
Feasible Arm Identification

Julian Katz-Samuels¹ Clayton Scott¹

Abstract

We introduce the feasible arm identification problem, a pure exploration multi-armed bandit problem where the agent is given a set of D -dimensional arms and a polyhedron $P = \{x : Ax \leq b\} \subset \mathbb{R}^D$. Pulling an arm gives a random vector and the goal is to determine, using a fixed budget of T pulls, which of the arms have means belonging to P . We propose three algorithms MD-UCBE, MD-SAR, and MD-APT and provide a unified analysis establishing upper bounds for each of them. We also establish a lower bound that matches up to constants the upper bounds of MD-UCBE and MD-APT. Finally, we demonstrate the effectiveness of our algorithms on synthetic and real-world datasets.

1. Introduction

Pure exploration multi-armed bandit (MAB) problems provide a framework for determining via a sequential experiment which of a set of distributions meet some criteria. In this setting, there are K distributions ν_1, \dots, ν_K and the agent sequentially chooses from which distribution to sample an observation. At the end of the sampling stage, the agent outputs the distributions which he believes meet the desired criteria and the performance of the agent is measured based on the quality of this decision. In the MAB literature, distributions are also referred to as arms, and sampling a realization from a distribution ν_i is referred to as pulling arm i . The most well-studied of these problems is top- k arm identification. In this problem, the goal is to find the k best arms, that is, k arms with the largest means. This problem and other pure exploration problems have applications in a wide range of areas, including crowdsourcing, A/B testing, and online advertising.

In many application domains, the arms and the criteria for

¹Department of Computer Science and Electrical Engineering, University of Michigan. Correspondence to: Julian Katz-Samuels <jkatzsam@umich.edu>.

a good arm are multi-dimensional in nature. For example, in crowdsourcing it is important to distinguish good workers from bad workers. For a multilabel classification task (where examples are associated with multiple labels), a worker can be modeled as a multi-dimensional arm where each dimension corresponds to her accuracy at identifying a particular label, and a natural definition for a “good worker” is that her accuracy is above some threshold for each label (e.g., 90%). A common approach for finding such workers is to use a collection of examples labeled by domain experts as a set of tests. Since workers are paid for each example that they label, often an organization is only willing to spend a limited number of queries to find good workers and an effective method under this budget constraint is needed.

As another example, consider A/B testing for designing products such as websites, ads, and video games. In this setting, there are several options for a product and a company diverts traffic to each of the options to determine which one to choose. Multi-dimensional criteria arise naturally in this domain, as well. For example, a company that wants to grow its user base for its website might desire the rate of new subscriptions to be above some level, while still maintaining a certain level of user retention among its current users. If the product is a video game, the company might also be interested in maintaining some metric of user engagement above a certain threshold.

The pure exploration MAB literature lacks (i) a simple framework for describing problems where the arms and criteria are multi-dimensional and (ii) practical algorithms for addressing these problems. In this paper, we aim to address this gap. We introduce the *feasible arm identification problem* in which arms are associated with multi-dimensional distributions and the goal is to find arms whose means belong to a given polyhedron¹ $P = \{x : Ax \leq b\}$. Polyhedra encompass a large class of regions that can model common user-defined constraints, including thresholds or ranges on individual dimensions and linear constraints involving multiple dimensions. We propose several algorithms for the fixed budget setting and provide upper and lower bounds. Finally, we demonstrate through experiments on synthetic and real-world datasets that by leveraging the geometry of the

¹There are several conflicting definitions of polyhedra. We define a polyhedron as the intersection of a finite number of closed halfspaces (Boyd and Vandenberghe, 2004).

problem, our methods significantly outperform a uniform allocation strategy. Indeed, in several of our experiments, our methods find the feasible arms with a probability that is a factor of 10 better than that of a uniform allocation strategy. All proofs are contained in the supplementary material.

2. Related Work

MABs have received a significant amount of attention. Most work considers minimizing the cumulative regret instead of a pure exploration objective. There have been relatively few works on multi-dimensional arms and criteria in this regime (Drugan and Now, 2013; Busa-Fekete et al., 2017; Tekin and Turgay, 2017). Drugan and Now (2013) modify a UCB algorithm to find all arms on the Pareto front. Busa-Fekete et al. (2017) use the Generalized Gini Index to optimize all objectives in a fair way. Tekin and Turgay (2017) consider a contextual MAB setting where the goal is to maximize the total reward in a non-dominant objective, subject to the constraint that the total reward in a dominant objective is maximized. These works differ from our work in that (i) they consider the cumulative regret setting, which is fundamentally different from the pure exploration setting (Bubeck et al., 2009), and (ii) they aim to either balance multiple objective functions or find arms on the Pareto front, whereas we aim to find feasible arms, where feasibility is defined by membership in a given polyhedron.

In recent years, there have been many advances in pure exploration MABs in the fixed confidence and fixed budget settings (Mannor and Tistisklis, 2004; Gabillon et al., 2012; Bubeck et al., 2013; Chen et al., 2014; Jamieson et al., 2014). A limited number of works have considered multi-dimensional feedback. Auer et al. (2016) considered a variant of the top arm identification problem where arms are multi-dimensional with each dimension corresponding to a distinct objective that an agent wishes to optimize, and the goal is to identify the Pareto front of the arms. In contrast to our work, they consider the fixed confidence setting. More importantly, Pareto front identification and feasible arm identification are mathematically very different problems and apply to distinct situations. Whereas Pareto front identification is relevant to multi-objective optimization, the feasible arm identification problem is useful for situations where there are user-defined criteria for what qualifies as a good arm.

Chen et al. (2017) recently proposed the *general sampling problem*, which can model a setting where arms are multi-dimensional and the goal is to find arms with means belonging to a given polyhedron. There are several major differences with our work. First, Chen et al. (2017) do not consider multi-dimensional feedback as the agent can sample from one dimension of one arm at a time. Second, whereas they study the fixed confidence setting, we

study the fixed budget setting. Third, they assume isotropic Gaussian arms, whereas we assume each arm is associated with a multi-dimensional sub-Gaussian distribution. Finally, their proposed algorithm (see their Algorithm 7) is sample-inefficient and impractical since in its first stage, it employs a uniform allocation strategy until the confidence bounds (defined with $\delta = 0.01$) of all of the means either intersect with the given polyhedron or do not intersect with the given polyhedron.

Locatelli et al. (2016) introduced the thresholding bandit problem (TBP), which is essentially the scalar version of the feasible arm identification problem, and the algorithm APT. In TBP, there are K scalar-valued distributions, a threshold τ , and a budget T . The goal is to identify all of the distributions with means above τ . Our work significantly generalizes TBP by considering multi-dimensional arms and the problem of identifying those arms with means belonging to a given polyhedron. Unlike Locatelli et al. (2016) who only analyze APT, we provide a unified analysis of three algorithms for the feasible arm identification problem. One of our algorithm, MD-APT, reduces to APT in the one-dimensional thresholding case and our upper bound also reduces to the upper bound of APT (up to constant factors). To deal with this general setting, we introduce a novel complexity measure that characterizes the hardness of determining whether an arm is in P . This measure is essentially the distance of the mean of an arm to the boundary of the polyhedron. In addition, our general setting introduces technical challenges for establishing upper and lower bounds. We overcome these by using tools from convex analysis, properties of multi-dimensional sub-Gaussian distributions, and change-of-measure arguments involving multi-dimensional distributions.

Recently, Zheng et al. (2017) considered a problem with a polyhedral constraint, but their setup is very different from our own. In their setting, the goal is to solve a linear program where either the constraints are not fully known or the cost function is not fully known but can be estimated by adaptive sampling. In our work, the constraints are known and we wish to learn which out of a collection of distributions have feasible means.

3. Setup

In this section, we formalize the feasible arm identification problem. To begin, we define some notation. For all $n \in \mathbb{N}$, let $[n] = \{1, \dots, n\}$. For any $\mathbf{x} \in \mathbb{R}^D$ and $A \subset \mathbb{R}^D$, let $\text{dist}(\mathbf{x}, A) = \inf_{\mathbf{y} \in A} \|\mathbf{x} - \mathbf{y}\|_2$. Let $\mathbf{1} = (1, \dots, 1)^t \in \mathbb{R}^D$ and $\mathbf{1}\{\cdot\}$ denote the indicator function. Define $\mathcal{S}^{D-1} = \{\mathbf{x} \in \mathbb{R}^D : \|\mathbf{x}\|_2 = 1\}$.

Suppose we are given K stochastic arms. When the i th arm is pulled, a reward is drawn i.i.d. from a D -dimensional

distribution ν_i . Denote $\boldsymbol{\mu}_i = \mathbb{E}_{\mathbf{X} \sim \nu_i} \mathbf{X}$. We assume that the agent is given a polyhedron $P = \{\mathbf{x} : A\mathbf{x} \leq \mathbf{b}\}$ where $A \in \mathbb{R}^{M \times D}$ and $\mathbf{b} \in \mathbb{R}^M$. Let \mathbf{a}_j^t denote the j th row of A . By dividing each constraint by $\|\mathbf{a}_j\|_2$, we can assume without loss of generality that $\|\mathbf{a}_j\|_2 = 1$ for all $j \in [M]$. Let ∂P denote the boundary of P , i.e., $\partial P = \bar{P} \setminus P^\circ$ where \bar{P} denotes the closure of P and P° denotes the interior of P . For simplicity, we assume that P has positive volume.

We consider the fixed budget setting. The game is as follows: there are T rounds and at each round t , the agent chooses an arm $I_t \in [K]$ and observes a realization $\mathbf{X}_t \sim \nu_{I_t}$. The goal is to identify all of the arms whose means belong to the polyhedron. To define a performance measure, let $\epsilon > 0$ denote the tolerance, and define $S_{P,\epsilon}^{\text{int}} := \{i \in [K] : \boldsymbol{\mu}_i \in P \text{ and } \text{dist}(\boldsymbol{\mu}_i, \partial P) \geq \epsilon\}$ and $S_{P,\epsilon}^{\text{out}} := \{i \in [K] : \text{dist}(\boldsymbol{\mu}_i, P) > \epsilon\}$. $S_{P,\epsilon}^{\text{int}}$ is the set of arms that lie in the interior of P by at least ϵ and $S_{P,\epsilon}^{\text{out}}$ is the set of arms that lie outside of P by at least ϵ . Let $\hat{S} \subset [K]$ denote the set of arms outputted by an algorithm. We define the following error measure:

$$\mathcal{L}_{T,P,\epsilon}(\hat{S}) := \mathbf{1}\{\hat{S} \cap S_{P,\epsilon}^{\text{out}} \neq \emptyset \vee \hat{S}^c \cap S_{P,\epsilon}^{\text{int}} \neq \emptyset\}$$

In words, the goal is to identify all of the arms with means belonging to the polyhedron up to tolerance ϵ in the sense that an algorithm is successful if its output includes every arm i such that $\boldsymbol{\mu}_i \in P$ and $\text{dist}(\boldsymbol{\mu}_i, \partial P) \geq \epsilon$ and excludes every arm l such that $\text{dist}(\boldsymbol{\mu}_l, P) > \epsilon$.

We define the *margin* of arm i as

$$\begin{aligned} \Delta_i^{(P,\epsilon)} &:= \text{dist}(\boldsymbol{\mu}_i, \partial P) + \epsilon \\ &= \begin{cases} \min_{j \in [M]} \text{dist}(\boldsymbol{\mu}_i, \{\mathbf{x} : \mathbf{a}_j^t \mathbf{x} = b_j\}) + \epsilon & : \boldsymbol{\mu}_i \in P \\ \text{dist}(\boldsymbol{\mu}_i, P) + \epsilon & : \boldsymbol{\mu}_i \notin P \end{cases} \end{aligned} \quad (1)$$

$$= \begin{cases} \min_{j \in [M]} b_j - \mathbf{a}_j^t \boldsymbol{\mu}_i + \epsilon & : \boldsymbol{\mu}_i \in P \\ \text{dist}(\boldsymbol{\mu}_i, P) + \epsilon & : \boldsymbol{\mu}_i \notin P \end{cases} \quad (2)$$

where line (1) follows by Lemma H.1 and line (2) follows by the closed form solution of the distance from a point to a hyperplane and $\|\mathbf{a}_j\|_2 = 1$ (Boyd and Vandenberghe, 2004).

The *complexity* of an instance of the feasible arm identification problem is defined to be:

$$H_{P,\epsilon} := \sum_{i \in [K]} [\Delta_i^{(P,\epsilon)}]^{-2}.$$

In words, an instance has low complexity if all of the arms are far from the boundary of the polyhedron and high complexity if some of the arms are very close to the boundary. The intuition behind this complexity measure is that for an algorithm to output the correct answer about arm i , it is sufficient to guarantee that an estimate $\hat{\boldsymbol{\mu}}_i$ is within a ball centered at $\boldsymbol{\mu}_i$ with radius $\frac{\Delta_i^{(P,\epsilon)}}{2}$ (see Lemma 3). For

the sake of brevity, we usually write $\mathcal{L}_{T,\epsilon}(\hat{S})$, $\Delta_i^{(\epsilon)}$, and H instead of $\mathcal{L}_{T,P,\epsilon}(\hat{S})$, $\Delta_i^{(P,\epsilon)}$, and $H_{P,\epsilon}$, respectively.

Our analysis assumes that each ν_i is a multi-dimensional sub-Gaussian distribution, which we now define (see Vershynin et al. (2017) for more details). Let X be a scalar random variable. We say that X is *R-sub-Gaussian* if $\mathbb{E} \exp(\frac{X^2}{R^2}) \leq 2$. We define the sub-Gaussian norm of X as the smallest R that satisfies the above requirement:

$$\|X\|_{\psi_2} = \inf\{R > 0 : \mathbb{E} \exp(\frac{X^2}{R^2}) \leq 2\}.$$

A random vector $\mathbf{X} \in \mathbb{R}^D$ is sub-Gaussian if $\mathbf{X}^t \mathbf{a}$ is sub-Gaussian for all $\mathbf{a} \in \mathbb{R}^D$. The sub-Gaussian norm of \mathbf{X} is defined as

$$\|\mathbf{X}\|_{\psi_2} = \sup_{\mathbf{a} \in S^{D-1}} \|\mathbf{X}^t \mathbf{a}\|_{\psi_2}.$$

We say that a random vector \mathbf{X} is *R-sub-Gaussian* if $\|\mathbf{X}\|_{\psi_2} \leq R$. Henceforth, we assume that ν_1, \dots, ν_K are *R-sub-Gaussian*. See Vershynin (2012) for a discussion of sub-Gaussian distributions.

4. Lower Bound

In this section, we establish a lower bound for the feasible arm identification problem. Our construction takes any polyhedron P and means $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K \in P^\circ$ and produces a collection of problems such that any algorithm makes a mistake on one of the problems with probability at least on the order of $\exp(-c\frac{T}{H})$ (where c is a constant). In fact, this lower bound holds even when the algorithm is given the distance of each arm to the boundary of the polyhedron. If $A \subset \mathbb{R}^D$ is closed and $\mathbf{x} \in \mathbb{R}^D$, let $\text{Proj}_A(\mathbf{x})$ denote the projection of \mathbf{x} onto A .

Theorem 1. *Let $P = \{\mathbf{x} \in \mathbb{R}^D : A\mathbf{x} \leq \mathbf{b}\}$ have positive volume and $\epsilon \geq 0$ such that $P_\epsilon^\circ := \{\mathbf{x} \in P : \text{dist}(\mathbf{x}, \partial P) > \epsilon\}$ is nonempty. Let $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K \in P_\epsilon^\circ$, $\boldsymbol{\tau}_i \in \text{Proj}_{\partial P}(\boldsymbol{\mu}_i)$ for all $i \in [K]$, and $\boldsymbol{\mu}'_i = \boldsymbol{\mu}_i + 2(\boldsymbol{\tau}_i - \boldsymbol{\mu}_i)$ for all $i \in [K]$. Let ν_i denote the distribution $N(\boldsymbol{\mu}_i, I)$ and ν'_i the distribution $N(\boldsymbol{\mu}'_i, I)$. Let \mathcal{B}^0 denote the product distribution $\nu_1 \otimes \dots \otimes \nu_K$ and \mathcal{B}^i denote the product distribution*

$$\nu_1 \otimes \dots \otimes \nu_{i-1} \otimes \nu'_i \otimes \nu_{i+1} \otimes \dots \otimes \nu_K.$$

Then, $\mathcal{B}^0, \dots, \mathcal{B}^K$ have the same problem complexity

$$H = \sum_{i=1}^K [\text{dist}(\boldsymbol{\mu}_i, \partial P) + \epsilon]^{-2}$$

and for any algorithm,

$$\begin{aligned} & \max_{i \in \{0, \dots, K\}} \mathbb{E}_{\mathcal{B}^i}(\mathcal{L}_{T,\epsilon}(\hat{S})) \\ & \geq \exp(-13\frac{T}{H} - 25D \log(48(\log(T) + 1)KD)). \end{aligned}$$

This lower bound is equal to the lower bound of [Locatelli et al. \(2016\)](#) (see their Theorem 1) up to the factor of D and constants. Since $D \log((\log(T) + 1)KD)$ grows very slowly as a function of T in comparison with $\frac{T}{\tilde{H}}$, the dependence on D is quite mild. We also note that the lower bound does not depend on the number of constraints M in the polyhedron P , which suggests that the number of constraints of P does not directly affect the statistical difficulty of the feasible arm identification problem. Since polyhedra approximate convex sets arbitrarily well, the independence of our lower bound from M enables us to derive a nearly identical lower bound for the setting where P is convex (see the supplementary material for details).

The proof of Theorem 1 is based on a novel lower bound construction with multidimensional distributions for MABs. Often, lower bounds in the bandit literature modify scalar distributions and the main idea is to perturb the mean of a scalar distribution by making it either larger or smaller. In the feasible arm identification problem, picking a direction to perturb the mean of a distribution is not so simple. Indeed, the direction depends on the polyhedron since for some polyhedra, changing the first coordinate does not produce points lying outside of the polyhedron. In our construction, we interchange a distribution ν_i with mean $\mu_i \in P^\circ$ with a distribution ν'_i with mean μ'_i that is shifted away from μ_i in the direction of its projection onto the boundary of P .

Theorem 1 also implies the following non-asymptotic minimax bound.

Corollary 1. *Let $P = \{x \in \mathbb{R}^D : Ax \leq b\}$ have positive volume, $\epsilon \geq 0$ such that P_ϵ° is nonempty, and $R > 0$. Let $\tilde{H} > 0$ such that there exists $\mu_1, \dots, \mu_K \in P_\epsilon^\circ$ with*

$$\tilde{H} = \sum_{i=1}^K [\text{dist}(\mu_i, \partial P) + \epsilon]^{-2}.$$

Let $\mathcal{B}_{P,\epsilon,\tilde{H},R}$ denote the set of feasible arm identification problems on polyhedron P , with tolerance ϵ , and with K arms such that the distributions are R -sub-Gaussian and the problem complexity is less than \tilde{H} . Then, $T \geq 25\tilde{H}R^2D \log(48(\log(T) + 1)KD)$ implies that, for any algorithm,

$$\sup_{\mathcal{B} \in \mathcal{B}_{P,\epsilon,\tilde{H},R}} \mathbb{E}_{\mathcal{B}}(\mathcal{L}_{T,\epsilon}(\hat{S})) \geq \exp(-14 \frac{T}{\tilde{H}R^2}).$$

In words, this result says essentially that for any polyhedron P and tolerance $\epsilon \geq 0$, the induced class of feasible arm identification problems with P and ϵ has a minimax lower bound on the order of $\exp(-c \frac{T}{\tilde{H}R^2})$ where c is a constant. Henceforth, we say that an algorithm is *nearly optimal* if for large enough T its expected loss decays as $\mathcal{O}(\exp(-c \frac{T}{\tilde{H}R^2}))$ where c is a constant.

5. Algorithms

In this section, we extend three algorithms to the feasible arm identification problem, namely, an upper confidence bound based algorithm (UCBE) ([Audibert and Bubeck, 2010](#)), a successive accepts and rejects algorithm (SAR) ([Bubeck et al., 2013](#); [Chen et al., 2014](#)), and the Any-time Parameter-free Thresholding algorithm (APT) ([Locatelli et al., 2016](#)). The main novelty of our approach is that our algorithms estimate the distance of the mean of each arm to the boundary of the polyhedron to decide which arm to pull. To begin, we introduce some notation. Let I_t denote the index of the arm chosen at time t . Let $X_{i,j,t}$ denote the t th realization of the j th coordinate of ν_i , $T_i(t) = \sum_{s=1}^{t-1} \mathbf{1}\{I_s = i\}$ denote the number of pulls of arm i at round t , and $\hat{\mu}_{i,t}$ denote the estimate of μ_i after t samples, i.e., $\hat{\mu}_{i,t} = (\hat{\mu}_{i,1,t}, \dots, \hat{\mu}_{i,D,t})^t$ where $\hat{\mu}_{i,j,t} = \frac{1}{t} \sum_{s=1}^t X_{i,j,s}$.

The key quantity in each of these algorithms is the following empirical estimator of the margin of each arm:

$$\hat{\Delta}_{i,t}^{(\epsilon)} = \begin{cases} \min_{j \in [M]} b_j - \mathbf{a}_j^t \hat{\mu}_{i,t} + \epsilon & : \hat{\mu}_{i,t} \in P \\ \text{dist}(\hat{\mu}_{i,t}, P) + \epsilon & : \hat{\mu}_{i,t} \notin P \end{cases}$$

Given $\hat{\mu}_{i,t}$, $\text{dist}(\hat{\mu}_{i,t}, P)$ can be computed by solving a quadratic program and, thus, the interior point method can compute $\hat{\Delta}_{i,t}^{(\epsilon)}$ in runtime polynomial in M and D . Each of our algorithms updates one $\hat{\Delta}_{i,t}^{(\epsilon)}$ in each round, thus solving at most T quadratic programs. Therefore, each algorithm can be implemented efficiently.

Algorithm 1 MD-UCBE: Multi-dimensional Upper Confidence Bound Exploration algorithm

- 1: **Input:** K arms, polyhedron P , tolerance ϵ , budget T , hyperparameter a
 - 2: **for** $t = 1, \dots, T$ **do**
 - 3: **if** $t \leq K$ **then**
 - 4: Sample $\mathbf{X}_t \sim \nu_t$.
 - 5: **else**
 - 6: Choose $I_t = \arg \min_i \hat{\Delta}_{i,T_i(t)}^{(\epsilon)} - \sqrt{\frac{a}{T_i(t)}}$ and sample $\mathbf{X}_t \sim \nu_{I_t}$.
 - 7: **end if**
 - 8: **end for**
 - 9: **Return:** $\hat{S} = \{i \in [K] : \hat{\mu}_{i,T_i(T+1)} \in P\}$
-

Next, we describe each of the algorithms and our results. Each algorithm outputs $\hat{S} = \{i \in [K] : \hat{\mu}_{i,T_i(T+1)} \in P\}$. The algorithms differ in how they decide which arm to pull. MD-UCBE (Algorithm 1) is a modification of the algorithm UCBE from [Audibert and Bubeck \(2010\)](#). At each time step t , it pulls an arm i that minimizes $\hat{\Delta}_{i,T_i(t)}^{(\epsilon)} - \sqrt{\frac{a}{T_i(t)}}$ breaking ties arbitrarily where a is a hyperparameter. Theorem 2 gives an upper bound on its expected loss.

Theorem 2. Let $K \geq 0, T \geq K$ and $\epsilon \geq 0$. Suppose $0 \leq a \leq \frac{25}{36} \frac{T-K}{H}$. Then, the expected loss of MD-UCBE satisfies:

$$\mathbb{E}[\mathcal{L}_{T,\epsilon}(\hat{S})] \leq 2(\log(T) + 1)K5^D \exp\left(-\frac{a}{1600R^2}\right).$$

Paralleling our upper bounds for MD-SAR and MD-APT, this result says that the degree of difficulty of a problem for MD-UCBE depends on H , i.e., the distance of the arms to the boundary of the polyhedron and the tolerance parameter ϵ . Theorem 2 suggests setting $a = \frac{25}{36} \frac{T-K}{H}$, in which case MD-UCBE is nearly optimal. One important shortcoming of this algorithm is that H is not known in practice, so it is unclear how to set the hyperparameter a . Indeed, in our experiments, we show that the performance of MD-UCBE is highly sensitive to the selection of a .

Algorithm 2 MD-SAR: Multi-dimensional Successive Accepts and Rejects algorithm

-
- 1: **Input:** K arms, polyhedron P , tolerance ϵ , budget T
 - 2: $\log(x) = \frac{1}{2} + \sum_{i=2}^x \frac{1}{i}, n_0 = 0, n_k = \left\lceil \frac{T-K}{\log(K)(K+1-k)} \right\rceil$
($k > 1$)
 - 3: $Q = [K]$
 - 4: **for** $k = 1, \dots, K-1$ **do**
 - 5: Query $n_k - n_{k-1}$ samples from all arms $i \in Q$
 - 6: $Q \leftarrow Q \setminus \arg \max_{i \in Q} \hat{\Delta}_{i,n_k}^{(\epsilon)}$
 - 7: **end for**
 - 8: **Return:** $\hat{S} = \{i \in [K] : \hat{\mu}_{i,T_i(T+1)} \in P\}$
-

MD-SAR (Algorithm 2) extends the SAR algorithm from Bubeck et al. (2013). It divides the budget T into $K-1$ rounds. In each round, it samples all of the arms belonging to $Q \subset [K]$ the same number of times. At the end of each round, it removes from Q an arm i that maximizes $\hat{\Delta}_{i,T_i(t)}^{(\epsilon)}$. Intuitively, MD-SAR stops sampling from an arm i for which there is the least amount of uncertainty about whether $\mu_i \in P$. Theorem 3 provides an upper bound on the expected loss of MD-SAR. It depends on a different complexity term that is nevertheless related to H . Let (i) denote the index of the arm with the i th smallest margin so that $\Delta_{(1)}^{(\epsilon)} \leq \Delta_{(2)}^{(\epsilon)} \leq \dots \leq \Delta_{(K)}^{(\epsilon)}$ and define the complexity parameter

$$H_2 = \max_{i \in [K]} i[\Delta_{(i)}^{(\epsilon)}]^{-2}.$$

The analysis of Audibert and Bubeck (2010) of the analogous quantities immediately implies that $H_2 \leq H \leq \log(2K)H_2$.

Theorem 3. Let $K \geq 0, T \geq K$ and $\epsilon \geq 0$. Then, the

expected loss of MD-SAR satisfies: $\mathbb{E}[\mathcal{L}_{T,\epsilon}(\hat{S})] \leq$

$$2(\log(T) + 1)K5^D \exp\left(-\frac{T-K}{1296 \log(2K)H_2 R^2}\right) + 4K^3 5^D \exp\left(-\frac{T-K}{512R^2 H_2}\right).$$

Similar to previous results on SAR-type algorithms in the fixed budget setting (Audibert and Bubeck, 2010; Chen et al., 2014), our upper bound on MD-SAR is loose by a factor of $\log(K)$ in the exponential. While the guarantee is not tight, it has the significant practical advantage over MD-UCBE that it does not involve a difficult-to-tune hyperparameter. On the other hand, MD-SAR has the limitation that it needs to know T in advance.

Algorithm 3 MD-APT: Multi-dimensional Anytime Parameter-Free Thresholding algorithm

-
- 1: **Input:** K arms, polyhedron P , tolerance ϵ , budget T
 - 2: **for** $t = 1, \dots, T$ **do**
 - 3: **if** $t \leq K$ **then**
 - 4: Sample $\mathbf{X}_t \sim \nu_t$.
 - 5: **else**
 - 6: Choose $I_t = \arg \min_i \hat{\Delta}_{i,T_i(t)}^{(\epsilon)} \sqrt{T_i(t)}$ and sample $\mathbf{X}_t \sim \nu_{I_t}$.
 - 7: **end if**
 - 8: **end for**
 - 9: **Return:** $\hat{S} = \{i \in [K] : \hat{\mu}_{i,T_i(t+1)} \in P\}$
-

MD-APT (Algorithm 3) is a modification of the APT algorithm in Locatelli et al. (2016). After an initialization phase in which it pulls each arm once, at each round t , it pulls an arm i that minimizes $\hat{\Delta}_{i,T_i(t)}^{(\epsilon)} \sqrt{T_i(t)}$. The intuition behind the algorithm is that if the margins $\Delta_i^{(\epsilon)}$ were known in advance, then a nearly optimal strategy would allocate samples to the arms proportionally to the $[\Delta_i^{(\epsilon)}]^{-2}$ s. For simplicity, let $\epsilon = 0$; the case $\epsilon > 0$ is not as clear since arms whose distance to the boundary is less than ϵ do not need to be sampled at all.

Proposition 1. Let $\epsilon = 0$. A static allocation strategy with a total of $\frac{T}{[\Delta_i^{(\epsilon)}]^2 H}$ pulls of the i th arm $\forall i \in [K]$ achieves

$$\mathbb{E}[\mathcal{L}_{T,\epsilon}(\hat{S})] \leq 2K5^D \exp\left(-\frac{1}{8} \frac{T}{HR^2}\right).$$

Thus, such a static allocation is nearly optimal. Since the $\Delta_i^{(\epsilon)}$ s are unknown, MD-APT samples the arms proportionally to the estimates $[\hat{\Delta}_{i,T_i(t)}^{(\epsilon)}]^{-2}$. Theorem 4 gives an upper bound on the expected loss of MD-APT.

Theorem 4. Let $K \geq 0, T \geq 2K$, and $\epsilon \geq 0$. Then, the expected loss of MD-APT satisfies:

$$\mathbb{E}[\mathcal{L}_{T,\epsilon}(\hat{S})] \leq 2(\log(T) + 1)K5^D \exp\left(-\frac{T}{1296R^2 H}\right).$$

This Theorem implies that MD-APT is nearly optimal. Further, unlike MD-UCBE, it is parameter-free and, unlike MD-SAR, it is an anytime algorithm in the sense that MD-APT does not require knowledge of the budget T . These properties make MD-APT practical for many applications (Jun and Nowak, 2016).

We note that although the runtime of our algorithms depends on M , our upper bounds on their statistical performance are independent of M . We leverage this result and the fact that one can approximate convex sets arbitrarily well with polyhedra to obtain a computationally inefficient algorithm with nearly the same guarantee as Theorem 4 for the setting where P is convex (see the supplementary material for details).

6. Analysis

Our analyses of the three algorithms are unified through a series of lemmas. The first key idea is a sufficient condition for $\widehat{\Delta}_{i,t}^{(\epsilon)}$ to concentrate around $\Delta_i^{(\epsilon)}$. Lemma 1 shows that concentration of $\widehat{\mu}_i(t)$ around its mean in the norm sense is sufficient.

Lemma 1. *Let $\gamma > 0$, $i \in [K]$, and $t \in [T]$. If $\|\widehat{\mu}_{i,t} - \mu_i\|_2 \leq \gamma$, then*

$$|\widehat{\Delta}_{i,t}^{(\epsilon)} - \Delta_i^{(\epsilon)}| \leq 2\gamma.$$

In the scalar case, concentration of the empirical margin around the true margin often follows by the triangle inequality. In our setting, because of the more complicated relationship between $\widehat{\mu}_{i,t}$ and $\widehat{\Delta}_{i,t}^{(\epsilon)}$ such an argument is not sufficient.

The second key idea is that with an appropriately high probability, $\widehat{\mu}_{i,t}$ concentrates around its mean in the norm sense. The main tools are Hoeffding's maximal inequality (see Lemma H.2) and an ϵ -net, which we now define (Vershynin et al., 2017).

Definition 1. *Let $A \subset \mathbb{R}^D$ and $\epsilon > 0$. $\mathcal{N} \subset A$ is an ϵ -net of A if $\forall \mathbf{x} \in A$, there exists $\mathbf{y} \in \mathcal{N}$ such that $\|\mathbf{x} - \mathbf{y}\|_2 \leq \epsilon$. Let $\mathcal{N} \subset A$ be an ϵ -net of A . We say that \mathcal{N} is minimal if, for any other ϵ -net \mathcal{M} of A , it holds that $|\mathcal{M}| \geq |\mathcal{N}|$.*

Lemma 2. *Let \mathcal{N} be a minimal $\frac{1}{2}$ -net on S^{D-1} . Let $\omega > 0$. Define the event*

$$\Xi = \{\forall i, \forall \mathbf{y} \in \mathcal{N}, \forall r \in [T] : |\mathbf{y}^t(\widehat{\mu}_{i,r} - \mu_i)| \leq \sqrt{\frac{\omega^2}{4r}}\}.$$

Then, on Ξ , for all $i \in [K]$ and for all $r \in [T]$,

$$\|\widehat{\mu}_{i,r} - \mu_i\|_2 \leq \sqrt{\frac{\omega^2}{r}}$$

and

$$\Pr(\Xi) \geq 1 - 2(\log(T) + 1)K5^D \exp\left(-\frac{\omega^2}{16R^2}\right).$$

In effect, Lemma 1 and Lemma 2 together imply that with high probability, (i) $\widehat{\mu}_{i,t}$ concentrates around μ_i in the norm sense and (ii) $\widehat{\Delta}_{i,t}^{(\epsilon)}$ concentrates around $\Delta_i^{(\epsilon)}$.

Finally, the third idea is the simple observation that if for all $i \in [K]$, $\widehat{\mu}_{i,t}$ lies in a ball centered at μ_i with radius $\frac{\Delta_i^{(\epsilon)}}{2}$, then an algorithm does not make a mistake.

Lemma 3. *Fix $t \in [T]$ and $i \in [K]$ and suppose that $\|\widehat{\mu}_{i,t} - \mu_i\|_2 < \frac{1}{2}\Delta_i^{(\epsilon)}$. Then, $A\mu_i \leq \mathbf{b} - \epsilon\mathbf{1}$ implies that $A\widehat{\mu}_{i,t} < \mathbf{b}$ and $\text{dist}(\mu_i, P) \geq \epsilon$ implies that $\widehat{\mu}_{i,t} \notin P$.*

The analysis of each algorithm then proceeds as follows. First, suppose some appropriately defined variant of the event Ξ in Lemma 2. Second, by Lemmas 1 and 2, (i) $\widehat{\mu}_{i,t}$ concentrates around μ_i in the norm sense and (ii) $\widehat{\Delta}_{i,t}^{(\epsilon)}$ concentrates around $\Delta_i^{(\epsilon)}$. Given these concentration results, it is shown that each algorithm pulls each arm a sufficient number of times so that Lemma 3 can be applied.

7. Experiments

In this section, we conduct experiments on synthetic and real-world datasets. In addition to the algorithms MD-UCBE, MD-SAR, and MD-APT, we consider a uniform allocation algorithm (UA), which samples the arms in a cyclic fashion. We consider the performance of MD-UCBE under four hyperparameter settings $a_i = i \frac{25}{36} \frac{T-K}{H}$ for $i \in \{.1, 1, 10, 100\}$. Let MD-UCBE[i] denote MD-UCBE with hyperparameter a_i . Note that the larger i is, the more MD-UCBE[i] explores and that our theoretical guarantee in Theorem 2 only covers $i \leq 1$. To calculate $\widehat{\Delta}_{i,t}^{(\epsilon)}$, we use the quadratic programming solver in the CVXOPT package for python. We average all experiments over 2000 trials.

7.1. Synthetic Experiments

Each experiment has 20 5-dimensional arms and is run for 2000 time steps. We use Gaussian distributions with variance $\frac{1}{4}$. For experiments 1, 2, and 3 we use a cube $P = \{\mathbf{x} \in \mathbb{R}^5 : 0 \leq x_i \leq 1\}$. In experiments 4 and 5, we use more complicated feasibility regions. In the following, we say an arm i is *irrelevant* if the error measure $\mathcal{L}_{T,\epsilon}(\cdot)$ does not depend on how i is categorized.

Experiment 1 (Four Groups with Irrelevant Arms): We set $\epsilon = 0.075$ and use $\mu_{0:1} = (.8)^{\otimes 5}$, $\mu_{2:3} = (.9)^{\otimes 5}$, $\mu_{4:5} = (1.1)^{\otimes 5}$, $\mu_{6:7} = (1.2)^{\otimes 5}$, $\mu_8 = (.975)^{\otimes 5}$, $\mu_9 = (1.025)^{\otimes 5}$, $\mu_{10:19} = (.3)^{\otimes 5}$. Note that this problem has two irrelevant arms, μ_8 and μ_9 .

Experiment 2 (Four Groups with no Irrelevant Arms): We set $\epsilon = 0$ and use $\mu_{0:1} = (.8)^{\otimes 5}$, $\mu_{2:3} = (.9)^{\otimes 5}$, $\mu_{4:5} = (1.1)^{\otimes 5}$, $\mu_{6:7} = (1.2)^{\otimes 5}$, $\mu_8 = (.95)^{\otimes 5}$, $\mu_9 = (1.05)^{\otimes 5}$, $\mu_{10:19} = (.3)^{\otimes 5}$. In comparison to experiment 1, we make it slightly easier to determine whether the arms μ_8 and μ_9

belong to the polyhedron because otherwise the difficulty of the problem prevents any algorithm from achieving substantial progress after 2000 time steps.

Experiment 3 (Linear Progression with Irrelevant Arms): We set $\epsilon = 0.075$ and use $\mu_{0:3} = (.75)^{\otimes 5} + (0 : 3) \times .05$, $\mu_4 = (.975)^{\otimes 5}$, $\mu_5 = (1.025)^{\otimes 5}$, $\mu_{6:9} = (1.25)^{\otimes 5} - (0 : 3) \times .05$, $\mu_{10:19} = (1.15)^{\otimes 5}$. Note that this problem has two irrelevant arms, μ_4 and μ_5 .

Experiment 4 (Four Groups on the Simplex): For this experiment, we use $P = \{x \in \mathbb{R}^5 : x_i \geq 0, \sum_i x_i \leq 2\}$. We set $\epsilon = .1$. Let $c = (.2)^{\otimes 5}$. We use $\mu_{0:4} = c$, $\mu_{5:9} = 1.85 \cdot c$, $\mu_{10:14} = 2.25 \cdot c$, and $\mu_{15:19} = 1.95 \cdot c$. $\mu_{0:9}$ are good arms, $\mu_{10:14}$ are bad arms, and $\mu_{15:19}$ are irrelevant.

Experiment 5 (Ordered Polyhedron): For this experiment, we use $P = \{x \in \mathbb{R}^5 : x_i \leq x_{i+1} \forall i \in [4]\}$ and $\epsilon = .1$. We use $\mu_{0:3} = (0, .2, .4, .6, .8)^t$, $\mu_{4:7} = (.0, .15, .3, .45, .6)^t$, $\mu_{8:11} = (0, .2, .15, .6, .8)^t$, $\mu_{12:15} = (0, .2, .05, .6, .8)^t$, and $\mu_{16:19} = (0, .2, .4, .2, 0)^t$. The arms $\mu_{8:11}$ are irrelevant.

The performance of MD-UCBE is very sensitive to the selection of its hyperparameter. MD-UCBE[1] and MD-UCBE[10] tend to do well, but MD-UCBE[100] explores too much so that it tends to perform only slightly better than UA and MD-UCBE[1] does not explore enough. Although MD-UCBE[1] has a theoretical guarantee, the constants are too large so that it never makes progress in solving the problems. MD-APT performs better than MD-SAR in experiments 1, 4, and 5 and worse than MD-SAR in experiments 2 and 3. In experiment 2, MD-APT pulls arm 8, which minimizes $\Delta_i^{(\epsilon)}$, too frequently. It pulls arm 8 on average 904.8125 times, whereas MD-SAR more evenly spreads out its pulls, pulling arm 8 on average 317.751 times. We observe a similar phenomenon in a variant of experiment 3 where we set $\epsilon = 0$ and which we defer to the supplementary material due to lack of space. This suggests that in certain problems MD-APT focuses too much on specific arms with means near the boundary and does not allocate enough samples to other arms. On the other hand, MD-SAR utilizes knowledge of the time horizon T to effectively spread out samples. MD-APT's agnosticism about T may put it at a disadvantage in the regime where some of the $\Delta_i^{(\epsilon)}$ are very small and T is small relative to H . As suggested by experiment 1, the ϵ parameter can be used to counteract the sensitivity of MD-APT to arms with means near the boundary.

7.2. Application 1: Dose-Finding

In clinical trials, an important challenge is determining the appropriate dosage of a drug. The main difficulty is the trade-off that as the dosage increases, the effectiveness of the drug tends to increase, but the likelihood of adverse

effects also increases. Thus, one must find a dosage that is sufficiently effective, but does not have too many side effects. We assume a situation where the side effects are mild enough not to be a concern for clinical trials, but could nevertheless be unacceptable for a final commercial product.

We investigate this problem by considering the data in [Genovese et al. \(2013\)](#) (see ARCR20 in week 16 in Table 2 and Table 3). In this study, the authors examine the drug secukinumab for treating rheumatoid arthritis. They consider four dosage levels (25mg, 75mg, 150mg, 300mg) and a placebo. We design a simulation based on their data where each arm corresponds to a drug and has two attributes, the likelihood of being effective and the likelihood of causing an adverse effect. Let $\mu_{i,1}$ denote the probability of being effective and $\mu_{i,2}$ the probability of causing an adverse effect. Then, dosage levels 25mg, 75mg, 150mg, and 300mg have means $\mu_1 = (.34, .519)^t$, $\mu_2 = (.469, .612)^t$, $\mu_3 = (.465, .465)^t$, $\mu_4 = (.537, .61)^t$, respectively, and the placebo has mean $\mu_5 = (.36, .58)^t$. We suppose that a drug is considered good if the probability of success is above .4 and the probability of adverse effects is below .5 and we set $\epsilon = 0$. Thus, only arm 3 is good and all other arms are bad. We chose these thresholds so that one drug is good; we did not try other threshold settings. We run the experiment for 1000 time steps.

Figure 6 gives the results of the experiment. MD-APT and MD-UCBE[10] perform better than the rest of the algorithms. MD-UCBE[1] performs slightly worse than UA, which may be because there are only 5 arms so that UA is not that bad of a strategy and MD-UCBE[1] does not explore sufficiently. MD-SAR only performs slightly better than UA. This may be because the time horizon is only 1000 time steps and there are only 5 arms.

7.3. Application 2: Crowdsourcing

We use a real-world dataset for the natural language processing task of affective text analysis ([Snow et al., 2008](#)). In this task, workers are asked to rate a short headline on valence and six emotions: disgust, fear, joy, anger, sadness and surprise. A group of experts also provide such ratings for the headlines.

We consider the problem of finding workers that tend to agree with the expert views on each of the tasks. We examine the deviation of a worker's ratings with the experts ratings. We normalize this deviation onto a scale of $[0, 1]$. Let $\mu_{i,j}$ denote the mean of worker i on task j and let $\bar{\mu}_j$ denote the mean of all of the workers on task j . We deem a worker i good if $\mu_{i,j} \leq \bar{\mu}_j$ for all $j \in [7]$. In words, a worker is good if for every task, he performs better than the average worker. To make this realistic, we assume that we are in a setting where the average worker performance on each task is known based on another pool of workers. We

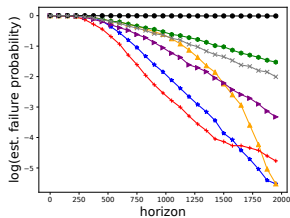


Figure 1. Four Groups on Cube with Irrelevant Arms

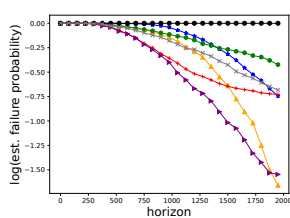


Figure 2. Four Groups on Cube, no Irrelevant Arms

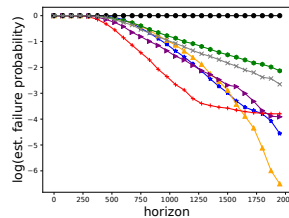


Figure 3. Linear Progression on Cube with Irrelevant Arms

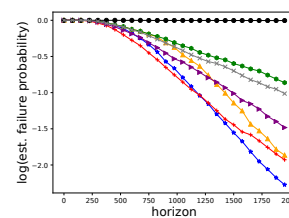


Figure 4. Four Groups on a Simplex

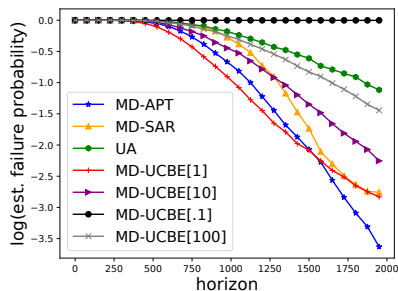


Figure 5. Ordered polyhedron

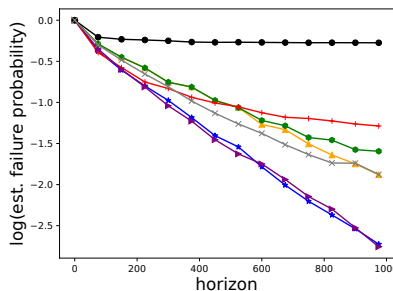


Figure 6. Dose-Finding Experiment

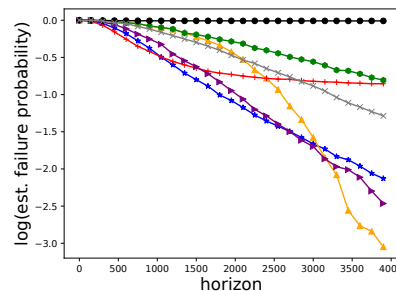


Figure 7. Crowdsourcing Experiment

use a tolerance of $\epsilon = 0.02$. There is a total of 38 workers, where 30 workers are bad arms, 3 workers are good arms, and 5 workers are irrelevant. Because each worker only provides a small number (at least 20) of ratings, whenever an arm is pulled, the algorithm observes an observation chosen uniformly at random with replacement from the data associated with the arm. We run each algorithm for 4000 time steps and in each trial, we randomly permute the samples of each worker. In the supplementary material, we repeat this experiment, but we simulate each arm as a Gaussian distribution (see Section J); the results are very similar.

Figure 7 gives the results of the experiment. Until roughly time step 3000, MD-APT and MD-UCBE[10] perform the best. Afterwards, MD-SAR does substantially better than MD-APT and MD-UCBE[10]. MD-UCBE[1] and MD-UCBE[100] perform only marginally better than UA.

7.4. Summary of Results

The experiments suggest that although MD-UCBE is a competitive algorithm, it is highly sensitive to hyperparameter selection, which limits its applicability in practice. MD-SAR and MD-APT tend to perform dramatically better than UA. For example, in the crowdsourcing experiment, UA has a final error rate of roughly 52%, whereas MD-SAR has a final error rate of roughly 5%. Further, our algorithms can handle complicated polyhedra such as the polyhedron that requires that coordinates are sorted in ascending order (see experiment 5). These results suggest that MD-APT tends to perform better than MD-SAR, but in some settings

(e.g., some arms with small $\Delta_i^{(\epsilon)}$ and H large relative to T) MD-APT focuses too much on some of the arms with means near the boundary. Because MD-SAR more evenly spreads out its pulls among the arms, it performs better in this regime.

8. Conclusion

In this paper, we introduced the feasible arm identification problem. This problem provides a flexible framework for settings where arms are multi-dimensional and it is of interest to determine whether each arm satisfies user-defined multi-dimensional criteria. We provided a characterization of the difficulty of these problems that yielded a lower bound and we provided a unified analysis of three algorithms MD-UCBE, MD-SAR, and MD-APT. Our experiments suggest that by leveraging the geometry of the feasible arm identification problem, MD-SAR and MD-APT are able to dramatically outperform a uniform allocation approach.

Our work also suggests several open directions for future research. For example, in many crowdsourcing problems, one does not ask workers to perform all tasks at once, but rather one task at a time and, yet, it may be of interest to find workers who excel at a collection of tasks. This suggests a variant of the feasible arm identification problem where the agent chooses one coordinate of one arm and observes a realization of the corresponding random variable in each round.

Acknowledgements

This work was supported in part by NSF grant 1422157. We thank the anonymous reviewers for their very helpful comments and Aditya Modi for his useful feedback.

References

- J.-Y. Audibert and S. Bubeck. Best arm identification in multi-armed bandits. *Conference on Learning Theory*, 2010.
- P. Auer, C.-K. Chiang, R. Ortner, and M. Drugan. Pareto front identification from stochastic bandit feedback. *Artificial Intelligence and Statistics*, pages 939–947, 2016.
- S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University press, 2004.
- S. Bubeck, R. Munos, and G. Stolz. Pure exploration in multi-armed bandits problems. *Algorithmic Learning Theory*, pages 23–37, 2009.
- S. Bubeck, T. Wang, and N. Viswanathan. Multiple identifications in multi-armed bandits. *International Conference on Machine Learning*, pages 258–265, 2013.
- R. Busa-Fekete, B. Szorenyi, P. Weng, and S. Mannor. Multi-objective bandits: Optimizing the generalized gini index. *ICML*, pages 625–634, 2017.
- L. Chen, A. Gupta, J. Li, M. Qiao, and R. Wang. Nearly optimal sampling algorithms for combinatorial pure exploration. *Proceedings of Machine Learning Research*, 65:1–53, 2017.
- S. Chen, T. Lin, I. King, M. Lyu, and W. Chen. Combinatorial pure exploration of multi-armed bandits. *Advances in Neural Information Processing Systems*, pages 379–387, 2014.
- M. Drugan and A. Now. Designing multi-objective multi-armed bandits algorithms: A study. *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pages 1–8, 2013.
- V. Gabillon, M. Ghavamzadeh, and A. Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. *Advances in Neural Information Processing Systems*, pages 3212–3220, 2012.
- M. Genovese, P. Durez, H. Richards, J. Supronik, E. Dokoupilova, V. Mazurov, and J. Aelion. Efficacy and safety of secukinumab in patients with rheumatoid arthritis: a phase ii, dose-finding, double-blind, randomised, placebo controlled study. *Annals of the rheumatic diseases*, 72: 863–869, 2013.
- K. Jamieson, M. Malloy, R. Nowak, and S. Bubeck. lil’ucb: An optimal exploration algorithm for multi-armed bandits. *Conference on Learning Theory*, pages 424–439, 2014.
- K.-S. Jun and R. Nowak. Anytime exploration for multi-armed bandits using confidence information. *ICML*, pages 974–982, 2016.
- A. Locatelli, M. Gutzeit, and A. Carpentier. An optimal algorithm for the thresholding bandit problem. *Proceedings of The 33rd International Conference on Machine Learning*, pages 1690–1698, 2016.
- S. Mannor and J. Tistisklis. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*, pages 623–648, 2004.
- R. Snow, B. O’Connor, D. Jurafsky, and A. Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. *Proceedings of the conference on empirical methods in natural language processing*, pages 254–263, 2008.
- C. Tekin and E. Turgay. Multi-objective contextual multi-armed bandit problem with a dominant objective. *arXiv preprint arXiv:1708.05655*, 2017.
- R. Vershynin. Introduction to the non-asymptotic analysis of random matrices. In Y. Eldar and G. Kutyniok, editors, *Compressed Sensing: Theory and Applications*, pages 210–268. Cambridge University Press, 2012.
- R. Vershynin, P. Hsu, C. Ma, J. Nelson, E. Schnoor, D. Stoger, T. Sullivan, and T. Tao. High-dimensional probability: An introduction with applications in data science. 2017.
- S. Zheng, B. Waggoner, Y. Liu, and Y. Chen. Active information acquisition for linear optimization. *arXiv preprint arXiv:1709.10061*, 2017.