Multitask Bandit Learning Through Heterogeneous Feedback Aggregation

Zhi Wang*[1], Chicheng Zhang*[2], Manish Kumar Singh[1], Laurel D. Riek[1], Kamalika Chaudhuri[1]
[1]University of California San Diego, [2]University of Arizona

Abstract

In many real-world applications, multiple agents seek to learn how to perform highly related yet slightly different tasks in an online bandit learning protocol. We formulate this problem as the ϵ-*multi-player multi-armed bandit* problem, in which a set of players concurrently interact with a set of arms, and for each arm, the reward distributions for all players are similar but not necessarily identical. We develop an upper confidence bound-based algorithm, RobustAgg(ϵ), that adaptively aggregates rewards collected by different players. In the setting where an upper bound on the pairwise dissimilarities of reward distributions between players is known, we achieve instance-dependent regret guarantees that depend on the amenability of information sharing across players. We complement these upper bounds with nearly matching lower bounds. In the setting where pairwise dissimilarities are unknown, we provide a lower bound, as well as an algorithm that trades off minimax regret guarantees for adaptivity to unknown similarity structure.

1 Introduction

Online multi-armed bandit learning has many important real-world applications (see Villar et al., 2015; Shen et al., 2015; Li et al., 2010, for a few examples). In practice, a group of online bandit learning agents are often deployed for similar tasks, and they learn to perform these tasks in similar yet nonidentical environments. For example, a group of assistive healthcare robots may be deployed to provide personalized cognitive training to people with dementia (PwD), e.g., by playing cognitive training games with people (Kubota et al., 2020). Each robot seeks to learn the preferences of its paired PwD so as to recommend tailored health intervention based on how the PwD reacts to and is engaged with the activities (as captured by sensors on the robots) (Kubota et al., 2020). As PwD may have similar preferences and may therefore exhibit similar reactions, one natural question arises—can the robots as a multi-agent system learn to perform their respective tasks faster through collaboration? In this paper, we develop multi-agent bandit learning algorithms where each agent can robustly aggregate data from other agents to better perform its respective task.

We generalize the multi-armed bandit problem (Auer et al., 2002) and formulate the ϵ-Multi-Player Multi-Armed Bandit (ϵ-MPMAB) problem, which models *heterogeneous multitask learning* in a multi-agent bandit learning setting. In an ϵ-MPMAB problem instance, a set of M players are deployed to perform similar tasks—simultaneously they interact with a set of actions/arms, and for each arm, different players receive feedback from similar but not necessarily identical reward distributions. In the above assistive robotics example, each player corresponds to a robot; each arm corresponds to one of the cognitive activities to choose from; for each player and each arm, there is a separate reward distribution which reflects a PwD's personal preferences. Informally, $\epsilon \geq 0$ is a *dissimilarity parameter* that upper bounds the pairwise distances between different reward distributions for different players on the same arm (see Definition 1 in the next section). The players can communicate and share information among each other, with a goal of maximizing their collective reward.

While multi-player bandit learning has been studied extensively in the literature (e.g., Landgren et al., 2016; Cesa-Bianchi et al., 2013; Gentile et al., 2014), warm-starting bandit learning using different feedback sources has been investigated (Zhang et al., 2019), and sequential transfer between similar tasks in a bandit learning setting has also been studied (Azar et al.,

*Equal contribution. Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

2013; Soare et al., 2014), to our knowledge, no prior work models multitask learning in a multi-player bandit learning perspective with a focus on adaptive and robust aggregation of player-dependent heterogeneous feedback. In Section 5, we further discuss and compare our problem formulation with related papers.

It is worth noting that naively utilizing data collected by other players may substantially hurt a player's regret (Zhang et al., 2019), if there are large disparities between the sources of feedback. This is also well-known as *negative transfer* in transfer learning (Rosenstein et al., 2005; Brunskill and Li, 2013).

Therefore, the main challenge of the ϵ-MPMAB problem is for the players to properly manage *when and how* to utilize auxiliary data shared by others—while auxiliary data can be useful to maintain more accurate estimates of the rewards for each player and each arm, they can also easily be inefficacious or even misleading. While transfer learning in the offline setting has been well studied, in this paper we seek to characterize the difficulty of the more challenging problem of learning through heterogeneous feedback aggregation in a multi-player online setting.

We will first study the ϵ-MPMAB problem when the dissimilarity parameter ϵ is known, and then move on to the harder setting in which ϵ is unknown. Here is a summary of our main contributions:

- We model online multitask bandit learning from heterogeneous data sources as the ϵ-MPMAB problem, with a goal of studying how to adaptively and robustly aggregate data to improve the collective performance of the players.
- In the setting where ϵ is known, we propose an upper confidence bound (UCB)-based algorithm, RobustAgg(ϵ), that adaptively aggregates rewards collected by different players.

  We provide (suboptimality)-gap-dependent and gap-independent upper bounds on the collective regret of RobustAgg(ϵ). Our regret bounds depend on the set of arms that admit information sharing among the players. When this set is large, RobustAgg(ϵ) can potentially improve the gap-dependent regret bound by nearly a factor of M compared to the baseline of players acting individually using UCB-1 (Auer et al., 2002).

  We complement these upper bounds with nearly matching gap-dependent and gap-independent lower bounds.
- In the setting where ϵ is unknown, we first establish a lower bound, showing that if an algorithm guarantees sublinear minimax regret with respect to all MPMAB instances, then it must be unable to significantly utilize inter-player similarity in a large collection of instances. To complement the above result, we use the framework of Corral (Agarwal et al., 2017; Pacchiano et al., 2020; Arora et al., 2020) and present an algorithm that trades off minimax regret guarantee for adaptivity to "easy" MPMAB problem instances.

2 Problem Specification

We formulate the ϵ-MPMAB problem, building on the standard model of stochastic multi-armed bandits (Lai and Robbins, 1985; Auer et al., 2002).

Throughout, we denote by $[n] = \{1, \ldots, n\}$. An *MPMAB problem instance* consists of a set of M players, labeled as elements in $[M]$, and a set of K arms, labeled as elements in $[K]$. In addition, each player $p \in [M]$ and each arm $i \in [K]$ is associated with an unknown reward distribution $\mathcal{D}_i^p$ with support $[0, 1]$ and mean μ_i^p. If all $\mathcal{D}_i^p$'s are Bernoulli distributions, we call this instance a *Bernoulli MPMAB problem instance*; under the Bernoulli reward assumption, $\mu = (\mu_i^p)_{i\in[K],p\in[M]}$ completely specifies the instance.

The reward distributions of the same arm are not necessarily identical for different players—we consider the following notion of dissimilarity between the reward distributions of the players. Related conditions have been considered in works on multi-task bandit learning (e.g., Azar et al., 2013; Soare et al., 2014).

Definition 1. *An MPMAB problem instance is said to be an ϵ-MPMAB problem instance, if for every pair of players $p, q \in [M]$, $\max_{i\in[K]} |\mu_i^p - \mu_i^q| \leq \epsilon$, where $\epsilon \in [0, 1]$. We call ϵ the dissimilarity parameter.*

Interaction protocol. Let $T > \max(M, K)$ be the horizon of an MPMAB (ϵ-MPMAB) problem instance. In each round $t \in [T]$, every player $p \in [M]$ pulls an arm i_t^p, and observes an independently-drawn reward $r_t^p \sim \mathcal{D}_{i_t^p}^p$. Once all the M players finish pulling arms in round t, each decision, i_t^p, together with the corresponding reward received, r_t^p, is immediately shared with all players.

Arm pulls, gaps, and performance measure. Let $\mu_*^p = \max_{i\in[K]} \mu_i^p$ be the optimal mean reward for every player $p \in [M]$. Denote by $n_i^p(t)$ the number of pulls of arm i by player p after t rounds, and $\Delta_i^p = \mu_*^p - \mu_i^p \geq 0$ the *suboptimality gap* (abbrev. gap) between the means of the reward distributions associated with some optimal arm i_*^p and arm i for player p. For any arm $i \in [K]$, define $\Delta_i^{\min} = \min_{p\in[M]} \Delta_i^p$. To measure the performance of MPMAB algorithms, we use the following notion of regret. The expected regret of

player p is defined as $\mathbb{E}[\mathcal{R}^p(T)] = \sum_{i\in[K]} \Delta_i^p \cdot \mathbb{E}[n_i^p(T)]$, and the players' *expected collective regret* is defined as $\mathbb{E}[\mathcal{R}(T)] = \sum_{p\in[M]} \mathbb{E}[\mathcal{R}^p(T)]$.

Bandit learning algorithms. A multi-player bandit learning algorithm $\mathcal{A}$ with horizon T is defined as a sequence of conditional probability distributions $\{\pi_t\}_{t=1}^T$, where for every t in $[T]$, π_t is the policy used in round t; specifically, $\pi_t(\cdot \mid (i_s^p, r_s^p)_{s\in[t-1],p\in[M]})$ is a conditional probability distribution of actions taken by all M players in round t, given historical data. A bandit learning algorithm is said to have *sublinear regret* for the ϵ-MPMAB (resp. MPMAB) problem, if there exists some $C > 0$ and $\alpha > 0$ such that $\mathbb{E}[\mathcal{R}(T)] \le CT^{1-\alpha}$ for all ϵ-MPMAB (resp. MPMAB) problem instances.

Miscellaneous notations. Throughout, we use $\tilde{O}$ notation to hide logarithmic factors. Given a universe set $\mathcal{H}$ and any $\mathcal{J} \subseteq \mathcal{H}$, we use $\mathcal{J}^C$ to denote the set $\mathcal{H} \setminus \mathcal{J}$.

Baseline: Individual UCB. We now consider a baseline algorithm that runs the UCB-1 algorithm individually for each player without communication—hereafter, we refer to it as IND-UCB. By (Auer et al., 2002, Theorem 1), and summing over the individual regret guarantees of all players, the expected collective regret of IND-UCB satisfies

$$\mathbb{E}[\mathcal{R}(T)] \le O\bigg(\sum_{i\in[K]} \sum_{p\in[M]:\Delta_i^p>0} \frac{\ln T}{\Delta_i^p}\bigg).$$

In addition, IND-UCB has a gap-independent regret bound of $\tilde{O}\left(M\sqrt{KT}\right)$ (e.g., Lattimore and Szepesvári, 2020, Theorem 7.2).

2.1 Can auxiliary data always help?

Since the interaction protocol allows information sharing among players, in any round $t > 1$, each player has access to more data than they would have without communication. Can the players always expect benefits from such auxiliary data and collectively perform better than IND-UCB?

Below we provide an example that illustrates that the role of auxiliary data depends on the dissimilarities between the player-dependent reward distributions, as indicated by ϵ, as well as the intrinsic difficulty of each multi-armed bandit problem each player faces individually, as indicated by the gaps Δ_i^p's. Specifically, we show in the example that when ϵ is much larger than the gaps Δ_i^p's, any sublinear-regret bandit learning algorithm for the ϵ-MPMAB problem cannot significantly take advantage of auxiliary data.

Example 2. *For a fixed $\epsilon \in (0, \frac{1}{8})$ and $\delta \le \epsilon/4$, consider the following Bernoulli MPMAB problem instance: for each $p \in [M]$, $\mu_1^p = \frac{1}{2} + \delta$, $\mu_2^p = \frac{1}{2}$. This is a 0-MPMAB instance, hence an ϵ-MPMAB problem instance. Also, note that ϵ is at least four times larger than the gaps $\Delta_2^p = \delta$.*

Claim 3. *For the above example, any sublinear regret algorithm for the ϵ-MPMAB problem must have $\Omega(\frac{M \ln T}{\delta})$ regret on this instance, matching the* IND-UCB *regret upper bound.*

The claim follows from Theorem 9 in Section 3.3; see Appendix B for details. The intuition is that any sublinear regret ϵ-MPMAB algorithm must have $\Omega\left(\frac{\ln T}{\delta^2}\right)$ pulls of arm 2 from every player; otherwise, as δ is small compared to ϵ, we can create a new ϵ-MPMAB instance such that arm 2 is optimal for some player and is sufficiently indistinguishable from the original MPMAB problem, causing the algorithm to fail its sublinear regret guarantee.

Complementary to the above negative result, in the next section, we establish algorithms and sufficient conditions for the players to take advantage of the auxiliary data to achieve better regret guarantees.

3 ϵ-MPMAB with Known ϵ

In this section, we study the ϵ-MPMAB problem with the dissimilarity parameter ϵ known to the players. We first present our main algorithm ROBUSTAGG(ϵ) in Section 3.1; Section 3.2 shows its regret guarantees; Finally, Section 3.3 provides nearly matching regret lower bounds. Our proofs are deferred to Appendices C, D and E.

3.1 Algorithm: RobustAgg(ϵ)

We present ROBUSTAGG(ϵ), an algorithm that adaptively and robustly aggregates rewards collected by different players in ϵ-MPMAB problem instances, given dissimilarity ϵ as an input parameter.

Intuitively, in any round, a player may decide to take advantage of data from other players who have similar reward distributions. Deciding how to use auxiliary data is tricky—on the one hand, they can help reduce variance and get a better mean reward estimate, but on the other hand, if the dissimilarity between players' reward distributions is large, auxiliary data can substantially bias the estimate. Our algorithm is built upon this insight of balancing bias and variance. A similar tradeoff in offline transfer learning for classification is studied in the work of Ben-David et al. (2010); we discuss the connection and differences between our work and theirs in Section 5.2.

Algorithm 1: RobustAgg(ϵ): Robust learning in ϵ-MPMAB

Input: Distribution dissimilarity parameter $\epsilon \in [0,1]$;

1 **Initialization:** Set $n_i^p = 0$ for all $p \in [M]$ and all $i \in [K]$.

2 **for** $t = 1, 2 \ldots, T$ **do**

3 **for** $p \in [M]$ **do**

4 **for** $i \in [K]$ **do**

5 Let $m_i^p = \sum_{q \in [M]: q \neq p} n_i^q$;

6 Let $\overline{n_i^p} = \max(1, n_i^p)$ and $\overline{m_i^p} = \max(1, m_i^p)$;

7 Let

$$\zeta_i^p(t) = \frac{1}{\overline{n_i^p}} \sum_{\substack{s<t \\ i_s^p = i}} r_s^p, \ \eta_i^p(t) = \frac{1}{\overline{m_i^p}} \sum_{\substack{q \in [M] \\ q \neq p}} \sum_{\substack{s<t \\ i_s^q = i}} r_s^q, \text{and } \kappa_i^p(t, \lambda) = \lambda \zeta_i^p(t) + (1-\lambda)\eta_i^p(t);$$

8 Let $F(\overline{n_i^p}, \overline{m_i^p}, \lambda, \epsilon) = 8\sqrt{13 \ln T \left[\frac{\lambda^2}{\overline{n_i^p}} + \frac{(1-\lambda)^2}{\overline{m_i^p}}\right]} + (1-\lambda)\epsilon$;

9 Compute $\lambda^* = \text{argmin}_{\lambda \in [0,1]} F(\overline{n_i^p}, \overline{m_i^p}, \lambda, \epsilon)$;

10 Compute an upper confidence bound of the reward of arm i for player p:

$$\text{UCB}_i^p(t) = \kappa_i^p(t, \lambda^*) + F(\overline{n_i^p}, \overline{m_i^p}, \lambda^*, \epsilon).$$

11 Let $i_t^p = \text{argmax}_{i \in [K]} \text{UCB}_i^p(t)$;

12 Player p pulls arm i_t^p and observes reward r_t^p;

13 **for** $p \in [M]$ **do**

14 Let $i = i_t^p$ and set $n_i^p = n_i^p + 1$.

Algorithm 1 provides a pseudocode of RobustAgg(ϵ). Specifically, it builds on the classic UCB-1 algorithm (Auer et al., 2002): for each player p and arm i, it maintains an upper confidence bound $\text{UCB}_i^p(t)$ for mean reward μ_i^p over time (lines 5 to 10), such that with high probability, $\mu_i^p \leq \text{UCB}_i^p(t)$, for all t.

To achieve the best regret guarantees, we would like our confidence bounds on μ_i^p to be as tight as possible. To this end, we consider a family of confidence intervals for μ_i^p, parameterized by a weighting factor $\lambda \in [0,1]$: $[\kappa_i^p(t,\lambda) \pm F(\overline{n_i^p}, \overline{m_i^p}, \lambda, \epsilon)]$.

In the above confidence interval formula, $\kappa_i^p(t,\lambda)$ estimates μ_i^p by taking a convex combination of $\xi_i^p(t)$ and $\eta_i^p(t)$, the empirical mean reward of arm i based on the player's own samples and the auxiliary samples, respectively (line 7). The width $F(\overline{n_i^p}, \overline{m_i^p}, \lambda, \epsilon)$ is a high-probability upper bound on $\left|\kappa_i^p(t,\lambda) - \mu_i^p\right|$ (line 8). Varying λ reveals the aforementioned bias-variance tradeoff: the first term, $8\sqrt{13 \ln T[\frac{\lambda^2}{\overline{n_i^p}} + \frac{(1-\lambda)^2}{\overline{m_i^p}}]}$, is a high probability upper bound on the deviation of $\kappa_i^p(t,\lambda)$ from its expectation $\mathbb{E}[\kappa_i^p(t,\lambda)]$; the second term, $(1-\lambda)\epsilon$, is an upper bound on the difference between $\mathbb{E}[\kappa_i^p(t,\lambda)]$ and μ_i^p. We choose $\lambda^* \in [0,1]$ to minimize the width of our confidence interval for μ_i^p (line 9), similar to the calculation in (Ben-David et al., 2010, Section 6).[1]

3.2 Regret analysis

We first define the notion of *subpar arms*. Let

$$\mathcal{I}_\epsilon = \left\{i : \exists p \in [M], \mu_*^p - \mu_i^p > 5\epsilon\right\}$$

be the set of subpar arms for an ϵ-MPMAB problem instance. Intuitively, $\mathcal{I}_\epsilon$ contains the set of "easier" arms for which data aggregation between players can be *effective*. For each arm $i \in \mathcal{I}_\epsilon$, the following fact shows that the gap $\Delta_i^p = \mu_*^p - \mu_i^p$ is sufficiently larger than the dissimilarity parameter ϵ for all players $p \in [M]$. This allows RobustAgg(ϵ) to exploit the "easiness" of these arms through data aggregation across players, thereby reducing avoidable individual explorations.

Fact 4. *$|\mathcal{I}_\epsilon| \leq K-1$. In addition, for each arm $i \in \mathcal{I}_\epsilon$, $\Delta_i^{\min} > 3\epsilon$; in other words, for all players p in $[M]$, $\Delta_i^p = \mu_*^p - \mu_i^p > 3\epsilon$; consequently, arm i is suboptimal for all players p in $[M]$.*

We now present regret guarantees of RobustAgg(ϵ).

Theorem 5. *Let* RobustAgg(ϵ) *run on an ϵ-MPMAB problem instance for T rounds. Then, its*

[1] See Appendix H for an analytical solution to the optimal weighting factor λ^*.

expected collective regret satisfies

$$\mathbb{E}[\mathcal{R}(T)] \le O\left(\sum_{i\in\mathcal{I}_\epsilon}\left(\frac{\ln T}{\Delta_i^{\min}} + M\Delta_i^{\min}\right) + \sum_{i\in\mathcal{I}_\epsilon^C}\sum_{p\in[M]:\Delta_i^p>0}\frac{\ln T}{\Delta_i^p}\right).$$

The first term in the above bound shows that the collective regret incurred by the players for the subpar arms $\mathcal{I}_\epsilon$ and the second term for arms in $\mathcal{I}_\epsilon^C = [K]\setminus\mathcal{I}_\epsilon$. Observe that for each subpar arm, the regret of the players *as a group* can be upper-bounded by $O\left(\frac{\ln T}{\Delta_i^{\min}} + M\Delta_i^{\min}\right)$, whereas for each arm in $\mathcal{I}_\epsilon^C$, the regret on *each* player is $O(\frac{\ln T}{\Delta_i^p})$ unless $\Delta_i^p = 0$.

Fact 6. *For any $i \in \mathcal{I}_\epsilon$, $\frac{1}{\Delta_i^{\min}} \le \frac{2}{M}\sum_{p\in[M]}\frac{1}{\Delta_i^p}$.*

Fallback guarantee. The regret guarantee of RobustAgg(ϵ) by Theorem 5 is always no worse than that of Ind-UCB by a constant factor, as from Fact 6, for all i in $\mathcal{I}_\epsilon$, $\frac{\ln T}{\Delta_i^{\min}} + M\Delta_i^{\min} = O\left(\sum_{p\in[M]}\frac{\ln T}{\Delta_i^p}\right)$.

Two extreme cases of $|\mathcal{I}_\epsilon|$. If $\mathcal{I}_\epsilon = \emptyset$, in which case we do not expect data aggregation across players to be beneficial, the above bound can be simplified to:

$$\mathbb{E}[\mathcal{R}(T)] \le O\left(\sum_{i\in[K]}\sum_{p\in[M]:\Delta_i^p>0}\frac{\ln T}{\Delta_i^p}\right).$$

In contrast, when $\mathcal{I}_\epsilon$ has a larger size, namely, more arms admit data aggregation across players, RobustAgg(ϵ) has an improved regret bound. The following corollary gives regret bounds in the most favorable case when $\mathcal{I}_\epsilon$ has size $K-1$. It is not hard to see that, in this case, $\mathcal{I}_\epsilon^C$ is equal to a singleton set $\{i_*\}$, where arm i_* is optimal for all players p.

Corollary 7. *Let* RobustAgg(ϵ) *run on an ϵ-MPMAB problem instance with $|\mathcal{I}_\epsilon| = K-1$ for T rounds. Then, its expected collective regret satisfies*

$$\mathbb{E}[\mathcal{R}(T)] \le O\left(\sum_{i\neq i_*}\frac{\ln T}{\Delta_i^{\min}} + M\sum_{i\neq i_*}\Delta_i^{\min}\right).$$

It can be observed that, compared to the Ind-UCB baseline, under the assumption that $|\mathcal{I}_\epsilon| = K-1$, RobustAgg($\epsilon$) improves the regret bound by nearly a factor of M: if we set aside the $O\left(M\sum_{i\neq i_*}\Delta_i^{\min}\right)$ term, which is of lower order than the rest under the mild assumption that $M = O\left(\min_{i\neq i^*}\frac{\ln T}{(\Delta_i^{\min})^2}\right)$, then the expected collective regret in Corollary 7 is a factor of $O(\frac{1}{M})$ times that of Ind-UCB, in light of Fact 6.

Gap-independent upper bound. We now provide an upper bound on the expected collective regret that is independent of the gaps Δ_i^p's.

Theorem 8. *Let* RobustAgg(ϵ) *run on an ϵ-MPMAB problem instance for T rounds. Then its expected collective regret satisfies*

$$\mathbb{E}[\mathcal{R}(T)] \le \tilde{O}\left(\sqrt{|\mathcal{I}_\epsilon|\,MT} + M\sqrt{(|\mathcal{I}_\epsilon^C| - 1)T} + M|\mathcal{I}_\epsilon|\right).$$

Recall that Ind-UCB has a gap-independent bound of $\tilde{O}\left(M\sqrt{KT}\right)$. By algebraic calculations, we can see that when $T = \Omega(KM)$, the regret bound of RobustAgg(ϵ) is a factor of $O\left(\max\left(\sqrt{\frac{|\mathcal{I}_\epsilon^C|-1}{K}}, \sqrt{\frac{1}{M}}\right)\right)$ times Ind-UCB's regret bound. Therefore, when $M = \omega(1)$ and $|\mathcal{I}_\epsilon^C| = o(K)$, i.e., when there is a large number of players, and an overwhelming portion of subpar arms, RobustAgg has a gap-independent regret bound of strictly lower order than Ind-UCB.

Observe that the above bound has a term $M\sqrt{(|\mathcal{I}_\epsilon^C| - 1)T}$ with a peculiar dependence on $|\mathcal{I}_\epsilon^C| - 1$; this is due to the fact that in the special case of $|\mathcal{I}_\epsilon| = K-1$, i.e., $|\mathcal{I}_\epsilon^C| = 1$, the contribution to the regret from arms in $\mathcal{I}_\epsilon^C$ is zero. Indeed, in this case, $\mathcal{I}_\epsilon^C$ is a singleton set $\{i_*\}$, where arm i_* is optimal for all players.

3.3 Lower bounds

Gap-dependent lower bound. To complement our gap-dependent upper bound in Theorem 5, we now present a gap-dependent lower bound. We show that, for any fixed ϵ, any sublinear regret algorithm for the ϵ-MPMAB problem must have regret guarantees not much better than that of RobustAgg(ϵ) for a large family of $\frac{\epsilon}{2}$-MPMAB problem instances.

Theorem 9. *Fix $\epsilon \ge 0$. Let $\mathcal{A}$ be an algorithm and $C > 0, \alpha > 0$ be constants, such that $\mathcal{A}$ has $CT^{1-\alpha}$ regret in all ϵ-MPMAB environments. Then, for any Bernoulli $\frac{\epsilon}{2}$-MPMAB instance $\mu = (\mu_i^p)_{i\in[K],p\in[M]}$ such that $\mu_i^p \in [\frac{15}{32}, \frac{17}{32}]$ for all i and p, we have:*

$$\mathbb{E}_\mu[\mathcal{R}(T)] \ge \Omega\left(\sum_{i\in\mathcal{I}_{\epsilon/20}^C}\sum_{p\in[M]:\Delta_i^p>0}\frac{\ln(\Delta_i^p T^\alpha/C)}{\Delta_i^p} + \sum_{i\in\mathcal{I}_{\epsilon/20}:\Delta_i^{\min}>0}\frac{\ln(\Delta_i^{\min} T^\alpha/C)}{\Delta_i^{\min}}\right).$$

Theorem 9 is nearly tight compared with the upper bound presented in Theorem 5 with two differences.

First, the upper bound is in terms of $\mathcal{I}_\epsilon$, while the lower bound is in terms of $\mathcal{I}_{\epsilon/20}$; we leave the possibility of exploiting data aggregation for arms in $\mathcal{I}_\epsilon \setminus \mathcal{I}_{\epsilon/20}$ as an open question. Second, the upper bound has an extra $O(\sum_{i\in\mathcal{I}_\epsilon} M\Delta_i^{\min})$ term, caused by the players issuing arm pulls in parallel in each round; we conjecture that it may be possible to remove this term by developing more efficient multi-player exploration strategies.

Gap-independent lower bound. The following theorem shows that, there exists a value of ϵ (that depends on T and $|\mathcal{I}_\epsilon|$), such that any algorithm must have a minimax collective regret not much lower than the upper bound shown in Theorem 8 in the family of all ϵ-MPMAB problems.

Theorem 10. *For any $K \geq 2, M, T \in \mathbb{N}$, and l, l^C in $\mathbb{N}$ such that $l \leq K-1, l+l^C = K$, there exists some $\epsilon > 0$, such that for any algorithm $\mathcal{A}$, there exists an ϵ-MPMAB problem instance, in which $|\mathcal{I}_\epsilon| = l$, and $\mathcal{A}$ has a collective regret at least $\Omega(M\sqrt{(l^C-1)T} + \sqrt{MlT})$.*

The above lower bound is nearly tight in light of the upper bound in Theorem 8: as long as $T = \Omega(KM)$, the upper and lower bounds match within a constant.

4 ϵ-MPMAB with Unknown ϵ

We now turn to the setting when ϵ is unknown to the learner. Unlike the RobustAgg(ϵ) algorithm developed in the last section, which only has nontrivial regret guarantees for all ϵ-MPMAB instances, in this section, we aim to design algorithms that have nontrivial regret guarantees for all MPMAB instances.

Recall that RobustAgg(ϵ) relies on the knowledge of ϵ to construct reward confidence intervals for each arm and player; when ϵ is unknown, constructing such confidence interval becomes a big challenge. In Appendix I, we give evidence showing that it may be impossible to design confidence interval-based algorithms that significantly benefit from inter-player information sharing. This suggests that new algorithmic ideas seem necessary to obtain nontrivial results in this setting.

4.1 Gap-dependent lower bound

Recall that Ind-UCB achieves a gap-dependent regret bound of $O\left(\sum_{i\in[K]}\sum_{p\in[M]:\Delta_i^p>0}\frac{\ln T}{\Delta_i^p}\right)$ for all MPMAB problems without knowing ϵ. Interestingly, we show in the following theorem that any sublinear regret algorithm for the MPMAB problem must have gap-dependent lower bound not much better than Ind-UCB for a large family of MPMAB problem instances, regardless of the value of ϵ and the size of $\mathcal{I}_\epsilon$ of that instance.

Theorem 11. *Let $\mathcal{A}$ be an algorithm and $C > 0, \alpha > 0$ be constants such that $\mathcal{A}$ has $CT^{1-\alpha}$ regret in all MPMAB problem instances. Then, for any Bernoulli MPMAB instance $\mu = (\mu_i^p)_{i\in[K],p\in[M]}$ such that $\mu_i^p \in [\frac{15}{32}, \frac{17}{32}]$ for all $i \in [K], p \in [M]$,*

$$\mathbb{E}_\mu[\mathcal{R}(T)] \geq \Omega\left(\sum_{i\in[K]}\sum_{p\in[M]:\Delta_i^p>0}\frac{\ln(T^\alpha\Delta_i^p/C)}{\Delta_i^p}\right).$$

4.2 Gap-independent upper bound

While we have shown gap-dependent lower bounds that nearly matches the upper bounds for Ind-UCB for sublinear regret MPMAB algorithms in Theorem 11, this does not rule out the possibility of achieving regret that improves upon Ind-UCB in small-gap instances. To see this, note that if Δ_i^p is of order $O(T^{-\alpha})$ for all i in $[K]$ and p in $[M]$, the above lower bound becomes vacuous. Therefore, it is still possible to get gap-independent upper bounds that improve over the $\tilde{O}(M\sqrt{KT})$ upper bound by Ind-UCB.

We present RobustAgg-Agnostic in Appendix F, an algorithm that achieves such guarantee: specifically, it achieves a gap-independent regret upper bound adaptive to $|\mathcal{I}_{2\epsilon}|$. In a nutshell, the algorithm aggregates over a set of RobustAgg(ϵ) base learners with different values of ϵ, using the strategy of Corral (Agarwal et al., 2017). We have the following theorem:

Theorem 12. *Let* RobustAgg-Agnostic *run on an ϵ-MPMAB problem instance with any $\epsilon \in [0,1]$. Its expected collective regret in a horizon of T rounds satisfies*

$$\mathbb{E}[\mathcal{R}(T)] \leq \tilde{O}\left(\left(|\mathcal{I}_{2\epsilon}| + M\left|\mathcal{I}_{2\epsilon}^C\right|\right)\sqrt{T} + M|\mathcal{I}_\epsilon|\right).$$

Under the mild assumption that $T = \Omega(\min(K^2, M^2))$, the above regret bound becomes $\tilde{O}\left(\left(|\mathcal{I}_{2\epsilon}| + M\left|\mathcal{I}_{2\epsilon}^C\right|\right)\sqrt{T}\right)$. If furthermore $|\mathcal{I}_{2\epsilon}| = K - o(\sqrt{K})$ and $M = \omega(\sqrt{K})$, the regret bound of RobustAgg-Agnostic is of lower order than Ind-UCB's $\tilde{O}(M\sqrt{KT})$ regret guarantee. In the most favorable case when $|\mathcal{I}_{2\epsilon}| = K-1$, RobustAgg-Agnostic has expected collective regret $\tilde{O}\left((M+K)\sqrt{T}\right)$.

Such adaptivity of RobustAgg-Agnostic to unknown similarity structure comes at a price of higher minimax regret guarantee: when $\mathcal{I}_\epsilon = \emptyset$, RobustAgg-Agnostic has a regret of $\tilde{O}\left(MK\sqrt{T}\right)$, a factor of $\sqrt{K}$ higher than $\tilde{O}(M\sqrt{KT})$, the worst-case regret of Ind-UCB. We conjecture that this may be unavoidable due to lack of knowledge of ϵ, similar to results in adaptive Lipschitz bandits (Locatelli and Carpentier, 2018; Krishnamurthy et al., 2019; Hadiji, 2019).

5 Related Work and Comparisons

5.1 Multi-agent bandits.

We first compare existing multi-agent bandit learning problems with the ϵ-MPMAB problem. We provide a more detailed review of the literature in Appendix A.

A large portion of prior studies (Kar et al., 2011; Szörényi et al., 2013; Landgren et al., 2016; Cesa-Bianchi et al., 2019; Kolla et al., 2018; Sankararaman et al., 2019; Wang et al., 2019; Dubey and Pentland, 2020a; Chawla et al., 2020; Wang et al., 2020) focuses on the setting where a set of players collaboratively work on one bandit learning problem instance, i.e., the reward distributions of an arm are identical across all players. In contrast, we study multi-agent bandit learning where the reward distributions across players can be different.

Multi-agent bandit learning with heterogeneous feedback has also been covered by previous studies. In (Shahrampour et al., 2017), a group of players seek to find the arm with the largest average reward over all players; however, in each round, the players have to reach a consensus and choose the same arm. Cesa-Bianchi et al. (2013) study a network of linear contextual bandit players with heterogeneous rewards, where the players can take advantage of reward similarities hinted by a graph. They use a Laplacian-based regularization, whereas we study when and how to use information from other players based on a dissimilarity parameter. Gentile et al. (2014); Li et al. (2016) assume that the players' reward distributions have a cluster structure; in addition, players that belong to one cluster share a *common* reward distribution; our paper do not assume such cluster structure. Dubey and Pentland (2020b) assume access to some side information for every player, and learns a reward predictor that takes both player's side information models and action as input. In comparison, our work do not assume access to such side information.

Similarities in reward distributions are explored in (Shivaswamy and Joachims, 2012; Zhang et al., 2019) to warm start bandit learning agents. Azar et al. (2013); Soare et al. (2014) investigate multitask learning in bandits through *sequential transfer* between tasks that have similar reward distributions. In contrast, we study the multi-player setting, where all players learn continually and concurrently.

There are other practical formulations of multi-player bandits with player-dependent reward distributions (Bistritz et al., 2020; Boursier et al., 2020), where the existence of collision is assumed; i.e., two players pulling the same arm in the same round receive zero reward. In comparison, collision is not modeled in this paper.

5.2 Learning using weighted data aggregation

Our design of confidence interval in Section 3.1 has resemblance to the weighted empirical risk minimization algorithm proposed for domain adaptation by Ben-David et al. (2010), but our purposes are different from theirs. Specifically, our choice of λ minimizes the length of the confidence intervals, whereas Ben-David et al. (2010) find λ that minimizes classification error in the target domain. Furthermore, our setting in Section 4 is more challenging: in offline domain adaptation, one may use a validation set drawn from the target domain to fine-tune the optimal weight λ^*, to adapt to unknown dissimilarity between the source and the target; however, in our setting (and online bandit learning in general), such tuning does not result in sample efficiency improvement.

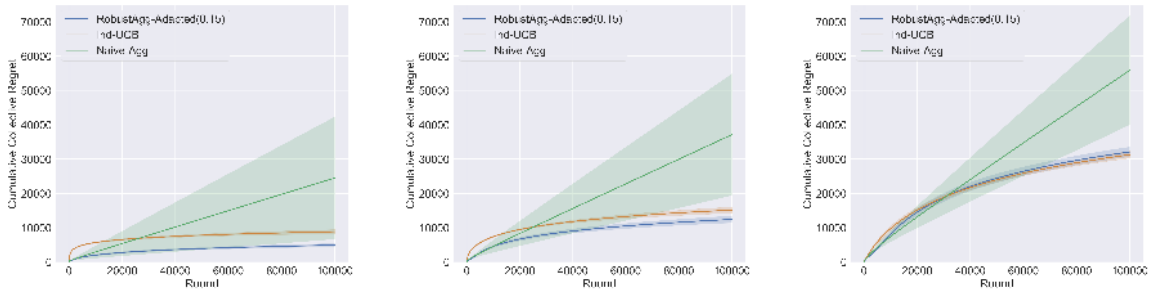
The idea of assigning weights to different sources of samples has also been studied by Zhang et al. (2019) for warm starting contextual bandit learning from misaligned distributions and by Russac et al. (2019) for online learning in non-stationary environments. Zhu et al. (2020) use a weighted compound of player-based estimator and cluster-based estimator for collaborative Thompson sampling, where the weights are given by a hyper-parameter; in contrast, we adaptively compute our weighting factor based on the numbers of samples collected by the players as well as the dissimilarity parameter ϵ.

6 Empirical Validation

We now validate our theoretical results with some empirical simulations using synthetic data[2]. Specifically, we seek to answer the following questions:

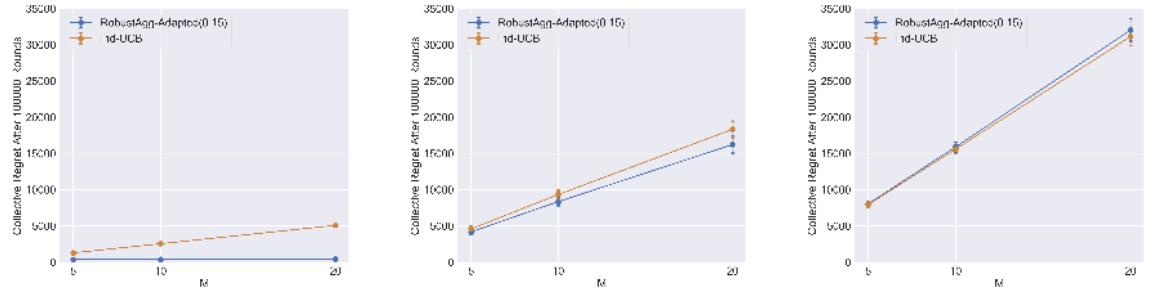
1. In practice, how does our proposed algorithm

[2]Our code is available at `https://github.com/zhiwang123/eps-MPMAB`.



(a) $|\mathcal{I}_\epsilon| = 8$ (b) $|\mathcal{I}_\epsilon| = 6$ (c) $|\mathcal{I}_\epsilon| = 0$

Figure 1: Compares the average performance of RobustAgg-Adapted(0.15), Ind-UCB, and Naive-Agg in randomly generated Bernoulli 0.15-MPMAB problem instances with $K = 10$ and $M = 20$. The x-axis shows a horizon of $T = 100,000$ rounds, and the y-axis shows the cumulative collective regret of the players.



(a) $|\mathcal{I}_\epsilon| = 9$ (b) $|\mathcal{I}_\epsilon| = 5$ (c) $|\mathcal{I}_\epsilon| = 0$

Figure 2: Compares the average performance of RobustAgg-Adapted(0.15) and Ind-UCB in randomly generated Bernoulli 0.15-MPMAB problem instances with $K = 10$. The x-axis shows different values of M, and the y-axis shows the cumulative collective regret of the players after $100,000$ rounds.

compare with algorithms that either do not take advantage of adaptive data aggregation or do not execute aggregation in a robust fashion?

2. How does the performance of our algorithm change with different numbers of subpar arms?

We note that these questions are considered in the setting where the dissimilarity parameter ϵ is known to the algorithms.

6.1 Experimental setup

We first describe the algorithms compared in the simulations. We then discuss the procedure we used for generating synthetic data.

RobustAgg-Adapted(ϵ). Since standard concentration bounds are loose in practice, we performed simulations on a more practical and aggressive variant of RobustAgg(ϵ), which we call RobustAgg-Adapted(ϵ). Our adaptation involves two minor modifications:

- We used a different constant "2" inside the square root in the UCBs; this constant was taken from the original UCB-1 algorithm, which is an ingredient of the baseline Ind-UCB, and we simply kept the default value.
- We also added an initialization phase, where each player pulls each arm once, to match with UCB-1 which has this phase.

A pseudocode of RobustAgg-Adapted(ϵ) can be found in Appendix G.

Baselines. We evaluate the following two algorithms as baselines: (a) Ind-UCB, described in Section 2; and (b) Naive-Agg, in which the players *naively* aggregate data assuming that their reward distributions are identical—in other words, Naive-Agg is equivalent to RobustAgg-Adapted(0).

Instance generation. We generated problem instances using the following *randomized* procedure. We first set $\epsilon = 0.15$. Then, given the number of players M, the number of arms K, and the number of subpar arms $|\mathcal{I}_\epsilon| \in \{0, 1, \ldots, K-1\}$, we first sampled the means of the reward distributions for player 1:

Let $c = K - |\mathcal{I}_\epsilon|$. For $i \in \{1, 2, \ldots, c\}$, we sampled $\mu_i^1 \overset{\text{i.i.d.}}{\sim} \mathcal{U}[0.8, 0.8 + \epsilon)$, where $\mathcal{U}[a, b)$ is the uniform distribution with support $[a, b]$. Let $d = \max_{i \in [c]} \mu_i^1$. Then, for $i \in \{c + 1, \ldots, K\}$, we sampled $\mu_i^1 \overset{\text{i.i.d.}}{\sim} \mathcal{U}[0, d - 5\epsilon)$.

We then sampled the means of the reward distributions for players $p \in \{2, \ldots, M\}$: For each $i \in [K]$, we sampled $\mu_i^p \overset{\text{i.i.d.}}{\sim} \mathcal{U}\big[\max(0, \mu_i^1 - \frac{\epsilon}{2}), \min(\mu_i^1 + \frac{\epsilon}{2}, 1)\big)$.

Fact 13. *The above construction gives a Bernoulli* 0.15*-MPMAB problem instance that has exactly* $(K - c)$ *subpar arms, namely,* $\mathcal{I}_\epsilon = \{i : c + 1 \le i \le K\}$.

6.2 Simulations and results

We ran two sets of simulations, and the results are shown in Figure 1 and Figure 2. More detailed results are deferred to Appendix G.

Experiment 1. We compare the cumulative collective regrets of the three algorithms in problem instances with *different numbers of subpar arms*. We set $M = 20$, $K = 10$ and $\epsilon = 0.15$. For each $v \in \{0, 1, 2, \ldots, 9\}$, we generated 30 Bernoulli 0.15-MPMAB problem instances, each of which has exactly v subpar arms, i.e., we generated instances with $|\mathcal{I}_\epsilon| = v$. Figures 1a, 1b and 1c show the average regrets in a horizon of $100,000$ rounds over these generated instances, in which $|\mathcal{I}_\epsilon| = 8, 6$ and 0, respectively. In the interest of space, figures in which $|\mathcal{I}_\epsilon|$ takes other values are deferred to Appendix G.3.

Notice that RobustAgg-Adapted(0.15) outperforms both baseline algorithms in Figures 1a and 1b when $|\mathcal{I}_\epsilon| = 8$ and 6. Figure 1c demonstrates that when $|\mathcal{I}_\epsilon| = 0$, i.e., when there is no arm that is amenable to data aggregation, the performance of RobustAgg-Adapted(0.15) is still on par with that of Ind-UCB. Also, as shown in Figure 1a, even when $|\mathcal{I}_\epsilon^C| = 2$, i.e., when there are only two "competitive" (not subpar) arms, the collective regret of Naive-Agg can still easily be nearly linear in the number of rounds.

Experiment 2. We study how the collective regrets of RobustAgg-Adapted(0.15) and Ind-UCB scale with the *number of players* in problem instances with different numbers of subpar arms. We set $K = 10$ and $\epsilon = 0.15$. For each combination of $M \in \{5, 10, 20\}$ and $v \in \{0, 1, 2, \ldots, 9\}$, we generated 30 Bernoulli 0.15-MPMAB problem instances with M players and exactly v subpar arms, that is, for each instance, $|\mathcal{I}_\epsilon| = v$. Figures 2a, 2b and 2c compare the average regrets after $100,000$ rounds in instances with different numbers of players M, in which $|\mathcal{I}_\epsilon|$ are set to be $9, 5$ and 0, respectively. Again, figures in which $|\mathcal{I}_\epsilon|$ takes other values are deferred to Appendix G.3.

Observe that when $|\mathcal{I}_\epsilon|$ is large, the collective regret of RobustAgg-Adapted(0.15) is less sensitive to the number of players. In the extreme case when $|\mathcal{I}_\epsilon| = 9$, all suboptimal arms are subpar arms, and Figure 2a shows that the collective regret of RobustAgg-Adapted(0.15) has negligible dependence on the number of players M.

6.3 Discussion

Back to the questions we raised earlier, our simulations show that RobustAgg-Adapted(ϵ), in general, outperforms the baseline algorithms Ind-UCB and Naive-Agg. When the set of subpar arms $\mathcal{I}_\epsilon$ is large, we showed that properly managing data aggregation can substantially improve the players' collective performance in an ϵ-MPMAB problem instance. When there is no subpar arm, we demonstrated the robustness of RobustAgg-Adapted(ϵ), that is, its performance is comparable with Ind-UCB, in which the players do not share information. These empirical results validate our theoretical analyses in Section 3.

7 Conclusion and Future Work

In this paper, we studied multitask bandit learning from heterogeneous feedback. We formulated the ϵ-MPMAB problem and showed that whether inter-player information sharing can boost the players' performance depends on the dissimilarity parameter ϵ as well as the intrinsic difficulty of each individual bandit problem that the players face. In particular, in the setting where ϵ is known, we presented a UCB-based data aggregation algorithm which has near-optimal instance-dependent regret guarantees. We also provided upper and lower bounds in the setting where ϵ is unknown.

There are many avenues for future work. For example, we are interested in extending our results to contextual bandits and Markov decision processes. Another direction is to study multitask bandit learning under other interaction protocols (e.g., only a subset of players take actions in each round). In the future, we would also like to evaluate our algorithms in real-world applications such as healthcare robotics (Riek, 2017).

8 Acknowledgments

We thank Geelon So and Gaurav Mahajan for insightful discussions. We also thank the National Science Foundation under IIS 1915734 and CCF 1719133 for research support. Chicheng Zhang acknowledges

startup funding support from the University of Arizona.

References

A. Agarwal, H. Luo, B. Neyshabur, and R. E. Schapire. Corralling a band of bandit algorithms. In *Conference on Learning Theory*, pages 12–38. PMLR, 2017.

R. Arora, T. V. Marinov, and M. Mohri. Corralling stochastic bandit algorithms. *arXiv preprint arXiv:2006.09255*, 2020.

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

M. G. Azar, A. Lazaric, and E. Brunskill. Sequential transfer in multi-armed bandit with finite set of models. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2220–2228. Curran Associates, Inc., 2013.

S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine learning*, 79(1-2): 151–175, 2010.

I. Bistritz, T. Baharav, A. Leshem, and N. Bambos. My fair bandit: Distributed learning of max-min fairness with multi-player bandits. In *Proceedings of the 37th International Conference on Machine Learning*, pages 11–21, 2020.

E. Boursier, E. Kaufmann, A. Mehrabian, and V. Perchet. A practical algorithm for multiplayer bandits when arm means vary among players. volume 108 of *Proceedings of Machine Learning Research*, pages 1211–1221, 2020.

E. Brunskill and L. Li. Sample complexity of multi-task reinforcement learning. *UAI*, 2013.

N. Cesa-Bianchi, C. Gentile, and G. Zappella. A gang of bandits. In *Advances in Neural Information Processing Systems*, pages 737–745, 2013.

N. Cesa-Bianchi, C. Gentile, and Y. Mansour. Delay and cooperation in nonstochastic bandits. *The Journal of Machine Learning Research*, 20(1):613–650, 2019.

R. Chawla, A. Sankararaman, A. Ganesh, and S. Shakkottai. The gossiping insert-eliminate algorithm for multi-agent bandits. volume 108 of *Proceedings of Machine Learning Research*, pages 3471–3481, 2020.

A. Dubey and A. Pentland. Cooperative multi-agent bandits with heavy tails. In *Proceedings of the 37th International Conference on Machine Learning*, pages 451–460, 2020a.

A. Dubey and A. Pentland. Kernel methods for cooperative contextual bandits. In *Proceedings of the 37th International Conference on Machine Learning*, pages 428–450, 2020b.

C. Gentile, S. Li, and G. Zappella. Online clustering of bandits. In *International Conference on Machine Learning*, pages 757–765, 2014.

H. Hadiji. Polynomial cost of adaptation for x-armed bandits. In *Advances in Neural Information Processing Systems*, pages 1029–1038, 2019.

S. Kar, H. V. Poor, and S. Cui. Bandit problems in networks: Asymptotically efficient distributed allocation rules. In *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 1771–1778. IEEE, 2011.

R. K. Kolla, K. Jagannathan, and A. Gopalan. Collaborative learning of stochastic bandits over a social network. *IEEE/ACM Transactions on Networking*, 26(4):1782–1795, 2018.

A. Krishnamurthy, J. Langford, A. Slivkins, and C. Zhang. Contextual bandits with continuous actions: Smoothing, zooming, and adapting. In *Conference on Learning Theory*, pages 2025–2027, 2019.

A. Kubota, E. I. Peterson, V. Rajendren, H. Kress-Gazit, and L. D. Riek. Jessie: Synthesizing social robot behaviors for personalized neurorehabilitation and beyond. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pages 121–130, 2020.

T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

P. Landgren, V. Srivastava, and N. E. Leonard. Distributed cooperative decision-making in multiarmed bandits: Frequentist and bayesian algorithms. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 167–172. IEEE, 2016.

T. Lattimore and C. Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, page 661–670, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781605587998. doi: 10.1145/1772690.1772758. URL `https://doi.org/10.1145/1772690.1772758`.

S. Li, A. Karatzoglou, and C. Gentile. Collaborative filtering bandits. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 539–548, 2016.

A. Locatelli and A. Carpentier. Adaptivity to smoothness in x-armed bandits. In *Conference on Learning Theory*, pages 1463–1492, 2018.

A. Pacchiano, M. Phan, Y. Abbasi-Yadkori, A. Rao, J. Zimmert, T. Lattimore, and C. Szepesvari. Model selection in contextual stochastic bandit problems. *arXiv preprint arXiv:2003.01704*, 2020.

L. D. Riek. Healthcare robotics. *Communications of the ACM*, 60(11):68–78, 2017.

M. T. Rosenstein, Z. Marx, L. P. Kaelbling, and T. G. Dietterich. To transfer or not to transfer. In *NIPS 2005 workshop on transfer learning*, 2005.

Y. Russac, C. Vernade, and O. Cappé. Weighted linear bandits for non-stationary environments. In *Advances in Neural Information Processing Systems*, pages 12017–12026, 2019.

A. Sankararaman, A. Ganesh, and S. Shakkottai. Social learning in multi agent multi armed bandits. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3):1–35, 2019.

S. Shahrampour, A. Rakhlin, and A. Jadbabaie. Multi-armed bandits in multi-agent networks. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2786–2790. IEEE, 2017.

W. Shen, J. Wang, Y.-G. Jiang, and H. Zha. Portfolio choices with orthogonal bandit learning. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 974–980. AAAI Press, 2015. ISBN 9781577357384.

P. Shivaswamy and T. Joachims. Multi-armed bandit problems with history. In *Artificial Intelligence and Statistics*, pages 1046–1054, 2012.

M. Soare, O. Alsharif, A. Lazaric, and J. Pineau. Multi-task linear bandits. *NIPS2014 Workshop on Transfer and Multi-task Learning : Theory meets Practice*, 2014.

B. Szörényi, R. Busa-Fekete, I. Hegedűs, R. Ormándi, M. Jelasity, and B. Kégl. Gossip-based distributed stochastic bandit algorithms. In *Journal of Machine Learning Research Workshop and Conference Proceedings*, volume 2, pages 1056–1064. International Machine Learning Society, 2013.

S. S. Villar, J. Bowden, and J. Wason. Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 30(2):199, 2015.

P.-A. Wang, A. Proutiere, K. Ariu, Y. Jedra, and A. Russo. Optimal algorithms for multiplayer multi-armed bandits. volume 108 of *Proceedings of Machine Learning Research*, pages 4120–4129, 2020.

Y. Wang, J. Hu, X. Chen, and L. Wang. Distributed bandit learning: How much communication is needed to achieve (near) optimal regret. *arXiv preprint arXiv:1904.06309*, 2019.

C. Zhang, A. Agarwal, H. Daumé III, J. Langford, and S. Negahban. Warm-starting contextual bandits: Robustly combining supervised and bandit feedback. In *International Conference on Machine Learning*, pages 7335–7344, 2019.

Z. Zhu, L. Huang, and H. Xu. Collaborative thompson sampling. *Mobile Networks and Applications*, 2020. doi: 10.1007/s11036-019-01453-x.