Efficient Distributionally Robust Bayesian Optimization with Worst-case Sensitivity

Sebastian Shenghong Tay [1 2] Chuan Sheng Foo [2] Daisuke Urano [3] Richalynn Chiu Xian Leong [3]
Bryan Kian Hsiang Low [1]

Abstract

In *distributionally robust Bayesian optimization* (DRBO), an exact computation of the worst-case expected value requires solving an expensive convex optimization problem. We develop a fast approximation of the worst-case expected value based on the notion of worst-case sensitivity that caters to arbitrary convex distribution distances. We provide a regret bound for our novel DRBO algorithm with the fast approximation, and empirically show it is competitive with that using the exact worst-case expected value while incurring significantly less computation time. In order to guide the choice of distribution distance to be used with DRBO, we show that our approximation implicitly optimizes an objective close to an interpretable risk-sensitive value.

1. Introduction

Bayesian optimization (BO) is a powerful paradigm for efficiently optimizing an unknown/black-box objective function $f(x)$ (Garnett, 2022) w.r.t. *action*/decision variable x with a limited budget of costly function evaluations. As a result, there is a fast growing interest in the application of BO to many complex real-world optimization problems like hyperparameter optimization of machine learning models (Chen et al., 2018), molecule search in automated chemical design (Griffiths & Hernández-Lobato, 2020), to name a few.

In practice, the black-box objective function often depends on a random *context*/environment variable c beyond our control s.t. the goal now is to maximize the expected value $\mathbb{E}_p\left[f(x,c)\right]$ w.r.t. some distribution p of c (Toscano-Palmerin & Frazier, 2018). For example, consider the problem of optimizing the expected size of a crop where x represents the soil nutrient concentrations and can be controlled, while c represents the average temperature in a month and is uncontrolled. The introduction of the random context c gives rise to new BO problem settings that aim to be risk-averse and avoid worst-case scenarios by optimizing *risk measures* of $f(x,c)$ instead (Cakmak et al., 2020; Nguyen et al., 2021a;b). Such works assume that the distribution of c is stationary and is either known or can be estimated well.

Distributionally robust BO (DRBO) was concurrently introduced by Kirschner et al. (2020) and Nguyen et al. (2020) and adapts the framework of *distributionally robust optimization* (DRO) in operations research (Rahimian & Mehrotra, 2019) to BO. DRBO considers the setting where c is subject to *distribution shift*, i.e., its distribution may be different from our prior knowledge. To be distributionally robust is thus to select the action x that maximizes the *worst-case expected value*, i.e., $\mathbb{E}_q\left[f(x,c)\right]$ under the worst-case distribution q of c selected by an adversary from some set of possible distributions. Such an approach provides another form of robustness, specifically, to an incorrect knowledge of the distribution of c rather than to the undesirable outcomes of c. In the earlier crop example, this corresponds to being robust to an incorrect knowledge of the distribution of the average temperature in a month which may be constantly shifting due to climate change.

A complex real-world optimization problem usually has several sources of randomness, so context c is multi-dimensional. For instance, the average temperature in a month, fertilizer composition, and crop genotypes may all be subject to distribution shift. If the outcomes of the multi-dimensional c are discretized into a finite set $\mathcal{C}$ and a reasonable discretization density is used for each dimension, then $|\mathcal{C}|$ can grow large quickly. This work allows DRBO to be scaled to a large $|\mathcal{C}|$. In contrast, existing DRBO algorithms scale poorly in $|\mathcal{C}|$: The algorithm of Kirschner et al. (2020) has to solve a convex optimization problem with $|\mathcal{C}|$ variables to obtain the worst-case expected value, which incurs up to $\mathcal{O}(|\mathcal{C}|^3)$ time. Nguyen et al. (2020) have devised an efficient method to do so based on Lagrange multipliers, but it is tied to a specific choice of distribution distance used to

[1] Department of Computer Science, National University of Singapore, Singapore [2] Institute for Infocomm Research, A*STAR, Singapore [3] Temasek Life Sciences Laboratory, Singapore. Correspondence to: Sebastian Shenghong Tay <seb.tsh@gmail.com>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

Table 1. Comparing time complexity of DRBO algorithms utilizing the EXACT worst-case expected value (Kirschner et al., 2020) vs. our fast approximation called MINIMAXAPPROX with various distribution distances d. The EXACT worst-case expected value is obtained by solving a general convex optimization problem with $|\mathcal{C}|$ variables using interior point methods which, we assume, incur $\mathcal{O}(|\mathcal{C}|^3)$ time.

Distribution distance d	EXACT	MINIMAXAPPROX
Maximum mean discrepancy (MMD)	$\mathcal{O}(\|\mathcal{C}\|^3)$	$\mathcal{O}(\|\mathcal{C}\|^2)$
Total variation (TV)	$\mathcal{O}(\|\mathcal{C}\|^3)$	$\mathcal{O}(\|\mathcal{C}\|)$
Modified χ^2-divergence (χ^2)	$\mathcal{O}(\|\mathcal{C}\|^3)$	$\mathcal{O}(\|\mathcal{C}\|)$
Wasserstein metric ($\mathcal{W}$)	$\mathcal{O}(\|\mathcal{C}\|^6)$	$\mathcal{O}(\|\mathcal{C}\|^2)$

construct the set of possible alternative distributions. When using general distances, their algorithm also needs to solve the expensive convex optimization problem. In conventional DRO where f is fully known, this may be tolerable since the optimization only has to be performed once. However, in DRBO where f needs to be learned and the optimization has to be performed separately for each action and BO iteration, the computational burden can become excessive.

This paper presents an efficient approximation of the worst-case expected value (i.e., solution of the convex optimization problem) by leveraging the notion of *worst-case sensitivity* introduced in a recent development in the operations research literature on DRO (Gotoh et al., 2020). This enables us to develop a novel DRBO algorithm for efficiently solving real-world problems where the context c has more than a few dimensions. In addition, our algorithm caters to arbitrary convex distribution distances: while the previous works have each utilized a specific distribution distance, we consider four distances, namely, *maximum mean discrepancy* (MMD) (Gretton et al., 2012), *total variation* (TV), χ^2*-divergence* (χ^2), and the Wasserstein metric ($\mathcal{W}$) (Mohajerin Esfahani & Kuhn, 2018) and discuss the model selection problem of choosing an appropriate distance for a given application. Specifically, our proposed approximation can be exploited for guiding this choice based on interpretable *risk-sensitive values* (i.e., trade-off between expected value and a notion of risk). For example, using χ^2 with our DRBO algorithm implicitly optimizes an objective close to a mean-variance trade-off which can be useful for growing crops if similarly-sized crops (e.g., fruits) are preferred for logistical reasons. Concretely, the contributions of our work here include the following:

- We propose a fast approximation of the worst-case expected value (i.e., solution of the inner convex optimization problem in DRBO) by leveraging the worst-case sensitivity, which reduces the incurred time as shown in Table 1 (Sec. 4);
- We derive a regret bound for our novel DRBO algorithm utilizing the fast approximation and show that its asymptotic regret is of the same order as that using the exact worst-case expected value (Sec. 5);
- To guide the choice of distribution distance for a given application, we show that our algorithm implicitly optimizes an objective close to an interpretable risk-sensitive value (Sec. 6); and
- We provide empirical results to show that our algorithm utilizing the fast approximation scales significantly better in the context set size $|\mathcal{C}|$ and yet performs comparably to that using the exact worst-case expected value, while outperforming non-robust ones (Sec. 7).

2. Related Work

DRBO was concurrently proposed by Kirschner et al. (2020) and Nguyen et al. (2020). These works (and ours) focus on robustness to distribution shift. Other prior works on BO with random context have more commonly focused on robustness to poor outcomes of c and assume a stationary distribution of c. Beyond the expected value (Toscano-Palmerin & Frazier, 2018), these works have considered risk measures like *value-at-risk* (VaR) and conditional VaR (Cakmak et al., 2020; Nguyen et al., 2021a;b), and risk-sensitive values like mean-variance (Iwazaki et al., 2021). The work of Bogunovic et al. (2018) considers robustness to adversarial perturbations of inputs, which translates to seeking ‘wider’ local maxima.

3. Distributionally Robust Bayesian Optimization (DRBO)

DRBO involves optimizing a black-box objective function $f : \mathcal{X} \times \mathcal{C} \rightarrow \mathbb{R}$ over the set $\mathcal{X} \subset \mathbb{R}^m$ of possible *controlled actions*/decisions x (i.e., *action set*) and the set $\mathcal{C} \subset \mathbb{R}^n$ of possible *uncontrolled contexts* c (i.e., *context set*) to be chosen by the environment. In this work, we treat $\mathcal{X}$ and $\mathcal{C}$ as finite sets and consider a probability simplex over the context set $\mathcal{C}$. Let $\mathcal{C}_j$ denote the j-th element of $\mathcal{C}$.

A conventional BO algorithm seeks the global maximum of f by sequentially querying f through noisy observations $y_t := f(x_t, c_t) + \xi_t$ for iteration $t = 1, \dots, T$ where $\xi \sim \mathcal{N}(0, \sigma^2)$. The queries are assumed to be costly in terms of time or other resources and it is thus preferred to find the maximum using as few queries as possible. Since f is unknown, we model f probabilistically using a *Gaussian process* (GP) (Williams & Rasmussen, 2006) belief whose pos-

Algorithm 1 Generalized DRBO (Kirschner et al., 2020)

```
1: Input: GP with kernel $k$, score function $\alpha$
2: for iteration $t = 1$ to $T$ do
3:     Obtain reference distribution $p_t$ and margin $\epsilon_t$
4:     Compute $\text{ucb}_x^t := (\mu_t(x, \mathcal{C}_j) + \beta_t \sigma_t(x, \mathcal{C}_j))_{j=1,\ldots,|\mathcal{C}|}^\top$
5:     Select action $x_t = \text{argmax}_{x \in \mathcal{X}}\, \alpha(\text{ucb}_x^t, p_t, \epsilon_t)$
6:     Observe $c_t \sim p_t^*$ and $y_t = f(x_t, c_t) + \xi_t$
7:     Update GP posterior with $\mathcal{D}_{t+1} := \{(x_i, c_i, y_i)\}_{i=1}^t$
8: end for
```

terior $p(f|\mathcal{D}_t, x, c) = \mathcal{N}(\mu_t(x, c), \sigma_t^2(x, c))$ of f at action-context query (x, c) and in iteration t has the following mean and variance given the dataset $\mathcal{D}_t := \{(x_i, c_i, y_i)\}_{i=1}^{t-1}$ of past queries and observations from iterations 1 to $t-1$:

$$\mu_t(x, c) := k_t(x, c)^\top (K_t + \sigma^2 I)^{-1} \varphi_t$$
$$\sigma_t^2(x, c) := k(x, c; x, c) - k_t(x, c)^\top (K_t + \sigma^2 I)^{-1} k_t(x, c)$$

where $\varphi_t := (y_i)_{i=1,\ldots,t-1}^\top$ is a column vector of observations, $k : \mathcal{X} \times \mathcal{C} \times \mathcal{X} \times \mathcal{C} \to \mathbb{R}$ is a positive definite kernel, $k_t(x, c) := (k(x, c; x_i, c_i))_{i=1,\ldots,t-1}^\top$, and $K_t := (k(x_i, c_i; x_j, c_j))_{i,j=1,\ldots,t-1}$. Briefly, the choice of k determines the *reproducing kernel Hilbert space* (RKHS) of functions that the GP posterior mean function will reside in (Schölkopf et al., 2002) and can be thought of as encoding our prior knowledge of f, which we assume to lie in the RKHS associated with k.

As mentioned above, the context vector c within a query is beyond our control. To describe the environment's random choice of c, in each iteration t, we associate a *reference distribution* $p_t \in \mathbb{R}^{|\mathcal{C}|}$ (subject to $\mathbf{1}^\top p_t = 1$ and $p_t \geq \mathbf{0}$)[1] with c, which represents a form of prior knowledge on the distribution of c. DRO considers the setting where c is subject to distribution shift, i.e., there exists an unknown *true distribution* p_t^* s.t. $c \sim p_t^*$ instead. We define an *uncertainty set* $\mathcal{U}_t \subset \mathbb{R}^{|\mathcal{C}|}$ to be containing all possible p_t^* and be an ϵ_t-ball around p_t w.r.t. *distribution distance* d and *margin* ϵ_t. Concretely, $q \in \mathcal{U}_t \iff d(p_t, q) \leq \epsilon_t$ subject to $\mathbf{1}^\top q = 1$ and $q \geq \mathbf{0}$.[1] We consider the worst-case scenario in which after an action $x_t \in \mathcal{X}$ is selected, an adversary chooses the worst possible $q \in \mathcal{U}_t$ to minimize the expected value. We refer to this quantity as the *worst-case expected value*:[2]

$$V_{d,t}(\epsilon_t, g) := \min_{q \in \mathcal{U}_t} \mathbb{E}_q\left[g\right] := \min_{q \in \mathcal{U}_t} q^\top g \qquad (1)$$

where $g \in \mathbb{R}^{|\mathcal{C}|}$ is in general a column vector of outcome values and in this work, $g := f(x, \cdot) := (f(x, \mathcal{C}_j))_{j=1,\ldots,|\mathcal{C}|}^\top$ after selecting action $x \in \mathcal{X}$. DRO then adopts a maximin approach by selecting the action that maximizes the worst-case expected value $\max_{x \in \mathcal{X}} V_{d,t}(\epsilon_t, f(x, \cdot)) = \max_{x \in \mathcal{X}} \min_{q \in \mathcal{U}_t} \mathbb{E}_q\left[f(x, \cdot)\right]$.[2] DRBO combines BO and DRO by introducing the additional challenge of learning f along with solving DRO. The work of Kirschner et al. (2020) has defined the following *cumulative robust regret* incurred by the selected actions x_t for $t = 1, \ldots, T$:

$$R_T = \textstyle\sum_{t=1}^T V_{d,t}(\epsilon_t, f(x_t^*, \cdot)) - V_{d,t}(\epsilon_t, f(x_t, \cdot)) \ ,$$
$$x_t^* := \text{argmax}_{x \in \mathcal{X}}\, V_{d,t}(\epsilon_t, f(x, \cdot)) \ . \qquad (2)$$

The goal is then to select x_t in each iteration $t = 1, \ldots, T$ s.t. R_T (2) is minimized. Algo. 1 details a generalized version of the DRBO algorithm based on the 'General' setting in (Kirschner et al., 2020): It exploits a popular BO acquisition function (Srinivas et al., 2010) called *upper confidence bound* (UCB) as a surrogate function of f for computing a column vector ucb_x^t of $|\mathcal{C}|$ UCB values over every possible context $\mathcal{C}_j$ for $j = 1, \ldots, |\mathcal{C}|$ given an action x and a choice of exploration parameter β_t for deriving the regret bound in Sec. 5, as defined in line 4 of Algo. 1. Our generalized version comes from the introduction of a *score function* $\alpha(g := \text{ucb}_x^t, p_t, \epsilon_t)$ that outputs a score for action x and is set as the worst-case expected value $V_{d,t}(\epsilon_t, g := \text{ucb}_x^t)$ (1) in (Kirschner et al., 2020). Since the constraints are convex, computing this exactly requires solving a convex optimization problem. However, standard solvers such as the commonly used interior-point methods involve solving systems of linear equations with at least $|\mathcal{C}|$ variables ($|\mathcal{C}|^2$ variables for the Wasserstein metric), which incurs $\mathcal{O}(|\mathcal{C}|^3)$ time if no additional structure in the system can be exploited (Boyd & Vandenberghe, 2004). This convex optimization problem has to be solved separately for each action and iteration, which incurs $\mathcal{O}(|\mathcal{C}|^3 |\mathcal{X}| T)$ time over all iterations and hence scales poorly in $|\mathcal{C}|$.[3] Since $|\mathcal{C}|$ grows exponentially with dimensionality n (by assuming that each dimension is discretized into at least some fixed number of points), the DRBO algorithm becomes computationally unwieldy if context $c \in \mathcal{C}$ has more than a few dimensions. To boost the scalability of Algo. 1 to a large $|\mathcal{C}|$, we will propose a fast approximation of the worst-case expected value (Sec. 4) to be used as the score function α, which reduces the incurred $\mathcal{O}(|\mathcal{C}|^3)$ time to either $\mathcal{O}(|\mathcal{C}|)$ or $\mathcal{O}(|\mathcal{C}|^2)$ time (depending on the distribution distance being used).

In each iteration t, the DRBO algorithm (Algo. 1) selects an action x_t to maximize the score function α (line 5), observes the context $c_t \sim p_t^*$ chosen by the environment and the resulting y_t (line 6), and updates the GP posterior belief with $\mathcal{D}_{t+1} = \mathcal{D}_t \bigcup \{(x_t, c_t, y_t)\}$ (line 7).

[1] We use $\geq$ to denote a component-wise inequality for vectors.

[2] To ease the notation, we abuse it slightly by letting $\mathbb{E}_q\left[g\right] := q^\top g$ and $\mathbb{E}_p\left[g\right] := p^\top g$.

[3] We have reported the worst-case time complexity here: In practice, repeatedly solving a convex optimization problem with different parameters can be sped up using techniques like caching the KKT matrix factorization when using interior-point methods.

4. Fast Approximation of Worst-case Expected Value with Worst-case Sensitivity

In this section, we will describe a fast approximation of the worst-case expected value $V_d(\epsilon_d, g)$ by leveraging the notion of *worst-case sensitivity* (Gotoh et al., 2020). From here on, we will drop the dependence on iteration t to ease notations unless necessary.

Note that $V_d(\epsilon_d, g)$ for some reference distribution p and vector g of outcome values is convex in the d-specific margin ϵ_d if d is also convex in ϵ_d (Ben-Tal et al., 2010) where

$$\epsilon_d := \begin{cases} \epsilon & \text{if } d = \text{MMD, TV, or } \mathcal{W} \,, \\ \sqrt{\epsilon} & \text{if } d = \chi^2 \,. \end{cases}$$

Such a convexity implies that the linear approximation of $V_d(\epsilon_d, g)$ at any ϵ_d is a lower bound. In particular, we can consider the linear approximation of $V_d(\epsilon_d, g)$ centered at $\epsilon_d = 0$ (i.e., when $V_d(\epsilon_d, g) = \mathbb{E}_p[g]$ since the adversary can only choose p):[2]

$$W_d(\epsilon_d, g) := \mathbb{E}_p[g] + \epsilon_d\, \mathcal{S}_d(g) \tag{3}$$

where $\mathcal{S}_d(g)$ is the gradient of $V_d(\epsilon_d, g)$ w.r.t. ϵ_d as $\epsilon_d \to 0$. Gotoh et al. (2020) have named $\mathcal{S}_d(g)$ the *worst-case sensitivity* and derived closed-form expressions of $\mathcal{S}_d(g)$ for various d. It turns out that $\mathcal{S}_d(g)$ can be interpreted as a *notion of risk* (e.g., variance, range) whose specific form depends on d. This turns (3) into a *risk-sensitive value* which is a trade-off between the expected value $\mathbb{E}_p[g]$ and a notion of risk $\mathcal{S}_d(g)$ to be further discussed in Sec. 6. Table 2 lists the distribution distances d considered in our work along with their worst-case sensitivities $\mathcal{S}_d(g)$.

Interestingly, the worst-case sensitivity for several distribution distances can be computed in closed form in $\mathcal{O}(|\mathcal{C}|)$ time. We leverage this key insight to efficiently compute a lower bound $W_d(\epsilon_d, g)$ (3) of $V_d(\epsilon_d, g)$, which can be optimized instead of solving the expensive convex optimization problem needed to compute $V_d(\epsilon_d, g)$. The worst-case sensitivity for TV and χ^2 can be computed in $\mathcal{O}(|\mathcal{C}|)$ time, and that of $\mathcal{W}$ can be computed in $\mathcal{O}(|\mathcal{C}|^2)$ time. Due to the Gram matrix inverse M^{-1} in the worst-case sensitivity for MMD, calculating it exactly incurs $\mathcal{O}(|\mathcal{C}|^3)$ time. So, we propose to approximate M with the identity matrix in a similar manner as that in (Staib & Jegelka, 2019) and our approximation of the worst-case sensitivity for MMD is

$$\widehat{\mathcal{S}}_{\text{MMD}}(g) = \left(g^\top g - g^\top \mathbf{1}/|\mathcal{C}|\right)^{1/2} \tag{4}$$

which can be computed in $\mathcal{O}(|\mathcal{C}|)$ time. We will account for this approximation in our regret bound (Sec. 5) and empirically show in Sec. 7 that it can still preserve the competitive performance. We will defer the discussion of its approximation quality to Appendix B.

However, $W_d(\epsilon_d, g)$ (3) is only accurate at small ϵ_d and decreases unboundedly as $\epsilon_d \to \infty$. So, we design a more refined approximation $\widehat{V}_d(\epsilon_d, g)$ at larger ϵ_d called MINIMAXAPPROX as it aims to minimize the maximum possible approximation error $\left|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)\right|$ incurred:[2]

$$\widehat{V}_d(\epsilon_d, g) := \begin{cases} \mathbb{E}_p[g] + \frac{\epsilon_d(\mathcal{T}_d + \mathcal{S}_d(g))}{2} & \text{if } 0 \le \epsilon_d < \epsilon'_d \,, \\ \frac{\mathbb{E}_p[g] + \epsilon_d \mathcal{T}_d + \min_i [g]_i}{2} & \text{if } \epsilon'_d \le \epsilon_d < \epsilon^*_d \,, \\ \min_i [g]_i & \text{if } \epsilon_d \ge \epsilon^*_d \,; \end{cases} \tag{5}$$

where

$$\begin{aligned} \mathcal{T}_d &:= (\min_i [g]_i - \mathbb{E}_p[g])/\epsilon^*_d \,, \\ \epsilon^*_d &:= \min\{\epsilon_d \mid V_d(\epsilon_d, g) = \min_i [g]_i\} \,, \\ \epsilon'_d &:= (\min_i [g]_i - \mathbb{E}_p[g])/\mathcal{S}_d(g) \,. \end{aligned} \tag{6}$$

MINIMAXAPPROX is a piecewise linear bisection of the region in which $V_d(\epsilon_d, g)$ (1) may lie, which we call the *valid region*. Fig. 1 illustrates an example of $V_d(\epsilon_d, g)$, $W_d(\epsilon_d, g)$ (3), valid region, and the MINIMAXAPPROX $\widehat{V}_d(\epsilon_d, g)$ (5). The construction of the valid region uses two additional pieces of available information: (a) minimum value $\min_i [g]_i$ of $V_d(\epsilon_d, g)$ (i.e., when q is a vector of 0's except for 1 at the $\text{argmin}_i [g]_i$-th component), which we refer to as the *worst value*, and (b) the corresponding $\epsilon_d = \epsilon^*_d$ at which that value is attained, i.e., $V_d(\epsilon^*_d, g) = \min_i [g]_i$ (6). The upper bounding line with gradient $\mathcal{T}_d$ (see Fig. 1) can be constructed due to the convex $V_d(\epsilon_d, g)$: Supposing $V_d(\epsilon_d, g)$ is above the upper bounding line, its gradient would have to decrease at some point to reach the worst value at ϵ^*_d, hence making it non-convex. Also, $W_d(\epsilon_d, g)$ (3) is a lower bounding line due to the convex $V_d(\epsilon_d, g)$. Finally, since $V_d(\epsilon_d, g)$ is non-increasing in ϵ_d, the worst value is a lower bounding line as well. So, $V_d(\epsilon_d, g)$ can in fact be any valid convex function that satisfies all these bounds. To minimize the maximum possible approximation error, MINIMAXAPPROX is a piecewise linear bisection of the valid region so that for any ϵ_d, the distance between the upper bound of $V_d(\epsilon_d, g)$ and $\widehat{V}_d(\epsilon_d, g)$ is equal to that between the lower bound of $V_d(\epsilon_d, g)$ and $\widehat{V}_d(\epsilon_d, g)$.

For $d =$ TV, χ^2, or $\mathcal{W}$, the required quantities $\min_i [g]_i$, $\mathbb{E}_p[g]$, and ϵ^*_d can be computed in $\mathcal{O}(|\mathcal{C}|)$ time. For $d =$ MMD, computing ϵ^*_{MMD} requires $\mathcal{O}(|\mathcal{C}|^2)$ time since $\text{MMD}(p, q)$ is required to compute ϵ^*_{MMD} and it involves a quadratic form with a $|\mathcal{C}| \times |\mathcal{C}|$ matrix. Together with our approximation $\widehat{\mathcal{S}}_{\text{MMD}}(g)$ (4), MINIMAXAPPROX $\widehat{V}_d(\epsilon_{\text{MMD}}, g)$ (5) incurs $\mathcal{O}(|\mathcal{C}|^2)$ time in total. In general, our proposed MINIMAXAPPROX can be used with any convex distribution distance d, provided that $\mathcal{S}_d$ can be efficiently computed. The time efficiency benefits will vary with the choice of d.

Table 2. Distribution distances d and their worst-case sensitivities $\mathcal{S}_d(g)$ in closed form considered in our work here. The worst-case sensitivity for MMD is from (Staib & Jegelka, 2019) while that for TV, χ^2, and $\mathcal{W}$ are from (Gotoh et al., 2020). The Gram matrix $M \in \mathbb{R}^{|\mathcal{C}|\times|\mathcal{C}|}$ is constructed by applying the MMD kernel k_M to all pairs of contexts in $\mathcal{C}$, i.e., $M = (k_M(\mathcal{C}_i, \mathcal{C}_j))_{i,j=1,\ldots,|\mathcal{C}|}$. For $\mathcal{W}$, γ is a discrete joint distribution with support $\mathcal{C} \times \mathcal{C}$ (represented as a $|\mathcal{C}| \times |\mathcal{C}|$ matrix) whose marginals are q and p.

Distribution distance d	$d(q,p)$	Worst-case sensitivity $\mathcal{S}_d(g)$
Maximum mean discrepancy (MMD)	$\|q-p\|_M := \sqrt{(q-p)^\top M(q-p)}$	$-\sqrt{g^\top M^{-1} g - (g^\top M^{-1}\mathbf{1})^2/(\mathbf{1}^\top M^{-1}\mathbf{1})}$
Total variation (TV)	$\|q-p\|_1$	$-0.5\,(\max_i\,[g]_i - \min_i\,[g]_i)$
χ^2-divergence (χ^2)	$\sum_{i=1}^{\|\mathcal{C}\|}[p]_i \cdot 0.5([q]_i/[p]_i - 1)^2$	$-(2\mathbb{V}_p[g])^{1/2}$
Wasserstein metric ($\mathcal{W}$)	$\min_{\gamma\in\triangle_{\mathcal{C}\times\mathcal{C}}} \sum_{i,j=1}^{\|\mathcal{C}\|}[\gamma]_{ij}\,\|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}}$	$-\left(\max_i \max_j\,([g]_i - [g_j])\,/\,\|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}}\right)$
with arbitrary norm $\|\cdot\|_{\mathcal{W}}$	where $\sum_{j=1}^{\|\mathcal{C}\|}[\gamma]_{ij} = [q]_i\,,\ \sum_{i=1}^{\|\mathcal{C}\|}[\gamma]_{ij} = [p]_j$	

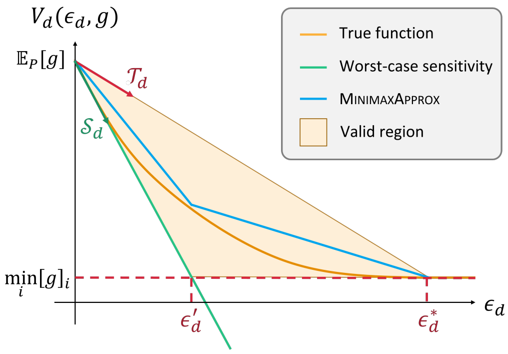


Figure 1. Illustrative example of a convex worst-case expected value $V_d(\epsilon_d, g)$ (1), $W_d(\epsilon_d, g)$ (3) around $\epsilon_d = 0$, and MinimaxApprox $\widehat{V}_d(\epsilon_d, g)$ (5).

5. Theoretical Analysis

In this section, we will first bound the error incurred by approximating the worst-case expected value (1) with MinimaxApprox (5). We will then exploit this error bound to derive the regret bound for Algo. 1 utilizing MinimaxApprox as the score function α instead. Our first result below derives an ϵ_d-dependent bound on the approximation error:

Proposition 1. *Suppose that a reference distribution p, a margin $\epsilon_d \geq 0$, and a vector g of outcome values are given. The error incurred by approximating $V_d(\epsilon_d, g)$ (1) with $\widehat{V}_d(\epsilon_d, g)$ (5) is bounded by*[2]

$$\left|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)\right| \leq \begin{cases} 0.5\epsilon_d(\mathcal{T}_d - \mathcal{S}_d(g)) & \text{if } 0 \leq \epsilon_d < \epsilon'_d\,, \\ 0.5(1 - \frac{\epsilon_d}{\epsilon^*_d})\,(\mathbb{E}_p\,[g] - \min_i [g]_i) & \text{if } \epsilon'_d \leq \epsilon_d < \epsilon^*_d\,, \\ 0 & \text{if } \epsilon_d \geq \epsilon^*_d\,. \end{cases}$$

Its proof is in Appendix A.1. From Proposition 1, when $0 \leq \epsilon_d < \epsilon'_d$, the error bound tightens with a decreasing ϵ_d, which preserves the fine approximation quality of $W_d(\epsilon_d, g)$ (3) when ϵ_d is small. When $\epsilon'_d \leq \epsilon_d < \epsilon^*_d$, the error bound tightens with an increasing ϵ_d, which aligns with the intuition that $V_d(\epsilon_d, g)$ is likely close to the worst value $\min_i [g]_i$ when ϵ_d is large. One can then expect that the maximum possible approximation error for any value of ϵ_d (i.e., an ϵ_d-independent bound) occurs exactly at $\epsilon_d = \epsilon'_d$, as formalized in the next result on the ϵ_d-independent bound:

Corollary 2. *Suppose that a reference distribution p, a margin $\epsilon_d \geq 0$, and a vector g of outcome values are given. The error incurred by approximating $V_d(\epsilon_d, g)$ (1) with $\widehat{V}_d(\epsilon_d, g)$ (5) is bounded by*[2]

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| \leq \frac{1}{2}(\mathbb{E}_p\,[g] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\mathcal{S}_d(g)}\right)$$

with equality when $\epsilon_d = \epsilon'_d$.

Its proof is in Appendix A.2. From Corollary 2, the maximum approximation error for any ϵ_d is small when $\mathbb{E}_p\,[g] - \min_i [g]_i$ is small, i.e., each outcome value $[g]_i$ is close to all other outcome values. In this case, the adversary cannot vary the worst-case expected value $V_d(\epsilon_d, g)$ much. The result below (see its proof in Appendix A.3) specifies the ϵ_d-independent bound for each distribution distance d considered in this work and accounts for our approximation $\widehat{\mathcal{S}}_{\text{MMD}}(g)$ (4) of the worst-case sensitivity for MMD:

Proposition 3. *Suppose that a reference distribution p, a margin $\epsilon_d \geq 0$, and a vector g of outcome values are given. The d-specific error incurred by approximating $V_d(\epsilon_d, g)$ with $\widehat{V}_d(\epsilon_d, g)$ and $\mathcal{S}_{\text{MMD}}(g)$ with $\min\left(\widehat{\mathcal{S}}_{\text{MMD}}(g), \mathcal{T}_d\right)$ is bounded by*[2]

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| \leq A_d(g) := E_d\,(\mathbb{E}_p\,[g] - \min_i [g]_i)$$

where

$$E_d := \begin{cases} 1 + \mathcal{T}_d / \max\left(\|g\|, \|g\|_{M^{-1}}\right) & \text{if } d = \text{MMD}, \\ 0.5 + \mathcal{T}_d / (\max_i [g]_i - \min_i [g]_i) & \text{if } d = \text{TV}, \\ 0.5 + \mathcal{T}_d / (8\mathbb{V}_p[g])^{1/2} & \text{if } d = \chi^2, \\ 0.5 + \mathcal{T}_d / \max\limits_i \max\limits_j \dfrac{[g]_i - [g_j]}{\|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}}} & \text{if } d = \mathcal{W}\,. \end{cases}$$

From Proposition 3, since $\mathcal{T}_d \geq \mathcal{S}_{\text{MMD}}(g)$, we approximate $\mathcal{S}_{\text{MMD}}(g)$ with $\min\left(\widehat{\mathcal{S}}_{\text{MMD}}(g), \mathcal{T}_d\right)$. The denominator of the second term in E_d can be interpreted as the 'complexity' of g whose definition changes with distribution distance d: For TV, the complexity is the range of values in g. For MMD, the complexity is the maximum between the L^2-norm of g vs. $\|g\|_{M^{-1}} := \sqrt{g^\top M^{-1} g}$, the latter of which is a finite-dimensional analog of the RKHS norm associated with the MMD kernel k_M (i.e., a measure of complexity for functions in an RKHS (Williams & Rasmussen, 2006)). As g increases in complexity, the error bound increases. By exploiting this error bound, we can derive a regret bound for Algo. 1 using MINIMAXAPPROX as the score function α:

Theorem 4. *Suppose that $p_t^* \in \mathcal{U}_t$ for $t = 1, \ldots, T$. With probability of at least $1-\delta$, the cumulative robust regret (2) for Algo. 1 utilizing* MINIMAXAPPROX *(5) is bounded by*

$$R_T \leq 4\beta_T \sqrt{T\left(\gamma_T + 4\log\frac{12}{\delta}\right)} + \sum_{t=1}^{T}(2B'_{d,t}\epsilon_{d,t} + 2A_{d,t}^{\max}) \tag{7}$$

where $\beta_t = \sigma\sqrt{\log\det(I+K_t) + 2\log(2/\delta)} + B$, B is the upper bound of the RKHS norm of f, γ_T is the kernel-dependent maximum information gain, $A_{d,t}^{\max} := \max(A_{d,t}(\mathrm{ucb}_{x_t}^t), A_{d,t}(\mathrm{ucb}_{x_t^*}^t))$ where $A_{d,t}(\cdot)$ is the error bound in Proposition 3, and $B'_{d,t}$ is a complexity parameter for $f_{x_t} := f(x_t, \cdot)$ and d given by

$$B'_{d,t} := \begin{cases} \|f_{x_t}\|_{M^{-1}} & \text{if } d = \text{MMD}\ , \\ \|f_{x_t}\| & \text{if } d = \text{TV}\ , \\ \sqrt{2}\,\|f_{x_t}\| & \text{if } d = \chi^2\ , \\ \dfrac{\|f_{x_t}\|}{\min_i \min_{j\neq i} \|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}}} & \textit{if } d = \mathcal{W}\ . \end{cases}$$

Its proof is in Appendix A.4. For the squared exponential kernel, $\gamma_T = \mathcal{O}((\log T)^{d+1}) \leq \mathcal{O}(\sqrt{T})$ (Srinivas et al., 2010). Also, $\log\det(I + K_T) \leq \max_{\{(x_t,c_t)\}_{t=1}^T} \log\det(I+K_t) = \mathcal{O}(\gamma_T)$ (Srinivas et al., 2010). The first term in (7) is thus sublinear in T. The second term, however, is linear in T. So, the regret bound for Algo. 1 utilizing the fast MINIMAXAPPROX (5) to achieve greater time efficiency is of the same order of T as that for the DRBO algorithm of Kirschner et al. (2020). While both algorithms are unfortunately not no-regret[4], using $\widehat{V}_d(\epsilon_d, g)$ (5) is theoretically no worse than using $V_d(\epsilon_d, g)$ (1) in terms of the dependence of their resulting regret bounds on T. Furthermore, Theorem 4 reveals the following insight: A more refined approximation entails a tighter regret bound, which suggests that future DRBO algorithms with improved approximations may perform better empirically.

6. Selection of Distribution Distance

We have used four distribution distances MMD, TV, χ^2, and $\mathcal{W}$ to construct the uncertainty set $\mathcal{U}_t$ in Algo. 1. A practitioner is not limited to these selections as any convex distribution distance d can be used, provided that $\mathcal{S}_d$ can be efficiently computed. A natural question is which distribution distance should be used for a particular application. In this section, we show that our approximation provides insights for a practitioner to select a suitable distance based on interpretable notions of risk. Gotoh et al. (2020) have noted that the DRO literature provides 'little guidance' on how to select a distribution distance and proposed worst-case sensitivity as the link between distribution distance and interpretable risk-sensitive values like mean-variance trade-off (χ^2) and mean-range trade-off (TV). Since $W_d(\epsilon_d, g)$ (3) has the form of a risk-sensitive value, it is more easily interpreted than DRO: If large variance in outcomes is undesirable (e.g., a farm wants large crops but roughly of the same size for logistical reasons), then one may optimize for mean-variance trade-off. If extreme outliers are undesirable (e.g., a portfolio manager wants good returns but avoid massive losses), then one may optimize for mean-range trade-off.

Without any assumptions on the DRO solution that can involve any convex function within the valid region (Fig. 1), it is difficult to formally link the DRO solution to that associated with the risk-sensitive value. However, the difference between MINIMAXAPPROX (5) (as a surrogate) and the risk-sensitive value (3) for $\epsilon_d < \epsilon'_d$ can be obtained:

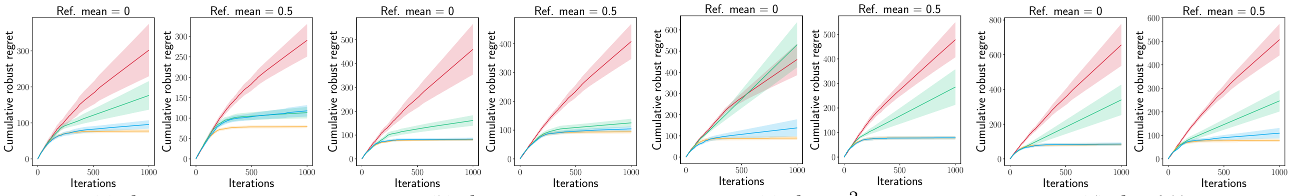
$$\widehat{V}_d(\epsilon_d, g) - W_d(\epsilon_d, g) = \epsilon_d\left(\mathcal{T}_d - \mathcal{S}_d(g)\right)/2\ . \tag{8}$$

Since $\mathcal{T}_d \geq \mathcal{S}_d(g)$ (6) (Fig. 1), the difference is linear and non-decreasing in ϵ_d, which indicates that MINIMAXAPPROX is likely to prefer similar actions (e.g., recall $g = \mathrm{ucb}_x^t$ when used with Algo. 1) as the risk-sensitive value when ϵ_d is small, and this preference is likely to diverge when ϵ_d is large. The same conclusion may be reached with the true DRO solution $V_d(\epsilon_d, g)$ by assuming that $V_d(\epsilon_d, g)$ is strongly convex with parameter $\mu > 0$. Hence, we conclude that the choice of distribution distance can indeed be guided by the risk-sensitive value associated with the worst-case sensitivity for each distance as the preferred actions are likely to be similar at least for small ϵ_d.

From (8), the difference between MINIMAXAPPROX and the risk-sensitive value depends on the worst value $\min_i[g]_i$ in $\mathcal{T}_d$. The influence of worst value is more easily interpreted by rewriting MINIMAXAPPROX (5) for $\epsilon_d < \epsilon'_d$ as[2]

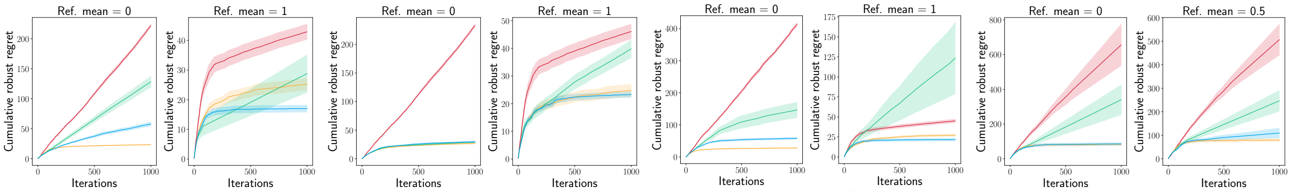
$$\widehat{V}_d(\epsilon_d, g) = \left(1 - \frac{\epsilon_d}{2\epsilon_d^*}\right)\mathbb{E}_p\left[g\right] + \frac{\epsilon_d}{2}\mathcal{S}_d(g) + \frac{\epsilon_d}{2\epsilon_d^*}\min_i[g]_i\ .$$

[4] Kirschner et al. (2020) and Nguyen et al. (2020) have developed no-regret algorithms for the 'Simulator' setting in which c_t can be selected. This differs from the 'General' setting considered in this work, which we find to be the most realistic in practice.



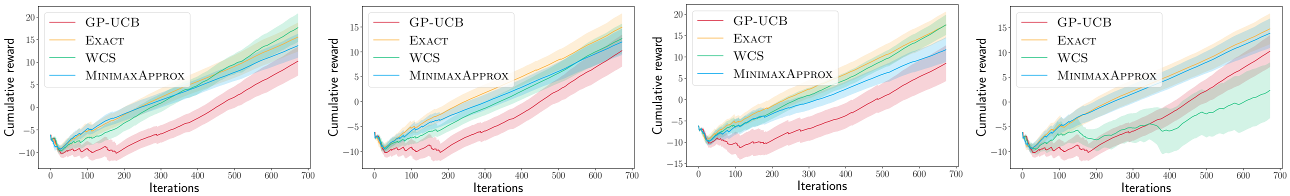
(a) $d =$ MMD (b) $d =$ TV (c) $d = \chi^2$ (d) $d = \mathcal{W}$

Figure 2. **Synthetic Random Functions**: Mean and standard error of cumulative robust regret (lower is better) for Algo. 1 utilizing MINIMAXAPPROX, WCS, EXACT, and GP-UCB with different distribution distances d and means of reference distribution.



(a) $d =$ MMD (b) $d =$ TV (c) $d = \chi^2$ (d) $d = \mathcal{W}$

Figure 3. **Plant Maximum Leaf Area**: Mean and standard error of cumulative robust regret (lower is better) for Algo. 1 utilizing MINIMAXAPPROX, WCS, EXACT, and GP-UCB with different distribution distances d and means of reference distribution.



(a) $d =$ MMD (b) $d =$ TV (c) $d = \chi^2$ (d) $d = \mathcal{W}$

Figure 4. **Wind Power Dataset**: Mean and standard error of cumulative reward (higher is better) for Algo. 1 utilizing MINIMAXAPPROX, WCS, EXACT, and GP-UCB with different distribution distances d.

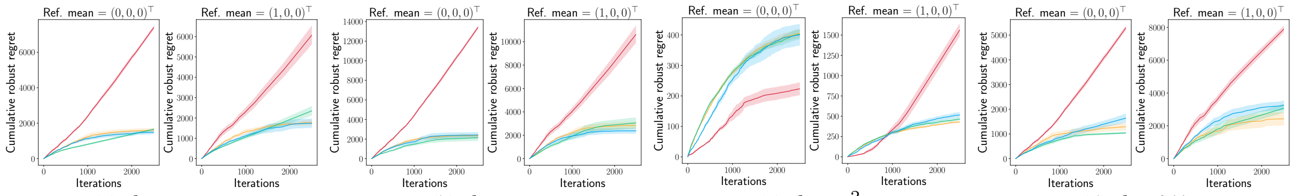
In this form, MINIMAXAPPROX can be interpreted as a weighted sum similar to the risk-sensitive value, except that we shift some weight from expected value $\mathbb{E}_p[g]$ and the notion of risk $S_d(g)$ (thereby forgoing optimality in the risk-sensitive value) to the worst value $\min_i [g]_i$ to be robust to distribution shift. We thus recommend practitioners to select d based on the relevance of its associated notion of risk to the application since optimizing for distributional robustness with d is implicitly optimizing an objective close to the associated risk-sensitive value. Note that there are also practical considerations when choosing d; we defer a full discussion of these practical considerations to Appendix C.

7. Experiments and Discussion

We empirically evaluate the performance of Algo. 1 utilizing MINIMAXAPPROX (5), the simpler $W_d(\epsilon_d, g)$ (3) (termed WCS), EXACT worst-case expected value (1) (Kirschner et al., 2020), or the non-robust $\mathbb{E}_{p_t}\left[\text{ucb}_x^t\right]$ (termed GP-UCB)[2] as a replacement of the score function α in line 5, the latter two of which are baselines. In all experiments except that with the wind power dataset, we use a Gaussian distribution as the reference distribution p_t (mixed with a low-weighted uniform distribution when using χ^2 for numerical reasons) and the uniform distribution as the true distribution p_t^*. We perform the experiments with multiple means for p_t. To ensure that the uncertainty set $\mathcal{U}_t$ includes p_t^*, we set the margin ϵ_t to be the distance between p_t and p_t^*. For the Wasserstein metric, we use the ℓ_2 norm. Refer to Appendix D for further experimental details. The code is available at https://github.com/sebtsh/fast-drbo.

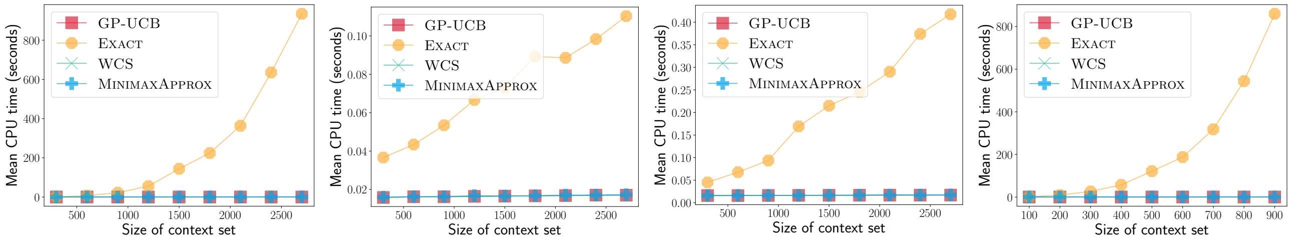
Synthetic Random Functions. We evaluate the performance of the tested algorithms on 2-D random functions drawn from a GP prior where the first dimension is the action and the second is the context. Fig. 2 shows results of the cumulative robust regret averaged over 10 random functions with 10 random initial observations each. MINIMAXAPPROX incurs a lower cumulative robust regret than both the non-robust GP-UCB and the simpler WCS and performs comparably to EXACT in several cases. We additionally show in Appendix E that the robust regret of EXACT using a 'wrong' distribution distance is higher when robust regret is measured w.r.t. some other 'true' distance. This confirms that a penalty is incurred when the wrong distance for an application is selected and highlights the importance of choosing the right distance for a given application.

Plant Maximum Leaf Area. We evaluate the performance of the tested algorithms in finding the acidity condition of a



(a) $d =$ MMD (b) $d =$ TV (c) $d = \chi^2$ (d) $d = \mathcal{W}$

Figure 5. **COVID-19 Test Allocation**: Mean and standard error of cumulative robust regret (lower is better) for Algo. 1 utilizing MINIMAXAPPROX, WCS, EXACT, and GP-UCB with different distribution distances d and means of reference distribution.



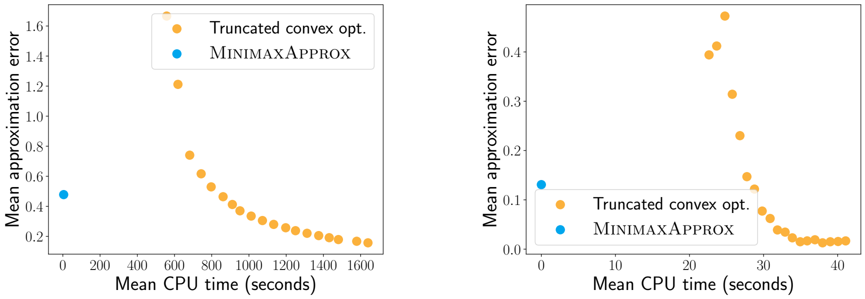
(a) $d =$ MMD (b) $d =$ TV (c) $d = \chi^2$ (d) $d = \mathcal{W}$

Figure 6. **Computation Time**: Mean CPU time (lower is better) incurred by MINIMAXAPPROX, WCS, EXACT, and GP-UCB vs. size of context set with different distribution distances d.

medium that maximizes the worst-case expected value of the leaf area of *Marchantia* plants (i.e., surrogate for crop yield) attained after a certain period. Precision agriculture is an important use case of DRBO as there may be multiple sources of distribution shift of the context variables (uncontrolled nutrients) such as unobservable variations in the fertilizer or soil due to manufacturer error. This objective function was built using kernel ridge regression on data collected from real-world experiments. The action variable is pH and the context variable is ammonium concentration. Fig. 3 shows results of the cumulative robust regret obtained over 10 random sets of 10 initial observations. MINIMAXAPPROX outperforms the non-robust GP-UCB and simpler WCS while performing competitively with EXACT in several cases.

Wind Power Dataset. To evaluate the performance of the tested algorithms on real-world distribution shift, we use wind power data from the Open Power System Data project (Wiese et al., 2019). Weather effects are particularly susceptible to distribution shift due to general unpredictability and climate change. The task is to predict the (scalar) amount of wind power generated in the next hour over a month. The reward obtained depends on the predicted value x (action) and the actual value c (context) and is modeled by the inventory cost function from (Kirschner et al., 2020): $r(x,c) = 0.1 \max(c-x,0) + \min(x,c) - 5\max(x-c,0)$. Ideally, x should be as close to c as possible without exceeding c as there is a large penalty for over-predicting. We use the empirical distribution of wind power in an hour from 2010-2012 as the reference distribution. We then run Algo. 1 with the environment providing the actual wind power in an hour over a month from 2013 and calculate the cumulative reward obtained. Fig. 4 shows results of the mean and standard error of the cumulative reward over the 12 months in 2013. The robustness to distribution shift allows the three robust algorithms to obtain a higher cumulative reward than the non-robust GP-UCB.

COVID-19 Test Allocation. To evaluate the performance of the tested algorithms on real-world use cases with multi-dimensional contexts, we use the COVID-19 epidemic model from (Cashore et al., 2020b). Data-driven epidemic policymaking is a compelling use case for DRBO as an incorrect estimation of the underlying distribution may lead to widespread illness. In this experiment, we have three populations and want to allocate a limited number of tests in order to minimize the number of COVID-19 cases in a given timeframe. The action variable is a 2-D vector indicating the proportion of tests to be allocated to the first and second populations; the proportion allocated to the third population is constrained by these proportions. The context variable is a 3-D vector indicating the proportions of initial COVID-19 cases distributed, respectively, among the first and second populations, and the transmission probability; the total number of initial cases is fixed. Fig. 5 shows results of the cumulative robust regret obtained over 5 random sets of 10 initial observations. MINIMAXAPPROX outperforms GP-UCB and WCS and is competitive with EXACT, except for χ^2 with mean $(0,0,0)^\top$ of reference distribution where GP-UCB performs well due to a case of the non-robust solution being close to the robust one. However, MINIMAXAPPROX incurs a lower final immediate regret than GP-UCB (resp., mean of 0.0146 and 0.0218 over the last 5 iterations), thus indicating that MINIMAXAPPROX finds the more robust solution in the end even though the cumulative robust regret is higher. With $|\mathcal{C}| = 550$, MINIMAXAPPROX and EXACT with $d =$ MMD took about 27 and 161 seconds in CPU time, respectively. When one more dimension



(a) $d = \text{MMD}$ (b) $d = \mathcal{W}$

Figure 7. **Error-time Trade-off**: Mean approximation error vs. CPU time incurred by MINIMAXAPPROX, and Pareto frontier of mean approximation error vs. CPU time incurred by truncated convex optimization.

is added (i.e., $|\mathcal{C}| = 5500$), their incurred times increase, respectively, to about 223 seconds and > 41 hours, hence demonstrating the infeasibility of EXACT when $|\mathcal{C}|$ is large.

Computation Time. We empirically confirm the time efficiency benefit of MINIMAXAPPROX by constructing context sets of various sizes (through increasing the discretization density) on a random function and measuring the incurred time to compute a solution. Fig. 6 shows results of the mean CPU time (seconds) vs. the size of the context set. The time incurred by EXACT noticeably increases as the size of the context set increases with a greater than linear scaling for MMD, while our approximations exhibit negligible growth in incurred time with the size of the context set.

Accuracy-time Trade-off. Since MINIMAXAPPROX trades off approximation quality of the worst-case expected value for greater time efficiency, a natural question is whether a better trade-off can be obtained by simply truncating (i.e., early stopping) a convex optimization solver used by Algo. 1 with EXACT worst-case expected value (1) (Kirschner et al., 2020) to produce intermediate solutions with reduced solution quality and incurred time. Fig. 7 shows a comparison between these two algorithms on a 2-D random function from a GP prior as the objective function. The mean approximation error and CPU time are computed over 20 actions with a context set size of 10000 and 900 using MMD and $\mathcal{W}$, respectively. It can be observed that the trade-off achieved by MINIMAXAPPROX lies well beyond the Pareto frontier of truncated convex optimization (i.e., truncated at different numbers of solver iterations), thus confirming that the time efficiency gain by MINIMAXAPPROX is significant.

8. Conclusion

This paper describes a novel fast approximation based on worst-case sensitivity that enables DRBO to scale to large context sets and provides interpretability to guide the choice of distribution distance. For future work, it is interesting to perform a meta-study over distribution distances to determine if one can be automatically selected based on a secondary optimization objective such as a particular risk-sensitive value. While this work and previous ones have focused on discrete action and context sets, future work can study DRBO over continuous sets. We may also extend our work to the 'Data-driven'/'Simulator' setting in (Kirschner et al., 2020) and look into better yet tractable approximations of worst-case sensitivity for MMD. Finally, we plan to generalize our DRBO algorithm to nonmyopic BO (Kharkovskii et al., 2020b; Ling et al., 2016), high-dimensional BO (Hoang et al., 2018), batch BO (Daxberger & Low, 2017), private outsourced BO (Kharkovskii et al., 2020a), preferential BO (Nguyen et al., 2021d), federated/collaborative BO (Dai et al., 2020b; 2021; Sim et al., 2021), meta-BO (Dai et al., 2022), and multi-fidelity BO (Zhang et al., 2017; 2019) settings, handle information-theoretic acquisition functions (Nguyen et al., 2021c;e), incorporate early stopping (Dai et al., 2019), delayed feedback (Verma et al., 2022), and/or recursive reasoning (Dai et al., 2020a), and consider its application to neural architecture search (Shu et al., 2022a;b) and inverse reinforcement learning (Balakrishnan et al., 2020). For applications with a huge budget of function evaluations, we like to couple our DRBO algorithm with the use of distributed/decentralized (Chen et al., 2012; 2013a;b; 2015; Hoang et al., 2016; 2019; Low et al., 2015; Ouyang & Low, 2018), online/stochastic (Hoang et al., 2015; 2017; Low et al., 2014; Xu et al., 2014; Yu et al., 2019b), or deep (Yu et al., 2019a; 2021) sparse GP models to represent the belief of the unknown objective function efficiently.

Acknowledgements

This research/project is supported by A*STAR under its RIE2020 Advanced Manufacturing and Engineering (AME) Industry Alignment Fund – Pre Positioning (IAF-PP) (Award A19E4a0101) and its RIE2020 Advanced Manufacturing and Engineering (AME) Programmatic Funds (Award A20H6b0151). Sebastian Shenghong Tay is supported by the Institute for Infocomm Research of Agency for Science, Technology and Research (A*STAR).

References

Abbasi-Yadkori, Y. *Online learning for linearly parametrized control problems*. Ph.D. Thesis, University of Alberta, 2012.

Agrawal, A., Verschueren, R., Diamond, S., and Boyd, S. A rewriting system for convex optimization problems. *Journal of Control and Decision*, 5(1):42–60, 2018.

Balakrishnan, S., Nguyen, Q. P., Low, B. K. H., and Soh, H. Efficient exploration of reward functions in inverse reinforcement learning via Bayesian optimization. In *Proc. NeurIPS*, pp. 4187–4198, 2020.

Ben-Tal, A., Bertsimas, D., and Brown, D. B. A soft robust model for optimization under ambiguity. *Operations Research*, 58(4-part-2):1220–1234, 2010.

Bogunovic, I., Scarlett, J., Jegelka, S., and Cevher, V. Adversarially robust optimization with Gaussian processes. In *Proc. NeurIPS*, pp. 5765–5775, 2018.

Boyd, S. P. and Vandenberghe, L. *Convex Optimization*. Cambridge University Press, 2004.

Cakmak, S., Astudillo Marban, R., Frazier, P., and Zhou, E. Bayesian optimization of risk measures. In *Proc. NeurIPS*, pp. 20130–20141, 2020.

Cashore, J. M., Duan, N., Janmohamed, A., Wan, J., Zhang, Y., Henderson, S., Shmoys, D., and Frazier, P. COVID-19 mathematical modeling for Cornell's fall semester. Technical report, 2020a. URL https://people.orie.cornell.edu/pfrazier/COVID_19_Modeling_Jun15.pdf.

Cashore, M., Wan, J., Zhang, Y., and Frazier, P. Group testing, 2020b. URL https://github.com/peter-i-frazier/group-testing.

Chen, J., Low, K. H., Tan, C. K.-Y., Oran, A., Jaillet, P., Dolan, J. M., and Sukhatme, G. S. Decentralized data fusion and active sensing with mobile sensors for modeling and predicting spatiotemporal traffic phenomena. In *Proc. UAI*, pp. 163–173, 2012.

Chen, J., Cao, N., Low, K. H., Ouyang, R., Tan, C. K.-Y., and Jaillet, P. Parallel Gaussian process regression with low-rank covariance matrix approximations. In *Proc. UAI*, pp. 152–161, 2013a.

Chen, J., Low, K. H., and Tan, C. K.-Y. Gaussian process-based decentralized data fusion and active sensing for mobility-on-demand system. In *Proc. RSS*, 2013b.

Chen, J., Low, K. H., Jaillet, P., and Yao, Y. Gaussian process decentralized data fusion and active sensing for spatiotemporal traffic modeling and prediction in mobility-on-demand systems. *IEEE Trans. Autom. Sci. Eng.*, 12: 901–921, 2015.

Chen, Y., Huang, A., Wang, Z., Antonoglou, I., Schrittwieser, J., Silver, D., and de Freitas, N. Bayesian optimization in AlphaGo. arXiv:1812.06855, 2018.

Chowdhury, S. R. and Gopalan, A. On kernelized multi-armed bandits. In *Proc. ICML*, pp. 844–853, 2017.

Dai, Z., Yu, H., Low, B. K. H., and Jaillet, P. Bayesian optimization meets Bayesian optimal stopping. In *Proc. ICML*, pp. 1496–1506, 2019.

Dai, Z., Chen, Y., Low, B. K. H., Jaillet, P., and Ho, T.-H. R2-B2: Recursive reasoning-based Bayesian optimization for no-regret learning in games. In *Proc. ICML*, pp. 2291–2301, 2020a.

Dai, Z., Low, B. K. H., and Jaillet, P. Federated Bayesian optimization via Thompson sampling. In *Proc. NeurIPS*, pp. 9687–9699, 2020b.

Dai, Z., Low, B. K. H., and Jaillet, P. Differentially private federated Bayesian optimization with distributed exploration. In *Proc. NeurIPS*, pp. 9125–9139, 2021.

Dai, Z., Chen, Y., Yu, H., Low, B. K. H., and Jaillet, P. On provably robust meta-Bayesian optimization. In *Proc. UAI*, 2022.

Daxberger, E. A. and Low, B. K. H. Distributed batch Gaussian process optimization. In *Proc. ICML*, pp. 951–960, 2017.

Diamond, S. and Boyd, S. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Domahidi, A., Chu, E., and Boyd, S. ECOS: An SOCP solver for embedded systems. In *Proc. European Control Conference (ECC)*, pp. 3071–3076, 2013.

Frazier, P., Zhang, Y., and Cashore, M. Feasibility of COVID-19 screening for the U.S. population with group testing, 2020. URL https://docs.google.com/document/d/1hw5K5V7XOug_r6CQ0UYt25szQxXFPmZmFhK15ZpH5U0.

Garnett, R. *Bayesian Optimization*. Cambridge University Press, 2022. In preparation.

Gotoh, J., Kim, M. J., and Lim, A. E. B. Worst-case sensitivity. arXiv:2010.10794, 2020.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

Griffiths, R.-R. and Hernández-Lobato, J. M. Constrained Bayesian optimization for automatic chemical design using variational autoencoders. *Chemical Science*, 11(2): 577–586, 2020.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. Array programming with NumPy. *Nature*, 585(7825):357–362, 2020.

Hoang, Q. M., Hoang, T. N., and Low, K. H. A generalized stochastic variational Bayesian hyperparameter learning framework for sparse spectrum Gaussian process regression. In *Proc. AAAI*, pp. 2007–2014, 2017.

Hoang, T. N., Hoang, Q. M., and Low, K. H. A unifying framework of anytime sparse Gaussian process regression models with stochastic variational inference for big data. In *Proc. ICML*, pp. 569–578, 2015.

Hoang, T. N., Hoang, Q. M., and Low, K. H. A distributed variational inference framework for unifying parallel sparse Gaussian process regression models. In *Proc. ICML*, pp. 382–391, 2016.

Hoang, T. N., Hoang, Q. M., and Low, B. K. H. Decentralized high-dimensional Bayesian optimization with factor graphs. In *Proc. AAAI*, pp. 3231–3238, 2018.

Hoang, T. N., Hoang, Q. M., Low, K. H., and How, J. P. Collective online learning of Gaussian processes in massive multi-agent systems. In *Proc. AAAI*, 2019.

Iwazaki, S., Inatsu, Y., and Takeuchi, I. Mean-variance analysis in Bayesian optimization under uncertainty. In *Proc. AISTATS*, pp. 973–981, 2021.

Kharkovskii, D., Dai, Z., and Low, B. K. H. Private outsourced Bayesian optimization. In *Proc. ICML*, pp. 5231–5242, 2020a.

Kharkovskii, D., Ling, C. K., and Low, B. K. H. Nonmyopic Gaussian process optimization with macro-actions. In *Proc. AISTATS*, pp. 4593–4604, 2020b.

Kirschner, J., Bogunovic, I., Jegelka, S., and Krause, A. Distributionally robust Bayesian optimization. In *Proc. AISTATS*, pp. 2174–2184, 2020.

Ling, C. K., Low, B. K. H., and Jaillet, P. Gaussian process planning with Lipschitz continuous reward functions: Towards unifying Bayesian optimization, active learning, and beyond. In *Proc. AAAI*, pp. 1860–1866, 2016.

Low, K. H., Xu, N., Chen, J., Lim, K. K., and Özgül, E. B. Generalized online sparse Gaussian processes with application to persistent mobile robot localization. In *Proc. ECML/PKDD Nectar Track*, pp. 499–503, 2014.

Low, K. H., Yu, J., Chen, J., and Jaillet, P. Parallel Gaussian process regression for big data: Low-rank representation meets Markov approximation. In *Proc. AAAI*, pp. 2821–2827, 2015.

Matthews, A. G. d. G., van der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z., and Hensman, J. GPflow: A Gaussian process library using TensorFlow. *Journal of Machine Learning Research*, 18(40):1–6, 2017. URL http://jmlr.org/papers/v18/16-537.html.

Mohajerin Esfahani, P. and Kuhn, D. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1):115–166, 2018.

Nguyen, Q. P., Dai, Z., Low, B. K. H., and Jaillet, P. Optimizing conditional value-at-risk of black-box functions. In *Proc. NeurIPS*, pp. 4170–4180, 2021a.

Nguyen, Q. P., Dai, Z., Low, B. K. H., and Jaillet, P. Value-at-risk optimization with Gaussian processes. In *Proc. ICML*, pp. 8063–8072, 2021b.

Nguyen, Q. P., Low, B. K. H., and Jaillet, P. An information-theoretic framework for unifying active learning problems. In *Proc. AAAI*, pp. 9126–9134, 2021c.

Nguyen, Q. P., Tay, S., Low, B. K. H., and Jaillet, P. Top-k ranking Bayesian optimization. In *Proc. AAAI*, pp. 9135–9143, 2021d.

Nguyen, Q. P., Wu, Z., Low, B. K. H., and Jaillet, P. Trusted-maximizers entropy search for efficient Bayesian optimization. In *Proc. UAI*, pp. 1486–1495, 2021e.

Nguyen, T., Gupta, S., Ha, H., Rana, S., and Venkatesh, S. Distributionally robust Bayesian quadrature optimization. In *Proc. AISTATS*, pp. 1921–1931, 2020.

O'Donoghue, B., Chu, E., Parikh, N., and Boyd, S. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 169(3):1042–1068, 2016.

Ouyang, R. and Low, K. H. Gaussian process decentralized data fusion meets transfer learning in large-scale distributed cooperative perception. In *Proc. AAAI*, pp. 3876–3883, 2018.

Rahimian, H. and Mehrotra, S. Distributionally robust optimization: A review. arXiv:1908.05659, 2019.

Schölkopf, B., Smola, A. J., Bach, F., et al. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.

Shu, Y., Cai, S., Dai, Z., Ooi, B. C., and Low, B. K. H. NASI: Label- and data-agnostic neural architecture search at initialization. In *Proc. ICLR*, 2022a.

Shu, Y., Chen, Y., Dai, Z., and Low, B. K. H. Neural ensemble search via Bayesian sampling. In *Proc. UAI*, 2022b.

Sim, R. H. L., Zhang, Y., Low, B. K. H., and Jaillet, P. Collaborative Bayesian optimization with fair regret. In *Proc. ICML*, pp. 9691–9701, 2021.

Srinivas, N., Krause, A., Kakade, S., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proc. ICML*, pp. 1015–1022, 2010.

Staib, M. and Jegelka, S. Distributionally robust optimization and generalization in kernel methods. In *Proc. NeurIPS*, pp. 9134–9144, 2019.

Toscano-Palmerin, S. and Frazier, P. I. Bayesian optimization with expensive integrands. arXiv:1803.08661, 2018.

Verma, A., Dai, Z., and Low, B. K. H. Bayesian optimization under stochastic delayed feedback. In *Proc. ICML*, 2022.

Wiese, F., Schlecht, I., Bunke, W.-D., Gerbaulet, C., Hirth, L., Jahn, M., Kunz, F., Lorenz, C., Mühlenpfordt, J., Reimann, J., et al. Open Power System Data–frictionless data for electricity system modelling. *Applied Energy*, 236:401–409, 2019.

Williams, C. K. and Rasmussen, C. E. *Gaussian processes for machine learning*. MIT Press, Cambridge, MA, 2006.

Xu, N., Low, K. H., Chen, J., Lim, K. K., and Özgül, E. B. GP-Localize: Persistent mobile robot localization using online sparse Gaussian process observation model. In *Proc. AAAI*, pp. 2585–2592, 2014.

Yu, H., Chen, Y., Dai, Z., Low, B. K. H., and Jaillet, P. Implicit posterior variational inference for deep Gaussian processes. In *Proc. NeurIPS*, pp. 14475–14486, 2019a.

Yu, H., Hoang, T. N., Low, K. H., and Jaillet, P. Stochastic variational inference for Bayesian sparse Gaussian process regression. In *Proc. IJCNN*, 2019b.

Yu, H., Liu, D., Low, K. H., and Jaillet, P. Convolutional normalizing flows for deep Gaussian processes. In *Proc. IJCNN*, 2021.

Zhang, Y., Hoang, T. N., Low, B. K. H., and Kankanhalli, M. Information-based multi-fidelity Bayesian optimization. In *Proc. NIPS Workshop on Bayesian Optimization*, 2017.

Zhang, Y., Dai, Z., and Low, B. K. H. Bayesian optimization with binary auxiliary information. In *Proc. UAI*, pp. 1222–1232, 2019.

A. Proofs

A.1. Proof of Proposition 1

Proposition 1. *Suppose that a reference distribution p, a margin $\epsilon_d \geq 0$, and a vector g of outcome values are given. The error incurred by approximating $V_d(\epsilon_d, g)$ (1) with $\widehat{V}_d(\epsilon_d, g)$ (5) is bounded by*[2]

$$\left|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)\right| \leq \begin{cases} 0.5\epsilon_d(\mathcal{T}_d - \mathcal{S}_d(g)) & \text{if } 0 \leq \epsilon_d < \epsilon'_d \;, \\ 0.5(1 - \epsilon_d/\epsilon^*_d)\left(\mathbb{E}_p\left[g\right] - \min_i [g]_i\right) & \text{if } \epsilon'_d \leq \epsilon_d < \epsilon^*_d \;, \\ 0 & \text{if } \epsilon_d \geq \epsilon^*_d \;. \end{cases}$$

Proof. When $0 \leq \epsilon_d < \epsilon'_d$, since $V_d(\epsilon_d, g)$ is convex in ϵ_d, $V_d(\epsilon_d, g)$ is lower bounded by the worst case sensitivity linear approximation (3) at $\epsilon_d = 0$ and upper bounded by the linear function $\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d$:

$$\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{S}_d(g) \leq V_d(\epsilon_d, g) \leq \mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d \tag{9}$$

$$\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{S}_d(g) - \left(\mathbb{E}_p\left[g\right] + \frac{1}{2}\epsilon_d(\mathcal{S}_d(g) + \mathcal{T}_d)\right) \leq V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g) \tag{10}$$

$$\leq \mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d - \left(\mathbb{E}_p\left[g\right] + \frac{1}{2}\epsilon_d(\mathcal{S}_d(g) + \mathcal{T}_d)\right) \tag{11}$$

$$\frac{1}{2}\epsilon_d(\mathcal{S}_d(g) - \mathcal{T}_d) \leq V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g) \leq \frac{1}{2}\epsilon_d(\mathcal{T}_d - \mathcal{S}_d(g)) \tag{12}$$

$$-\frac{1}{2}\epsilon_d(\mathcal{T}_d - \mathcal{S}_d(g)) \leq V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g) \leq \frac{1}{2}\epsilon_d(\mathcal{T}_d - \mathcal{S}_d(g)) \;. \tag{13}$$

Since $\mathcal{T}_d \geq \mathcal{S}_d(g)$,

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| \leq \frac{1}{2}\epsilon_d(\mathcal{T}_d - \mathcal{S}_d(g)), \text{ for } 0 \leq \epsilon_d < \epsilon'_d \;. \tag{14}$$

When $\epsilon'_d \leq \epsilon_d < \epsilon^*_d$, $V_d(\epsilon_d, g)$ is lower bounded by the minimum possible value of V given by the worst value $\min_i [g]_i$ and upper bounded by the linear function $\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d$:

$$\min_i [g]_i \leq V_d(\epsilon_d, g) \leq \mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d \tag{15}$$

$$\frac{1}{2}(\min_i [g]_i - \mathbb{E}_p\left[g\right] - \epsilon_d \mathcal{T}_d) \leq V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g) \leq \frac{1}{2}(\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d - \min_i [g]_i) \tag{16}$$

$$-\frac{1}{2}(\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d - \min_i [g]_i) \leq V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g) \leq \frac{1}{2}(\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d - \min_i [g]_i) \;. \tag{17}$$

Since $\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d \geq \min_i [g]_i$,

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| \leq \frac{1}{2}(\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d - \min_i [g]_i) \tag{18}$$

$$= \frac{1}{2}(\mathbb{E}_p\left[g\right] + \epsilon_d \left(\frac{\min_i [g]_i - \mathbb{E}_p\left[g\right]}{\epsilon^*_d}\right) - \min_i [g]_i) \tag{19}$$

$$= \frac{\epsilon^*_d - \epsilon_d}{2\epsilon^*_d}\left(\mathbb{E}_p\left[g\right] - \min_i [g]_i\right), \text{ for } \epsilon'_d \leq \epsilon_d < \epsilon^*_d \;. \tag{20}$$

When $\epsilon_d \geq \epsilon^*_d$, $V_d(\epsilon_d, g) = \min_i [g]_i$ from the definition of ϵ^*_d and the fact that $V_d(\epsilon_d, g)$ is non-increasing in ϵ_d. Hence,

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| = |\min_i [g]_i - \min_i [g]_i| \tag{21}$$

$$= 0, \text{ for } \epsilon_d \geq \epsilon^*_d \;. \tag{22}$$

□

A.2. Proof of Corollary 2

Corollary 2. *Suppose that a reference distribution* p*, a margin* $\epsilon_d \geq 0$*, and a vector* g *of outcome values are given. The error incurred by approximating* $V_d(\epsilon_d, g)$ (1) *with* $\widehat{V}_d(\epsilon_d, g)$ (5) *is bounded by*[2]

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| \leq \frac{1}{2}(\mathbb{E}_p\,[g] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\mathcal{S}_d(g)}\right)$$

with equality when $\epsilon_d = \epsilon'_d$.

Proof. Since $\mathcal{T}_d \geq \mathcal{S}_d(g)$, the upper bound $\frac{1}{2}\epsilon_d(\mathcal{T}_d - \mathcal{S}_d(g))$ from (14) is non-decreasing in ϵ_d. When $0 \leq \epsilon_d < \epsilon'_d$, we thus have the ϵ_d-independent upper bound

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| \leq \frac{1}{2}\epsilon'_d(\mathcal{T}_d - \mathcal{S}_d(g)), \text{ for } 0 \leq \epsilon_d < \epsilon'_d\ . \tag{23}$$

Since $\mathcal{T}_d \leq 0$, the upper bound $\frac{1}{2}(\mathbb{E}_p\,[g] + \epsilon_d\mathcal{T}_d - \min_i[g]_i)$ from (18) is non-increasing in ϵ_d. When $\epsilon'_d \leq \epsilon_d < \epsilon^*_d$, we thus have the ϵ_d-independent upper bound

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| \leq \frac{1}{2}(\mathbb{E}_p\,[g] + \epsilon'_d\mathcal{T}_d - \min_i [g]_i), \text{ for } \epsilon'_d \leq \epsilon_d < \epsilon^*_d\ . \tag{24}$$

Subtracting the upper bound in (24) from (23),

$$\begin{aligned}
\frac{1}{2}\epsilon'_d(\mathcal{T}_d - \mathcal{S}_d(g)) - \frac{1}{2}(\mathbb{E}_p\,[g] + \epsilon'_d\mathcal{T}_d - \min_i [g]_i) &= \frac{1}{2}(\min_i [g]_i - \mathbb{E}_p\,[g] - \epsilon'_d\mathcal{S}_d(g)) && (25)\\
&= \frac{1}{2}(\min_i [g]_i - \mathbb{E}_p\,[g] - (\min_i [g]_i - \mathbb{E}_p\,[g])) && (26)\\
&= 0\ . && (27)
\end{aligned}$$

Both upper bounds thus have the same value. Since the approximation error is zero for $\epsilon_d \geq \epsilon^*_d$, we conclude that for all $\epsilon_d \geq 0$,

$$\begin{aligned}
|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| &\leq \frac{1}{2}\epsilon'_d(\mathcal{T}_d - \mathcal{S}_d(g)) && (28)\\
&= \frac{1}{2}(\min_i [g]_i - \mathbb{E}_p\,[g])\left(\frac{\mathcal{T}_d}{\mathcal{S}_d(g)} - 1\right) && (29)\\
&= \frac{1}{2}(\mathbb{E}_p\,[g] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\mathcal{S}_d(g)}\right)\ . && (30)
\end{aligned}$$

□

A.3. Proof of Proposition 3

Proposition 3. *Suppose that a reference distribution* p*, a margin* $\epsilon_d \geq 0$*, and a vector* g *of outcome values are given. The* d*-specific error incurred by approximating* $V_d(\epsilon_d, g)$ *with* $\widehat{V}_d(\epsilon_d, g)$ *and* $\mathcal{S}_{\text{MMD}}(g)$ *with* $\min\left(\widehat{\mathcal{S}}_{\text{MMD}}(g), \mathcal{T}_d\right)$ *is bounded by*[2]

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| \leq A_d(g) := E_d\left(\mathbb{E}_p\,[g] - \min_i [g]_i\right)$$

where

$$E_d := \begin{cases} 1 + \mathcal{T}_d / \max\left(\|g\|, \|g\|_{M^{-1}}\right) & \text{if } d = \text{MMD}\ ,\\ 0.5 + \mathcal{T}_d / (\max_i [g]_i - \min_i [g]_i) & \text{if } d = \text{TV}\ ,\\ 0.5 + \mathcal{T}_d / (8\mathbb{V}_p[g])^{1/2} & \text{if } d = \chi^2\ ,\\ 0.5 + \mathcal{T}_d / \max\limits_i \max\limits_j \dfrac{[g]_i - [g_j]}{\|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}}} & \text{if } d = \mathcal{W}\ . \end{cases}$$

Proof. We begin with the general approximation error from Corollary 2:

$$|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)| \leq \frac{1}{2}(\mathbb{E}_p\,[g] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\mathcal{S}_d(g)}\right), \text{ for } \epsilon_d \geq 0\ . \tag{31}$$

The bounds for TV, χ^2, and $\mathcal{W}$ are attained by simply substituting in the closed form for their worst-case sensitivities (listed in Table 2). The proof for MMD is more involved as we have to account for the approximation of $\mathcal{S}_{\text{MMD}}(g)$ with $\widetilde{\mathcal{S}}_{\text{MMD}}(g) := \min\left(\{\widehat{\mathcal{S}}_{\text{MMD}}(g), \mathcal{T}_d\}\right)$. We first define

$$\mathcal{S}_{\text{min}} := \min\left(\{\mathcal{S}_{\text{MMD}}(g), \widetilde{\mathcal{S}}_{\text{MMD}}(g)\}\right) \tag{32}$$

$$\mathcal{S}_{\text{max}} := \max\left(\{\mathcal{S}_{\text{MMD}}(g), \widetilde{\mathcal{S}}_{\text{MMD}}(g)\}\right) \tag{33}$$

$$\epsilon'_{\text{min}} := \frac{\min_i [g]_i - \mathbb{E}_p\left[g\right]}{\mathcal{S}_{\text{min}}} \tag{34}$$

$$\epsilon'_{\text{max}} := \frac{\min_i [g]_i - \mathbb{E}_p\left[g\right]}{\mathcal{S}_{\text{max}}} \tag{35}$$

$$\tilde{\epsilon}'_d := \frac{\min_i [g]_i - \mathbb{E}_p\left[g\right]}{\widetilde{\mathcal{S}}_{\text{MMD}}(g)} \tag{36}$$

$$\widehat{V}_{\text{min}}(\epsilon_d, g) := \begin{cases} \mathbb{E}_p\left[g\right] + \frac{1}{2}\epsilon_d(\mathcal{T}_d + \mathcal{S}_{\text{min}}), & \text{if } 0 \leq \epsilon_d < \epsilon'_{\text{min}} \\ \frac{1}{2}\left(\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d + \min_i [g]_i\right), & \text{if } \epsilon'_{\text{min}} \leq \epsilon_d < \epsilon^*_d \\ \min_i [g]_i, & \text{if } \epsilon_d \geq \epsilon^*_d \end{cases} \tag{37}$$

$$\widehat{V}_{\text{max}}(\epsilon_d, g) := \begin{cases} \mathbb{E}_p\left[g\right] + \frac{1}{2}\epsilon_d(\mathcal{T}_d + \mathcal{S}_{\text{max}}), & \text{if } 0 \leq \epsilon_d < \epsilon'_{\text{max}} \\ \frac{1}{2}\left(\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d + \min_i [g]_i\right), & \text{if } \epsilon'_{\text{max}} \leq \epsilon_d < \epsilon^*_d \\ \min_i [g]_i, & \text{if } \epsilon_d \geq \epsilon^*_d \end{cases} \tag{38}$$

$$\widetilde{V}_d(\epsilon_d, g) := \begin{cases} \mathbb{E}_p\left[g\right] + \frac{1}{2}\epsilon_d(\mathcal{T}_d + \widetilde{\mathcal{S}}_{\text{MMD}}(g)), & \text{if } 0 \leq \epsilon_d < \tilde{\epsilon}'_d \\ \frac{1}{2}\left(\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d + \min_i [g]_i\right), & \text{if } \tilde{\epsilon}'_d \leq \epsilon_d < \epsilon^*_d \\ \min_i [g]_i, & \text{if } \epsilon_d \geq \epsilon^*_d \end{cases} \tag{39}$$

where $d = \text{MMD}$. In particular, $\widehat{V}_d$ is equivalent to either $\widehat{V}_{\text{min}}$ or $\widehat{V}_{\text{max}}$, and $\widetilde{V}_d$ is equivalent to the other. The proof considers the general case where it is unknown which corresponds to which. Since $\mathcal{S}_{\text{min}}$ and $\mathcal{S}_{\text{max}}$ are negative, $0 \leq \epsilon'_{\text{min}} \leq \epsilon'_{\text{max}} \leq \epsilon^*_d$. When $0 \leq \epsilon_d < \epsilon'_{\text{min}}$,

$$\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g) = \mathbb{E}_p\left[g\right] + \frac{1}{2}\epsilon_d(\mathcal{T}_d + \mathcal{S}_{\text{max}}) - \left(\mathbb{E}_p\left[g\right] + \frac{1}{2}\epsilon_d(\mathcal{T}_d + \mathcal{S}_{\text{min}})\right) \tag{40}$$

$$= \frac{1}{2}\epsilon_d(\mathcal{S}_{\text{max}} - \mathcal{S}_{\text{min}})\ . \tag{41}$$

Since $\mathcal{S}_{\text{max}} \geq \mathcal{S}_{\text{min}}$, $\frac{1}{2}\epsilon_d(\mathcal{S}_{\text{max}} - \mathcal{S}_{\text{min}})$ is non-decreasing in ϵ_d. By substituting $\epsilon_d = \epsilon'_{\text{min}}$, we obtain the ϵ_d-independent upper bound

$$\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g) \leq \frac{1}{2}\left(\frac{\min_i [g]_i - \mathbb{E}_p\left[g\right]}{\mathcal{S}_{\text{min}}}\right)(\mathcal{S}_{\text{max}} - \mathcal{S}_{\text{min}}) \tag{42}$$

$$= \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{S}_{\text{max}}}{\mathcal{S}_{\text{min}}}\right),\ \text{for } 0 \leq \epsilon_d < \epsilon'_{\text{min}} \tag{43}$$

and the corresponding lower bound by substituting $\epsilon_d = 0$

$$\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g) \geq 0,\ \text{for } 0 \leq \epsilon_d < \epsilon'_{\text{min}}\ . \tag{44}$$

When $\epsilon'_{\text{min}} \leq \epsilon_d < \epsilon'_{\text{max}}$,

$$\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g) = \mathbb{E}_p\left[g\right] + \frac{1}{2}\epsilon_d(\mathcal{T}_d + \mathcal{S}_{\text{max}}) - \frac{1}{2}\left(\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{T}_d + \min_i [g]_i\right) \tag{45}$$

$$= \frac{1}{2}\left(\mathbb{E}_p\left[g\right] + \epsilon_d \mathcal{S}_{\text{max}} - \min_i [g]_i\right)\ . \tag{46}$$

Since $\mathcal{S}_{\text{max}} \leq 0$, $\epsilon_d \mathcal{S}_{\text{max}}$ is non-increasing in ϵ_d. By substituting $\epsilon_d = \epsilon'_{\text{min}}$, we obtain the ϵ_d-independent upper bound

$$\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g) \leq \frac{1}{2}\left(\mathbb{E}_p\left[g\right] + \left(\frac{\min_i [g]_i - \mathbb{E}_p\left[g\right]}{\mathcal{S}_{\text{min}}}\right)\mathcal{S}_{\text{max}} - \min_i [g]_i\right) \tag{47}$$

$$= \frac{1}{2}\left(\mathbb{E}_p\left[g\right] - \min_i [g]_i - \left(\mathbb{E}_p\left[g\right] - \min_i [g]_i\right)\frac{\mathcal{S}_{\text{max}}}{\mathcal{S}_{\text{min}}}\right) \tag{48}$$

$$= \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{S}_{\text{max}}}{\mathcal{S}_{\text{min}}}\right), \text{ for } \epsilon'_{\text{min}} \leq \epsilon_d \leq \epsilon'_{\text{max}} \tag{49}$$

and the corresponding lower bound by substituting $\epsilon_d = \epsilon'_{\text{max}}$

$$\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g) \geq \frac{1}{2}\left(\mathbb{E}_p\left[g\right] + \epsilon'_{\text{max}}\mathcal{S}_{\text{max}} - \min_i [g]_i\right) \tag{50}$$

$$= 0, \text{ for } \epsilon'_{\text{min}} \leq \epsilon_d \leq \epsilon'_{\text{max}}\ . \tag{51}$$

When $\epsilon_d \geq \epsilon'_{\text{max}}$,

$$\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g) = 0\ . \tag{52}$$

From Equations (43), (49) and (52), we obtain the global upper bound

$$\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g) \leq \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{S}_{\text{max}}}{\mathcal{S}_{\text{min}}}\right)\ . \tag{53}$$

From Equations (44), (51) and (52), we obtain the global lower bound

$$\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g) \geq 0\ . \tag{54}$$

We thus have that

$$\left|\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g)\right| \leq \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{S}_{\text{max}}}{\mathcal{S}_{\text{min}}}\right), \text{ for } \epsilon_d \geq 0\ . \tag{55}$$

Since $V_d(\epsilon_d, g)$ is convex, $\mathcal{T}_d \geq \mathcal{S}_{\text{MMD}}(g)$. By definition, $\mathcal{T}_d \geq \widetilde{\mathcal{S}}_{\text{MMD}}(g)$. Hence,

$$\left|\widehat{V}_{\text{max}}(\epsilon_d, g) - \widehat{V}_{\text{min}}(\epsilon_d, g)\right| \leq \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\mathcal{S}_{\text{min}}}\right) \tag{56}$$

$$= \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\min\left(\{\widehat{\mathcal{S}}_{\text{MMD}}(g), \mathcal{S}_{\text{MMD}}(g)\}\right)}\right) \tag{57}$$

$$\leq \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\min\left(\{-\left\|g\right\|, -\left\|g\right\|_{M^{-1}}\}\right)}\right) \tag{58}$$

where the first inequality arises since $\mathcal{S}_{\text{min}} \leq 0$ and the last inequality applies Lemma 7. Finally,

$$\left|V_d(\epsilon_d, g) - \widetilde{V}_d(\epsilon_d, g)\right| \leq \left|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)\right| + \left|\widehat{V}_d(\epsilon_d, g) - \widetilde{V}_d(\epsilon_d, g)\right| \tag{59}$$

$$\leq \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\mathcal{S}_{\text{MMD}}(g)}\right) + \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\min\left(\{-\left\|g\right\|, -\left\|g\right\|_{M^{-1}}\}\right)}\right) \tag{60}$$

$$\leq \frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\min\left(\{\widehat{\mathcal{S}}_{\text{MMD}}(g), \mathcal{S}_{\text{MMD}}(g)\}\right)}\right) + \tag{61}$$

$$\frac{1}{2}(\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\min\left(\{-\left\|g\right\|, -\left\|g\right\|_{M^{-1}}\}\right)}\right) \tag{62}$$

$$\leq (\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 - \frac{\mathcal{T}_d}{\min\left(\{-\left\|g\right\|, -\left\|g\right\|_{M^{-1}}\}\right)}\right) \tag{63}$$

$$= (\mathbb{E}_p\left[g\right] - \min_i [g]_i)\left(1 + \frac{\mathcal{T}_d}{\max\left(\{\left\|g\right\|, \left\|g\right\|_{M^{-1}}\}\right)}\right) \tag{64}$$

where the first inequality applies the triangle inequality, the second inequalty uses Corollary 2 to bound $\left|V_d(\epsilon_d, g) - \widehat{V}_d(\epsilon_d, g)\right|$ and the fact that the bound for $\left|\widehat{V}_{\max}(\epsilon_d, g) - \widehat{V}_{\min}(\epsilon_d, g)\right|$ bounds $\left|\widehat{V}_d(\epsilon_d, g) - \widetilde{V}_d(\epsilon_d, g)\right|$, the third inequality arises since $\mathcal{T}_d \leq 0$, and the fourth inequality applies Lemma 7 again, which concludes the proof. □

A.4. Proof of Theorem 4

Theorem 4. Suppose that $p_t^* \in \mathcal{U}_t$ for $t = 1, \ldots, T$. With probability of at least $1 - \delta$, the cumulative robust regret (2) for Algo. 1 utilizing MINIMAXAPPROX (5) *is bounded by*

$$R_T \leq 4\beta_T \sqrt{T\left(\gamma_T + 4\log\frac{12}{\delta}\right)} + \sum_{t=1}^{T}(2B'_{d,t}\epsilon_{d,t} + 2A_{d,t}^{\max}) \tag{65}$$

where $\beta_t = \sigma\sqrt{\log\det(I + K_t) + 2\log(2/\delta)} + B$, B is the upper bound of the RKHS norm of f, γ_T is the kernel-dependent maximum information gain, $A_{d,t}^{\max} := \max(A_{d,t}(\mathrm{ucb}_{x_t}^t), A_{d,t}(\mathrm{ucb}_{x_t^*}^t))$ where $A_{d,t}(\cdot)$ is the error bound in Proposition 3, and $B'_{d,t}$ is a complexity parameter for $f_{x_t} := f(x_t, \cdot)$ and d given by

$$B'_{d,t} := \begin{cases} \|f_{x_t}\|_{M^{-1}} & \text{if } d = \text{MMD}\ , \\ \|f_{x_t}\| & \text{if } d = \text{TV}\ , \\ \sqrt{2}\,\|f_{x_t}\| & \text{if } d = \chi^2\ , \\ \dfrac{\|f_{x_t}\|}{\min_i \min_{j\neq i} \|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}}} & \textit{if } d = \mathcal{W}\ . \end{cases}$$

Proof. The proof follows the steps of the proof of Theorem 2 from (Kirschner et al., 2020) except for the introduction of the approximation error induced by our approximation of $V_d(\epsilon_d, g)$ with $\widehat{V}_d(\epsilon_d, g)$, and the complexity parameter $B'_{d,t}$ that depends on the choice of d.

We begin by recalling the definition of the vector of upper confidence bound values associated with action x at iteration t $\mathrm{ucb}_x^t \in \mathbb{R}^{|\mathcal{C}|}$:

$$[\mathrm{ucb}_x^t]_j := \mu_t(x, \mathcal{C}_j) + \beta_t\sigma_t(x, \mathcal{C}_j),\ \forall j \in [|\mathcal{C}|]\ . \tag{66}$$

In this proof, we switch to the inner product notation to ease the notation.

We define the worst-case distributions associated with an action x and a set of outcome values (which may be either the true function values or the upper confidence bound values):

$$q_x^{\mathrm{ucb}_t} \in \underset{q\in\mathcal{U}_t}{\operatorname{argmin}}\langle q, \mathrm{ucb}_x^t\rangle \tag{67}$$

$$q_x^f \in \underset{q\in\mathcal{U}_t}{\operatorname{argmin}}\langle q, f_x\rangle \tag{68}$$

The robust regret at iteration t is

$$r_t = \min_{q\in\mathcal{U}_t}\mathbb{E}_q\left[f(x_t^*, \cdot)\right] - \min_{q\in\mathcal{U}_t}\mathbb{E}_q\left[f(x_t, \cdot)\right] \tag{69}$$

$$\leq \langle q_{x_t^*}^{\mathrm{ucb}_t}, \mathrm{ucb}_{x_t^*}^t\rangle - \langle q_{x_t}^f, f_{x_t}\rangle \tag{70}$$

since $\mathrm{ucb}_{x_t^*}^t \geq f_{x_t^*}$. We now introduce the approximation error upper bound $A_{d,t}$. We have that

$$\widehat{V}_d(\epsilon_d, \mathrm{ucb}_{x_t}^t) \leq \langle q_{x_t}^{\mathrm{ucb}_t}, \mathrm{ucb}_{x_t}^t\rangle + A_{d,t} \tag{71}$$

$$\langle q_{x_t^*}^{\mathrm{ucb}_t}, \mathrm{ucb}_{x_t^*}^t\rangle - A_{d,t} \leq \widehat{V}_d(\epsilon_d, \mathrm{ucb}_{x_t^*}^t) \tag{72}$$

$$\widehat{V}_d(\epsilon_d, \mathrm{ucb}_{x_t^*}^t) \leq \widehat{V}_d(\epsilon_d, \mathrm{ucb}_{x_t}^t) \tag{73}$$

where the last inequality uses the fact that at iteration t we choose the x_t that maximizes $\widehat{V}_d(\epsilon_d, \mathrm{ucb}_{x_t}^t)$. Adding these 3 inequalities together,

$$\widehat{V}_d(\epsilon_d, \mathrm{ucb}_{x_t}^t) + \langle q_{x_t^*}^{\mathrm{ucb}_t}, \mathrm{ucb}_{x_t^*}^t\rangle - A_{d,t} + \widehat{V}_d(\epsilon_d, \mathrm{ucb}_{x_t^*}^t) \leq \langle q_{x_t}^{\mathrm{ucb}_t}, \mathrm{ucb}_{x_t}^t\rangle + A_{d,t} + \widehat{V}_d(\epsilon_d, \mathrm{ucb}_{x_t^*}^t) + \widehat{V}_d(\epsilon_d, \mathrm{ucb}_{x_t}^t) \tag{74}$$

$$\langle q_{x_t^*}^{\mathrm{ucb}_t}, \mathrm{ucb}_{x_t^*}^t\rangle \leq \langle q_{x_t}^{\mathrm{ucb}_t}, \mathrm{ucb}_{x_t}^t\rangle + 2A_{d,t} \tag{75}$$

Applying this inequality to (70),

$$\begin{aligned} r_t &\le \langle q^{\text{ucb}_t}_{x_t}, \text{ucb}^t_{x_t}\rangle + 2A_{d,t} - \langle q^f_{x_t}, f_{x_t}\rangle & (76)\\ &\le \langle p^*_t, \text{ucb}^t_{x_t}\rangle - \langle q^f_{x_t}, f_{x_t}\rangle + 2A_{d,t} & (77)\\ &= \langle p^*_t, f_{x_t}\rangle + \langle p^*_t, \text{ucb}^t_{x_t} - f_{x_t}\rangle - \langle q^f_{x_t}, f_{x_t}\rangle + 2A_{d,t} & (78)\\ &= \langle p^*_t, \text{ucb}^t_{x_t} - f_{x_t}\rangle + \langle p^*_t - q^f_{x_t}, f_{x_t}\rangle + 2A_{d,t} & (79)\\ &\le 2\beta_t\langle p^*_t, \sigma_t(x_t,\cdot)\rangle + \langle p^*_t - q^f_{x_t}, f_{x_t}\rangle + 2A_{d,t} & (80) \end{aligned}$$

where the second inequality uses the fact that $q^{\text{ucb}_t}_{x_t}$ is the worst-case distribution and the third inequality uses the fact that f_{x_t} lies within the confidence bounds with probability at least $1-\delta$ (Lemma 9).
We now consider the term $\langle p^*_t - q^f_{x_t}, f_{x_t}\rangle$. The bound for this term depends on the distribution distance d.
For MMD, applying Lemma 5,

$$\begin{aligned} \langle p^*_t - q^f_{x_t}, f_{x_t}\rangle &\le \left\|p^*_t - q^f_{x_t}\right\|_M \left\|f_{x_t}\right\|_{M^{-1}} & (81)\\ &\le 2\epsilon_{\text{MMD},t}\left\|f_{x_t}\right\|_{M^{-1}} \ . & (82) \end{aligned}$$

For TV, applying the Cauchy-Schwarz inequality,

$$\begin{aligned} \langle p^*_t - q^f_{x_t}, f_{x_t}\rangle &\le \left\|p^*_t - q^f_{x_t}\right\| \left\|f_{x_t}\right\| & (83)\\ &\le \left\|p^*_t - q^f_{x_t}\right\|_1 \left\|f_{x_t}\right\| & (84)\\ &\le 2\epsilon_{\text{TV},t}\left\|f_{x_t}\right\| & (85) \end{aligned}$$

where the second inequality arises from the fact that the L^1-norm of a vector is greater than or equal to its L^2-norm.
For χ^2, again applying the Cauchy-Schwarz inequality,

$$\begin{aligned} \langle p^*_t - q^f_{x_t}, f_{x_t}\rangle &\le \left\|p^*_t - q^f_{x_t}\right\| \left\|f_{x_t}\right\| & (86)\\ &\le \left(\left\|p^*_t - p\right\| + \left\|q^f_{x_t} - p\right\|\right) \left\|f_{x_t}\right\| & (87)\\ &\le \left(\sqrt{2\chi^2(p^*_t, p)} + \sqrt{2\chi^2(q^f_{x_t}, p)}\right) \left\|f_{x_t}\right\| & (88)\\ &\le 2\sqrt{2\epsilon_{\chi^2,t}}\left\|f_{x_t}\right\| & (89) \end{aligned}$$

where the second inequality applies the triangle inequality and the third inequality applies Lemma 6.
For $\mathcal{W}$, from (84),

$$\begin{aligned} \langle p^*_t - q^f_{x_t}, f_{x_t}\rangle &\le \left\|p^*_t - q^f_{x_t}\right\|_1 \left\|f_{x_t}\right\| & (90)\\ &\le \frac{2}{\min_i \min_{j\neq i}\left\|\mathcal{C}_i - \mathcal{C}_j\right\|_{\mathcal{W}}}\epsilon_{\mathcal{W},t}\left\|f_{x_t}\right\| & (91) \end{aligned}$$

where the second inequality applies Lemma 8. We thus have the generalized regret bound

$$r_t \le 2\beta_t\langle p^*_t, \sigma_t(x_t,\cdot)\rangle + 2B'_{d,t}\epsilon_{d,t} + 2A_{d,t} \tag{92}$$

$$B'_{d,t} := \begin{cases} \left\|f_{x_t}\right\|_{M^{-1}} & \text{if } d = \text{MMD}\ , \\ \left\|f_{x_t}\right\| & \text{if } d = \text{TV}\ , \\ \sqrt{2}\left\|f_{x_t}\right\| & \text{if } d = \chi^2\ , \\ \left\|f_{x_t}\right\| / \min_i \min_{j\neq i}\left\|\mathcal{C}_i - \mathcal{C}_j\right\|_{\mathcal{W}} & \text{if } d = \mathcal{W}\ . \end{cases} \tag{93}$$

$$\epsilon_{d,t} := \begin{cases} \epsilon_t, & \text{if } d = \text{MMD, TV, or } \mathcal{W}\ , \\ \sqrt{\epsilon_t}, & \text{if } d = \chi^2\ . \end{cases} \tag{94}$$

The cumulative regret R_T is then given by

$$R_T = \sum_{t=1}^{T} r_t \tag{95}$$

$$\leq \sum_{t=1}^{T} 2\beta_t \langle p_t^*, \sigma_t(x_t, \cdot)\rangle + \sum_{t=1}^{T}(2B'_{d,t}\epsilon_{d,t} + 2A_{d,t}) \tag{96}$$

$$\leq 4\beta_T \sqrt{T\left(\gamma_T + 4\log\frac{12}{\delta}\right)} + \sum_{t=1}^{T}(2B'_{d,t}\epsilon_{d,t} + 2A_{d,t}) \tag{97}$$

where $\sum_{t=1}^{T} 2\beta_t\langle p_t^*, \sigma_t(x_t, \cdot)\rangle$ is bounded by the same method as in the proof of Theorem 2 in (Kirschner et al., 2020), which completes the proof. □

A.5. Lemmas

Lemma 5. *For any two vectors $q, p \in \mathbb{R}^n$ and positive definite symmetric matrix $M \in \mathbb{R}^{n\times n}$,*

$$\langle q, p\rangle \leq \|q\|_M \|p\|_{M^{-1}} \tag{98}$$

where $\|q\|_M := \sqrt{q^\top M q}$.

Proof.

$$(\|q\|_M \|p\|_{M^{-1}})^2 = q^\top M q p^\top M^{-1} p\ . \tag{99}$$

Since M is a positive definite symmetric matrix, it can be decomposed into $M = UDU^\top$ where $U \in \mathbb{R}^{n\times n}$ is an orthogonal matrix with columns that together form an orthonormal basis of $\mathbb{R}^n$ and $D \in \mathbb{R}^{n\times n}$ is a diagonal matrix with positive entries that are the eigenvalues of M. M^{-1} can then be decomposed into $M^{-1} = UD^{-1}U^\top$. Since the diagonal entries of D and D^{-1} are positive, they may be further decomposed as $D = D^{\frac{1}{2}}D^{\frac{1}{2}}$ and $D^{-1} = D^{-\frac{1}{2}}D^{-\frac{1}{2}}$ where $D^{\frac{1}{2}}, D^{-\frac{1}{2}} \in \mathbb{R}^{n\times n}$ are diagonal matrices with entries as the square roots of the corresponding entries in D and D^{-1} respectively.

$$q^\top M q p^\top M^{-1} p = q^\top UDU^\top q p^\top UD^{-1}U^\top p \tag{100}$$

$$= q^\top UD^{\frac{1}{2}}D^{\frac{1}{2}}U^\top q p^\top UD^{-\frac{1}{2}}D^{-\frac{1}{2}}U^\top p\ . \tag{101}$$

Define $\hat{q} := D^{\frac{1}{2}}U^\top q$ and $\hat{p} := D^{-\frac{1}{2}}U^\top p$.

$$q^\top UD^{\frac{1}{2}}D^{\frac{1}{2}}U^\top q p^\top UD^{-\frac{1}{2}}D^{-\frac{1}{2}}U^\top p = \hat{q}^\top\hat{q}\hat{p}^\top\hat{p} \tag{102}$$

$$= (\|\hat{q}\|\,\|\hat{p}\|)^2\ . \tag{103}$$

By the Cauchy-Schwarz inequality,

$$(\|\hat{q}\|\,\|\hat{p}\|)^2 \geq (\hat{q}^\top\hat{p})^2 \tag{104}$$

$$= \left(q^\top UD^{\frac{1}{2}}D^{-\frac{1}{2}}U^\top p\right)^2 \tag{105}$$

$$= (q^\top p)^2 \tag{106}$$

$$= \langle q, p\rangle^2\ . \tag{107}$$

Taking the square root on both sides of the inequality completes the proof. □

Lemma 6. *For any two discrete probability vectors $q, p \in \mathbb{R}^n$, $\sum_{i=1}^{n}[q]_i = \sum_{i=1}^{n}[p]_i = 1$,*

$$\|q - p\|^2 \leq 2\chi^2(q, p) \tag{108}$$

where $\chi^2(q, p) := \sum_{i=1}^{n}[p]_i\phi(\frac{[q]_i}{[p]_i})$, $\phi(x) := \frac{1}{2}(x-1)^2$ is the χ^2-divergence.

Proof.

$$\|q-p\|^2 = \langle q-p, q-p\rangle \tag{109}$$
$$= \sum_{i=1}^{n}([q]_i - [p]_i)^2 \ . \tag{110}$$

Since $([q]_i - [p]_i)^2$ and $[p]_i$ are positive for all i,

$$\sum_{i=1}^{n}([q]_i - [p]_i)^2 \le \sum_{i=1}^{n}\frac{1}{[p]_i}([q]_i - [p]_i)^2 \tag{111}$$
$$= \sum_{i=1}^{n}\frac{1}{[p]_i}([q]_i^2 - 2[q]_i[p]_i + [p]_i^2) \tag{112}$$
$$= \sum_{i=1}^{n}\left(\frac{[q]_i^2}{[p]_i} - 2[q]_i + [p]_i\right) \tag{113}$$
$$= \sum_{i=1}^{n}[p]_i\left(\frac{[q]_i^2}{[p]_i^2} - 2\frac{[q]_i}{[p]_i} + 1\right) \tag{114}$$
$$= 2\sum_{i=1}^{n}[p]_i\phi\left(\frac{[q]_i}{[p]_i}\right) \tag{115}$$
$$= 2\chi^2(q,p) \ . \tag{116}$$

□

Lemma 7. *Let $\mathcal{S}_{MMD}(g) := -\sqrt{g^\top M^{-1}g - \frac{(g^\top M^{-1}\mathbf{1})^2}{\mathbf{1}^\top M^{-1}\mathbf{1}}}$ be the MMD worst-case sensitivity from (Staib & Jegelka, 2019), where $g \in \mathbb{R}^n$, $M \in \mathbb{R}^{n\times n}$ is a positive definite matrix and $\mathbf{1} \in \mathbb{R}^n$ has all entries equal to 1. $\mathcal{S}_{MMD}(g)$ is bounded by*

$$-\|g\|_{M^{-1}} \le \mathcal{S}_{MMD}(g) \le 0 \tag{117}$$

where $\|g\|_M := \sqrt{g^\top M g}$.

Proof. From Appendix C in (Staib & Jegelka, 2019), $-\mathcal{S}_{\text{MMD}}(g)$ can be re-written as

$$-\mathcal{S}_{\text{MMD}}(g) = \sqrt{(g-\gamma\mathbf{1})^\top M^{-1}(g-\gamma\mathbf{1})} \tag{118}$$
$$\gamma := \frac{\mathbf{1}^\top M^{-1}g}{\mathbf{1}^\top M^{-1}\mathbf{1}} \ . \tag{119}$$

Since M is positive definite, M^{-1} is also positive definite. The above re-written form thus implies that $-\mathcal{S}_{\text{MMD}}(g) \ge 0$. Since M^{-1} is positive definite, $\mathbf{1}^\top M^{-1}\mathbf{1} \ge 0$ and hence $\frac{(g^\top M^{-1}\mathbf{1})^2}{\mathbf{1}^\top M^{-1}\mathbf{1}} \ge 0$. Together, these imply that

$$g^\top M^{-1}g - \frac{(g^\top M^{-1}\mathbf{1})^2}{\mathbf{1}^\top M^{-1}\mathbf{1}} \le g^\top M^{-1}g \tag{120}$$
$$= \|g\|_{M^{-1}}^2 \ . \tag{121}$$

We thus have the following bounds on the MMD worst-case sensitivity:

$$0 \le -\mathcal{S}_{\text{MMD}}(g) \le \|g\|_{M^{-1}} \tag{122}$$
$$-\|g\|_{M^{-1}} \le \mathcal{S}_{\text{MMD}}(g) \le 0 \ . \tag{123}$$

□

Lemma 8. *For any two discrete probability vectors $q, p \in \mathbb{R}^n$, $\sum_{i=1}^n [q]_i = \sum_{i=1}^n [p]_i = 1$ with support $\mathcal{C}$,*

$$\|q-p\|_1 \le \frac{2}{c_{\text{min}}} \mathcal{W}(q,p) \tag{124}$$

$$c_{\text{min}} := \min_i \min_{j \neq i} \|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}} \tag{125}$$

where $\mathcal{W}(q,p) := \min_{\gamma \in \triangle_{n\times n}} \sum_{i=1}^n \sum_{j=1}^n [\gamma]_{ij} \cdot \|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}}$ is the Wasserstein metric with arbitrary norm $\|\cdot\|_{\mathcal{W}}$, and γ is a discrete joint distribution with support $\mathcal{C} \times \mathcal{C}$ (represented as a $|\mathcal{C}| \times |\mathcal{C}|$ matrix) whose marginals are q and p, i.e., $\sum_{i=1}^{|\mathcal{C}|} [\gamma]_{ij} = [p]_j, \sum_{j=1}^{|\mathcal{C}|} [\gamma]_{ij} = [q]_i$.

Proof. Define $\gamma^* := \text{argmin}_{\gamma \in \triangle_{n\times n}} \sum_{i=1}^n \sum_{j=1}^n [\gamma]_{ij} \cdot \|(\mathcal{C}_i - \mathcal{C}_j)\|_{\mathcal{W}}$ to be the joint distribution that achieves the minimum of the expected cost. For all $1 \le i \le n$,

$$[\gamma^*]_{ii} \le \min([q]_i, [p]_i) \tag{126}$$

since either $\sum_{j=1}^n [\gamma^*]_{ij} > [q]_i$ or $\sum_{i=1}^n [\gamma^*]_{ij} > [p]_i$ otherwise. Since $\max([q]_i, [p]_i) = \min([q]_i, [p]_i) + |[q]_i - [p]_i|$,

$$\sum_{i=1}^n [\gamma^*]_{ii} + |[q]_i - [p]_i| \le \sum_{i=1}^n \max([q]_i, [p]_i). \tag{127}$$

Since $\sum_{i=1}^n \min([q]_i, [p]_i) + \max([q]_i, [p]_i) = 2$,

$$\sum_{i=1}^n [\gamma^*]_{ii} + \max([q]_i, [p]_i) \le 2. \tag{128}$$

Taking (128) - (127),

$$\begin{aligned}
\sum_{i=1}^n [\gamma^*]_{ii} &\le 2 - \left(\sum_{i=1}^n [\gamma^*]_{ii} + |[q]_i - [p]_i| \right) && (129)\\
2\sum_{i=1}^n [\gamma^*]_{ii} &\le 2 - \sum_{i=1}^n |[q]_i - [p]_i| && (130)\\
\|q-p\|_1 &\le 2\left(1 - \sum_{i=1}^n [\gamma^*]_{ii}\right) && (131)\\
&\le 2\left(\sum_{i=1}^n \sum_{j=1, j\neq i}^n [\gamma^*]_{ij}\right) && (132)\\
\|q-p\|_1 \cdot c_{\text{min}} &\le 2\left(\sum_{i=1}^n \sum_{j=1, j\neq i}^n [\gamma^*]_{ij} \cdot c_{\text{min}}\right) && (133)\\
&\le 2\left(\sum_{i=1}^n \sum_{j=1, j\neq i}^n [\gamma^*]_{ij} \cdot \|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}}\right) && (134)\\
&= 2\left(\sum_{i=1}^n \sum_{j=1}^n [\gamma^*]_{ij} \cdot \|\mathcal{C}_i - \mathcal{C}_j\|_{\mathcal{W}}\right) && (135)\\
&= 2\mathcal{W}(q,p) && (136)\\
\|q-p\|_1 &\le \frac{2}{c_{\text{min}}} \mathcal{W}(q,p)\ . && (137)
\end{aligned}$$

□

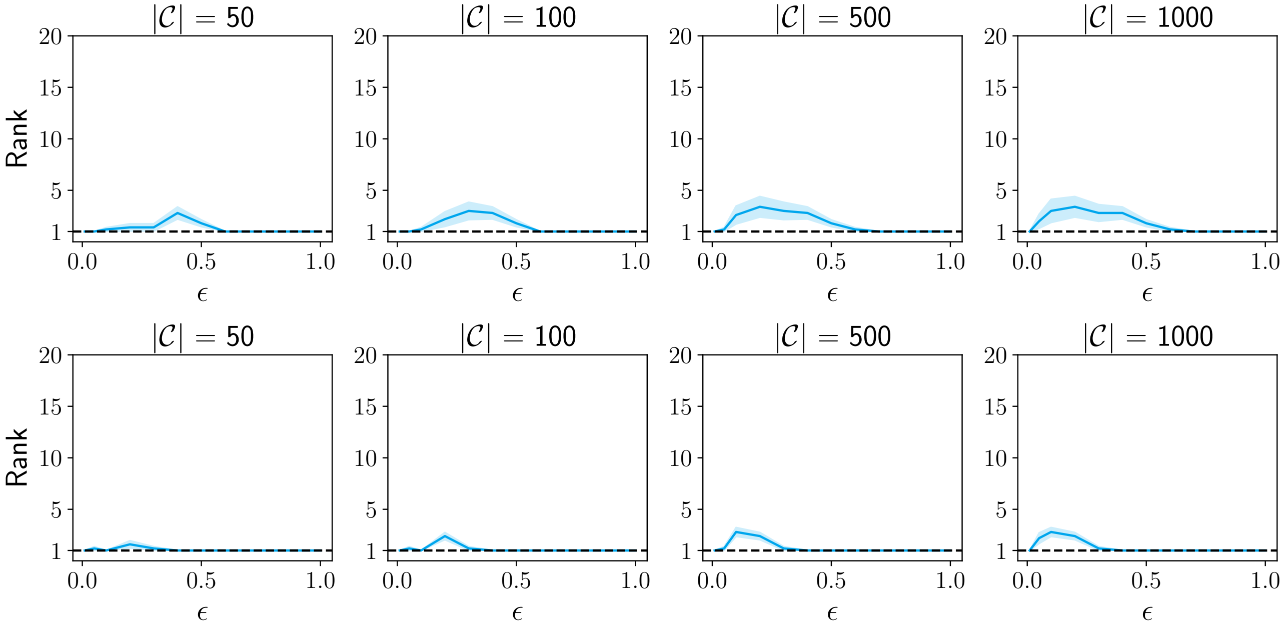


Figure 8. The mean and standard error of the rank of the best action selected by MINIMAXAPPROX with the MMD worst-case sensitivity approximation $\widehat{\mathcal{S}}_{\mathrm{MMD}}(g)$ (4), w.r.t. the ranking of actions induced by MINIMAXAPPROX with the true MMD worst-case sensitivity $\mathcal{S}_{\mathrm{MMD}}(g)$ (1 is lowest, lower is better). The top row shows the results with a reference mean of 0 and the bottom row shows the results with a reference mean of 1. The results are averaged across 5 different 2-D random functions drawn from a GP prior, with 20 actions.

Theorem 4 relies on the following result on a high probability bound on the absolute difference between the GP posterior mean and the true function at any action, context and iteration. This result has been presented in various forms in various works on online optimization (Srinivas et al., 2010; Abbasi-Yadkori, 2012; Chowdhury & Gopalan, 2017). We reproduce the result as stated in (Kirschner et al., 2020):

Lemma 9. *With probability at least $1-\delta$, for any $x \in \mathcal{X}$, $c \in \mathcal{C}$ at any iteration $t \geq 1$,*

$$|\mu_t(x,c) - f(x,c)| \leq \beta_t \sigma_t(x,c) \tag{138}$$

where $\beta_t = \sigma\sqrt{\log\det(I+K_t) + 2\log\left(\frac{1}{\delta}\right)} + B$ and B is the upper bound of the RKHS norm of f.

B. MMD Worst-case Sensitivity Approximation Quality

In this section, we discuss the quality of the worst-case sensitivity approximation $\widehat{\mathcal{S}}_{\mathrm{MMD}}(g)$ (4) which replaces the Gram matrix M in the MMD worst-case sensitivity $\mathcal{S}_{\mathrm{MMD}}(g)$ with the identity matrix, in order to avoid the $\mathcal{O}(|\mathcal{C}|^3)$ time matrix inversion.

Observe that the term $g^\top g = \|g\|^2$ appears in $\widehat{\mathcal{S}}_{\mathrm{MMD}}(g)$. If we use a larger discretization density over the context, the size of the context set $|\mathcal{C}|$ will increase. This causes $\|g\|^2$ to increase unboundedly as $|\mathcal{C}|$ increases, since $g \in \mathbb{R}^{|\mathcal{C}|}$. In the true MMD worst-case sensitivity $\mathcal{S}_{\mathrm{MMD}}(g)$ (Table 2), the corresponding term is a finite-dimensional analog of the RKHS norm squared $g^\top M^{-1} g = \|g\|^2_{M^{-1}}$. We empirically observe that $\|g\|^2_{M^{-1}}$ increases much less as $|\mathcal{C}|$ increases. This occurs because $g^\top M^{-1} g$ becomes a better approximation of the RKHS norm squared (which we assume to be bounded) of the underlying function that produces g as $|\mathcal{C}| \to \infty$. This suggests that the approximation $\widehat{\mathcal{S}}_{\mathrm{MMD}}(g)$ becomes a worse one as $|\mathcal{C}|$ increases, and indeed we empirically observe this effect. This aligns with intuition since the larger $|\mathcal{C}|$ is, the more time we save by avoiding the matrix inversion.

Nevertheless, we observe that the approximation does not degrade performance much in the experiments in Sec. 7 using $\widehat{\mathcal{S}}_{\mathrm{MMD}}(g)$, including the COVID-19 test allocation experiments with a relatively large context set size of $|\mathcal{C}| = 550$.

We hypothesize that, although $\widehat{\mathcal{S}}_{\text{MMD}}(g)$ may be a poor approximation in numerical value, using MINIMAXAPPROX with $\widehat{\mathcal{S}}_{\text{MMD}}(g)$ preserves the ranking of actions induced by using MINIMAXAPPROX with $\mathcal{S}_{\text{MMD}}(g)$, i.e., a 'good' action according to $\mathcal{S}_{\text{MMD}}(g)$ is also a 'good' action according to $\widehat{\mathcal{S}}_{\text{MMD}}(g)$. Figure 8 shows the mean and standard error of rank of the best action selected by MINIMAXAPPROX with $\widehat{\mathcal{S}}_{\text{MMD}}(g)$, w.r.t. the ranking of actions induced by MINIMAXAPPROX with $\mathcal{S}_{\text{MMD}}(g)$, averaged over 5 different 2-D random functions drawn from a GP prior. We observe that MINIMAXAPPROX with $\widehat{\mathcal{S}}_{\text{MMD}}(g)$ tends to select the best action when ϵ is small or large, since MINIMAXAPPROX is most accurate at those values of ϵ. As $|\mathcal{C}|$ increases, the range of ϵ at which MINIMAXAPPROX with $\widehat{\mathcal{S}}_{\text{MMD}}(g)$ picks suboptimal actions (i.e. actions with rank higher than 1) increases, and the average rank appears to increase as well, which is explained by the poorer approximation as $|\mathcal{C}|$ increases. Nevertheless, we do observe that MINIMAXAPPROX with $\widehat{\mathcal{S}}_{\text{MMD}}(g)$ generally picks high ranking actions which empirically supports our hypothesis that MINIMAXAPPROX with $\widehat{\mathcal{S}}_{\text{MMD}}(g)$ approximately preserves the ranking of actions induced by using MINIMAXAPPROX with $\mathcal{S}_{\text{MMD}}(g)$ and thus leads to good performance. Other tractable approximations of M, such as with block-diagonal or banded matrices, are possible as well and the quality of such approximations is left to future work.

C. Practical Considerations for Distribution Distance Selection

In this section, we discuss some practical considerations when selecting d for an application and give examples with popular distances, namely the Wasserstein metric, MMD, TV and χ^2.

Computational considerations. For example, general ϕ-divergences of the form $\sum_{i=1}^{\mathcal{C}}[p]_i\phi([q]_i/[p]_i)$ are undefined when $\text{supp}\,(p) \nsubseteq \text{supp}\,(q)$, which is likely to cause issues if p is an empirical distribution. DRO with the Wasserstein distance involves solving a linear optimization problem with variable size $|\mathcal{C}|^2$ which has time complexity greater than $\mathcal{O}(|\mathcal{C}|^4)$ and is intractable for large context sets. The MMD constraint has a full-rank, dense Hessian which precludes efficient methods of solving the KKT system with interior-point methods for convex optimization (Boyd & Vandenberghe, 2004), compared to the TV and χ^2 constraints have either 0 or a diagonal Hessian and hence can be solved relatively efficiently.

Modelling correlations between contexts. For example, the Wasserstein distance requires a metric between context points and (informally) models the cost to turn one probability distribution into another using the metric as the cost. MMD uses a kernel to model the correlations between context points. TV and χ^2 do not model the correlations between points. For example, consider the distribution that has almost all probability mass on $\mathcal{C}_j$, and a small probability mass distributed evenly among all other context points, denoted δ_j. $\text{TV}(\delta_j, \delta_{j+1}) = \text{TV}(\delta_j, \delta_{j+i}) \forall i, 1-j \leq i \leq |\mathcal{C}| - j$. Similarly, $\chi^2(\delta_j, \delta_{j+1}) = \chi^2(\delta_j, \delta_{j+i}) \forall i, 1-j \leq i \leq |\mathcal{C}| - j$. This may be undesirable for a particular application: For example, if the context were average temperature, a distribution of this form with most of the mass on $25.1°C$ should certainly be close to one with most of the mass on $25.2°C$ and not to one with most of the mass on $35°C$, yet TV and χ^2 would give the same distance.

D. Experiment Details

In all experiments (except Computation Time), $\beta_t = 2\,\forall t \in [T]$, $\sigma^2 = 0.001$. We use the ECOS convex optimization solver (Domahidi et al., 2013) was used for the EXACT acquisition function and computing the true robust solution. If ECOS fails for any reason, we use the SCS convex optimization solver (O'Donoghue et al., 2016) instead.

D.1. Random Functions

The 2-D random functions are drawn from a GP prior using an ARD squared exponential kernel with lengthscale 0.05 in each dimension. The domain of each function is $[0, 1]^2$ and each dimension uses a discretization density of 20. The reference distributions were $\mathcal{N}(0, 0.02)$ and $\mathcal{N}(0.5, 0.02)$ truncated outside the domain and normalized. The true distribution was the uniform distribution. The modelling GP uses the ground-truth kernel and observational variance.

D.2. Plant Maximum Leaf Area

The domain of the function is $[0, 1]^2$, normalized from the objective function's true domain of $[2.5, 6.5]$ for pH and $[0, 30]$ for ammonium in mM. Each dimension uses a discretization density of 50. The reference distributions were $\mathcal{N}(0, 0.02)$ and $\mathcal{N}(1, 0.02)$ truncated outside the domain and normalized. The true distribution was the uniform distribution. The modelling GP uses an ARD squared exponential kernel with lengthscale 0.1 in each dimension and the ground-truth observational variance. The objective function values were normalized by subtracting the data mean and dividing by the data standard deviation.

D.3. Wind Power Dataset

The domain of the function is $[0, 1]^2$. The actual power in a month is then normalized to this range. Each dimension uses a discretization density of 50. The modelling GP uses an ARD squared exponential kernel with lengthscale 0.1 in each dimension and the ground-truth observational variance.

D.4. COVID-19 Test Allocation

This objective function uses the simulator from (Cashore et al., 2020b) which was used to study group testing protocols for the reopening of Cornell University (Cashore et al., 2020a) and for the general U.S. population (Frazier et al., 2020).

The domain of the function is $[0, 1]^5$. Each dimension uses a discretization density of 10. The first two dimensions is the action variable x determining the proportion of tests to be allocated to the first and second populations respectively, and the remaining proportion is allocated to the third. Hence, $[x]_1 + [x]_2 \leq 1$ and the invalid actions are removed from the domain, leading to an action set of size 55. The remaining three dimensions is the context variable c. The first two dimensions of the context are the proportions of COVID-19 initial cases allocated to the first and second populations respectively, and the remaining proportion is allocated to the third. Hence, $[c]_1 + [c]_2 \leq 1$ and the invalid contexts are removed from the domain. The last dimension of the context determines the transmission probability given by $0.1[c]_3 + 0.1$. The context set has size 550. Each population has size 10000, there are 5000 tests and 1250 initial COVID-19 cases. Instead of running the simulator for every query, the objective function was constructed with kernel ridge regression on data collected by running the simulator on inputs in a Sobol sequence over the domain. For the full list of simulator hyperparameters, refer to the code repository. The reference distributions were $\mathcal{N}((0, 0, 0)^\top, 0.02I)$ and $\mathcal{N}((1, 0, 0)^\top, 0.02I)$ truncated outside the domain and normalized. The modelling GP uses an ARD squared exponential kernel with lengthscale 0.2 in each dimension and the ground-truth observational variance. The objective function values were normalized by subtracting the data mean and dividing by the data standard deviation.

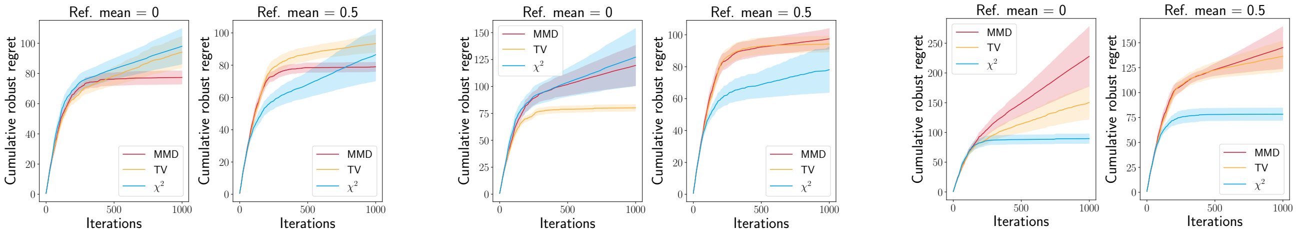
D.5. Computation Time and Time-accuracy Trade-off

All measured CPU times were averaged across 20 actions. The SCS convex optimization solver (O'Donoghue et al., 2016) was used to construct the Pareto frontier for truncated convex optimization.

D.6. Implementation

The experiments were implemented in Python, and the major packages used were NumPy (Harris et al., 2020), GPflow (Matthews et al., 2017) and CVXPY (Diamond & Boyd, 2016; Agrawal et al., 2018). For a full list of packages and versions, refer to the code repository.

E. EXACT with Wrong Distances



(a) d = MMD

(b) d = TV

(c) $d = \chi^2$

Figure 9. **Random functions from GP prior**: Mean and standard error of cumulative robust regret (lower is better) for Algo. 1 utilizing EXACT worst-case expected value (1) (Kirschner et al., 2020) with different distribution distances d and means of reference distribution. The distribution distance d in the sub-caption indicates the true distribution distance with which the cumulative robust regret is calculated.

To test the hypothesis that using the wrong distance leads to sub-optimal performance when the robust regret is calculated using some true distance, we run Algo. 1 with EXACT on 2-D random functions drawn from a GP prior. Fig. 9 shows that using the true distance largely leads to lower cumulative robust regret, compared to when the wrong distance is used. While the cumulative robust regret of χ^2 is lower than that of TV when TV is the true distance, the immediate regret of TV (averaged over the last 5 iterations) is 0, indicating that in all random functions the algorithm converges to the robust action at the end when the true distance is used, while that of χ^2 is non-zero, indicating that the algorithm does not converge to the robust action at the end when the wrong distance is used. We observe this effect for all true distances: When the true distance is used, the immediate regret is 0 at the end, while it is non-zero when wrong distances are used. This experiment illustrates the importance of choosing the right distance when there is some true (possibly unknown) objective that is the most suitable for a particular application.