustness in the homoscedastic GP setting. Instead, in our setting, we account for the risk that comes from the realization of random noise with *unknown* distribution. Rather than optimizing the expected performance, in our risk-averse setting, we prefer inputs with *lower variance*. To this end, we incorporate the learning of the noise distribution into the optimization procedure via a *mean-variance* objective. The closest to our setting is risk-aversion with respect to noise in multi-armed bandits [32]. Their approach, however, fails to exploit reward dependence among similar arms.

Contributions. We propose a novel *Risk-averse Heteroscedastic Bayesian optimization* (RAHBO) approach based on the optimistic principle that trades off the expectation and uncertainty of the mean-variance objective function. We model both expectation and variance as (unknown) RKHS functions, and propose a practical risk-aware algorithm in the heteroscedastic GP setting. In our theoretical analysis, we establish no-regret guarantees for our approach and provide a robust reporting rule for applications where only a single solution must be provided. We demonstrate the effectiveness of RAHBO on synthetic benchmarks, as well as on frequently considered hyperparameter tuning tasks.

2 Problem setting

Let $\mathcal{X}$ be a given compact set of inputs (typically $\mathcal{X} \subset \mathbb{R}^d$ for some $d \in \mathbb{N}$). We consider a problem of sequentially interacting with a fixed and unknown objective $f : \mathcal{X} \to \mathbb{R}$. At every round of this

procedure, the learner selects an action $x_t \in \mathcal{X}$, and obtains a noisy observation

$$y_t = f(x_t) + \xi(x_t), \tag{1}$$

where $\xi(x_t)$ is zero-mean noise independent across different time steps t. In this work, we consider sub-Gaussian heteroscedastic noise that depends on the query location.

Definition 1. *A zero-mean real-valued random variable ξ is ρ–sub-Gaussian, if there exists variance-proxy ρ^2 such that* $\forall \lambda \in \mathbb{R},\ \ \mathbb{E}[e^{\lambda \xi}] \leq e^{\frac{\lambda^2 \rho^2}{2}}$.

The variance $\mathbb{V}ar[\xi]$ lower bounds any valid variance-proxy, i.e., $\mathbb{V}ar[\xi] \leq \rho^2$. In case $\mathbb{V}ar[\xi] = \rho^2$ holds, ξ is said to be *ρ–strictly sub-Gaussian*. Besides zero mean Gaussian random variables, most standard symmetric bounded random variables (e.g., Bernoulli, beta, uniform, binomial) are strictly sub-Gaussian (see [1, Proposition 1.1]). Throughout the paper, we consider sub-Gaussian noise, and in Section 3.3, we specialize to the case of strictly sub-Gaussian noise.

Unlike the previous works that mostly focus on sequential optimization of f in the homoscedastic noise case, in this work we consider the trade-off between risk and return in the heteroscedastic case. While there exist a number of risk-averse objectives, we consider the simple and frequently used mean-variance model (MV) [32]. Here, the objective value at $x \in \mathcal{X}$ is a trade-off between the mean return $f(x)$ and the risk expressed by its variance-proxy $\rho^2(x)$:

$$\text{MV}(x) = f(x) - \alpha \rho^2(x), \tag{2}$$

where $\alpha \geq 0$ is a so-called *coefficient of absolute risk tolerance*. In this work, we assume α is fixed and known to the learner. Note that in the case of $\alpha = 0$, the objective coincides with the standard BO objective.

Performance metrics. We aim to construct a sequence of input evaluations x_t that eventually maximizes the risk-averse objective $\text{MV}(x_t)$. To assess this convergence, we consider two metrics. The first one corresponds to the notion of cumulative regret similar to the one used in standard Bayesian optimization and bandits. Here, the learner's goal is to maximize its risk-averse cumulative reward over a time horizon T, or equivalently minimize its *risk-averse cumulative regret*:

$$R_T = \sum_{t=1}^{T} \Big[\text{MV}(x^*) - \text{MV}(x_t) \Big], \tag{3}$$

where $x^* \in \arg\max_{x \in \mathcal{X}} \text{MV}(x)$. A sublinear growth of R_T with T implies vanishing average regret $R_T/T \to 0$ as $T \to \infty$. Intuitively, this implies the existence of some t such that $\text{MV}(x_t)$ is arbitrarily close to the optimal value $\text{MV}(x^*)$.

Besides the risk-averse cumulative regret, for a given accuracy $\epsilon \geq 0$, we also seek to report a single "good" risk-averse point $\hat{x}_T \in \mathcal{X}$ after a total of T rounds, that satisfies:

$$\text{MV}(\hat{x}_T) \geq \text{MV}(x^*) - \epsilon. \tag{4}$$

Here, the learner seeks to simultaneously minimize the number of expensive function evaluations T.

The latter metric Eq. (4) is especially useful, for example, in tuning machine learning hyperparameters, where one seeks to determine high-quality inputs by the end of the optimization procedure and then deploy them in production. Alternatively, the cumulative regret metric is important in calibration problems taking place during the system usage. We consider both performance metrics in our experiments (see Section 4).

Regularity assumptions. We consider standard smoothness assumptions [35, 7] when it comes to the unknown function $f : \mathcal{X} \to \mathbb{R}$. In particular, we assume that $f(\cdot)$ belongs to a reproducing kernel Hilbert space (RKHS) $\mathcal{H}_\kappa$ (a space of smooth and real-valued functions defined on $\mathcal{X}$), i.e., $f \in \mathcal{H}_\kappa$, induced by a kernel function $\kappa(\cdot, \cdot)$. Moreover, we assume that the RKHS norm of $f(\cdot)$ is bounded $\|f\|_\kappa \leq \mathcal{B}_f$ for some finite constant $\mathcal{B}_f > 0$. We assume that the noise $\xi(x)$ is $\rho(x)$-sub-Gaussian with variance proxy $\rho^2(x)$. Moreover, we assume that the variance-proxy is uniformly upper-bounded by some fixed constant $\varrho > 0$, that is, for all $x \in \mathcal{X}$, we have $\rho^2(x) \leq \varrho^2$. We also use ϱ_A^2 to denote the largest value of $\rho^2(x)$ among the points in set A, i.e., $\varrho_A^2 = \max_{x \in A} \rho^2(x)$.

3 Algorithms

We first recall the GP-based framework for sequential learning of RKHS functions from observations with heteroscedastic noise. Then, in Section 3.2, we consider a simple risk-averse Bayesian optimization problem with *known* variance-proxy, and later on in Section 3.3, we focus on our main problem setting in which the variance-proxy is *unknown*.

3.1 Bayesian optimization with heteroscedastic noise

Before addressing the risk-averse objective, we briefly recall the standard GP-UCB algorithm [35] in the setting of heteroscedastic sub-Gaussian noise. The regularity assumptions permits the construction of confidence bounds via Gaussian process (GP) model. Particularly, to decide which point to query at every round, GP-UCB makes use of the posterior GP mean and variance denoted by $\mu_t(\cdot)$ and $\sigma_t^2(\cdot)$, respectively. They are computed based on the previous measurements $y_{1:t} = [y(x_1), \dots, y(x_t)]^\top$ and the given kernel $\kappa(\cdot,\cdot)$:

$$\mu_t(x) = K_t(x)^T (K_t + \lambda \Sigma_t)^{-1} y_{1:t}, \tag{5}$$

$$\sigma_t^2(x) = \frac{1}{\lambda}(\kappa(x,x) - \kappa_t(x)^\top (K_t + \lambda\Sigma_t)^{-1}\kappa_t(x)), \tag{6}$$

where $\Sigma_t := \mathrm{diag}(\rho^2(x_1), \dots, \rho^2(x_t))$, $(K_t)_{i,j} = \kappa(x_i, x_j)$, $\kappa_t(x)^T = [\kappa(x_1,x), \dots, \kappa(x_t,x)]^T$ and $\lambda > 0$ is a regularization parameter. At time t, GP-UCB chooses the point maximizing the upper confidence bound of the unknown function $f(\cdot)$, i.e.,

$$x_t \in \arg\max_{x\in\mathcal{X}} \underbrace{\mu_{t-1}(x) + \beta_t \sigma_{t-1}(x)}_{=:\mathrm{ucb}_t^f(x)}. \tag{7}$$

If the noise $\xi_t(x_t)$ is heteroscedastic (i.e., input-dependent) and $\rho(x_t)$-sub-Gaussian, the following confidence bounds hold:

Lemma 1 (Lemma 7 in [19]). *Let $f \in \mathcal{H}_\kappa$, and $\mu_t(\cdot)$ and $\sigma_t^2(\cdot)$ be defined as in Eqs. (5) and (6) with $\lambda > 0$. Assume that the process $(x_t, y_t)_{t\geq 1}$ satisfies the noise model from Eq. (1). Then the following holds for all $t \geq 1$ and $x \in \mathcal{X}$ with probability at least $1-\delta$:*

$$|\mu_{t-1}(x) - f(x)| \leq \underbrace{\left(\sqrt{2\log\left(\frac{\det(\lambda\Sigma_t + K_t)^{1/2}}{\delta\det(\lambda\Sigma_t)^{1/2}}\right)} + \sqrt{\lambda}\|f\|_\kappa\right)}_{:=\beta_t} \sigma_{t-1}(x) \tag{8}$$

Here, β_t stands for exploration parameter that balances between exploration vs. exploitation and ensures the validity of confidence bounds.

Failure of GP-UCB in the risk-averse setting. GP-UCB is guaranteed to achieve sublinear cumulative regret with high probability in the risk-neutral (homoscedastic/heteroscedastic) Bayesian optimization setting [35, 12]. However, in the risk-averse setting (Eq. (2)), the maximizers $x^* \in \arg\max_{x\in\mathcal{X}} \mathrm{MV}(x)$ and $x_f^* \in \arg\max_{x\in\mathcal{X}} f(x)$ might not coincide, and consequently, $\mathrm{MV}(x^*)$ can be significantly larger than $\mathrm{MV}(x_f^*)$. This is illustrated in Fig. 1, where GP-UCB most frequently chooses optimum A of the highest risk.

3.2 Warm up: Known variance-proxy

To remedy the previous issue with GP-UCB, we propose a natural *Risk-Averse Heteroscedastic BO* (RAHBO) approach in case the proxy variance is known. At every round t, our algorithm chooses the action:

$$x_t \in \arg\max_{x\in\mathcal{X}} \mu_{t-1}(x) + \beta_t\sigma_{t-1}(x) - \alpha\rho^2(x), \tag{9}$$

where β_t is from Lemma 1 and α is from Eq. (2). Here, we assume that the learner knows the variance-proxy $\rho^2(\cdot)$. In the next section, we relax this assumption and consider a more practical setting when $\rho^2(\cdot)$ is unknown to the learner.

The performance of RAHBO is formally captured in the following proposition.

Proposition 1. *Consider any function $f(\cdot) \in \mathcal{H}_\kappa$, and let $A = \{x_t\}_{t=1}^T$ denote the set of actions chosen by* RAHBO *when run on such $f(\cdot)$ with $\{\beta_t\}_{t=1}^T$ as defined in Lemma 1, and known variance-proxy $\rho^2(\cdot)$. Then, with probability at least $1-\delta$,* RAHBO *attains cumulative regret $R_T = \mathcal{O}\big(\varrho_A \beta_T \sqrt{T\lambda\gamma_T}\big)$.*

The proof of this proposition is provided in Appendix A.1. Here, γ_T denotes the *maximum information gain* [35] at time T, that measures the informativeness of the sampled points $A_T = \{x_1, \dots, x_T\} \subset D$ about f defined by

$$\gamma_T := \max_{A \subset \mathcal{D},\, |A|=T} I(y_{1:T}, f_{1:T}), \tag{10}$$

where $I(y_{1:T}, f_{1:T})$ is the *mutual information* between evaluations $y_{1:T}$ and unknown $f_{1:T} = [f(x_1), \dots, f(x_T)]^\top$ at points $A \subset D$. In the case of heteroscedastic noise, we have $I(y_{1:T}, f_{1:T}) = \frac{1}{2}\sum_{t=1}^T \ln\big(1 + \frac{\sigma_t(x_t)^2}{\lambda\rho(x_t)^2}\big)$ (see Appendix A.1 for details). We note that analytical upper bounds on γ_T are provided in [35] for most of the popularly used kernels. These upper bounds typically scale sublinearly in T; for linear kernel $\gamma_T = \mathcal{O}(d \log T)$, and in case of squared exponential kernel $\gamma_T = \mathcal{O}(d(\log T)^{d+1})$. While these bounds are derived assuming the homoscedastic GP setting with some fixed constant noise variance σ^2, we note that that the same upper bounds are still applicable in the considered heteroscedastic case (i.e., by setting σ^2 to ϱ^2).

3.3 RAHBO for unknown variance-proxy

In the case of unknown variance-proxy $\rho^2(x)$, the confidence bounds for the unknown $f(x)$ in Lemma 1 can not be readily used, and we construct new ones on the combined mean-variance objective. To learn about the unknown $\rho^2(x)$, we make some further assumptions.

Assumption 1. *The variance-proxy belongs to a reproducing kernel Hilbert space induced by some kernel κ^{var}, i.e., $\rho^2 \in \mathcal{H}_{\kappa^{var}}$, and $\|\rho^2\|_{\kappa^{var}} \le B_{var}$. For every $x \in \mathcal{X}$, the observational noise $\xi(x)$ from Eq. (1) is* strictly *$\rho(x)$-sub-Gaussian, i.e., $\mathbb{V}ar[\xi(x)] = \rho^2(x)$.*

Since $\mathbb{V}ar[\xi(\cdot)]$ and $\rho^2(\cdot)$ coincide, we can focus on estimating the variance. In particular, we estimate the variance $\mathbb{V}ar[\xi(x)]$ in a *repeated experiment setting*, where for each x_t we collect $k > 1$ evaluations $\{y_i(x_t)\}_{i=1}^k$. Then the sample variance and sample mean of $\xi(x_t)$ are given as:

$$\hat{m}_k(x_t) = \frac{1}{k}\sum_{i=1}^k y_i(x_t) \quad \text{and} \quad \hat{s}_k^2(x_t) = \frac{1}{k-1}\sum_{i=1}^k (y_i(x_t) - \hat{m}_k(x_t))^2. \tag{11}$$

The key idea is that for strictly sub-Gaussian noise, $\hat{s}_{1:t}^2 = [\hat{s}_k^2(x_1), \dots, \hat{s}_k^2(x_t)]^\top$ yields *unbiased, but noisy* evaluations of the unknown variance-proxy $\rho_{1:t}^2 = [\rho^2(x_1), \dots, \rho^2(x_t)]^\top$, i.e., for $1 \le i \le t$, we have

$$\hat{s}_k^2(x_i) = \rho^2(x_i) + \eta(x_i), \tag{12}$$

with noise $\eta(x_i)$. In order to efficiently estimate the variance, we need an additional assumption.

Assumption 2. *The noise $\eta(x_t)$ in Eq. (12) is $\rho_\eta(x_t)$-sub-Gaussian with known $\rho_\eta^2(x_t)$ and independence between times t.*

A similar assumption is made in [32] in the multi-armed bandit setting. The fact that $\rho_\eta^2(\cdot)$ is known is rather mild as Assumption 1 allows to control its value. For example, for strictly sub-Gaussian $\eta(x)$ we show (in Appendix A.2) that $\mathbb{V}ar(\eta(x)) = \rho_\eta^2(x) \le \frac{2\rho^4(x)}{k-1}$. Then, given that $\rho^2(x) \le \varrho^2$, we can utilize the following (rather conservative) bound as a variance-proxy, i.e., $\rho_\eta^2(x) = \frac{2\varrho^4}{k-1}$. As we refine our variance estimates, we can exploit them to learn faster about the mean.

RAHBO algorithm. We present our risk-averse heteroscedastic Bayesian Optimization approach for unknown variance-proxy in Algorithm 1. Our method relies on building two GP models.

Firstly, we use sample variance evaluations $\hat{s}_{1:t}^2$ to construct a GP model for $\rho^2(\cdot)$ where the mean $\mu_{t-1}^{var}(\cdot)$ and variance $\sigma_{t-1}^{var}(\cdot)$ are computed in the same way as in Eqs. (5) and (6) (by using κ^{var}, $\rho_\eta^2(\cdot)$ as the variance-proxy, and treating $\hat{s}_{1:t}^2$ as noisy observations). Accordingly, we build the

Algorithm 1 Risk-averse Heteroscedastic Bayesian Optimization (RAHBO)

Require: Parameters $\alpha, \{\beta_t\}_{t\geq 1}, \lambda$, kernel functions κ, κ^{var}

for $t = 1, 2, \ldots$ **do**
  Select $x_t \in \arg\max_{x\in\mathcal{X}} \mathrm{ucb}^f_{t-1}(x) - \alpha\, \mathrm{lcb}^{var}_{t-1}(x)$
  Observe k samples $\{y_i(x_t)\}_{i=1}^k$: $y_i(x_t) = f(x_t) + \xi_i(x_t)$ for every $i \in [k]$
  Use $\{y_i(x_t)\}_{i=1}^k$ to compute $\hat{m}_k(x_t)$ and $\hat{s}^2_k(x_t)$ as in Eq. (11)
  Use $\{x_t, \hat{s}^2_k(x_t)\}$ to update $\mu^{var}_t(\cdot)$ and $\sigma^{var}_t(\cdot)$ according to Eq. (5) and Eq. (6)
  Construct $\mathrm{ucb}^{var}_t(\cdot)$ and $\mathrm{lcb}^{var}_t(\cdot)$ as in Eq. (13) and Eq. (14)
  Use $\mathrm{ucb}^{var}_t(\cdot)$ to compute $\hat{\Sigma}_t$ as in Eq. (15)
  Use $\{x_t, \hat{m}_k(x_t)\}$ and $\hat{\Sigma}_t$ to update $\mu_t(x)$ and $\sigma_t(x)$ according to Eqs. (5) and (6)
  Construct $\mathrm{ucb}^f_t(\cdot)$ as in Eq. (7)
end for

upper and lower confidence bounds $\mathrm{ucb}^{var}_t(\cdot)$ and $\mathrm{lcb}^{var}_t(\cdot)$ on the variance-proxy as follows (see Appendix A.4 for more details):

$$\mathrm{ucb}^{var}_t(x) := \mu^{var}_{t-1}(x) + \beta^{var}_t \sigma^{var}_{t-1}(x) \tag{13}$$

$$\mathrm{lcb}^{var}_t(x) := \mu^{var}_{t-1}(x) - \beta^{var}_t \sigma^{var}_{t-1}(x), \tag{14}$$

where once again β^{var}_t is set according to Lemma 1.

Secondly, we use sample mean evaluations $\hat{m}_{1:t} = [\hat{m}_k(x_1), \ldots, \hat{m}_k(x_t)]^\top$ as our observed data to construct a model for f. To compute $\mu_t(\cdot)$ and $\sigma_t(\cdot)$ (via Eq. (6) and Eq. (44)), instead of the unknown variance-proxy $\rho(\cdot)$ and Σ_t, we use its truncated (with ϱ^2) upper confidence bound and set

$$\hat{\Sigma}_t := \tfrac{1}{k}\mathrm{diag}\big(\min\{\mathrm{ucb}^{var}_t(x_1), \varrho^2\}, \ldots, \min\{\mathrm{ucb}^{var}_t(x_1), \varrho^2\}\big). \tag{15}$$

Note that $\mu_t(\cdot)$ and $\sigma_t(\cdot)$ are then computed by using $\hat{\Sigma}_t$ that is corrected by k (since every point is sampled k times). Hence, the previous substitution of the unknown variance-proxy by its conservative estimate in Eq. (15) guarantees (conditioning on the confidence bounds for $\rho(\cdot)$ holding true) that the confidence bounds $\mathrm{ucb}^f_t(x) := \mu_{t-1}(x) + \beta_t \sigma_{t-1}(x)$ on f also hold (with high probability).

Finally, we define our acquisition function as maximization of $\mathrm{ucb}^{\mathrm{MV}}_t(x) := \mathrm{ucb}^f_t(x) - \alpha \mathrm{lcb}^{var}_t(x)$, i.e., select $x_t \in \arg\max_{x\in\mathcal{X}} \mathrm{ucb}^{\mathrm{MV}}_t(x)$ at every round t. The performance of RAHBO is formally captured in the following theorem.

Theorem 2. *Let the set of actions $A_T = \{x_t\}_{t=1}^T$ be chosen with* RAHBO *with unknown variance-proxy (Algorithm 1) with each action taken $k > 1$ times. Let $\mathcal{R}^2 = \max_{x\in A_T} \rho^2_\eta(x_t)$. Then, with probability at least $1-\delta$, the risk-averse regret R_T of* RAHBO *is bounded by:*

$$\Pr\left\{ R_T \leq \sqrt{4kT\lambda(\beta^2_T \varrho^2 \hat{\gamma}_T + k(\beta^{var}_T)^2 \mathcal{R}^2 \Gamma_T)} \quad \forall T \geq 1 \right\} \geq 1-\delta. \tag{16}$$

Here, $\hat{\gamma}_T$ denotes maximum information gain $\hat{\gamma}_T = \max_{A_T} I((\hat{m}_k)_{1:T}, f_{1:T})$ assuming $(\hat{m}_k)_{1:T}$ are coming from the distribution with constant variance-proxy ϱ^2/k. Also, Γ_T denotes maximum information gain $\Gamma_T = \max_{A_T} I((\hat{s}^2_k)_{1:T}, \rho^2_{1:T})$ assuming that $(\hat{s}^2_k)_{1:T}$ are coming from the distribution with variance-proxy ρ^2_η. For the proof see Appendix A.5.

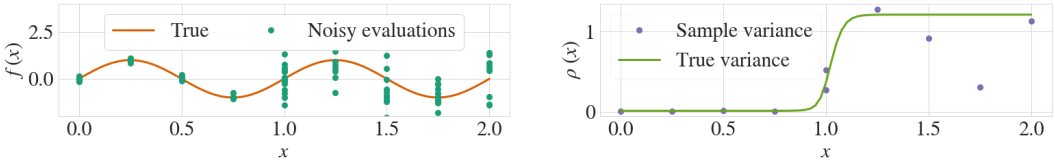
Furthermore, we obtain a bound on the number of iterations required for identifying an ϵ-optimal point defined in Eq. (4):

Corollary 2.1. *Let Algorithm 1 (*RAHBO*) after T iterations output the point $\hat{x}_T$ such that: $\hat{x}_T := x_{t^*}$, where $t^* := \arg\max_t \{\mathrm{lcb}^{MV}_t(x_t)\}$. Then $\hat{x}_T$ achieves ϵ-accuracy, i.e., MV$(x^*) -$ MV$(\hat{x}_T) \leq \epsilon$, after $T \geq \frac{\lambda(\beta^2_T \varrho^2 \hat{\gamma}_T + k(\beta^{var}_T)^2 \mathcal{R}^2 \Gamma_T)}{\epsilon^2}$ samples.*

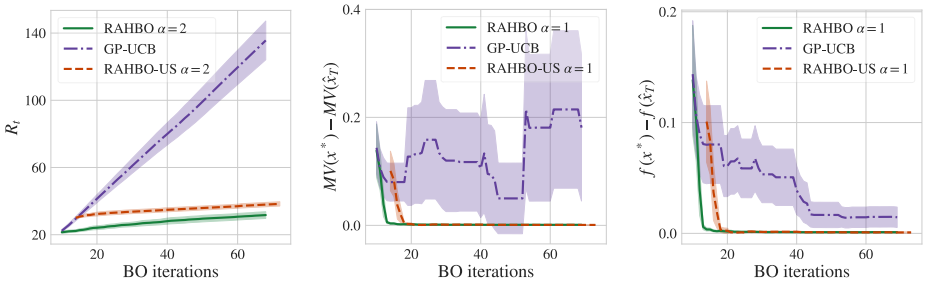
4 Experiments

In this section, we experimentally validate RAHBO and the baselines on two synthetic examples and two real hyperparameter tuning tasks. We provide an open-source implementation of our method. [1]

[1] https://anonymous.4open.science/r/Risk-averse-BO-1C3F



(a) Illustration of the sine function (left) and noise variance (right)



(b) Cumulative regret (c) Suboptimality w.r.t. MV (d) Suboptimality w.r.t. f

Figure 2: (a) Unknown true objective along with noisy evaluations with varying noise level (left) and unknown true noise variance and its evaluations (right). (b) Cumulative regret. (c) Simple MV regret for reporting rule $\hat{x}_T = \arg\max_{x_t} \mathrm{lcb}_T(x_t)$. (c) Simple regret $f(x^*) - f(\hat{x}_T)$ for the unknown function at the reported point $\hat{x}_T$ from (d). RAHBO not only leads to strong results in terms of MV but also in terms of the mean objective $f(x)$.

Baselines. We compare against two baselines: First, we use GP-UCB with heteroscedastic noise as a standard risk-neutral algorithm that optimizes the unknown $f(x)$. Second, we consider a risk-averse baseline that uniformly learns variance-proxy $\rho^2(x)$ *before* the optimization procedure. This prior learning is in contrast to RAHBO that learns it on the fly. We call it RAHBO-US, standing for RAHBO with uncertainty sampling. It consists of two stages: (i) uniformly learning of $\rho^2(x)$ via uncertainty sampling followed by (ii) GP-UCB for the risk-averse objective, where instead of the unknown $\rho^2(x)$, we use the mean of the learned model. Note that RAHBO-US is closest to the contextual BO setting in [17], where the context distribution is assumed to be known.

Experimental setup. At each iteration t, an algorithm queries a point x_t and observes sample mean and sample variance of k observations $\{y_i(x_t)\}_{i=1}^k$. We use a heteroscedastic GP for modelling $f(x)$ and a homoscedastic GP for $\rho^2(x)$. We set $\beta_t = 2$, which is commonly used in practice to improve performance over the theoretical results. Before the BO procedure, we determine the GP hyperparameters maximizing the marginal likelihood. To this end, we use initial points that are the same for all baselines and are chosen via Sobol sequence that generates low discrepancy quasi-random samples. We repeat each experiment several times, generating new initial points for every repetition. We use two metrics described in Section 2: (a) risk-averse cumulative regret R_t computed for the acquired inputs; (b) simple regret $\mathrm{MV}(x^*) - \mathrm{MV}(\hat{x}_T)$ computed for inputs as reported via Corollary 2.1. For each metric, we report its mean $\pm$ two standard errors over the repetitions.

Example function. We first illustrate the methods on a sine function depicted in Fig. 2a. It has two global optimizers, but we control the evaluation noise such that for each input, there is a choice between a low and a high noise version. We use a sigmoid function as noise variance, as depicted in Fig. 2a, that induces little noise on $[0, 1]$ and increased noise on $(1, 2]$. We initialize the algorithms by selecting 10 inputs x at random, and keep these points the same for all algorithms. We use $k = 10$ samples at each chosen x_t. The number of acquisition rounds is $T = 60$. We repeat the experiment 30 times and show the average performance in Fig. 2.

Branin benchmark Next, we evaluate the methods on the (negated) Branin benchmark function in Fig. 1a, achieving its optimum value $f^* = -0.4$ at $(-\pi, 12.3), (\pi, 2.3), (9.4, 2.5)$. The heteroscedastic variance function illustrated in Fig. 1b defines different noise for the three optima. We initialize all algorithms by selecting 10 inputs. We use $k = 10$ samples to estimate the noise variance. The number of acquisition rounds is $T = 150$. We repeat BO 25 times and show the results in Figs. 1c and 5a. Fig. 1c provides more intuition behind the observed regret: UCB exploits the noisiest maxima the most, while RAHBO prefers smaller variance.

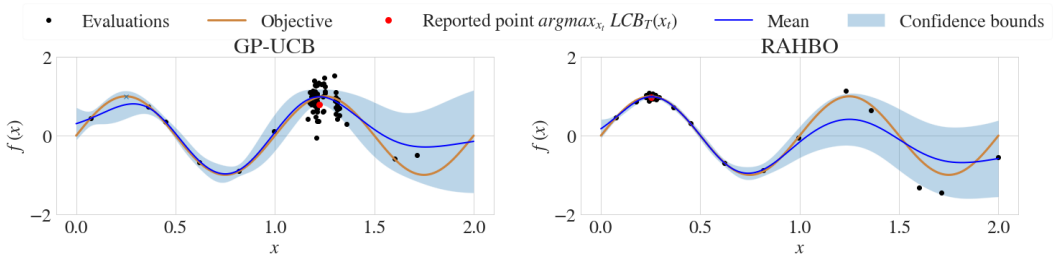


Figure 3: GP models fitted for GP-UCB (left) and RAHBO (right) for sine function. After initialization with the same sampled points, GP-UCB concentrates on the high-noise region whereas RAHBO prefers small variance. Additional plots are presented in Appendix A.7.

Tuning a Free Electron Laser We tune parameters of a Free Electron Laser (FEL), an important scientific instrument[2]. Here, the main objective is to maximize the pulse energy measured by a gas detector. Parameter tuning is a time-consuming and repetitive task during operation. Tuning the FEL signal is an application where the cumulative regret is the relevant performance metric, since (re-)calibration takes place while user experiments are running. This is in contrast to deploying optimal hyperparameters for machine learning systems in production.

We use real FEL measurements to train a neural network surrogate model, and use it to simulate the FEL objective $f(x)$ for new settings x. We similarly fit a model of the heteroscedastic variance by regressing the squared residuals via a GP model. Here, we focus on the four most sensitive parameters.

We report our comparison in Fig. 4 where we also assess the effect of varying the coefficient of absolute risk tolerance α. We use 30 points to initialize the baselines and then perform 200 acquisition rounds. We repeat each experiment 15 times. In Fig. 4a we plot the empirical frequency of the true (unknown to the methods) values $f(x_t)$ and $\rho^2(x_t)$ at the inputs x_t acquired by the methods. The empirical frequency for $\rho^2(x)$ illustrates the tendency of risk-neutral GP-UCB to query points with higher noise, while risk-averse achieves substantially reduced variance and minimal reduction in mean performance. Sometimes, risk-neutral GP-UCB also fails to succeed in querying points with the highest f-value. That tendency results in lower cumulative regret for RAHBO in Figs. 4c and 4d. We also compare the performance of the reporting rule from Corollary 2.1 in Fig. 4b, where we plot error bars with standard deviation both for $f(\hat{x}_T)$ and $\rho^2(\hat{x}_T)$ at the reported point $\hat{x}_T$. As before, RAHBO drastically reduces the variance compared to GP-UCB, while having only slightly lower mean performance. Additional results are presented in Appendix Fig. 7c.

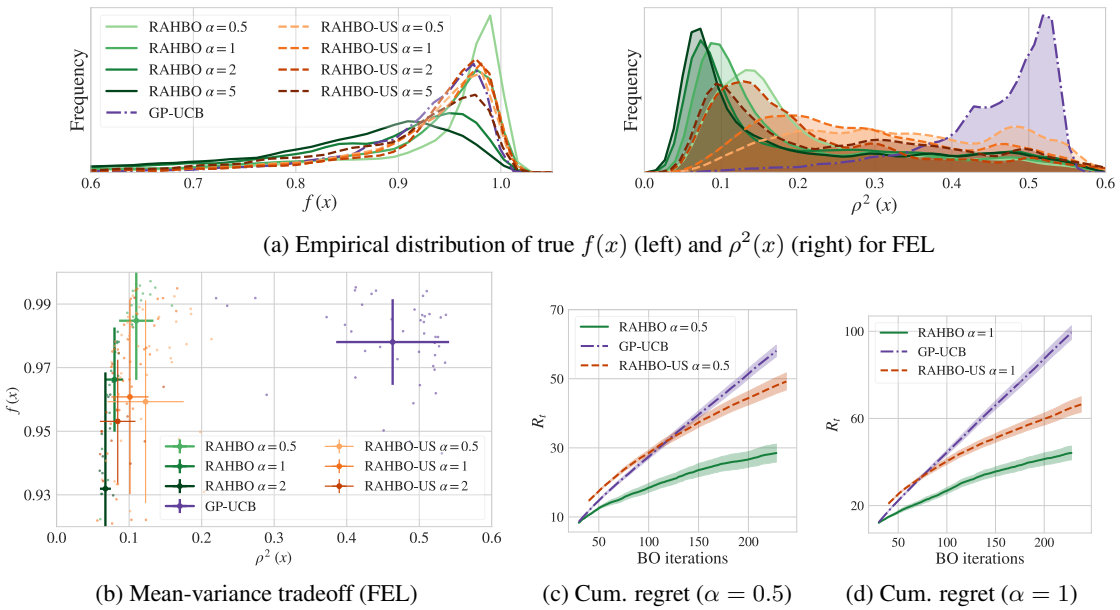
Random Forest tuning BO is widely used by cloud services for tuning machine learning hyperparameters and the resulting models might be then used in high-stakes applications such as credit scoring or fraud detection. In k-fold cross-validation, the average metric over the validation sets is optimized – a canonical example of the *repeated experiment setting* that we consider in the paper. High across-folds variance is a practical problem [23] where the mean-variance approach might be beneficial.

In our experiment, we tune hyperparameters of a random forest classifier (RF) on a dataset of fraudulent credit card transactions [22].[3] It consist of 285k transactions with 29 features (processed due to confidentiality issues) that are distributed over time, and only 0.2% are fraud examples (see Appendix for more details). The search space for the RF hyperparameters is also provided in the Appendix. We use the balanced accuracy (BA) score and 5 validation folds, i.e., $k = 5$, and each validation fold is shifted in time with respect to the training data. We seek not only for high performance *on average* but also for low variance across the validation folds that have different time shifts with respect to the training data.

We initialize the algorithms by selecting 10 hyperparameter settings and keep these points the same for all algorithms. We use Matérn 5/2 kernels with Automatic Relevance Discovery (ARD) and normalize the input features to the unit cube. The number of acquisition rounds in one experiment is 50 and we repeat each experiment 15 times. We report results in Figs. 5b and 5c where we plot mean $\pm$ 2 standard errors. While both RAHBO and GP-UCB perform comparable in terms of the mean error, its standard deviation for RAHBO is smaller.

[2] Details withheld to maintain anonymity during review.

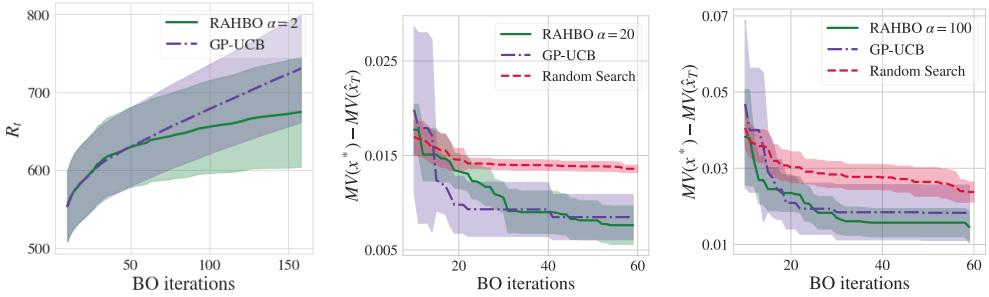
[3] https://www.kaggle.com/mlg-ulb/creditcardfraud



(a) Empirical distribution of true $f(x)$ (left) and $\rho^2(x)$ (right) for FEL

(b) Mean-variance tradeoff (FEL) (c) Cum. regret ($\alpha = 0.5$) (d) Cum. regret ($\alpha = 1$)

Figure 4: **(a)** Distributions of $f(x)$ and $\rho^2(x)$ for *all points* queried by BO for the FEL. Clearly, GP-UCB queries points with higher noise (but not necessarily high return f) in contrast to the risk-averse methods. Moreover, greater α leads to lower variance. A good trade-off is achieved at $\alpha = 0.5$. **(b)** Standard deviation error bars for $f(\hat{x}_T)$ and $\rho^2(\hat{x}_T)$ at the *reported* $\hat{x}_T = \arg\max_{x_t} \mathrm{lcb}_T(x_t)$ for FEL computed over the repeated BO. Though points reported by GP-UCB have slightly higher $f(x)$ than most of the risk-averse methods, its noise variance is much higher. Again, $\alpha = 0.5$ yields a good mean-variance trade-off. **(c-d)** Cum. regret for $\alpha = 0.5$ and $\alpha = 1$ (see more in Appendix.)



(a) Branin benchmark (b) RF Tuning (c) RF Tuning

Figure 5: **Branin**: (a) Cumulative regret. **Random Forest**: (b-c) Simple regret fat the reprted $\hat{x}_T = \arg\max_{x_t} MV(x_t)$ for (b) $\alpha = 20$ and (c) $\alpha = 100$. While both methods have comparable mean, RAHBO has consistently lower variance.

5 Conclusion

In this work, we generalized BO to trade the mean and input-dependent variance of the objective, both of which we assume to be unknown a priori. Our novel risk-averse upper confidence bound (RAHBO) algorithm enjoys sublinear cumulative regret and performs well on synthetic benchmarks and hyperparameter tuning tasks.

Limitations. In settings where the noise is homoscedastic, our approach does not bring additional value. Moreover, our proof of Theorem 2 relies on the Assumption 2 with the bounds for noise tails. Though this assumption is valid for important applications such as hyperparameter tuning where the noise is bounded, it fails to apply if $\eta(x)$ has heavy tails. Chowdhury [31] shows how GP-UCB can be adapted to this setting, and addressing the mean variance tradeoffs in this case is an interesting direction for future work.

References

[1] J. Arbel, O. Marchal, and H. Nguyen. On strict sub-gaussianity, optimal proxy variance and symmetry for bounded random variables. *arXiv: Probability*, 2019.

[2] E. Benhamou. A few properties of sample variance, 2018.

[3] F. Berkenkamp, A. Krause, and A. P. Schoellig. Bayesian optimization with safety constraints: Safe and automatic parameter tuning in robotics, 2020.

[4] M. Binois, R. Gramacy, and M. Ludkovski. Practical heteroscedastic gaussian process modeling for large simulation experiments. *Journal of Computational and Graphical Statistics*, 11 2016.

[5] M. Binois, J. Huang, R. B. Gramacy, and M. Ludkovski. Replication or exploration? sequential design for stochastic simulation experiments. *Technometrics*, 61:7–23, 2019.

[6] I. Bogunovic, A. Krause, and J. Scarlett. Corruption-tolerant gaussian process bandit optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 1071–1081. PMLR, 2020.

[7] I. Bogunovic, J. Scarlett, S. Jegelka, and V. Cevher. Adversarially Robust Optimization with Gaussian Processes. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[8] I. Bogunovic, J. Scarlett, A. Krause, and V. Cevher. Truncated variance reduction: A unified approach to bayesian optimization and level-set estimation. *arXiv preprint arXiv:1610.07379*, 2016.

[9] S. Cakmak, R. Astudillo, P. Frazier, and E. Zhou. Bayesian optimization of risk measures, 2020.

[10] R. Calandra, A. Seyfarth, J. Peters, and M. P. Deisenroth. Bayesian Optimization for Learning Gaits under Uncertainty. *Annals of Mathematics and Artificial Intelligence*, 76, 2016.

[11] Y. Chen, A. Huang, Z. Wang, I. Antonoglou, J. Schrittwieser, D. Silver, and N. de Freitas. Bayesian optimization in alphago, 2018.

[12] S. R. Chowdhury and A. Gopalan. On kernelized multi-armed bandits. In *International Conference on Machine Learning*, pages 844–853. PMLR, 2017.

[13] S. R. Chowdhury and A. Gopalan. Bayesian optimization under heavy-tailed payoffs. *arXiv preprint arXiv:1909.07040*, 2019.

[14] A. I. Cowen-Rivers, W. Lyu, R. Tutunov, Z. Wang, A. Grosnit, R. R. Griffiths, H. Jianye, J. Wang, and H. B. Ammar. An empirical study of assumptions in bayesian optimisation, 2021.

[15] J. González, J. Longworth, D. C. James, and N. D. Lawrence. Bayesian optimization for synthetic gene design, 2015.

[16] H.-L. J. M. Griffiths, Ryan-Rhys. Constrained Bayesian optimization for automatic chemical design using variational autoencoders. *Chem. Sci.*, 11(2):577–586, 2020.

[17] S. Iwazaki, Y. Inatsu, and I. Takeuchi. Mean-variance analysis in bayesian optimization under uncertainty, 2020.

[18] J. Kirschner, I. Bogunovic, S. Jegelka, and A. Krause. Distributionally robust bayesian optimization, 2020.

[19] J. Kirschner and A. Krause. Information directed sampling and bandits with heteroscedastic noise, 2018.

[20] J. Kirschner, M. Mutny, N. Hiller, R. Ischebeck, and A. Krause. Adaptive and Safe {B}ayesian Optimization in High Dimensions via One-Dimensional Subspaces. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3429–3438. PMLR, 2019.

[21] K. Korovina, S. Xu, K. Kandasamy, W. Neiswanger, B. Poczos, J. Schneider, and E. P. Xing. Chembo: Bayesian optimization of small organic molecules with synthesizable recommendations, 2019.

[22] Y.-A. Le Borgne and G. Bontempi. *Machine Learning for Credit Card Fraud Detection - Practical Handbook*. 05 2021.

[23] A. Makarova, H. Shen, V. Perrone, A. Klein, J. B. Faddoul, A. Krause, M. Seeger, and C. Archambeau. Overfitting in bayesian optimization: an empirical study and early-stopping solution, 2021.

[24] A. Marco, P. Hennig, J. Bohg, S. Schaal, and S. Trimpe. Automatic lqr tuning based on gaussian process global optimization. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016.

[25] J. Mockus. On bayesian methods for seeking the extremum. In *Proceedings of the IFIP Technical Conference*, page 400–404, Berlin, Heidelberg, 1974. Springer-Verlag.

[26] J. Močkus. On Bayesian methods for seeking the extremum. In *Optimization Techniques*, 1975.

[27] H. Moss, D. Leslie, D. Beck, J. González, and P. Rayson. BOSS: Bayesian Optimization over String Spaces. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 15476–15486. Curran Associates, Inc., 2020.

[28] D. M. Negoescu, P. I. Frazier, and W. B. Powell. The knowledge-gradient algorithm for sequencing experiments in drug discovery. *INFORMS J. on Computing*, 23(3):346–363, July 2011.

[29] T. T. Nguyen, S. Gupta, H. Ha, S. Rana, and S. Venkatesh. Distributionally robust bayesian quadrature optimization, 2020.

[30] Z. D. Quoc Phong Nguyen, B. K. H. Low, and P. Jaillet. Value-at-risk optimization with gaussian processes, 2021.

[31] G. A. Ray Chowdhury, Sayak. Bayesian optimization under heavy-tailed payoffs, 09 2019.

[32] A. Sani, A. Lazaric, and R. Munos. Risk-aversion in multi-armed bandits. In *NIPS*, 2012.

[33] P. G. Sessa, I. Bogunovic, M. Kamgarpour, and A. Krause. Mixed strategies for robust optimization of unknown objectives, 2020.

[34] J. Snoek, H. Larochelle, and R. P. Adams. Practical bayesian optimization of machine learning algorithms, 2012.

[35] N. Srinivas, A. Krause, S. Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, page 1015–1022, Madison, WI, USA, 2010. Omnipress.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Section 1 and the abstract
   (b) Did you describe the limitations of your work? [Yes] See Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [No] We are not aware of any societal impact of our work
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Sections 2 and 3.3
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See the URL at the beginning of Section 4
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Section 4 plots
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4
   (b) Did you mention the license of the assets? [Yes] See the URL at the beginning of Section 4
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] They were obtained personally or the asset is in the open access
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] No, the data does not contain these
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]