Collaborative Learning and Personalization in Multi-Agent Stochastic Linear Bandits

Avishek Ghosh[⋆,†], Abishek Sankararaman[⋆,‡] and Kannan Ramchandran[†]

Department of Electrical Engineering and Computer Sciences, UC Berkeley[†]
AWS AI, Palo Alto, USA[‡]

email: avishek_ghosh@berkeley.edu, abisanka@amazon.com,
kannanr@eecs.berkeley.edu

June 17, 2021

Abstract

We consider the problem of minimizing regret in an N agent heterogeneous stochastic linear bandits framework, where the agents (users) are similar but not all identical. Our problem is motivated by recommendation systems that enjoy the *more the merrier phenomenon*; as more people join the system, the service improves for everyone, as the system can learn *across heterogenous users*. We model user heterogeneity using two popularly used ideas in practice; *(i)* A clustering framework where users are partitioned into groups with users in the same group being identical to each other, but different across groups, and *(ii)* a personalization framework where no two users are necessarily identical, but a user's parameters are close to that of the population average. In the clustered users' setup, we propose a novel algorithm, based on successive refinement of cluster identities and regret minimization. We show that, for any agent, the regret scales as $\mathcal{O}(\sqrt{T/N})$, if the agent is in a 'well separated' cluster, or scales as $\mathcal{O}(T^{\frac{1}{2}+\varepsilon}/(N)^{\frac{1}{2}-\varepsilon})$ if its cluster is not well separated, where ε is positive and arbitrarily close to 0. Our algorithm is adaptive to the cluster separation, and is parameter free—it does not need to know the number of clusters, separation and cluster size —yet the regret guarantee adapts to the inherent complexity. In the personalization framework, we introduce a natural algorithm where, the personal bandit instances are initialized with the estimates of the global average model. We show that, an agent i whose parameter deviates from the population average by ϵ_i, attains a regret scaling of $\widetilde{O}(\epsilon_i\sqrt{T})$. This demonstrates that if the user representations are close (small ϵ_i), the resulting regret is low, and vice-versa. The results are empirically validated and we observe superior performance of our adaptive algorithms over non-adaptive baselines.

1 Introduction

Large scale web recommendation systems have become ubiquitous in the modern day, due to a myriad of applications that use them including online shopping services, video streaming services, news and article recommendations, restaurant recommendations etc, each of which are used by thousands, if not more users, across the world. For each user, these systems make repeated decisions under uncertainty, in order to better learn the preference of each individual user and serve them. A unique feature these large platforms have is that of *collaborative learning* —namely applying the learning from one user to improve the performance on another [Lee01]. However, the sequential online setting renders this complex, as two users are seldom identical [PEZ+20].

⋆Equal contribution.

We study the problem of multi-user contextual bandits [CMB20], and quantify the gains obtained by collaborative learning under user heterogeneity. Towards this end, we propose two models of user-heterogeneity: (a) clustering framework where only users in the same group are identical (b) personalization framework where no two users are necessarily identical, but are close to the population average. Both these models are widely used in practical systems involving a large number of users (ex. [PEZ+20, LSY03, SKKR01, SKKR02, LK03]). User clustering in such systems can be induced from a variety of factors such as affinity to similar interests, age-groups etc [Ozs16, LLCS15, SM14]. The personalization framework in these systems is also a natural in many neural network models, wherein users represented by learnt embedding vectors are not identical; nevertheless similar users are embedded nearby [XDZ+17, ZDF17, OTOT17, Ozs16].

Formally, our model consists of N users, all part of a common platform. The interaction between the agents and platform proceeds in a sequence of rounds. At the beginning of each round, the platform receives K context vectors corresponding to K items from the environment. The platform then recommends an item to each user and receives feedback from the users about the item. We posit that associated with each user i, is an preference vector θ_i^*, initially unknown to the platform. In any round, the average reward (the feedback) received by agent i for a recommendation of item, is the inner product of θ_i^* with the context vector of the recommended item. The goal of the platform is to maximize the reward collected over a time-horizon of T rounds. Following standard terminology, we henceforth refer to an "arm" and item interchangeably, and thus "recommending item k" is synonymous to "playing arm k". We also use agents and users interchangeably.

Example Application: Our setting is motivated through a caricature of a news recommendation system serving N users and K publishers [LCLS10]. Each day, each of the K publishers, publishes a news article, which corresponds to the context vector in our contextual bandit framework. In practice, one can use standard tools to embed articles in vector spaces, where the dimensions correspond to topics such as politics, religion, sports etc ([WTAL16]). The user preference indicates the interest of a user, and the reward, being computed as an inner product of the context vector and the user preference, models the observation that the more aligned an article is to a user's interest, the higher the reward.

For both frameworks, we propose *adaptive* algorithms; in the clustering setup, we propose Successive Clustering of Linear Bandits (SCLB), which is agnostic to the number of clusters, the gap between clusters and the cluster size. Yet SCLB yields regret that depends on these parameters, and is thus adaptive. In the personalization framework, our proposed algorithm, namely Personalized Multi-agent Linear Bandits (PMLB) adapts to the level of common representation across users. In particular, if an agents' preference vector is close to the population average, PMLB exploits that and incurs low regret for this agent due to collaboration. On the other hand if an agent's preference vector is far from the population average, PMLB yields a regret similar to that of OFUL [CMB20] or Linear Bandit algorithms [AyPS11] that do not benefit from multi-agent collaboration.

1.1 Our Contributions:

In the clustering framework, we give a novel multi-phase, successive refinement based algorithm, SCLB, that achieves collaborative gain, even in the most adverse settings. SCLB does not need any knowledge of the instance, such as the number of clusters, cluster sizes or the cluster separation; but is adaptive. Formally, we show that the regret of SCLB is $\mathcal{O}(\sqrt{T/N})$, if the agent is in a 'well separated' cluster, or $\mathcal{O}(T^{\frac{1}{2}+\varepsilon}/(N)^{\frac{1}{2}-\varepsilon})$ if its cluster is not well separated, where ε is positive and arbitrarily close to 0. *This result holds true, even in the limit when the cluster separation approaches* 0. This shows that when the underlying instance gets harder to cluster, the regret is increased. Nevertheless, despite the clustering being hard to accomplish, every user still experiences collaborative gain of $N^{1/2-\varepsilon}$ and regret sub-linear in T. On the other hand if the clustering task is easy i.e., well-separated, then the regret rate *matches that of an oracle that knows the cluster identities.*

It is worth pointing out that SCLB works for *all* ranges of separation, which is starkly different from standard algorithms in bandit clustering ([GLZ14, GLK+17, KSL16]) and statistics ([BWY+17, KC20]).

We now compare our results to CLUB, proposed in [GLZ14], that can be modified to be applicable to our setting (c.f. Section 6). First, CLUB is non-adaptive and its regret guarantees hold only when the clusters are separated. Second, even in the separated setting, the separation (gap) cannot be lower than $\mathcal{O}(1/T^{1/4})$ for CLUB, while it can be as low as $\mathcal{O}(1/T^{\alpha})$, where $\alpha < 1/2$ for SCLB. Moreover, in simulations (Section 6) we observe that SCLB outperforms CLUB in a variety of synthetic and a real data setting.

In the personalization framework, we define the *factor of common representation* for agent i as $\epsilon_i := \|\theta_i^* - \frac{1}{N}\sum_{l=1}^{N}\theta_l^*\|$, where $\theta_i^* \in \mathbb{R}^d$ is agent i's representation, and $\frac{1}{N}\sum_{l=1}^{N}\theta_i^*$ is the average representation of N agents. We propose an algorithm PMLB, which adapts to ϵ_i gracefully (without knowing it apriori) and yields a regret of $\mathcal{O}(\epsilon_i\sqrt{dT})$. Hence, if the agents share more and more representations, i.e., ϵ_i is small, then PMLB obtains low regret. On the other hand, if ϵ_i is large, say $\mathcal{O}(1)$, the agents do not share a common representation, the regret of PMLB is $\mathcal{O}(\sqrt{dT})$, which matches that obtained by each agent playing OFUL, independently of other agents. Thus, PMLB benefits from collaborative learning and obtains small regret, if the problem structure admits, else the regret matches the baseline strategy of every agent running an independent bandit instance.

Empirical Validation: We verify our theoretical insights through simulations, both on synthetic and Last.FM real dataset [GLZ14]. We compare with two baselines —one is CLUB of [GLZ14], and the other is the baseline where every agent runs an independent bandit model, i.e., no collaboration. We observe that our algorithms have superior performance compared to the baselines in a variety of settings.

2 Related Work

Collaborative gains in multi-user recommendation systems have long been studied in Information retrieval and recommendation systems community (ex. [LK03, SKKR02, LSY03, Lee01]). The focus has been in developing effective ideas to help practitioners deploy large scale systems. Empirical studies of recommendation system has seen renewed interest lately due to the integration of deep learning techniques with classical ideas (ex. [MNLD20, ZHW$^+$19, YYC$^+$20, CAS16, OTOT17, NMS$^+$19]). Motivated by the empirical success, we undertake a theoretical approach to quantify collaborative gains achievable in a contextual bandit setting. Contextual bandits has proven to be fruitful in modeling sequential decision making in many applications [LCLS10, CBGZ13, GLZ14].

The paper of [GLZ14] is closest to our clustering setup, where in each round, the platform plays an arm for a single randomly chosen user. As outlined before, our algorithm obtains a superior performance, both in theory and empirically. For personalization, the recent paper of [YHLD21] is the closest, which posits all users's parameters to be in a common low dimensional subspace. [YHLD21] proposes a learning algorithm under this assumption. In contrast, we make no parametric assumptions, and demonstrate an algorithm that achieves collaboration gain, if there is structure, while degrading gracefully to the simple baseline of independent bandit algorithms in the absence of structure.

The framework of personalized learning has been exploited in a great detail in representation learning and meta-learning. While [DTB$^+$19, LR11, RCG$^+$15, HPR$^+$17, PBS15] learns common representation across agents in the RL framework, [ADK$^+$20] uses it for imitation learning. We remark that representation learning is also closely connected to meta-learning [DCGP19, FRKL19, KBT19], where close but a common initialization is learnt from leveraging non identical but similar representations. Furthermore, in Federated learning, the problem of personalization is a well studied problem (see [MMRS20, FMO20b, FMO20a]).

3 Problem Setup

Our system consists of N users, interacting with a centralized system (termed as 'center' henceforth) repeatedly over T rounds. At the beginning of each round, environment provides the center with K context vectors corresponding to K arms, and for each user, the center recommends one of the K arms to play. At the end of the round, every user receives a reward for the arm played, which is observed by the center. The K context vectors in round t are denoted by $\beta_t = [\beta_{1,t}, \ldots, \beta_{K,t}] \in \mathbb{R}^{d\times K}$. Each user i, is associated with a preference

Algorithm 1: Successive Clustering of Linear Bandits (SCLB)

1: **Input:** No. of users N, horizon T, parameter $\alpha < 1/2$, constant C, high probability bound δ
2: **for** phases $1 \le i \le \log_2(T)$ **do**
3: Play CMLB ($\gamma = 3/(N2^i)^\alpha$, horizon $T = 2^i$, high probability $\delta/2^i$, cluster-size $p^* = i^{-2}$)
4: **end for**

vector $\theta_i^* \in \mathbb{R}^d$, and the reward user i obtains from playing arm j at time t is is given by $\langle \beta_{j,t}, \theta^* \rangle + \xi_t$. The $(\xi_t)_{t\ge1}$ and $(\beta_t)_{t\ge1}$ are random variables, whose distribution is described below.

We follow the stochastic framework [AyPS11, CMB20] and denote by $\mathcal{F}_{t-1}$, as the sigma algebra generated by all noise random variables upto and including time $t-1$. We denote by $\mathbb{E}_{t-1}(.)$ as the conditional expectation operator with respect to $\mathcal{F}_{t-1}$. We assume that the $(\xi_t)_{t\ge1}$ are conditionally sub-Gaussian noise with known parameter σ, conditioned on all the arm choices and realized rewards in the system upto and including time $t-1$. Without loss of generality, we assume $\sigma = 1$ throughout. The contexts $\beta_{i,t} \in \mathbb{B}_2^d(1)$ are assumed to be drawn independent of both the past and $\{\beta_{j,t}\}_{j\neq i}$, satsifying

$$\mathbb{E}_{t-1}[\beta_{i,t}] = 0 \qquad \mathbb{E}_{t-1}[\beta_{i,t}\,\beta_{i,t}^\top] \succeq \rho_{\min} I. \tag{1}$$

This means that the conditional mean of the covariance matrix is zero and the conditional covariance matrix is positive definite with minimum eigenvlaue at least $\rho_{\min}$. These assumptions are standard and widely used in past literature on stochastic linear bandits [FKL19, CMB20]. Also, since the context vectors are drawn from unit sphere (and hence sub-Gaussian), we have $\rho_{\min} \le 1/d$, and hence one needs to track the dependence on $\rho_{\min}$. Observe that our stochastic assumption also includes the simple setting where the contexts evolve according to a random process independent of the actions and rewards from the learning algorithm.

Throughout the paper, we use the linear bandit algorithm, namely OFUL (see [AyPS11]) as a blackbox, and use it in a judicious way for clustering and personalization. In particular we use a variant of the OFUL as prescribed in [CMB20][1]. At time t, we denote by $B_{i,t} \in [K]$ to be the arm played by any agent in cluster i with preference vector θ_i^*. The corresponding regret, over a time horizon of T is given by

$$R_i(T) = \sum_{t=1}^{T} \mathbb{E} \max_{j\in[K]} \langle \theta_i^*, \beta_{j,t} - \beta_{B_{i,t},t} \rangle \tag{2}$$

4 Clustering framework for multi-agent contextual bandits

In this section, we assume that the users are clustered into L groups, with $p_i \in (0,1]$ denoting the fraction of users in cluster i. All users in the same cluster have the same context vector, and thus without loss of generality, for all clusters $i \in [L]$, we denote by $\theta_i^* \in \mathbb{R}^d$, to be the preference vector of any user of cluster i. We define the *separation parameter*, or SNR (signal to noise ratio) of cluster $i \in [L]$ as $\Delta_i := \min_{j\in[L]\setminus\{i\}} \|\theta_i^* - \theta_j^*\|$, smallest distance to another cluster.

Learning Algorithm: We propose the Successive Clustering of Linear Bandits (SCLB) algorithm in Algorithm 1. SCLB does not need any knowledge of the gap $\{\Delta_i\}_{i=1}^L$, the number of clusters L or the cluster size fractions $\{p_i\}_{i=1}^L$. Nevertheless, SCLB adapts to the problem SNR and yields regret accordingly. One attractive feature of Algorithm 1 is that it works uniformly *for all* ranges of the gap $\{\Delta_i\}_{i=1}^L$. This is in sharp contrast with the existing algorithms [GLZ14] which is only guaranteed to give good performance when the gap $\{\Delta_i\}_{i=1}^L$ are large enough. Furthermore, our uniform guarantees are in contrast with the works in standard clustering algorithms, where theoretical guarantees are only given for a sufficiently large separation (see [KC20, BWY$^+$17].)

[1]We use OFUL as used in the OSOM algorithm of [CMB20] for the linear contextual setting.

Algorithm 2: Clustered Multi-Agent Bandits (CMLB)

Input: No. of users N, horizon T, parameter $\alpha < 1/2$, constant C, high probability bound δ, threshold γ, cluster-size parameter p^*

Individual Learning Phase

$T_{\text{Explore}} \leftarrow C^{(2)} d (NT)^{2\alpha} \log(1/\delta)$

All agents play OFUL(δ) independently for T_{explore} rounds

$\{\hat{\theta}^{(i)}\}_{i=1}^{N} \leftarrow$ All agents' estimates at the end of round T_{explore}.

Cluster the Users

`User-Clusters` $\leftarrow$ `MAXIMAL-CLUSTER`$(\{\hat{\theta}^{(i)}\}_{i=1}^{N}, \gamma,\ p^*)$

Collaborative Learning Phase

Initialize one OFUL(δ) instance per-cluster

for clusters $\ell \in \{1, \ldots, |$`User-Clusters`$|\}$ **in parallel do**

  for times $t \in \{T_{\text{explore}} + 1, \cdots, T\}$ **do**

    All users in the ℓ-th cluster play the arm given by the OFUL algorithm of cluster l.

    Average of the observed rewards of all users of cluster l is used to update the OFUL(δ) state of cluster l

  end for

end for

SCLB is a multi-phase algorithm, which invokes Clustered Multi-agent Linear Bandits (CMLB) (Algorithm 2) repeatedly, by decreasing the size parameter, namely p^* polynomially and high probability parameter δ_i exponentially. Algorithm 1 proceeds in phases of exponentially growing phase length with phase $j \in \mathbb{N}$ lasting for 2^j rounds. In each phase, a fresh instance of CMLB is instantiated with high probability parameter $\delta/2^j$ and the minimum size parameter j^{-2}. Thus, as the phase length grows, the size parameter sent as input to Algorithm 2 decays. We show that this simple strategy suffices to show that the size parameter converges to p_i, and we obtain collaborative gains without knowledge of p_i.

The CMLB Sub-routine (Algorithm 2) : CMLB works in the three phases: (a) (Individual Learning) the N users play an independent linear bandit algorithm to (roughly) learn their preference; (b) (Clustering) users are clustered based on their estimates using `MAXIMAL CLUSTER` (see Algorithm 3); and (c) (Collaborative Learning) one Linear Bandit instance per cluster is initialized and all users of a cluster play the same arm. The average reward over all users in the cluster is used to update the per-cluster bandit instance. When clustered correctly, the learning is faster, as the noise variance is reduced due to averaging across users. Note that `MAXIMAL CLUSTER` algorithm requires a size parameter p^*. However, in simulations (Sec. 6), we observe that CMLB works without the size parameter. Hence, empirically we show that CMLB is sufficient for collaborative learning and successive refinement is not necessary.

4.1 Regret guarantee of SCLB

As mentioned earlier, SCLB is an adaptive algorithm that yields provable regret for *all ranges* of $\{\Delta_i\}_{i=1}^{L}$. When $\{\Delta_i\}_{i=1}^{L}$ are large, SCLB can cluster the agents perfectly, and thereafter exploit the collaborative gains across users in same cluster. On the other hand, if $\{\Delta_i\}_{i=1}^{L}$ are small, SCLB still adapts to the gap, and yields a non-trivial (but sub-optimal) regret. As a special case, if all the clusters are very close to one another, SCLB identifies that setting and treats all N agents as *one big* cluster, yielding highest collaborative gain.

Without loss of generality, in what follows, we focus on an arbitrary agent belonging to cluster i and characterize her regret. Throughout this section, we assume

$$T \geq C \frac{1}{N} \left[\frac{\tau_{\min}(\delta) \rho_{\min}}{d \log(1/\delta)} \right]^{\frac{1}{2\alpha}}, \text{ and } \tau_{\min}(\delta) = \left[\frac{16}{\rho_{\min}^2} + \frac{8}{3\rho_{\min}} \right] \log(\frac{2dT}{\delta}) \tag{3}$$

Algorithm 3: MAXIMAL-CLUSTER

1: **Input:** All estimates $\{\hat{\theta}^{(i)}\}_{i=1}^{N}$, size parameter $p^* > 0$, threshold $\gamma \geq 0$.
2: Construct an undirected Graph G on N vertices as follows: $||\widehat{\theta}_i^* - \widehat{\theta}_j^*|| \leq \gamma \Leftrightarrow i \sim_G j$
3: $\mathcal{C} \leftarrow \{C_1, \cdots, C_k\}$ all the connected components of G
4: $\mathcal{S}(p^*) \leftarrow \{C_j : |C_j| < p^* N\}$ {All Components smaller than $p^* N$}
5: $C^{(p)} \leftarrow \cup_{C \in \mathcal{S}(p^*)} C$ {Collapse all small components into one}
6: **Return :** $\mathcal{C} \setminus \mathcal{S}(p^*) \bigcup C^{(p)}$ {Each connected component larger than $p^* N$ is a cluster, and all small components are a single cluster}

Definition 1 (α-Separable Cluster)**.** *For a fixed $\alpha < 1/2$, cluster $i \in [L]$ is termed α-separable if $\Delta_i \geq \frac{5}{(NT)^\alpha}$. Otherwise, it is termed as α-inseparable.*

Lemma 1. *If CMLB is run with parameters $\gamma = 3/(NT)^\alpha$ and $p^* \leq p_i$ and $\alpha < \frac{1}{2}$, then with probability at least $1 - 2\binom{N}{2}\delta$, any cluster i that is α-separable is clustered correctly. Furthermore, the regret of any user in the α-separated cluster i satisfies,*

$$R_i(T) \leq C_1[\frac{d}{\rho_{\min}}(NT)^\alpha + \sqrt{\frac{d}{\rho_{\min}}}\,(\sqrt{\frac{T - \frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{p_i N}})]\log(1/\delta),$$

with probability exceeding $1 - 4\binom{N}{2}\delta$.

We now present the regret of SCLB for the setting with separable cluster

Theorem 1. *If Algorithm 1 is run for T steps with parameter $\alpha < \frac{1}{2}$, then the regret of any agent in a cluster i that is α-separated satisfies*

$$R_i(T) \leq 4\left(2^{\frac{1}{\sqrt{p_i}}}\right) + C_2\left[\frac{d}{\rho_{\min}}(NT)^\alpha + \sqrt{\frac{dT}{\rho_{\min} N}}\right]\log^2(T)\log(1/\delta),$$

with probability at-least $1 - cN^2\delta$. Moreover, if $\alpha \leq \frac{1}{2}(\frac{\log\left\lfloor\frac{\rho_{\min} T}{dp_i N}\right\rfloor}{\log(NT)})$, we have $R_i(T) \leq \tilde{\mathcal{O}}[2^{\frac{1}{\sqrt{p_i}}} + \sqrt{\frac{d}{\rho_{\min}}}\,\sqrt{\frac{T}{N}}]\log(1/\delta)$.

Remark 1. *Note that we obtain the regret scaling of $\tilde{\mathcal{O}}(\sqrt{T/N})$, which is optimal, i.e., the regret rate matches an oracle that knows cluster membership. The cost of successive clustering is $\mathcal{O}(2^{\frac{1}{\sqrt{p_i}}})$, which is a T-independent (problem dependent) constant.*

Remark 2. *Note that the separation we need is only $5/(NT)^\alpha$. This is a weak condition since in a collaborative system with large N and T, this quantity is sufficiently small.*

Remark 3. *Observe that $R_i(T)$ is a decreasing function of N. Hence, more users in the system ensures that the regret decreases. This is collaborative gain we obtain.*

Remark 4. *(Comparison with [GLZ14]) Note that in a setup where clusters are separated, [GLZ14] also yields a regret of $\tilde{\mathcal{O}}(\sqrt{T/N})$. However, the separation between the parameters (gap) for [GLZ14] cannot be lower than $\mathcal{O}(1/T^{1/4})$, in order to maintain order-wise optimal regret. On the other hand, we can handle separations of the order $\mathcal{O}(1/T^\alpha)$, and since $\alpha < 1/2$, this is a strict improvement over [GLZ14].*

Remark 5. *Note that the constant term $\mathcal{O}(2^{\frac{1}{\sqrt{p_i}}})$ can be removed if we have an estimate of the p_i. In this case, instead of SCLB, we simply run CMLB with the estimate of p_i and obtain the regret of Lemma 1, without the term $\mathcal{O}(2^{\frac{1}{\sqrt{p_i}}})$. Note that in simulations (Sec. 6), we observe that the size input to CMLB is not needed.*

We now present our results when cluster i is α-inseparable.

Lemma 2. *If CMLB is run with input $\gamma = 3/(NT)^\alpha$ and $p^* \leq p_i$ and $\alpha < \frac{1}{2}$, then any user in a cluster i that is α-inseparable satisfies*

$$R(T) \leq C_1 L(\frac{T^{1-\alpha}}{N^\alpha}) + C_2\sqrt{\frac{d}{\rho_{\min}}}\,[\sqrt{\frac{T - \frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{p^* N}}]\log(1/\delta),$$

with probability at least $1 - 4\binom{N}{2}\delta$.

Theorem 2. *If Algorithm 1 is run for T steps with parameter $\alpha < \frac{1}{2}$, then the regret of any agent in a cluster i that is α-inseparable satisfies*

$$R_i(T) \leq 4(2^{\frac{1}{\sqrt{p_i}}}) + C\,L(\frac{T^{1-\alpha}}{N^\alpha})\log(T) + C_1\sqrt{\frac{dT}{N\rho_{\min}}}\log(1/\delta)\ \log^2(T),$$

with probability at-least $1 - cN^2\delta$. Moreover, if If $\alpha = \frac{1}{2} - \varepsilon$, where ε is a positive constant arbitrarily close to 0, we obtain, $R(T) \leq \tilde{\mathcal{O}}\left[2^{\frac{1}{\sqrt{p_i}}} + L(\frac{T^{\frac{1}{2}+\varepsilon}}{N^{\frac{1}{2}-\varepsilon}}) + \sqrt{\frac{d}{\rho_{\min}}}\,(\sqrt{\frac{T}{N}})\log(1/\delta)\right]$.

Remark 6. *As $\varepsilon > 0$, the regret scaling of $\tilde{\mathcal{O}}(\frac{T^{\frac{1}{2}+\varepsilon}}{N^{\frac{1}{2}-\varepsilon}})$ is strictly worse than the optimal rate of $\tilde{\mathcal{O}}(\sqrt{T/N})$. This can be attributed to the fact that the gap (or SNR) can be arbitrarily close to 0, and inseparability of the clusters makes the problem harder to address.*

Remark 7. *Note that similar to the previous setting, the regret decreases with N, which shows the benefit of collaborative learning.*

Remark 8. *In this setting of low gap (or SNR), where the clusters are inseparable, most existing algorithms (for example [GLZ14]) are not applicable. However, we still manage to obtain sub-optimal but non-trivial regret with high probability.*

Special case: all the clusters are 'sufficiently' close We now treat the special case where all the clusters are reasonably close, *but not necessarily identical*. In particular, if $\max_{i\neq j}\|\theta_i^* - \theta_j^*\| \leq 1/(NT)^\alpha$, CMLB puts all the users in one big cluster. The collaborative gain in this setting is the largest. Here the regret guarantee of SCLB will be similar to that of Theorem 2 with $p_i = 1$. We defer to Appendix 11 for a detailed analysis.

Remark 9. *Note that if $\max_{i\neq j}\|\theta_i^* - \theta_j^*\| = 0$, all agents are identical, we cannot match the guarantee of an* oracle *which knows such information. The oracle guarantee would be $\mathcal{O}(\sqrt{T/N})$, whereas our guarantee is strictly worse. The additional regret stems from the universality of our algorithm as it works for all ranges of Δ_i.*

5 Personalization in multi-agent contextual bandits

In this section, we present the formulation for personalized learning in multi-agent contextual Linear Bandit framework. Suppose we have N agents, with optimal parameters $\{\theta_i^*\}_{i=1}^N$. Unlike the clustering structure in the above mentioned sections, here, $\{\theta_i^*\}_{i=1}^N$ may all be different from one another. Of course, without any structural similarity among $\{\theta_i^*\}_{i=1}^N$, the only way-out is to learn the parameters separately for each user. In the setup of personalized learning, it is typically assumed that (see [YHLD21, CHMS21, FMO20c, LHBS20] and the references therein) that the parameters $\{\theta_i^*\}_{i=1}^N$ share some commonality, and the job is to learn the shared components or representations of $\{\theta_i^*\}_{i=1}^N$ collaboratively. After learning the common part, the individual representations can be learnt locally at each agent.

Algorithm 4: Personalized Multi-agent Linear Bandits (PMLB)

Input: Agents N, Horizon T

Common representation learning : Estimate $\bar{\theta}^* = \frac{1}{N}\sum_{i=1}^{N}\theta_i^*$

Initialize a single instance of OFUL(δ), called common OFUL

for times $t \in \{1, \cdots, \sqrt{T}\}$ **do**

  All agents play the action given by the common OFUL

  Common OFUL's state is updated by the average of observed rewards at all agents

end for

$\widehat{\theta}^* \leftarrow$ the parameter estimate of Common OFUL at the end of round $\sqrt{T}$

Personal Learning

for agents $i \in \{1, \ldots, N\}$ **in parallel do**

  Initialize one ALB-Norm(δ) of [GSK21] instance per agent

  for times $t \in \{\sqrt{T}+1, \ldots, T\}$ **do**

    Agents play arm specified by their personal copy of ALB-Norm (denoted as $\beta_{b_t^{(i)},t}$) and receive reward y_t

    Every agent updates their ALB-Norm state with corrected reward $\tilde{y}_i^{(t)} = y_i^{(t)} - \langle \beta_{b_t^{(i)},t}, \hat{\theta}^* \rangle$

  end for

end for

Similar to Section 3, the contexts $\beta_{i,t}$-s are drawn independent of the past from a distribution such that $\beta_{i,t}$ is independent of $\{\beta_{j,t}\}_{j\neq i}$. The only difference is that, instead of sampling the contexts from unit ball $\mathbb{B}_d^{(1)}$, we assume that $\beta_{i,t}$-s are assumed to be drawn from scaled Gaussian $\mathcal{N}(0, \frac{1}{d}\mathbb{I}_d)$. Hence, the conditions of equation (1) are satisfied with $\rho_{\min} = 1/d$. This is without loss of generality, as it simplifies the exposition.

We now define the notion of common representation across users. For consistency, similar to Section 3, we assume $\|\theta_l^*\| \leq 1$ for all $l \in [N]$. We define $\bar{\theta}^* = \frac{1}{N}\sum_{l=1}^{N}\theta_l^*$ as the average parameter.

Definition 2. *(ϵ common representation) An agent i has ϵ_i common representation across N agents if $\|\theta_i^* - \bar{\theta}^*\| \leq \epsilon_i$, where ϵ_i is defined as the common representation factor.*

The above definition characterizes how far the representation of agent i is from the average representation $\bar{\theta}^*$. Note that since $\|\theta_l^*\| \leq 1$ for all l, we have $\epsilon_i \leq 2$. Furthermore, if ϵ_i is small, one can hope to exploit the common representation across users. On the other hand, if ϵ_i is large (say $\mathcal{O}(1)$), there is no hope to leverage collaboration across agents.

We now present Personalized Multi-agent Linear Bandits (PMLB) (Algorithm 4), which adapts to the common representation factor ϵ_i, without knowing it apriori. In particular, we show that when ϵ_i is small, Algorithm 4 exploits the common representation across users and obtains a low regret. On the other hand, if ϵ_i is large, the regret of Algorithm 4 matches to the setup where agent i learns θ_i^* without interacting with other agents in the system. Note that this is intuitive since with high ϵ_i, the agents share no common representation, and so we do not get a regret improvement in this case by exploiting the actions of other agents.

Algorithm 4 works in two phases. In the first phase, it learns the average representation $\bar{\theta}^*$. Since the algorithm aggregates the reward from all N agents, it turns out that the common representation learning phase can be restricted to $\sqrt{T}$ time steps. At the end of this phase, the center has the estimate $\hat{\theta}^*$ of the average representation $\bar{\theta}^*$. In the personal learning phase, which lasts for $T-\sqrt{T}$ time steps, each agent tries to learn the vector $\theta_i^* - \hat{\theta}^*$, since the center plays the OFUL algorithm of [CMB20, AyPS11] with the shifted reward. However, our goal is to characterize the regret of agent i. To achieve this, we analyze the regret of the shifted OFUL algorithm in Appendix 15. In particular exploiting the anti-concentration property of Chi-squared distribution along with some standard results from optimization, we show that the regret of the

shifted system is worse that the regret of agent i (both in expectation and in high probability)[2].

For learning $\theta_i^* - \hat{\theta}^*$, we employ the Adaptive Linear Bandits-norm (`ALB-norm`) algorithm of [GSK21]. `ALB-norm` is adaptive, yielding a norm dependent regret. The idea here is to exploit the fact that in the common learning phase we have a good estimate of $\bar{\theta}^*$. Hence, if the common representation factor ϵ_i is small, then $\|\theta_i^* - \hat{\theta}^*\|$ is small, and it reflects in the regret expression. On the other hand, if ϵ_i is large, even with the good estimation of $\bar{\theta}^*$, the quantity $\|\theta_i^* - \hat{\theta}^*\|$ an be large, which worsens the regret scaling. We have the following regret guarantee for Algorithm 4.

Theorem 3. *If we play Algorithm 4 for upto T time and δ, where $T \geq \tau_{\min}^2(\delta)$ (where $\tau_{\min}(\delta)$ is defined in eqn. (3)) and $d \geq C\log(K^2T)$, then the regret of agent i satisfies*

$$R_i(T) \leq \tilde{\mathcal{O}}(\epsilon_i \sqrt{dT} + \ T^{1/4} \sqrt{\frac{d^2}{\rho_{\min} N}}) \log^2(1/\delta),$$

with probability at least $1 - c\delta - \frac{1}{\text{poly}(T)}$.

Remark 10. *Note that the leading term in the regret expression is $\tilde{\mathcal{O}}(\epsilon_i\sqrt{dT})$. If the common representation factor ϵ_i is small, PMLB exploits the common representation across agents and as a result the regret is small as well.*

Remark 11. *On the other hand, if ϵ_i is big enough, say $\mathcal{O}(1)$, this implies that there is no common representation across users, and hence collaborative learning is meaning less. In this case, the agents learn individually (by running OFUL), and obtain a regret of $\tilde{\mathcal{O}}(\sqrt{dT})$ with high probability. Note that this is being reflected in Theorem 3, as the regret is $\tilde{\mathcal{O}}(\sqrt{dT})$, when $\epsilon_i = \mathcal{O}(1)$.*

The above remarks imply the adaptivity of PMLB. Without knowing the common representation factor ϵ_i, PMLB indeed adapts to it—meaning that yields a regret that depends on ϵ_i. If ϵ_i is small, PMLB leverages common representation learning across agents, otherwise when ϵ_i is large, it yields a performance equivalent to the individual learning.

The requirement on d in Theorem 3 can be removed if we consider the expected regret.

Corollary 1. *(Expected Regret) Suppose $T \geq \tau_{\min}^2(\delta)$ for $\delta > 0$. The expected regret of the i-th agent after running Algorithm 4 for T time steps is given by*

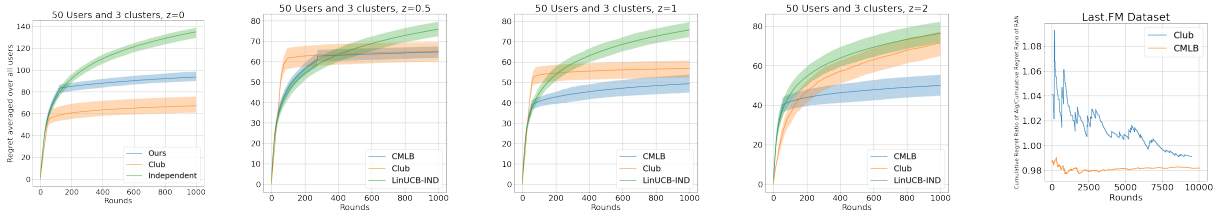
$$\mathbb{E}[R_i(T)] \leq \tilde{\mathcal{O}}(\epsilon_i \sqrt{dT} + \ T^{1/4} \sqrt{\frac{d^2}{\rho_{\min} N}}).$$

Remark 12. *When $\epsilon_i = 0$, i.e., in the case when all agents have the identical vectors θ_i^*, then Theorem 3 gives a regret scaling as $R_i(T) \leq \widetilde{\mathcal{O}}(T^{1/4} d\sqrt{\frac{1}{\rho_{\min} N}})$. This does not contradict the well known lower bound of $R(T) \geq \Omega(\sqrt{dT})$ ([LS20, Chapter 24] with finite action set) as the lower bound is valid only in the regime $d \leq O(\sqrt{T})$, in which case our bound in Theorem 3 has regret scaling larger than $\mathcal{O}(T^{3/4})$ with respect to T.*

6 Experiments

We simulate CMLB with p^* set to 0. Although in theory (Lemmas 1, 2), we need $0 < p^* \leq p$ to be non-trivial, we do not require it so in simulations as demonstrated in this section. Thus, we only report results from CMLB and not SCLB as its performance is worse than CMLB.

[2]This is intuitive since, otherwise one can find *appropriate shifts* to reduce the regret of OFUL, which contradicts the optimality of OFUL.



(a) (b) (c) (d) (e)

Figure 1: Fig $(a), (b), (c), (d)$ are on synthetic data and (e) is on Last.FM dataset. In fig (e), the ratio of the cumulative regret of CLUB and CMLB with that of recommending arms at random is plotted.

6.1 Synthetic Simulations

Data : For each plot of Figures 1 (panel (a) - (d)), users are clustered such that the frequency of cluster i is proportional to i^{-z} (identical to that done in [GLZ14]), where z is mentioned in the figures. Thus for $z = 0$, all clusters are balanced, and for larger z, the clusters become imbalanced. For each cluster, the unknown parameter vector θ^* is chosen uniformly at random from the unit sphere.

Setup : We compare CMLB (Algorithm 2) with CLUB [GLZ14] and the simple baseline of no collaboration, where every agent has an independent copy of OFUL, a Linear Bandit algorithm. This is called as `LinUCB-Ind` in the plots above. In each setting, we simulate all algorithms with the 25 context-vectors, each of dimension 15, sampled at random from the scaled Gaussian distribution $\mathcal{N}(0, d^{-1}\mathbb{I}_d)$. The plots in Figure 1 (a) - (d) show the regret averaged over all users, after each algorithm has taken 1000 steps for all users. CMLB and LinUCB-Ind take a total of 1000 rounds, while CLUB takes $1000 \times$ `num-agents` rounds. For CLUB, users are picked in a round-robin fashion, with all users shown the same set of contexts in a batch. Thus at the t-th arm-pull in all algorithms, all users have the same set of contexts. We repeat 30 times and plot 95-th percentile confidence bounds of the regret averaged over users.

Hyper-parameters : For CMLB, in all experiments, we use $\delta = 0.4$, $\alpha = 0.2$, $C = 0.2$ and $p^* = 0$. For LinUCB, we use $\lambda = 1$. For CLUB, we tuned the two hyper-parameters α and α_2 for each setting, by considering the performance over the first 500 rounds and choosing the best one.

Results : We observe that our algorithm is competitive with respect to CLUB, and is superior compared to the baseline where each agent is playing an independent copy of OFUL. In particular, we observe either as the clusters become more imbalanced, or as the number of users increases, CMLB has a superior performance compared to CLUB. Furthermore, since CMLB only clusters users logarithmically many number of times, its run-time is faster compared to CLUB.

6.2 Last.FM Dataset

Data : LastFM is a collection of 1892 users and 17632 artists. This dataset contains records of `(user, artist, tags)` denoting that a user listened to an artist and assigned a tag. We convert this into a multi-agent recommendation task, identical to the setting considered in [GLZ14] and [CBGZ13]. We break down all tags into atomic units, exactly as suggested in [GLZ14, CBGZ13], and assign to every artist, the collection of assigned atomic tags by all users. We then extract the top 25 principal components from the *tf-idf* matrix of artists and atomic tags as the context vectors for artists. Thus, each artist is a 25 dimensional vector. The reward for a `(user,artist)` pair is 1 if present in the dataset; else 0.

Setup : We consider a time-horizon of 10000 - CMLB was simulated for 10000 rounds and CLUB until all users had taken 10000 steps. At a given time instant, a set of 25 randomly sampled items was shown as contexts for CMLB. For CLUB, we first chose a user by picking them in a round robin fashion, and choose 24 items at random and one item at random from among those the user had listened to. This way, we ensure that at each time, the best reward for CLUB is at-least one. However, for CMLB, as at each time step all users play, such a guarantee cannot be made. This makes the learning setting harder for CMLB since at

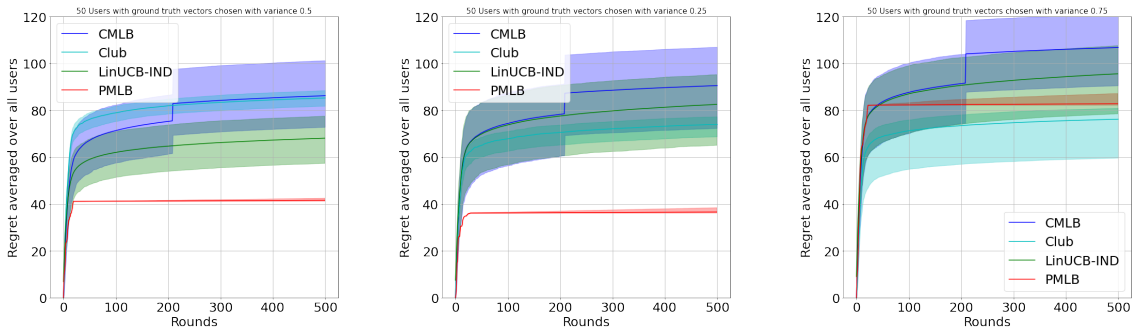


Figure 2: Synthetic data, where the ground truth vectors for users are chosen as described in Section 6.3. We observe that for small variance, PMLB outperforms the baselines, while for larger variance, the performance of PMLB gracefully degrades and matches the baselines.

every time, a large fraction of users have the best-reward of 0, i.e., the arm separation is 0, while the best reward is always 1 for CLUB.

Hyper-parameters : We use $\delta = 0.3$, $\alpha = C = 0.5$ for CMLB. For CLUB, we use $\alpha = 1$ and $\alpha_2 = 2$ chosen from a burn-in period of 500 arm-pulls of all agents.

Results : We compare the two algorithms by plotting the ratio of cumulative regret to that obtained by recommending an artist at random each time in Figure 1 in panel (e). We see that CMLB is competitive. However, the sparsity, renders the task quite challenging and our results indicate that neither algorithms are particularly appealing for this dataset.

6.3 Evaluation of PMLB on Synthetic Data

In simulating PMLB, we replace ALB-Norm with vanilla LinUCB. We do this, because we observe that this simplification itself yields good empirical performance.

Data : For each plot, we consider a system where the N ground-truth θ^* vectors are sampled independently from $\mathcal{N}(\mu, \sigma\mathbb{I})$, the normal distribution in d dimensions with mean $\mu \in \mathbb{R}^d$, and variance σ. The parameter μ was chosen from the standard normal distirbution in each experiement. We test performance for different values of σ. Observe that for small σ, all the ground-truth vectors will be close-by (high structure) and when σ is large, the ground-truth vectors are more spread out.

Setup : We compare the performance of PMLB with that of CMLB, CLUB and the baseline algorithm of playing an independent LinUCB at each agent. The setup is identical to that described in Section 6.

Hyperparameters : For the LinUCB of PMLB, we use the same hyper-parameters as that used in CMLB described in Section 6.1.

Observations : We observe in Figure 2 that PMLB adapts to the available structure. In the case when σ is low, in which case every user is close to the average, the regret of PMLB is much lower compared to the baselines. On the other hand, when σ is large, i.e., there is no structure to exploit, the regret of PMLB is comparable to the baselines. This demonstrates empirically that PMLB adapts to the problem structure and exploits it whenever present.

7 Societal impact in practice

Although motivated by recommendation systems, our work is theoretical in nature, and analyzes simple caricature models to quantify the power of collaboration in an online setting. As such, we do not recommend directly implementing our algorithms in practice. Personalization in practical recommendation systems needs to address several challenging issues beyond average performance; for ex. is a certain user or a minority

group of users always receiving poor performance? Quantifying and mitigating such issues is beyond this paper's scope, but is necessary for practical deployments.

8 Conclusion

We considered the problem of leveraging user heterogeneity to reduce regret in a multi-agent stochastic bandit problem. We studied two frameworks – *(i)* a clustering framework where users within a group are identical and, *(ii)* a personalization framework where users preference vectors are close to the population average. In both cases, we give novel adaptive algorithms that, without any knowledge of the underlying instance, provides regret guarantees that are sub-linear in T and N. We further verify this in simulations, both on real and synthetic data. A natural avenue for future work will be to combine the two frameworks, where users are all not necessarily identical, but at the same time, their preferences are spread out in space (for example the perference vectors are sampled from a Gaussian mixture model). Natural algorithms here will involve first performing a clustering on the population, followed by algorithms such as PMLB. Characterizing performance and demonstrating adaptivity in such settings is left to future work.

References

[ADK+20] Sanjeev Arora, Simon Du, Sham Kakade, Yuping Luo, and Nikunj Saunshi. Provable representation learning for imitation learning via bi-level optimization. In *International Conference on Machine Learning*, pages 367–376. PMLR, 2020.

[AyPS11] Yasin Abbasi-yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24, pages 2312–2320. Curran Associates, Inc., 2011.

[BWY+17] Sivaraman Balakrishnan, Martin J Wainwright, Bin Yu, et al. Statistical guarantees for the em algorithm: From population to sample-based analysis. *Annals of Statistics*, 45(1):77–120, 2017.

[CAS16] Paul Covington, Jay Adams, and Emre Sargin. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*, pages 191–198, 2016.

[CBGZ13] Nicolo Cesa-Bianchi, Claudio Gentile, and Giovanni Zappella. A gang of bandits. *arXiv preprint arXiv:1306.0811*, 2013.

[CHMS21] Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. *arXiv preprint arXiv:2102.07078*, 2021.

[CMB20] Niladri Chatterji, Vidya Muthukumar, and Peter Bartlett. Osom: A simultaneously optimal algorithm for multi-armed and linear contextual bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 1844–1854. PMLR, 2020.

[DCGP19] Giulia Denevi, Carlo Ciliberto, Riccardo Grazzi, and Massimiliano Pontil. Learning-to-learn stochastic gradient descent with biased regularization. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1566–1575. PMLR, 09–15 Jun 2019.

[DTB+19] Carlo D'Eramo, Davide Tateo, Andrea Bonarini, Marcello Restelli, and Jan Peters. Sharing knowledge in multi-task deep reinforcement learning. In *International Conference on Learning Representations*, 2019.

[FKL19] Dylan J. Foster, Akshay Krishnamurthy, and Haipeng Luo. Model selection for contextual bandits, 2019.

[FMO20a] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. On the convergence theory of gradient-based model-agnostic meta-learning algorithms. In *International Conference on Artificial Intelligence and Statistics*, pages 1082–1092. PMLR, 2020.

[FMO20b] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*, 2020.

[FMO20c] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 3557–3568. Curran Associates, Inc., 2020.

[FRKL19] Chelsea Finn, Aravind Rajeswaran, Sham Kakade, and Sergey Levine. Online meta-learning. In *International Conference on Machine Learning*, pages 1920–1930. PMLR, 2019.

[GLK+17] Claudio Gentile, Shuai Li, Purushottam Kar, Alexandros Karatzoglou, Giovanni Zappella, and Evans Etrue. On context-dependent clustering of bandits. In *International Conference on Machine Learning*, pages 1253–1262. PMLR, 2017.

[GLZ14] Claudio Gentile, Shuai Li, and Giovanni Zappella. Online clustering of bandits. In *International Conference on Machine Learning*, pages 757–765. PMLR, 2014.

[GSK21] Avishek Ghosh, Abishek Sankararaman, and Ramchandran Kannan. Problem-complexity adaptive model selection for stochastic linear bandits. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 1396–1404. PMLR, 13–15 Apr 2021.

[HPR+17] Irina Higgins, Arka Pal, Andrei Rusu, Loic Matthey, Christopher Burgess, Alexander Pritzel, Matthew Botvinick, Charles Blundell, and Alexander Lerchner. Darla: Improving zero-shot transfer in reinforcement learning. In *International Conference on Machine Learning*, pages 1480–1490. PMLR, 2017.

[KBT19] Mikhail Khodak, Maria-Florina Balcan, and Ameet Talwalkar. Adaptive gradient-based meta-learning methods. *arXiv preprint arXiv:1906.02717*, 2019.

[KC20] Jeongyeol Kwon and Constantine Caramanis. The em algorithm gives sample-optimality for learning mixtures of well-separated gaussians. In *Conference on Learning Theory*, pages 2425–2487. PMLR, 2020.

[KSL16] Nathan Korda, Balazs Szorenyi, and Shuai Li. Distributed clustering of linear bandits in peer to peer networks. In *International conference on machine learning*, pages 1301–1309. PMLR, 2016.

[LCLS10] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

[Lee01] Wee Sun Lee. Collaborative learning for recommender systems. In *ICML*, volume 1, pages 314–321. Citeseer, 2001.

[LHBS20] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Federated multi-task learning for competing constraints. *CoRR*, abs/2012.04221, 2020.

[LK03] Qing Li and Byeong Man Kim. Clustering approach for hybrid recommender system. In *Proceedings IEEE/WIC International Conference on Web Intelligence (WI 2003)*, pages 33–38. IEEE, 2003.

[LLCS15] Yang Liu, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Topical word embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[LM00] B. Laurent and P. Massart. Adaptive estimation of a quadratic functional by model selection. *The Annals of Statistics*, 28(5):1302 – 1338, 2000.

[LR11] Alessandro Lazaric and Marcello Restelli. Transfer from multiple mdps. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011.

[LS20] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[LSY03] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1):76–80, 2003.

[MMRS20] Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. Three approaches for personalization with applications to federated learning. *CoRR*, abs/2002.10619, 2020.

[MNLD20] Yifei Ma, Balakrishnan Narayanaswamy, Haibin Lin, and Hao Ding. Temporal-contextual recommendation in real-time. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2291–2299, 2020.

[NMS+19] Maxim Naumov, Dheevatsa Mudigere, Hao-Jun Michael Shi, Jianyu Huang, Narayanan Sundaraman, Jongsoo Park, Xiaodong Wang, Udit Gupta, Carole-Jean Wu, Alisson G Azzolini, et al. Deep learning recommendation model for personalization and recommendation systems. *arXiv preprint arXiv:1906.00091*, 2019.

[OTOT17] Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. Embedding-based news recommendation for millions of users. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1933–1942, 2017.

[Ozs16] Makbule Gulcin Ozsoy. From word embeddings to item recommendation. *arXiv preprint arXiv:1601.01356*, 2016.

[PBS15] Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. Actor-mimic: Deep multitask and transfer reinforcement learning. *arXiv preprint arXiv:1511.06342*, 2015.

[PEZ+20] Aditya Pal, Chantat Eksombatchai, Yitong Zhou, Bo Zhao, Charles Rosenberg, and Jure Leskovec. Pinnersage: Multi-modal user embedding framework for recommendations at pinterest. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2311–2320, 2020.

[RCG+15] Andrei A Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. *arXiv preprint arXiv:1511.06295*, 2015.

[SKKR01] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295, 2001.

[SKKR02] Badrul M Sarwar, George Karypis, Joseph Konstan, and John Riedl. Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. In *Proceedings of the fifth international conference on computer and information technology*, volume 1, pages 291–324. Citeseer, 2002.

[SM14] Martin Saveski and Amin Mantrach. Item cold-start recommendations: learning local collective embeddings. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 89–96, 2014.

[WTAL16] Suhang Wang, Jiliang Tang, Charu Aggarwal, and Huan Liu. Linked document embedding for classification. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 115–124, 2016.

[XDZ+17] Hong-Jian Xue, Xinyu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. Deep matrix factorization models for recommender systems. In *IJCAI*, volume 17, pages 3203–3209. Melbourne, Australia, 2017.

[YHLD21] Jiaqi Yang, Wei Hu, Jason D. Lee, and Simon Shaolei Du. Impact of representation learning in linear bandits. In *International Conference on Learning Representations*, 2021.

[YYC+20] Tiansheng Yao, Xinyang Yi, Derek Zhiyuan Cheng, Felix Yu, Aditya Menon, Lichan Hong, Ed H Chi, Steve Tjoa, Evan Ettinger, et al. Self-supervised learning for deep models in recommendations. *arXiv preprint arXiv:2007.12865*, 2020.

[ZDF17] Handong Zhao, Zhengming Ding, and Yun Fu. Multi-view clustering via deep matrix factorization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[ZHW+19] Zhe Zhao, Lichan Hong, Li Wei, Jilin Chen, Aniruddh Nath, Shawn Andrews, Aditee Kumthekar, Maheswaran Sathiamoorthy, Xinyang Yi, and Ed Chi. Recommending what video to watch next: a multitask ranking system. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 43–51, 2019.

Appendix

We present the proofs of our main results in this section. Throughout this section, we assume $C, C_1, C_2, ..., c, c_1, ...$ as universal constants, the value of which may change from instanc to instance.

9 Proof of Lemma 1

Here, there is a gap between the optimal parameters. In this case, suppose the Individual Learning phase lasts for $C^{(2)} \frac{d(NT)^{2\alpha}}{\rho_{\min}} \log(1/\delta)$ time steps. Following the analysis of OFUL [AyPS11, CMB20], alonf=g with the condition in equation 3, after t instances, we have

$$\|\hat{\theta}_t^{(i)} - \theta_1^*\| \leq \frac{D}{\sqrt{1 + \rho_{\min} t/2}},$$

with probability at least $1 - \delta$, where $D = \tilde{\mathcal{O}}(\sqrt{d}) \log(1/\delta)$.

If agents i and j fall in same cluster, we have,

$$\|\hat{\theta}^{(j)} - \hat{\theta}^{(i)}\| \leq 2/(NT)^{\alpha}.$$

with probability at least $1 - 2\delta$.

Otherwise we have

$$\|\hat{\theta}^{(j)} - \hat{\theta}^{(i)}\| \geq \Delta_i - 2/(NT)^{\alpha}.$$

Now, suppose $3/(NT)^{\alpha} \leq \Delta_i - 2/(NT)^{\alpha}$, or in other words, $\Delta_i \geq 5/(NT)^{\alpha}$. In that case,

$$\|\hat{\theta}^{(j)} - \hat{\theta}^{(i)}\| \geq 3/(NT)^{\alpha}.$$

with high probability. So, if we threshold $3/(NT)^\alpha$, we can find out the cluster perfectly with probability exceeding $1-2\delta$. Since we want this to hold for every pair of agents, a simple union bound yields the lemma.

Since, there is no clustering error, we have

$$R(T) = R(ind-learn) + R(coll-learn)$$

The Individual Learning phase continues until $C^{(2)}\frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)$ time steps. Hence, according to [CMB20], we have

$$\begin{aligned} R(ind-learn) &\leq C\sqrt{\frac{d}{\rho_{\min}}}\left(\sqrt{C^{(2)}\frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}\right)\sqrt{\log(1/\delta)} \\ &\leq C_1\ \frac{d}{\rho_{\min}}(NT)^\alpha\ \log(1/\delta) \end{aligned}$$

with probability greater than $1-\delta$. To avoid clutter, we have only considered the leading term in the above regret.

We now characterize the regret in the collaborative learning phase. Here, the regret depends on the cluster size. Since the center averages the mean reward from all the users in a cluster, it effectively reduces the noise variance by a factor of the cluster size. Hence, the regret upper bound is we have (using [CMB20]),

$$R(coll-learn) \leq C_1\sqrt{\frac{d}{\rho_{\min}}}\ \left(\sqrt{\frac{T-\frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{p_iN}}\right)\sqrt{\log(1/\delta)}$$

with probability at least $1-\delta$. Since, the size of the i-th cluster is p_i.

Hence, with probability at least $1-2\delta$ total regret is given by

$$R(T) \leq C_1\left[\frac{d}{\rho_{\min}}(NT)^\alpha + \sqrt{\frac{d}{\rho_{\min}}}\ \left(\sqrt{\frac{T-\frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{p_iN}}\right)\right]\log(1/\delta).$$

Suppose α satisfies

$$\alpha \leq \frac{1}{2}\left(\frac{\log\left[\frac{\rho_{\min}T}{dp_iN}\right]}{\log(NT)}\right).$$

Then, the first term in the above regret expression can be upper bounded by the second term, and the resulting regret is given by

$$R(T) \leq C\left(\sqrt{\frac{d}{\rho_{\min}}}\ \sqrt{\frac{T}{p_iN}}\right)\log(1/\delta)$$

with probability at least $1-2\delta$.

10 Proof of Lemma 2

In this case, we have $\Delta_i \leq 5/(NT)^\alpha$. In this case, we show that the maximal-cluster subroutine of Algorithm 2 treats the neighboring clusters of cluster i, also together with i, as a single cluster, with high probability. It may happen that some of the clusters are left out owing to being far from cluster i. Let $\mathcal{S}$ be the set of cluster indices that Algorithm 2 clubs with cluster i. It is easy to see $\max_{j\in\mathcal{S}}\|\theta_i^* - \theta_j^*\| \leq 5L/(NT)^\alpha$; otherwise we will be in separable cluster setting.

Note that in this case, we have no high probability guarantees on the cluster assignment by Algorithm 2. However, in this situation also, we argue that the regret suffered by the users are not very large. This is because the maximum separation between the clusters (and hence the clustering error) is $\mathcal{O}(L/(NT)^{\alpha})$, which is quite small.

Let us now focus on the regret upper-bound. The regret is given by

$$R(T) = R(ind - learn) + R(coll - learn) + R(cluster - error)$$

The first term comes from the initial phase of our algorithm. The second term comes after the (one-phase) clustering, and thereby exploiting the clustered OFUL algorithm. The third term comes when the algorithm makes an error in parameter estimates. Here Algorithm 2 clubs several clusters with cluster i, and hence one needs to address the clustering error. This clustering error indeed accumulates over the rest of the play.

The regret in the individual learning follows analysis similar to Lemma 1. We obtain

$$\begin{aligned} R(ind - learn) &\leq C\sqrt{\frac{d}{\rho_{\min}}}\left(\sqrt{C^{(2)}\frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}\right)\sqrt{\log(1/\delta)} \\ &\leq C_1\ \frac{d}{\rho_{\min}}(NT)^{\alpha}\ \log(1/\delta) \end{aligned}$$

with probability greater than $1-\delta$.

Let us now consider the collaborative learning phase. In this case, the maximal-cluster subroutine of CMLB treats the neighboring clusters of cluster i, also together with i, as a single cluster, with high probability. It may happen that some of the clusters are left out owing to being far from cluster i. Let $\mathcal{S}$ be the set of cluster indices that Algorithm 2 clubs with cluster i. It is easy to see $\max_{j\in\mathcal{S}}\|\theta_i^* - \theta_j^*\| \leq 5L/(NT)^{\alpha}$; otherwise we will be in Case I (separable clusters).

Note that in this case, we have no high probability guarantees on the cluster assignment by CMLB. However, in this situation also, the regret suffered by the users are not very large. This is because the maximum separation between the clusters (and hence the clustering error) is $\max_{j\in\mathcal{S}}\|\theta_i^* - \theta_j^*\| \leq 5/(NT)^{\alpha}$. Furthermore, since we have no control on how many clusters CMLB club, in the worst case, the minimum cluster size will be p^*N (the input size parameter to the CMLB subroutine). Hence, we obtain

$$R(coll - learn) \leq C\sqrt{\frac{d}{\rho_{\min}}}\ \left(\sqrt{\frac{T - \frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{p^*N}}\right)\sqrt{\log(1/\delta)}$$

with probability at least $1-\delta$.

We now characterize the regret from the cluster miss-specification. This term occurs since $\Delta \leq 5/(NT)^{\alpha}$. For agent i, from the OFUL algorithm (see [CMB20, AyPS11], we see that the regret in linearly dependent on θ_1^*.

Hence, following the regret analysis of stochastic linear bandits (see [CMB20], using triangle inequality, and the condition $\max_{i\neq j}\|\theta_i^* - \theta_j^*\| \leq 5L/(NT)^{\alpha}$), we have

$$R(cluster - error) \leq CL\left(\frac{T - \frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{(NT)^{\alpha}}\right).$$

Now, combining all 3 components, we have

$$\begin{aligned} R(T) \leq C_1\ \frac{d}{\rho_{\min}}(NT)^{\alpha}\ \log(1/\delta) + C_2\sqrt{\frac{d}{\rho_{\min}}}\ \left(\sqrt{\frac{T - \frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{p^*N}}\right)\sqrt{\log(1/\delta)} \\ + C_3L\left(\frac{T - \frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{(NT)^{\alpha}}\right). \end{aligned}$$

Rewriting, we have

$$R(T) \leq C_2\sqrt{\frac{d}{\rho_{\min}}}\left(\sqrt{\frac{T-\frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{p^*N}}\right)\sqrt{\log(1/\delta)} + C_3L\left(\frac{T}{(NT)^\alpha}\right) + (C_1-C_3L)\left(\frac{T-\frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{(NT)^\alpha}\right).$$

Since $L \geq 1$, choosing $C_3 > C_1$, we obtain

$$R(T) \leq C_2\sqrt{\frac{d}{\rho_{\min}}}\left(\sqrt{\frac{T-\frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{p^*N}}\right)\sqrt{\log(1/\delta)} + C_3\left(\frac{T}{(NT)^\alpha}\right).$$

Now, suppose $\alpha = \frac{1}{2}-\varepsilon$, where ε is a positive constant arbitrarily close to 0. In that case, we obtain

$$\begin{aligned} R(T) &\leq C\left[L\left(\frac{T^{\frac{1}{2}+\varepsilon}}{N^{1-\varepsilon}}\right) + \sqrt{\frac{d}{\rho_{\min}}}\left(\sqrt{\frac{T-\frac{d(NT)^{2\alpha}}{\rho_{\min}}\log(1/\delta)}{p^*N}}\right)\sqrt{\log(1/\delta)}\right] \\ &\leq C\left[L\left(\frac{T^{\frac{1}{2}+\varepsilon}}{N^{1-\varepsilon}}\right) + \sqrt{\frac{d}{\rho_{\min}}}\left(\sqrt{\frac{T}{p^*N}}\right)\sqrt{\log(1/\delta)}\right] \end{aligned}$$

with probability at least $1-4\binom{N}{2}\delta$.

11 Special Case: all clusters are close

Let us consider the pairwise differences this setting where we treat all the agents as one big cluster. Without loss of generality, we focus on $\{\|\hat{\theta}^{(i)}-\hat{\theta}^{(j)}\|\}_{j\neq i}$, and assume that the agent belongs to cluster 1 with parameter θ_1^*. In this setting, if the j-th agent falls in cluster 1, we have

$$\|\hat{\theta}^{(j)}-\hat{\theta}^{(i)}\| \leq 2/(NT)^\alpha.$$

with probability at least $1-2\delta$. Otherwise, we obtain

$$\begin{aligned} \|\hat{\theta}^{(j)}-\hat{\theta}^{(i)}\| &\leq 2/(NT)^\alpha + \max_{i,j}\|\theta_i^*-\theta_j^*\| \\ &\leq 3/(NT)^\alpha \end{aligned}$$

with probability exceeding $1-2\delta$. Ignoring the constants for now, if we have

$$\max_{j\neq i}\{\|\hat{\theta}^{(i)}-\hat{\theta}^{(j)}\|\} \leq 3/(NT)^\alpha,$$

then with probability at least $1-2\binom{N}{2}\delta$, and everyone belongs to the same cluster. So, we can put the threshold as $3/(NT)^\alpha$ to identify whether there is a cluster structure present or not.

The regret computation in this setup follows from Lemma 2, with 2 differences:
(a) The clustering error is $\max_{i,j}\|\theta_i^*-\theta_j^*\| \leq 1/(NT)^\alpha$. Note that order-wise it is same as the misclustering error for Lemma 2.
(b) Here, since all the clusters are close and CMLB puts everyone in the same cluster, the collaborative learning gain will be $\sqrt{\frac{1}{N}}$. Hence, the regret bound follows from Theorem 2 with these modifications.

12 Analysis of SCLB in Algorithm 1

Proof of Theorem 1: Minimum Cluster size is larger than $\frac{5}{(NT)^\alpha}$

We give the proof in the case when $\Delta_i > \frac{5}{(NT)^\alpha}$. The other setting follows identically. In each phase i, the size parameter used in Line 3 of Algorithm 1 is i^{-2}. Thus for all phases $i \geq \lceil \frac{1}{\sqrt{p}} \rceil$, the input size parameter to Algorithm 1 invoked in Line 3 of Algorithm 1 is correct, i.e., $i^{-2} \leq p$, and thus satisfies the conditions of Lemma 1.

In the rest of the proof, denote by $i^* := \lceil \frac{1}{\sqrt{p}} \rceil$ and by $T_i = 2^i$, for all $i \geq 1$. Lemma 2 states that, for any $i \geq i^*$, the regret incurred by any agent in phase i satisfies

$$R(T_i) \leq C_1 \left[\frac{d}{\rho_{\min}} (NT_i)^\alpha + \sqrt{\frac{d}{\rho_{\min}}} \left(\sqrt{\frac{T_i - \frac{d(NT_i)^{2\alpha}}{\rho_{\min}} \log(2^i/\delta)}{i^{-2}N}} \right) \right] \log(2^i/\delta), \tag{4}$$

with probability at-least $1 - 4\binom{N}{2} 2^{-i}\delta$. We now use a simple regret decomposition and an union bound to conclude the proof of Theorem 1.

Observe that, in a time horizon of T, there are at-most $\lceil \log_2(T) \rceil$ number of phases. The total regret can be decomposed as

$$\begin{aligned} R(T) &\leq \sum_{i=1}^{\lceil \log_2(T) \rceil} R(T_i), \\ &\leq \sum_{i=1}^{i^*} 2^i + \sum_{i=i^*}^{\lceil \log_2(T) \rceil} R(T_i), \\ &\leq 2^{i^*+1} + \sum_{i=i^*}^{\lceil \log_2(T) \rceil} R(T_i). \end{aligned}$$

In the first equality, we upper bound by assuming that the agent incurs a regret of 1, in all time steps till phase i^*. Now from an union bound, we can conclude that with probability at-least $1 - \sum_{i=1}^{\lceil \log_2(T) \rceil} cN^2 2^{-i}\delta \geq 1 - 2cN^2\delta$, Equation (4) is satisfied for all $i \geq i^*$.

Combining the above facts, along with the definition that $T_i = 2^i$, we have,

$$\begin{aligned} R(T_i) &\leq 2^{i^*+1} + C \log\left(\frac{2}{\delta}\right) \left[\sum_{i=1}^{\lceil \log_2(T) \rceil} \left(\frac{dN^\alpha}{\rho_{min}} i 2^{i\alpha} + \sqrt{\frac{d}{\rho_{min} N}} i^2 \sqrt{2^i} \right) \right], \\ &\leq 2^{i^*+1} + C_1 \log\left(\frac{2}{\delta}\right) \log^2(T) \left[\sum_{i=1}^{\lceil \log_2(T) \rceil} \left(\frac{dN^\alpha}{\rho_{min}} 2^{i\alpha} + \sqrt{\frac{d}{\rho_{min} N}} \sqrt{2^i} \right) \right] \end{aligned}$$

with probability at-least $1 - 2cN^2\delta$. The second inequality follows by upper bounding $i \leq \lceil \log_2(T) \rceil$. Observe from the definition of i^* that $2^{i^*+1} \leq 4\left(2^{\frac{1}{\sqrt{p}}}\right)$, the regret is bounded by

$$R(T) \leq 4\left(2^{\frac{1}{\sqrt{p}}}\right) + C_2 \log\left(\frac{2}{\delta}\right) \log^2(T) \left[(NT)^\alpha \frac{d}{\rho_{min}} + \sqrt{T \frac{d}{\rho_{min} N}} \right].$$

Here C_1, C_2 are universal constants.

Theorem 2: Minimum Cluster size is smaller than or equal to $\frac{5}{(NT)^\alpha}$

Following the identical steps as for Case I, where we use Lemma 2 to bound the regret in a phase, we get that with probability at-least $1 - 2cN^2\delta$

$$R(T) \leq C_1 L \log^2(T) 2^{1-\alpha} \frac{T^{1-\alpha}}{N^\alpha} + C_2 \log^2(T) \sqrt{T \frac{d}{N \rho_{min}}} \log(2/\delta).$$

13 Proof of Theorem 3

Collaboratively learn the common representations: Recall the setup of collaborative learning; at each time t, out of K contexts available at the center, $\{\beta_{r,t}\}_{r=1}^{K}$, the center chooses a context vector, (call it $\beta_{r,t}$, corresponding to the r-th arm), and broadcasts to all the agents. Agent i, using the context $\beta_{r,t}$, observes the following reward:

$$y_t^{(i)} = \langle \beta_{r,t}, \theta_i^* \rangle + \eta_{i,t},$$

and sends this to the center. Similarly, all the N agents observes their reward and send those to the center. The center then averages this rewards and obtain

$$\frac{1}{N}\sum_{l=1}^{N} y_t^{(l)} = \langle \beta_{r,t}, \frac{1}{N}\sum_{l=1}^{N} \theta_l^* \rangle + \frac{1}{N}\sum_{l=1}^{N} \eta_{l,t}.$$

Since, we are averaging i.i.d noise, the variance decreases by a factor of N. Now, based on the average reward, the center choosing the next arm by playing the stochastic contextual Bandit algorithm OFUL. Hence, in this phase, the center indeed learns the parameter $\bar{\theta}^* := \frac{1}{N}\sum_{l=1}^{N} \theta_l^*$. We let this phase run for $\sqrt{T}$ rounds, and let $\hat{\theta}^*$ be the corresponding estimate. Provided, $T > \tau_{\min}^2(\delta)$, from [CMB20], we have,

$$\|\hat{\theta}^* - \bar{\theta}^*\| \leq \tilde{O}\left(\sqrt{\frac{d}{\rho_{\min} N\sqrt{T}}}\right)\log(1/\delta),$$

with probability at least $1-\delta$. The corresponding regret (call it $R_{c,1}$) is

$$R_{c,1} = \tilde{O}\left(\sqrt{\frac{d\sqrt{T}}{\rho_{\min} N}}\right)\log(1/\delta),$$

with probability at least $1-\delta$.

Note that additional to the above, we incur a regret since instead of learning θ_i^*, we are actually learning $\bar{\theta}^*$. This is equivalent to clustering with miss-specification. Following the proofs similar to Section 4, we obtain

$$R_{c,2} = \|\theta_i^* - \bar{\theta}^*\| T_1 \leq \epsilon_i \sqrt{T},$$

where we use the fact that $\|\theta_l^*\| \leq 1$ for all $l \in [N]$. Hence, the total regret in this phase is

$$R_{c,1} + R_{c,2} \leq \tilde{O}\left(\sqrt{\frac{d\sqrt{T}}{\rho_{\min} N}}\right)\log(1/\delta) + \epsilon_i\sqrt{T},$$

with probability at least $1-\delta$.

Personal Learning: At each time t, out of K contexts available at the center, $\{\beta_{r,t}\}_{r=1}^{K}$, suppose the center chooses a context vector, $\beta_{r,t}$, (corresponding to the r-th arm) and recommends it to agent i. Thereafter, agent i generates the reward $y_t^{(i)} = \langle \beta_{r,t}, \theta_i^* \rangle + \xi_{i,t}$, and sends it to the center. Subsequently, the center calculates the corrected reward

$$\tilde{y}_t^{(i)} = y_t^{(i)} - \langle \beta_{r,t}, \hat{\theta}^* \rangle.$$

Note that the center has the information about $(\beta_{r,t}, \hat{\theta}^*)$ and so it can compute $\tilde{y}_t^{(i)}$. With this shift, the center basically learns the vector $\theta_i^* - \hat{\theta}^*$.

In this phase we use the `ALB-norm` algorithm of [GSK21][3]. Note that the `ALB-norm` algorithm is a norm adaptive algorithm, which is particularly useful when the parameter norm is small. `ALB-norm` uses the OFUL algorithm of [CMB20, AyPS11] repeatedly over epochs. At the beginning of each epoch, it estimates the parameter norm, and runs OFUL with the norm estimate (see [GSK21, Algorithm 1]). Hence, it is shown in [GSK21, Algorithm 1] that while estimating the parameter Ψ^*, with high probability, the regret of `ALB-norm` is

$$R_{\texttt{ALB-norm}} \leq \|\Psi^*\| \; R_{OFUL}.$$

In Appendix 15, we present an analysis of shifted OFUL. In particular we show that shifts (by a fixed vector) can not reduce the regret (which is intuitive). Note that we learn $\hat{\theta}^*$ in the common learning phase, and fix it throughout the personal learning phase. Hence, conditioned on the observations of the common learning phase, $\hat{\theta}^*$ is a fixed (deterministic) vector. Also,

$$\begin{aligned}\|\theta_i^* - \hat{\theta}^*\| &\leq \|\theta_i^* - \bar{\theta}^*\| + \|\hat{\theta}^* - \bar{\theta}^*\| \\ &\leq \epsilon_i + \tilde{O}\left(\sqrt{\frac{d}{\rho_{\min} N\sqrt{T}}}\right)\log(1/\delta),\end{aligned}$$

with probability at least $1-\delta$. Hence, using Lemma 6 of Appendix 15, the regret in the personal learning phase (call it $R_i^{(p)}$) is given by

$$R_i^{(p)} \leq \tilde{\mathcal{O}}\left(\|\theta_i^* - \hat{\theta}^*\|\sqrt{d(T-\sqrt{T})}\log(1/\delta)\right),$$

with probability at least $1 - c\delta - \frac{1}{\text{poly}(T)}$, provided $d \geq C\log(K^2T)$. Substituting, we obtain

$$\begin{aligned}R_i^{(p)} &\leq \tilde{\mathcal{O}}\left(\left[\epsilon_i + \tilde{O}\left(\sqrt{\frac{d}{\rho_{\min} N\sqrt{T}}}\right)\log(1/\delta)\right]\sqrt{d(T-\sqrt{T})}\log(1/\delta)\right) \\ &\leq \tilde{\mathcal{O}}\left(\left[\epsilon_i + \sqrt{\frac{d}{\rho_{\min} N\sqrt{T}}}\right]\sqrt{dT}\right)\log^2(1/\delta),\end{aligned}$$

with probability exceeding $1 - c\delta - \frac{1}{\text{poly}(T)}$.

Total Regret: We now characterize the total regret of agent i. We have

$$\begin{aligned}&R_i(T) = R_i^{(c,1)} + R_i^{(c,2)} + R_i^{(p)} \\ &\leq \tilde{O}\left(\sqrt{\frac{d\sqrt{T}}{\rho_{\min} N}}\right)\log(1/\delta) + \epsilon_i\sqrt{T} + \tilde{\mathcal{O}}\left(\left[\epsilon_i + \sqrt{\frac{d}{\rho_{\min} N\sqrt{T}}}\right]\sqrt{dT}\right)\log^2(1/\delta) \\ &\leq \tilde{\mathcal{O}}\left[\left(\epsilon_i\sqrt{T} + \epsilon_i\sqrt{dT}\right) + T^{1/4}\left(\sqrt{\frac{d}{\rho_{\min} N}} + \sqrt{\frac{d^2}{\rho_{\min} N}}\right)\right]\log^2(1/\delta) \\ &\leq \tilde{\mathcal{O}}\left[\epsilon_i\sqrt{dT} + \; T^{1/4}\sqrt{\frac{d^2}{\rho_{\min} N}}\right]\log^2(1/\delta)\end{aligned}$$

with probability at least $1 - c_1\delta - \frac{1}{\text{poly}(T)}$.

[3] In particular we use `ALB-norm` algorithm of [GSK21], with $\tau = \mathcal{O}(1)$, and zero arm biases, and hence no pure exploration to estimate the arm biases.

14 Proof of Corollary 1

In order to obtain the expected regret, one writes expectation as an integral of the tail probabilities and use the high probability bound to compute the tail probability. With this, in the common learning phase, we have

$$\mathbb{E}R_i^{(c,1)} \leq \tilde{O}\left(\sqrt{\frac{d\sqrt{T}}{\rho_{\min}N}}\right),$$

and

$$\mathbb{E}R_i^{(c,2)} \leq \epsilon_i \sqrt{T}.$$

In the personal learning phase we use Corollary 2 of Appendix 15, which says that shifting makes the regret worse in expectation. Hence, using [GSK21, Theorem 1] and converting it to an expected regret, we have

$$\mathbb{E}R_i^{(p)} \leq \tilde{\mathcal{O}}\left(\left[\epsilon_i + \sqrt{\frac{d}{\rho_{\min}N\sqrt{T}}}\right]\sqrt{dT}\right).$$

The final regret bound follows from summing up the above 3 expressions.

15 Shifted OFUL Regret

In this section, we want to establish a relationship between the regret of the standard OFUL algorithm and the shift compensated algorithm. We define the shifted version of OFUL below.

Definition 3. *The OFUL algortihm is used to make a decision of which action to take at time-step t, given the history of past actions $X_1, \cdots, X_{t-1}$ and observed rewards $Y_1, \cdots, Y_{t-1}$. The Γ shifted OFUL is an algorithm identical to OFUL that describes the action to take at time step t, based on the past actions $X_1, \cdots, X_{t-1}$ and the observed rewards $\widetilde{Y}_1^{(\Gamma)}, \cdots, \widetilde{Y}_{t-1}^{(\Gamma)}$, where for all $1 \leq s \leq t-1$, $\widetilde{Y}_s = Y_s - \langle X_s, \Gamma\rangle$.*

Definition 4. *For a linear bandit instance with unknown parameter θ^*, and a sequence of (possibly random) actions $X_{1:T} := X_1, \cdots, X_T$, denote by $\mathcal{R}_T(X_{1:T}) := \sum_{t=1}^T \max_{1\leq j\leq K}\langle \beta_{j,t} - X_t, \theta^*\rangle$.*

Definition 5. *For a linear bandit system with unknown parameter θ^*, and a sequence of (possibly random) actions $X_{1:T} := X_1, \cdots, X_T$, denote by $\mathcal{R}_T^{(\Gamma)}(X_{1:T}) := \sum_{t=1}^T \max_{1\leq j\leq K}\langle \beta_{j,t} - X_t, \theta^* - \Gamma\rangle$.*

Proposition 1. *Suppose for a linear bandit instance with parameter θ^*, an algorithm plays the sequence of actions $X_1, \cdots, X_T$, then*

$$\mathcal{R}_T(X_{1:T}) \leq \mathcal{R}_T^{(\Gamma)}(X_{1:T}) + \sum_{t=1}^T \left(\langle X_t - \underset{\beta\in\{\beta_{1,t},\cdots,\beta_{K,t}\}}{\operatorname{argmax}} \langle \beta, \theta^*\rangle, \Gamma\rangle\right).$$

Proof. From the definition of $\mathcal{R}_T^{(\Gamma)}$, we can write the regret as

$$\begin{aligned}\mathcal{R}_T^{(\Gamma)}(X_{1:T}) &= \sum_{t=1}^T \max_{1\leq j\leq K}\langle \beta_{j,t} - X_t, \theta^* + \Gamma\rangle, \\ &\overset{(a)}{\leq} \sum_{t=1}^T \max_{1\leq j\leq K}\langle \beta_{j,t}, \theta^*\rangle + \langle \beta_t^*, \Gamma\rangle - \langle X_t, \theta^*\rangle - \langle X_t, \Gamma\rangle, \end{aligned} \tag{5}$$

where, $\beta_t^* := \operatorname{argmax}_{\beta\in\{\beta_{1,t},\cdots,\beta_{K,t}\}}\langle \beta, \theta^*\rangle$. The inequality (a) follows from the following elementary fact.

Lemma 3. *Let $\mathcal{X}$ be a compact set, and functions $f, g : \mathcal{X} \to \mathbb{R}$, such that $\sup_{x\in\mathcal{X}} |f(x)| < \infty$ and $\sup_{x\in\mathcal{X}} |g(x)| < \infty$. Then,*

$$\max_{x\in\mathcal{X}}(f(x) + g(x)) \geq \max_{x\in\mathcal{X}} f(x) + \min_{x\in\mathcal{X}} g(x).$$

that Rewriting Equation (5), we see that

$$\mathcal{R}_T^{(\Gamma)}(X_{1:T}) \leq \mathcal{R}_T + \sum_{t=1}^{T} \langle \beta_t^* - X_t, \Gamma \rangle,$$

and thus the proposition is proved. □

Corollary 2. *If for every time $t \geq 1$, the set of K context vectors $\beta_{1,t}, \cdots, \beta_{K,t}$ are all 0 mean random variables, then*

$$\mathbb{E}[\mathcal{R}_T] \leq \mathbb{E}[\mathcal{R}_T^{(\Gamma)}].$$

Corollary 3. *Suppose for all time t, $\operatorname{argmax}_{\beta\in\{\beta_{1,t},\cdots,\beta_{K,t}\}} \langle \beta, \theta^* \rangle = \operatorname{argmax}_{\beta\in\{\beta_{1,t},\cdots,\beta_{K,t}\}} \langle \beta, \Gamma \rangle$. Then,*

$$\mathcal{R}_T(X_{1:T}) \leq \mathcal{R}_T^{(\Gamma)}(X_{1:T}).$$

Proof. From the hypothesis of the theorem, we can observe the following,

$$\begin{aligned}\sum_{t=1}^{T} \left(\langle X_t - \operatorname*{argmax}_{\beta\in\{\beta_{1,t},\cdots,\beta_{K,t}\}} \langle \beta, \theta^* \rangle, \Gamma \rangle \right) &= \sum_{t=1}^{T} \left(\langle X_t - \operatorname*{argmax}_{\beta\in\{\beta_{1,t},\cdots,\beta_{K,t}\}} \langle \beta, \Gamma \rangle, \Gamma \rangle \right), \\ &\leq 0.\end{aligned}$$

Plugging the above bound into Proposition 1 completes the proof. □

15.0.1 High Probability Bound on $\mathcal{R}_T^{(\Gamma)}$

Lemma 4. *Suppose the K context vectors $\beta_1, \cdots, \beta_K$ are such that for all i, $||\beta_i|| \leq 2$ and for all $i \neq j$, $|\langle \beta_i - \beta_j, \theta^* \rangle| \geq 4||\theta^* - \Gamma||$, where θ^* is the unknown linear bandit parameter and Γ is a fixed vector. Then*

$$\operatorname*{argmax}_{1\leq j\leq K} \langle \beta_j, \theta_i^* \rangle = \operatorname*{argmax}_{1\leq j\leq K} \langle \beta_j, \Gamma \rangle.$$

Proof. We will prove the following more stronger statement. Let $i \neq j \in [K]$ be such that $\langle \theta^*, \beta_i \rangle \geq \langle \theta^*, \beta_j \rangle$. Then, under the hypothesis of the proposition statement, we have $\langle \theta^*, \beta_i - \beta_j \rangle \geq 4||\theta^* - \Gamma||$. Thus, the following chain holds,

$$\begin{aligned}\langle \beta_i - \beta_j, \Gamma \rangle &= \langle \beta_i - \beta_j, \theta^* \rangle + \langle \beta_i - \beta_j, \Gamma - \theta^* \rangle, \\ &\geq 4||\theta^* - \Gamma|| + \langle \beta_i - \beta_j, \Gamma - \theta^* \rangle, \\ &\geq 4||\theta^* - \Gamma|| - ||\beta_i - \beta_j||||\Gamma - \theta^*||, \\ &\geq 0.\end{aligned}$$

The first inequality follows from the hypothesis of the proposition statement, the second follows from Cauchy Schwartz inequality and the last follows from the fact that $||\beta_i - \beta_j|| \leq 2$. Thus, we have shown that under the hypothesis of the Proposition, the ordering of the coordinates whether by inner product with θ^* or with Γ remains unchanged. In particular, the argmax is identical. □

Lemma 5. *Let θ^* be a fixed vector with $\|\theta^*\| \leq 1$, and $\Gamma \in \mathbb{R}^d$ be any arbitrary vector such that $||\theta^* - \Gamma|| \leq \psi$, for some $\psi < \frac{1}{2\sqrt{2}}$. Let $\beta_1, \cdots, \beta_K$ be i.i.d. vectors, each distributed as the normalized standard Gaussian vector in d dimensions, i.e., $\beta_1 \sim \mathcal{N}\left(0, \frac{1}{d}\mathbb{I}_d\right)$. Then,*

$$\mathbb{P}\left[\operatorname*{argmax}_{1\leq j\leq K}\langle\beta_j, \theta_i^*\rangle = \operatorname*{argmax}_{1\leq j\leq K}\langle\beta_j, \Gamma\rangle\right] \geq \left(1 - \binom{K}{2}e^{-\frac{d}{4}(1-8\psi^2)^2} - Ke^{-\frac{\sqrt{5}-1}{2}d}\right).$$

Proof. Denote by the *Good event* $\mathcal{E} := \left\{\operatorname{argmax}_{1\leq j\leq K}\langle\beta_j, \theta_i^*\rangle = \operatorname{argmax}_{1\leq j\leq K}\langle\beta_j, \Gamma\rangle\right\}$ From Lemma 4, we know that a sufficient condition for event $\mathcal{E}$ to hold is that for all $i \neq j$, we have $\left|\langle\theta^*, \beta_i - \beta_j\rangle\right| \geq 2||\theta^* - \Gamma||$ and for all i, $||\beta_i|| < 2$. Thus, from a simple union bound, we get

$$\begin{aligned}\mathbb{P}[\mathcal{E}^c] &\leq \sum_{1\leq i<j\leq K}\mathbb{P}\left[\left|\langle\theta^*, \beta_i - \beta_j\rangle\right| \leq 4||\theta^* - \Gamma||\right] + \sum_{i=1}^{K}\mathbb{P}[||\beta_i|| \geq 2], \\ &= \binom{K}{2}\mathbb{P}\left[\left|\langle\theta^*, \beta_1 - \beta_2\rangle\right| \leq 4||\theta^* - \Gamma||\right] + K\mathbb{P}[||\beta_1|| \geq 2].\end{aligned}$$

The second equality follows from the fact that $\beta_1, \cdots, \beta_K$ are i.i.d. Now, since $||\theta^*|| \leq 1$, we have from Cauchy Schwartz that, almost-surely, $\left|\langle\theta^*, \beta_1 - \beta_2\rangle\right| \leq ||\beta_1 - \beta_2||$. Thus,

$$\begin{aligned}\mathbb{P}\left[|\langle\theta^*, \beta_1 - \beta_2\rangle| \leq 4||\theta^* - \Gamma||\right] &\leq \mathbb{P}[||\beta_1 - \beta_2|| \leq 4||\theta^* - \Gamma||], \\ &\leq \mathbb{P}[||\beta_1 - \beta_2|| \leq 4\psi], \\ &= \mathbb{P}[||\beta_1 - \beta_2||^2 \leq 16\psi^2], \\ &= \mathbb{P}\left[\frac{d}{2}||\beta_1 - \beta_2||^2 \leq 8\psi^2 d\right], \\ &\overset{(a)}{\leq} e^{-\frac{d}{4}(1-8\psi^2)^2}\end{aligned}$$

The first inequality follows from Cauchy Schwartz, and the fact that $||\theta^*|| \leq 1$. The last inequality follows from the classical anti-concentration inequality of Massart and Laurent [LM00]. As $\beta_1 - \beta_2 \sim \mathcal{N}(0, 2d^{-1}\mathbb{I}_d)$. Thus, $\frac{d}{2}||\beta_1 - \beta_2||^2$ is a Chi-square distributed random variable with d-degrees of freedom. The inequality in [LM00] gives that for any χ^2 random variable with d degrees of freedom U, and $x \in \mathbb{R}_+$,

$$\mathbb{P}[U \leq d - 2\sqrt{dx}] \leq e^{-x}.$$

Now, plugging in $x = \frac{d}{4}(1 - 8\psi^2)^2$ in the above inequality yields step (a).

Finally, we also need to ensure that the context vectors $\beta_1, \cdots, \beta_K$ have norms bounded by $2\sqrt{d}$. This can also be similarly be bounded by the upper tail inequality as

$$\begin{aligned}\mathbb{P}[||\beta_1|| \geq 2] &= \mathbb{P}[d||\beta_1||^2 \geq 4d], \\ &\overset{(b)}{\leq} e^{-\frac{\sqrt{5}-1}{2}d}.\end{aligned}$$

The inequality (b) follows from the upper-tail concentration bound for χ^2 random variables. Observe that $d||\beta_1||^2$ is a χ^2 random variable with d degrees of freedom. For any χ^2 random variable U is a with d degrees of freedom, and for any $x > 0$, it is known that [LM00]

$$\mathbb{P}[U \geq d + 2x + 2\sqrt{dx}] \leq e^{-x}.$$

Substituting $x = \frac{\sqrt{5}-1}{2}$ in the above equation yields step (b). Putting this all together concludes the proof. □

Lemma 6. *Consider a linear bandit instance with parameter θ^* with $||\theta^*|| \leq 1$ and the context vectors at each time are sampled uniformly and independently from the scaled normal distribution in d dimensions, i.e., the contexts are i.i.d. across time and arms from $\mathcal{N}\left(0, d^{-1}\mathbb{I}_d\right)$. Let $\Gamma \in \mathbb{R}^d$ be such that $||\theta^* - \Gamma|| \leq \psi$ with $\psi < \frac{1}{2\sqrt{2}}$, and $X_{1:T} = (X_1, \cdots, X_T)$ be the set of actions chosen by the Γ shifted OFUL. Then, with probability at-least $\left(1 - \binom{K}{2} e^{-\frac{d}{4}(1-8\psi^2)^2} - Ke^{-\frac{\sqrt{5}-1}{2}d}\right)$,*

$$\mathcal{R}_T(X_{1:T}) \leq \mathcal{R}_T^{(\Gamma)}(X_{1:T}).$$

Proof. This follows by combining Lemma 5 and 4. □