Non-Stationary Representation Learning in Sequential Multi-Armed Bandits

Yuzhen Qin [1] **Tommaso Menara** [1] **Samet Oymak** [2] **ShiNung Ching** [3] **Fabio Pasqualetti** [1]

Abstract

Most of the existing theoretical studies on representation learning are focused on batch tasks. However, in practical decision-making scenarios, the learner often observes tasks in a sequential fashion. In such sequential problems, learning good representations becomes more challenging as the underlying task representation may change over time. In this paper, we address non-stationary representation learning in sequential multi-armed linear bandits. We introduce an online algorithm that is able to detect task switches and learn and transfer a non-stationary representation in an adaptive fashion. We derive a regret upper bound for our algorithm, which significantly outperforms the existing ones that do not learn the representation. Our bound provides theoretical insights into problem-dependent quantities and reveals the excess regret incurred by representation learning, non-stationarity, and task switch detection.

1. Introduction and Problem Setup

Representation learning is an important tool to perform transfer learning, wherein common features shared by tasks are extracted and generalized. Humans naturally transform experiences into compact internal representations that guide actions in future complex environments (Radulescu et al., 2021). Recent years have witnessed an increasing interest in studying representation learning (see (Bengio et al., 2013)).

Due to promising seminal results, theoretical research on representation learning has sparked considerable interest (Du et al., 2020; Tripuraneni et al., 2020a; Bouniot et al., 2020; Yang et al., 2021; Hu et al., 2021). Existing studies focus on representation learning through batch tasks, and are restricted to static representations, relying on the working assumption that *one representation fits all tasks*. However, in realistic decision-making scenarios, the learner often faces tasks that appear in sequence, where the underlying representation may change over time. Such scenarios necessitate human-like reasoning and a more fluid approach to representation learning. To this end, this paper takes an important step towards a deeper theoretical understanding of representation learning in non-stationary environments. To model the sequential decision-making scenario, we consider a series of multi-task linear bandits. Further, to model the non-stationary environment, we assume that the underlying representation shared by the sequential bandits is time-varying, but the learner is not explicitly informed of representation changes. We introduce an online algorithm (illustrated in Fig. 1) that is able to learn and transfer non-stationary representations in an adaptive fashion.

Related Work A seminal theoretical contribution on representation learning can be found in (Baxter, 2000), where tasks are sampled from the same underlying environment. Some recent relevant studies are found in (Maurer et al., 2016; Balcan et al., 2019; Tripuraneni et al., 2020a; Du et al., 2020; Tripuraneni et al., 2020b; Bouniot et al., 2020). Representation learning is also applied to sequential decision-making problems. Particularly, some recent studies have revealed the benefits of representation learning in playing multi-task linear bandits (Lattimore et al., 2020; Yang et al., 2021; Li et al., 2021), where bandit tasks are played simultaneously and are assumed to share a single linear representation. In (Azar et al., 2013), sequential transfer is studied with a finite set of tasks. In (Soare et al., 2014), the authors address sequential transfer across tasks that are close in ℓ_2 distance. We depart from existing results by addressing sequential transfer in representation learning, where the underlying representation is allowed to be non-stationary.

Problem Setup In this paper, we consider a sequential multi-task linear bandit problem, where the agent plays multiple bandits that appear in sequence. The learning agent is initially given an action set $\mathcal{A} \subset \mathbb{R}^d$. At each round t, the agent chooses an action $x_t \in \mathcal{A}$ and receives a reward $y_t = x_t^\top \theta_{s(t)} + \eta_t$, where $\{\eta_t\}$ is a random noise sequence, and the unknown coefficient $\theta_{s(t)} \in \mathbb{R}^d$ is randomly drawn from the task set $\mathcal{T} := \{\omega_1, \omega_2, \dots, \omega_K\}$ (which can consist of infinite tasks). We assume that each bandit is played

[1]Department of Mechanical Engineering, University of California, Riverside, Riverside, CA, 92521 [2]Department of Electrical and Computer Engineering, University of California, Riverside, Riverside, CA, 92521 [3]Department of Electrical and Systems Engineering and Biomedical Engineering, Washington University in St. Louis, St. Louis, MO, USA. Correspondence to: Yuzhen Qin <yuzhenq@ucr.edu>, Fabio Pasqualetti <fabiopas@engr.ucr.edu>.

ICML 2021 Workshop on Reinforcement Learning Theory

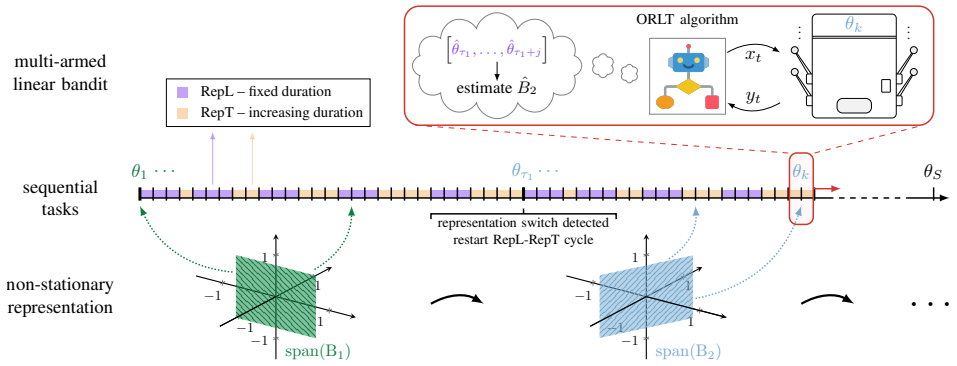

Figure 1. Non-stationary representation learning algorithm. Two key features: 1) representation switch detection, and 2) balancing Representation Learning (RepL) and Transfer (RepT).

for N rounds, after which a new task is sampled from $\mathcal{T}$. However, the agent does not know exactly when a task switch occurs, and only knows that each task is played about $\Theta(N)$ rounds. The agent plays S bandits in total, and the goal is to maximize the cumulative reward after playing the random sequence of bandits $\mathcal{S} := \{\theta_1, \theta_2, \ldots, \theta_S\}$ for a total of $T := NS$ rounds. To measure the agent's performance, we utilize the (pseudo-)regret, which is defined as: $R_T = \sum_{t=1}^{T}(g(\theta_{s(t)})^\top \theta_{s(t)} - x_t^\top \theta_{s(t)})$, where $g(\theta) := \arg\max_{x \in \mathcal{A}} x^\top \theta$ is the optimal arm given a task θ. The agent goal is equivalent to minimizing R_T.

As typically done in the literature for continuous armed bandits (e.g., see (Rusmevichientong and Tsitsiklis, 2010; Yang et al., 2021)), we first assume that the action set $\mathcal{A}$ is an ellipsoid of the form $\mathcal{A} = \{x \in \mathbb{R}^d : x^\top Q^{-1} x \leq 1\}$, where Q is a symmetric positive definite matrix. Second, we assume that there exist $\phi_{\min}$ and $\phi_{\max}$ so that $\Theta(1) = \phi_{\min} \leq \|\omega_s\| \leq \phi_{\max} = \Theta(1)$ for any $s \in [K]$. Third, the additive noise $\eta_t \in \mathbb{R}$ is i.i.d. δ^2-sub-Gaussian with zero mean and satisfies $\mathbb{E}[e^{\lambda \eta_t}] \leq \exp(\delta^2 \lambda^2 / 2)$ for any $\lambda > 0$. We also make the following assumption on $\mathcal{S}$.

Assumption 1 (Non-Stationary Representation) *For the bandit task sequence $\mathcal{S}$, we assume that there exist some positive integers $\tau_1, \tau_2, \ldots, \tau_{n_c}$ such that for any subsequence $\mathcal{S}_i = \{\theta_{\tau_{i-1}+1}, \theta_{\tau_{i-1}+1}, \ldots, \theta_{\tau_i}\}$, there is a linear feature extractor $B_i \in \mathbb{R}^{d \times r_i}$ with orthonormal columns so that $\theta_s = B_i \alpha_s$ for all $s = \tau_{i-1}+1, \ldots, \tau_i$, where $\alpha_s \in \mathbb{R}^{r_i}$. Also, there exists $r \ll d$ such that $r \geq r_i$ for any i, and $\tau_i - \tau_{i-1} \geq D \gg r$ for some D.*

The matrix B_i corresponds to a *linear representation* for the bandit tasks in the corresponding subsequence $\mathcal{S}_i$. With a slight abuse of terminology, we also refer to each B_i as a representation. Assumption 1 states that the representation is non-stationary since each B_i has a *duration* of $D_i = \tau_i - \tau_{i-1}$, which is the number of consecutive bandit tasks from $\text{span}(B_i)$. To infer B_i, the agent must play a sufficient number of tasks from $\mathcal{S}_i$. Thus, we lower-bound the duration of each representation as $D_i \geq D$ for all i. Note that the learner has no prior knowledge of the number of total representations n_c and their respective durations D_i. We further make the following assumptions.

Assumption 2 (Detectability and Task Diversity) *There exists a constant $\kappa_1 > 0$ such that $\left\| B_{i+1}^\top [B_i]_\perp / \sqrt{n - r_i} \right\| \geq \kappa_1 \delta / \phi_{\min}$ for any i (B_{i+1} appears right after B_i). Moreover, we assume that $\|g(\theta)^\top (\theta' - \theta)\| \geq \kappa_2 \delta$ for some $\kappa_2 > 0$ for any $\theta, \theta' \in \mathcal{T}$. Finally, there exists a constant $\ell = \Theta(r)$ such that any subsequence of length ℓ in $\mathcal{S}_i$ (whose representation is $B_i \in \mathbb{R}^{d \times r_i}$) satisfies $\sigma_{r_i}(W_{s,l} W_{s,\ell}^\top) \geq \nu > 0$ for any s, where $W_{s,l} = [\theta_{\tau_{i-1}+s+1}, \ldots, \theta_{\tau_{i-1}+s+\ell}]$.*

The first (resp. second) statement ensures that two consecutive representations (resp. bandit tasks) are sufficiently different. These two assumptions guarantee that reward changes caused by representation or task switches can be distinguished from the ordinary fluctuations due to noise, which is important for detecting the underlying representations or tasks switches. Intuitively, if a new representation or a new task brings changes that are not even distinguishable from noise, there may be no need to detect the new representation or the new task. The third statement guarantees that the sequential tasks are well "spread out" in each subspace $\text{span}(B_i)$ so that representations can be recovered. Although our theoretical results rely on these assumptions, the key idea in this paper can be generalized to more general situations even if these assumptions are not satisfied.

Contributions The contribution of this paper is threefold. First, in sharp contrast to existing results on representation learning, our algorithm is adaptive to non-stationary representations. Inspired by the observation that humans shift their attention when the environment changes (Radulescu et al., 2021), our algorithm can detect representation switches and learn the new representations. Second, our algorithm balances representation learning (RepL) and representation transfer (RepT) in an adaptive fashion by alter-

Algorithm 1 RepL algorithm

Input: Approx. horizon $\Theta(N)$, exploration length N_1.
for $t = 1 : N_1$ **do**
$x_t = \lambda_0 a_i, i = (t-1 \mod d) + 1$;
end for
compute $\hat{\theta} = (XX^\top)^{-1}XY$
for $t = N_1 + 1 : N$ **do** $x_t = \arg\max_{\mathrm{x}\in\mathcal{A}} \mathrm{x}^\top \hat{\theta}$; **end for**

nating between them. In comparison to existing algorithms, no prior knowledge of the number of tasks that share the same representation is required. Third, we provide an upper bound for our algorithm, which grows as

$$R_T \lesssim \underbrace{Sr\sqrt{N}}_{\text{Oracle}} + \sum_{i=1}^{n_c} \underbrace{dr\sqrt{D_i N}}_{\text{cost of RepL}} + \underbrace{Sn_{\text{rsd}}}_{\text{Non-stationarity}} + \underbrace{Sn_{\text{tsd}}}_{\text{Task Switches}}.$$

Our bound explicitly reveals the excess regret incurred by representation learning, non-stationarity, and detection of task switches. Our algorithm outperforms the algorithms that do not learn a representation significantly (such algorithms have a regret bound $\tilde{\Theta}(Sd\sqrt{N})$, e.g., see (Dani et al., 2008; Rusmevichientong and Tsitsiklis, 2010; Li et al., 2021)). Notice that, if there is only one representation, our upper bound becomes $\tilde{O}(Sr\sqrt{N} + dr\sqrt{SN} + Sn_{\text{rsd}} + Sn_{\text{tsd}})$. As we mentioned earlier, it even outperforms the algorithm in (Yang et al., 2021) that plays bandits simultaneously. This is because the entire set of the S bandit tasks may not share a common representation, even if some of its subsets do. Finally, comparing our upper bound with the lower bound for sequential bandits, we find that there is only a gap of $\sqrt{r}$ between the former and the latter.

2. Online Representation Learning and Transfer Algorithm

In this section, we introduce our main algorithm for Online Representation Learning and Transfer (ORLT), which we illustrate in Algorithm. 1. This algorithm consists of two base algorithms: 1) Representation Learning (RepL), and 2) Representation Transfer (RepT). It strikes the balance in an online fashion by alternating between RepL and RepT. Furthermore, ORLT has two key abilities: 1) task switch detection, and 2) representation switch detection. Let us first discuss these four key components separately.

Representation Learning Representation learning is performed by collecting sequential data generated by playing bandit tasks in sequence. Given a sequence of tasks $\theta_1, \theta_2, \ldots, \theta_k$, let $B \in \mathbb{R}^{d\times\hat{r}}$ be their linear representation. To learn B, the agent plays these tasks sequentially using the RepL algorithm in Algorithm. 1 and obtains their respective estimate $\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_k$. Given a bandit task $\theta \in \mathbb{R}^d$, the agent first explores it for N_1 rounds by repeatedly

Algorithm 2 RepT algorithm

Input: Approx. horizon $\Theta(N)$, $\hat{B} \in \mathbb{R}^{d\times\hat{r}}$
set the exploration length $N_2 = c\hat{r}\sqrt{N}$
for $t = 1 : N_2$ **do**
$x_t = \lambda_0 a'_i, i = (t-1 \mod \hat{r}) + 1$
end for
Compute $\hat{\alpha} = (\hat{B}^\top X_{N_2} X_{N_2}^\top \hat{B})^{-1}\hat{B}^\top X_{N_2} Y_{N_2}$, $\hat{\theta} = \hat{B}\hat{\alpha}$
for $t = N_2 + 1 : N$ **do** $x_t = \arg\max_{\mathrm{x}\in\mathcal{A}} \mathrm{x}^\top \hat{\theta}$ **end for**

Algorithm 3 Outlier Detection algorithm (OD)

Input: $\hat{B} \in \mathbb{R}^{d\times\hat{r}}$ and n_{rsd}
Generate a random orthonormal matrix $P \in \mathbb{R}^{(d-\hat{r})\times n_{\text{rsd}}}$, let $M = \hat{B}_\perp P$.
for $t = 1, \ldots, n_{\text{rsd}}$ **do**
$x_t = \lambda_0 [M]_t$, collect y_t
end for
if $Y_{n_{\text{rsd}}} \notin \mathcal{C}_{n_{\text{rsd}}}$ **then**
outlier detected
end if

taking d independent actions $\lambda_0 a_1, \lambda_0 a_2, \ldots, \lambda_0 a_d$. Here, $A = [a_1, \ldots, a_d]$ can be any orthonormal basis of the space $\mathbb{R}^d$, and the scalar λ_0 ensures that each action is in the action set $\mathcal{A}$. After N_1 rounds, the agent estimates the task coefficient and takes the greedy action for $N - N_1$ rounds.

Let $\hat{W}_k = [\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_k]$. We perform singular value decomposition (SVD) of the matrix $\hat{W}_k\hat{W}_k^\top/k$, and let $\hat{B}$ be the top $\hat{r}$ singular vectors of $\hat{W}_k\hat{W}_k^\top/k$. Note that $\hat{r}$ can be time-varying due to the non-stationary environment. To compute the current $\hat{B}$, we *ignore the singular vectors that are associated with small singular values* (in theory, those that are much less than ν for ν introduced in Assumption 2) to handle the non-stationarity in an adaptive way. This yields $\hat{B}$ as an estimate of B.

Representation Transfer Once the agent has obtained an estimated representation $\hat{B}$ with dimension $\hat{r}$, i.e., $\hat{B} \in \mathbb{R}^{d\times\hat{r}}$, it can now generalize $\hat{B}$ to other bandit tasks by invoking the RepT algorithm in Algorithm. 2. The RepT algorithm is an ETC-like algorithm with an input $\hat{B}$ comprising two stages: exploration and commitment. The main feature of RepT is that the exploration is conducted in the subspace $\text{span}(\hat{B})$. Therefore, fewer steps of exploration are actually needed. Specifically, the exploration length N_2 is set to $c\hat{r}\sqrt{N}$ with a constant $c > 0$ and $\hat{r}$ is the dimension of $\text{span}(\hat{B})$. In fact, this choice of N_2 finds an optimal balance between exploration and exploitation. Note that $a'_i = [\hat{B}]_i$ for any i in Algorithm 2.

Representation Switch Detection To deal with the non-stationary environments, our algorithm detects representation switches. To do that, the agent makes an assessment

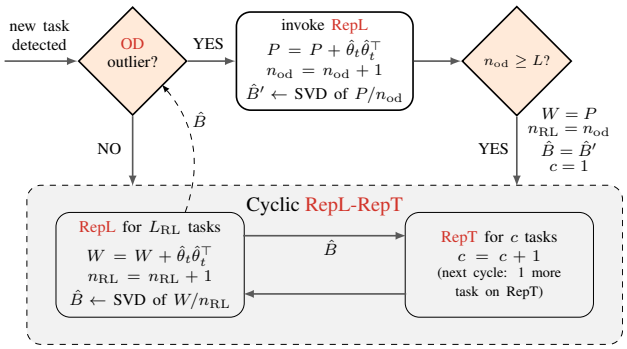

Figure 2. The ORLT algorithm

on every new bandit task by initially taking *probing actions* using the knowledge of the current representation (i.e., $\hat{B} \in \mathbb{R}^{d\times\hat{r}}$). If abnormal rewards are generated by these trials (the ones beyond the confidence interval $\mathcal{C}_{n_{\text{rsd}}} = \{Y_{n_{\text{rsd}}} \in \mathbb{R}^{n_{\text{rsd}}} : \left| \|Y_{n_{\text{rsd}}}\|_2 - \delta\sqrt{n_{\text{rsd}}} \right| \leq \gamma_1 \delta \sqrt{n_{\text{rsd}}}\}$), the agent concludes that the new task is an outlier (see Algorithm. 3). Only if several outliers appear consecutively, the agent decides that a representation switch has happened. This makes our algorithm robust to occasional outliers.

Task Switch Detection Our algorithm is also able to detect task switches by distinguishing the reward changes caused by task switches from the fluctuations due to noise.

We start by constructing confidence intervals for the rewards after some initial rounds of commitment (say, n_1). For all $t_0 \geq n_1$, we first compute the average of the reward received so far in the commitment phase $\bar{y}_{t_0} = \frac{1}{t_0}\sum_{i=1}^{t_0} y_i$. Then, we monitor rewards in a moving window of length n_{tsd}, i.e., $y_{t_0+1}, \ldots, y_{t_0+n_{\text{tsd}}}$. Likewise, the average reward in the time window $Y_{n_{\text{tsd}}} = \frac{1}{n_{\text{tsd}}}\sum_{i=1}^{n_{\text{tsd}}} y_{t_0+i}$ belongs to the interval $[g(\hat{\theta})^\top\theta - \xi_2, g(\hat{\theta})^\top\theta + \xi_2]$ with high probability. Therefore, we can construct a confidence interval for the reward as $\mathcal{C}_{\text{tsd}}(t_0) = [\bar{y}_{t_0} - \xi_1 - \xi_2, \bar{y}_{t_0} + \xi_1 + \xi_2]$ at each round t_0. If the agent observes rewards whose average in the considered window goes beyond this confidence interval, there is a task switch with high probability. If κ_2 in Assumption 2 is sufficiently large and one selects large ξ_1, ξ_2, each task switch can be detected with high probability.

The Main Algorithm: ORLT The main algorithm (ORLT) of this paper is illustrated in Fig. 2. In ORLT, the agent keeps monitoring the rewards to detect task switches. After detecting tasks, ORLT operates in a *cyclic manner* to deal with the sequential non-stationary setting. In each cycle there are two phases: 1) RepL phase, and 2) RepT phase. In the c-th cycle, for the RepL phase, the agent plays L_{RL} bandit tasks (we set $L_{\text{RL}} = \ell$ with ℓ defined in Assumption 2) by invoking Algorithm 1, and for the RepT phase, c tasks are played using Algorithm 2. The length of the RepT phase increases with time (i.e., with c increasing), which means that the RepL phase is activated less and less frequently. In the RepL phase, we set $N_1 = \Theta(d\sqrt{N})$, which is optimal given that we just know each bandit is played about $\Theta(N)$ rounds. Another key feature of the ORLT algorithm is its capability of detecting representation switches (see Algorithm. 3). If a representation switch is detected, we ignore the data collected in the previous representation (removing the collected $\hat{\theta}_t$'s from the matrix W as in Fig. 2) and completely restart the alternating cycle between RepL and RepT (starting RepL and resetting $c = 1$).

The following theorem provides an upper bound for the regret of the ORLT algorithm.

Theorem 1 (Upper bound for the regret) *With Assumptions 1 and 2, let the agent play the S sequential bandit tasks using the ORLT algorithm in Fig. 2. Suppose that n_{c} underlying representations, i.e., $B_1, B_2, \ldots, B_{n_{\text{c}}}$, appear in sequence, where each $B_i \in \mathbb{R}^{d\times r_i}$ is shared by D_i consecutive bandit tasks. Then, the regret of ORLT satisfies the following upper bound*

$$\mathbb{E}R_T = \tilde{O}\big(Sr\sqrt{N} + \sum\nolimits_{i=1}^{n_{\text{c}}} dr\sqrt{D_i N} + Sn_{\text{tsd}} + Sn_{\text{rsd}}\big),$$

where n_{rsd} is the number of trial actions for outlier detection, n_{tsd} is the number of rounds for task switch detection, κ_1 and κ_2 are given in Assumption 2, and c_1 and c_2 are positive constants.

Lower bound. Adapted from (Yang et al., 2021), a natural lower bound for our sequential bandits problem is $\Omega(d\sum_i^{n_{\text{c}}}\sqrt{rD_iN} + Sr\sqrt{N})$. Notice that there is only a gap of $\sqrt{r}$ between the lower bound and our upper bound derived in Theorem 1, which is surprising given the fact that our algorithm learns and transfers representation online.

Numerical Experiments We perform numerical experiments to validate our theoretical results and demonstrate the efficacy of our algorithm.

Fig. 3 (a) contains the comparison of our ORLT algorithms with other algorithms on synthetic data. We consider a sequence $\{\theta_1, \ldots, \theta_S\}$ of bandit tasks with $S = 6000$ and $\theta_i \in \mathbb{R}^d$, where $d = 20$, and each task is played for 1000 rounds. To model the non-stationary representation, we construct 6 representation matrices $B_1 \in \mathbb{R}^{d\times 2}, B_2 \in \mathbb{R}^{d\times 4}, B_3 \in \mathbb{R}^{d\times 2}, B_4 \in \mathbb{R}^{d\times 4}, B_5 \in \mathbb{R}^{d\times 2}, B_6 \in \mathbb{R}^{d\times 2}$ (satisfying Assumption 2), each with 1000 consecutive bandits. We let the following 4 algorithms play these sequential bandits: 1) our ORLT, 2) ETC, which is optimal for individual tasks but does not learn any representation, 3) subspace-oracle, where the representation $[B_1, B_2, \ldots, B_6] \in \mathbb{R}^{d\times 16}$ of all the bandits is known, and 4) the oracle algorithm, where both the non-stationary representations and the switches times are known. We show in Fig. 3 (a) that our algorithm significantly outperforms ETC, demonstrating the benefits of learning non-stationary

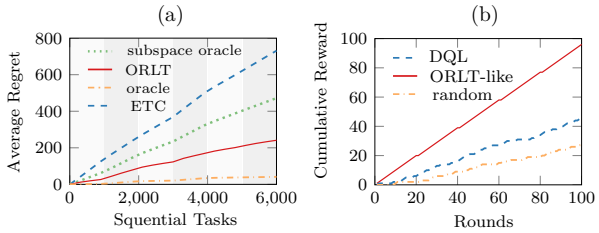

Figure 3. Performance comparison between different algorithms. (a) Synthetic Data: different shades of gray distinguish distinct underlying representations. Average regret is obtained by dividing the cumulative reward by the task number S. (b) WCST.

representations in sequential bandits. Further, our algorithm even outperforms "subspace-oracle". This verifies our earlier theoretical observation: playing bandits sequentially can yield less regret than doing that simultaneously. This is because subsets of a given set of bandits may share much lower-dimensional representations than the one (if it exists) shared by the whole set. The gap observed with respect to the oracle algorithm is explained by the fact that the ORLT must spend some rounds to detect representation switches and to learn new representations sequentially.

We also utilize a slightly modified version of the ORLT algorithm to play the Wisconsin Card Sorting Test (WCST), which is typically utilized to assess "human abstraction and shift of set" (Grant and Berg, 1948). In WCST, a participant is given 4 different cards at the beginning of the test. Then, a number of stimulus cards containing symbols of varying shape, number, and color are presented to the participant in sequence. The participant is asked to associate the stimulus cards to one of the 4 cards on the table according to different rules (i.e., shape, color, number). The underlying rule changes over time, and is not known by the participant. The only feedback the participant receives is whether the classification is correct or not (e.g., receiving reward 1 for correct action, 0 otherwise). By interacting with the sequential tasks, the participant needs to infer which rule dictates the correct association. As we show in the Appendix H, the WCST can be model as a linear bandit problem. Fig. 3 (b) illustrates our algorithm outperforming Deep Q learning and a baseline algorithm. All details can be found in the Appendix.

3. Conclusions and Future Work

In this work, we address representation learning in sequential multi-armed bandits. Unlike most existing studies, the underlying representation is allowed to be non-stationary. We introduce an online algorithm that is able to handle the non-stationarity and outperforms the ones that do not learn representations, or learn static representations. We also provide a regret upper bound for our algorithm.

We have assumed that the representation changes sufficiently slowly so that every representation can be learned with high probability. An interesting case that we leave as a topic for future research would be to guarantee high rewards when playing sequential tasks sampled from different representations that appear in a mixed sequence.

References

Angela Radulescu, Yeon Soon Shin, and Yael Niv. Human representation learning. *Annual Reviews in Neuroscience*, 2021.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

Simon S Du, Wei Hu, Sham M Kakade, Jason D Lee, and Qi Lei. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*, 2020.

Nilesh Tripuraneni, Chi Jin, and Michael I Jordan. Provable meta-learning of linear representations. *arXiv preprint arXiv:2002.11684*, 2020a.

Quentin Bouniot, Ievgen Redko, Romaric Audigier, Angélique Loesch, Yevhenii Zotkin, and Amaury Habrard. Towards better understanding meta-learning methods through multi-task representation learning theory. *arXiv preprint arXiv:2010.01992*, 2020.

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.

Jiachen Hu, Xiaoyu Chen, Chi Jin, Lihong Li, and Liwei Wang. Near-optimal representation learning for linear bandits and linear rl. *arXiv preprint arXiv:2102.04132*, 2021.

Jonathan Baxter. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12:149–198, 2000.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17 (81):1–32, 2016.

Maria-Florina Balcan, Mikhail Khodak, and Ameet Talwalkar. Provable guarantees for gradient-based meta-learning. In *International Conference on Machine Learning*, pages 424–433. PMLR, 2019.

Nilesh Tripuraneni, Michael Jordan, and Chi Jin. On the theory of transfer learning: The importance of task diversity. In H. Larochelle, M. Ranzato, R. Hadsell, M. F.

Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 7852–7862. Curran Associates, Inc., 2020b.

Tor Lattimore, Csaba Szepesvari, and Gellert Weisz. Learning with good feature representations in bandits and in RL with a generative model. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5662–5670, 13–18 Jul 2020.

Yingkai Li, Yining Wang, Xi Chen, and Yuan Zhou. Tight regret bounds for infinite-armed linear contextual bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 370–378. PMLR, 2021.

Mohammad Gheshlaghi Azar, Alessandro Lazaric, and Emma Brunskill. Sequential transfer in multi-armed bandit with finite set of models. In *Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 2*, pages 2220–2228, 2013.

Marta Soare, Ouais Alsharif, Alessandro Lazaric, and Joelle Pineau. Multi-task linear bandits. In *NIPS2014 Workshop on Transfer and Multi-task Learning: Theory meets Practice*, 2014.

Paat Rusmevichientong and John N Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010.

Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. In *the 21st Annual Conference on Learning Theory*, pages 355–366, 2008.

David A Grant and Esta Berg. A behavioral analysis of degree of reinforcement and ease of shifting to new responses in a weigl-type card-sorting problem. *Journal of experimental psychology*, 38(4):404, 1948.

Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*, volume 47. Cambridge University Press, 2018.

Rajendra Bhatia. *Matrix analysis*, volume 169. Springer Science & Business Media, 2013.

Appendix

A. Extended Literature Review

Representation Learning underlies major advances in language processing (Ando et al., 2005; Liu et al., 2019; Lee et al., 2020), drug discovery ((Ramsundar et al., 2015)), and reinforcement learning (Baevski et al., 2019; Teh et al., 2017; D'Eramo et al., 2019). For other empirical studies, we refer the reader to a survey (Bengio et al., 2013). Theoretical studies on representation representation have attracted much attention (Balcan et al., 2019; Tripuraneni et al., 2020a; Du et al., 2020; Tripuraneni et al., 2020b; Bouniot et al., 2020). For instance, (Du et al., 2020) shows that representation learning improves data efficiency, contributing to few-shot learning. In (Tripuraneni et al., 2020a), the method of moments is presented to learn the representation in the multi-task linear regression problem.

Representation learning is also applied to sequential decision-making problems. In (D'Eramo et al., 2020) and (Arora et al., 2020), representation learning is shown to be beneficial in multi-task reinforcement learning and imitation learning tasks. As a popular model for sequential decision-making scenarios, multi-armed bandits have drawn intense attention in the past decades due to their wide-ranging applications (see (Auer, 2002; Dani et al., 2008; Rusmevichientong and Tsitsiklis, 2010; Abbasi-Yadkori et al., 2011; Chu et al., 2011)). Some work investigates low-rank structure (Lale et al., 2019; Jun et al., 2019; Lu et al., 2021) or sparse structure (Abbasi-Yadkori et al., 2012; Hao et al., 2020) on linear and generalized linear bandits. Meta-learning or transfer learning is addressed in (Zhang and Bareinboim, 2017; Cella et al., 2020).

Also, there has been increasing interest in studying bandit problems in non-stationary environments. A large body of work has studied the bandit problem with time-varying task coefficient (e.g., see (Gupta et al., 2011; Besbes et al., 2014; Cheung et al., 2018; Luo et al., 2018; Besbes et al., 2019; Besson and Kaufmann, 2019; Russac et al., 2019; 2020; Fei et al., 2020)), where the reward-generating function is $y_t = x_t^\top \theta_s + \eta_t$ with varying θ_s. Existing results rely on the assumption of a variation bound $V_S = \sum_{s=1}^{S} \|\theta_{s+1} - \theta_s\|$ for the time-varying coefficients (e.g., see (Besbes et al., 2014; Cheung et al., 2018)). Various approaches, including sliding-window (Cheung et al., 2018), exponential discounting (Garivier and Moulines, 2011; Russac et al., 2019), and restarted strategy (Zhao et al., 2020), have been proposed to address the non-stationary nature. Thompson sampling with discounting factors has also been used (Gupta et al., 2011; Raj and Kalyani, 2017; Kim and Tewari, 2020) in non-stationary environments. Particularly, some studies rely on detection of changing points (Cao et al., 2019; Auer et al., 2018; Wu et al., 2018), which is akin to our method to detect task switches. In this paper, we detect task switches by monitoring abrupt reward changes, and under the assumption that tasks are sufficiently mutually different, task switches can be detected with high probability. Moreover, we depart from previous work by addressing an additional type of non-stationarity, as in our framework the representation underlying time-varying task coefficients is also allowed to change over time. A recent work on linear supervised

learning investigates the situation where tasks' distribution cannot be captured by a single representation (Denevi et al., 2021). A conditional meta-learning approach is introduced to generate a representation tailored to the task at hand. Different from this work, we focus on the sequential decision-making scenario (i.e, sequential bandits) and do not require side information about tasks.

B. A General Instrumental Lemma

The following lemma will be utilized throughout this Appendix.

Lemma 1 ((Vershynin, 2018)) *Let $X_1, X_2, \dots, X_n$ be n independent random variables such that X_i is δ-sub-Gaussian. Then, for any $\xi \geq 0$, the average $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$ satisfies*

$$\Pr\left[\bar{X} \geq \xi\right] \leq \exp\left(-\frac{n\xi^2}{2\delta^2}\right),$$

and

$$Pr\left[\bar{X} \leq -\xi\right] \leq \exp\left(-\frac{n\xi^2}{2\delta^2}\right).$$

C. Supporting Results for RepL

The following theorem captures the angle distance between the two subspaces described by $\hat{B}$ and B.

Theorem 2 (Accuracy of learned representation) *Given $k \geq \hat{r}$ bandits $\theta_1, \theta_2, \dots, \theta_k$ drawn from $\mathcal{T}$, suppose $B \in \mathbb{R}^{d\times\hat{r}}$ is their representation. Let $\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_k$ be their respective estimate after playing each of them for N rounds using the RepL algorithm. Suppose $\sigma_{\hat{r}}(W_k) \geq \nu > 0$. Then,*

$$\sin\boldsymbol{\theta}(\hat{B}, B) \leq \tilde{O}\left(\frac{d\delta}{\lambda_0\nu}\sqrt{\frac{1}{kN_1}}\right), \tag{C.1}$$

with probability at least $1 - \frac{1}{kN_1}$, where δ is the variance parameter of the sub-Gaussian noise η_t.

Notice that unlike the results in (Tripuraneni et al., 2020a; Yang et al., 2021), (C.1) has no obvious dependence on $\hat{r}$. That is because in those studies it is assumed that $\mathrm{tr}(W_k W_k/k) = \mu\hat{r}\nu$ where μ is the average condition number. However, in our case $\mathrm{tr}(W_k W_k/k) = \Theta(1)$ since $\phi^2_{\min} \leq \mathrm{tr}(W_k W_k/k) \leq \phi^2_{\max}$. In comparison to (Yang et al., 2021), our estimate $\hat{B}$ has a smaller angle distance to the true B by a factor of $\sqrt{d}$.

To prove Theorem 2, we will use the following lemma.

Lemma 2 (Bernstein's inequality (Vershynin, 2018)) *Let $X_1, X_2, \dots, X_k$ be independent zero-mean $n \times n$ symmetric random matrices, such that there exists $M > 0$ such that $\|X_i\| \leq M$ almost surely for all $i = 1, 2, \dots, k$. Then, for any $t \geq 0$, it holds that*

$$\Pr\left[\left\|\sum_{i=1}^{k} X_i\right\| \geq t\right] \leq 2d\exp\left(\frac{-2t^2}{\sigma^2 + Mt/3}\right),$$

where $\sigma^2 = \left\|\sum_{i=1}^{k}\mathbb{E}X_i\right\|$.

Proof of Theorem 2 For each bandit θ_i, the model is described by $y_t = x_t^\top\theta_i + \eta_t$. Pre-multiplying both sides with x_t yields $x_t y_t = x_t x_t^\top\theta_i + x_t\eta_t$. In the RepL algorithm (see Fig. 1), x_t periodically picks each column in the matrix $\lambda_0[a_1, a_2, \dots, a_d]$, with $[a_1, a_2, \dots, a_d]$ forming an orthonormal basis of the space $\mathbb{R}^d$. Since $[a_1, a_2, \dots, a_d]$ can be any orthonormal basis, we let it be the standard basis of $\mathbb{R}^d$ without loss of generality. Since N_1 rounds are spent on exploration in the RepL algorithm, it follows that

$$\sum_{t=1}^{N_1} x_t y_t = \sum_{t=1}^{N_1} x_t x_t^\top\theta_i + \sum_{t=1}^{N_1} x_t\eta_t.$$

Denote $X := [x_1, x_2, \dots, x_{N_1}]$, $Y := [y_1, \dots, y_{N_1}]^\top$, and $\eta := [\eta_1, \eta_2, \dots, \eta_{N_1}]^\top$. Note that $\hat{\theta}$ is computed by $\hat{\theta}_i = (XX^\top)^{-1}XY$. Since $Y = X^\top\theta_i + \eta$, it can be observed that

$$\hat{\theta}_i = (XX^\top)^{-1}X(X^\top\theta_i + \eta) = \theta_i + (XX^\top)^{-1}X\eta.$$

Algebraic computations yield

$$\hat{\theta}_i\hat{\theta}_i^\top = \theta_i\theta_i^\top + \theta_i\eta^\top X^\top(XX^\top)^{-1} + (XX^\top)^{-1}X\eta\theta_i^\top + (XX^\top)^{-1}X\eta\eta^\top X^\top(XX^\top)^{-1}.$$

Recall that η_i is an independent sub-Gaussian random variable with zero mean, so it follows that $\mathbb{E}\eta = 0$ and $\mathbb{E}\eta\eta^\top = \delta^2 I_{N_1}$. Then, the expectation of $\hat{\theta}_i\hat{\theta}_i^\top$ can be computed as

$$\begin{aligned}\mathbb{E}\hat{\theta}_i\hat{\theta}_i^\top &= \theta_i\theta_i^\top + (XX^\top)^{-1}X\mathbb{E}\eta\eta^\top X^\top(XX^\top)^{-1}\\ &= \theta_i\theta_i^\top + \delta^2(XX^\top)^{-1}.\end{aligned}$$

Since $[a_1, a_2, \dots, a_d]$ is the standard basis of $\mathbb{R}^d$, it holds that $\sum_{i=1}^{d} a_i a_i^\top = \lambda_0 I_d$. Without loss of generality, we assume that N_1 is a multiple of d, then it follows that $XX^\top = \frac{N_1}{d}\lambda_0^2 I_d$. Therefore, we have $(XX^\top)^{-1} = \frac{d}{\lambda_0^2 N_1}$. For the purpose of notational simplicity, denote $D := \frac{d}{\lambda_0^2 N_1} I_d$, then it follows that

$$\mathbb{E}\hat{\theta}_i\hat{\theta}_i^\top = \theta_i\theta_i^\top + \delta^2 D,$$

and

$$\hat{\theta}_i\hat{\theta}_i^\top = \theta_i\theta_i^\top + D\underbrace{\left(\theta_i\eta^\top X^\top + X\eta\theta_i^\top\right)}_{A} + D^2\underbrace{\left(X\eta\eta^\top X^\top\right)}_{C}. \tag{C.2}$$

Define a set of new variables $z_i = \frac{1}{k}\hat{\theta}_i\hat{\theta}_i^\top - \frac{1}{k}\left(\theta_i\theta_i^\top + D\right)$. From (C.2), we have

$$z_i = \frac{1}{k}(DA + D^2C - D).$$

Then, the expectation of z_i satisfies

$$\mathbb{E}z_i^2 = \frac{1}{k^2}\mathbb{E}\left[D^2A^2 + D^4C^2 + D^2 + D^3(AC + CA) - 2D^2A - 2D^3C\right]. \quad \text{(C.3)}$$

Since D is deterministic, to compute $\mathbb{E}z_i^2$, it suffices to calculate $\mathbb{E}A^2, \mathbb{E}C^2, \mathbb{E}(AC + CA), \mathbb{E}A$, and $\mathbb{E}C$. Next, we calculate these terms one by one.

For $\mathbb{E}A^2$, it holds that

$$\begin{aligned}\mathbb{E}A^2 &= \mathbb{E}(\theta_i\eta^\top X^\top)^2 + \mathbb{E}(X\eta\theta_i^\top)^2 \\ &\quad + \mathbb{E}(\theta_i\eta^\top X^\top X\eta\theta_i^\top) + \mathbb{E}(X\eta\theta_i^\top\theta_i\eta^\top X^\top) \\ &= 2\frac{\lambda_0^2\delta^2N_1}{d}\theta_i\theta_i^\top + N_1\lambda_0^2\delta^2\theta_i\theta_i^\top + \theta_i^\top\theta_i\frac{\lambda_0^2\delta^2N_1}{d}I_d.\end{aligned}$$

For $\mathbb{E}C^2$, we have

$$\begin{aligned}\mathbb{E}C^2 &= \mathbb{E}\left[X\eta\eta^\top X^\top X\eta\eta^\top X^\top\right] \\ &= \mathbb{E}\left[\eta^\top X^\top X\eta \cdot X\eta\eta^\top X^\top\right] \\ &= \mathbb{E}\left[\sum_{t=1}^{N_1}\eta_t^4 x_t^\top x_t x_t x_t^\top\right] \\ &= \psi_4\lambda_0^2\left(\sum_{t=1}^{N_1}x_t x_t^\top\right) \\ &= \psi_4\lambda_0^2 XX^\top = \frac{\psi_4\lambda_0^4N_1}{d}I_d,\end{aligned}$$

where $\psi_4 = \mathbb{E}\eta_t^4$ (ψ_4 always exists since η_t is a sub-Gaussian random variable).

For $\mathbb{E}(AC + CA)$, it holds that

$$\begin{aligned}\mathbb{E}(AC + CA) =& \mathbb{E}\left[\left(\theta_i\eta^\top X^\top + X\eta\theta_i^\top\right)X\eta\eta^\top X^\top \right. \\ &\left. + X\eta\eta^\top X^\top\left(\theta_i\eta^\top X^\top + X\eta\theta_i^\top\right)\right] \\ =& \mathbb{E}\left[\eta^\top X^\top X\eta \cdot \theta_i\eta^\top X^\top\right] \\ &+ \mathbb{E}\left[\theta_i^\top X\eta \cdot X\eta\eta^\top X^\top\right] \\ &+ \mathbb{E}\left[\eta^\top X^\top\theta_i \cdot X\eta\eta^\top X^\top\right] \\ &+ \mathbb{E}\left[\eta^\top X^\top X\eta \cdot X\eta\theta_i^\top\right] \\ =& \frac{\lambda_0^3\psi_3N}{d}\theta_i\mathbf{1}_d^\top + \frac{\lambda_0^3\psi_3N}{d}\mathrm{diag}(\theta_i) \\ &+ \frac{\lambda_0^3\psi_3N}{d}\mathrm{diag}(\theta_i) + \frac{\lambda_0^3\psi_3N}{d}\mathbf{1}_d\theta_i^\top \\ =& \frac{\lambda_0^3\psi_3N}{d}(\theta_i\mathbf{1}_d^\top + \mathbf{1}_d\theta_i^\top) \\ &+ \frac{2\lambda_0^3\psi_3N}{d}\mathrm{diag}(\theta_i).\end{aligned}$$

Notice that $\mathbb{E}A = 0$. It remains the calculate $\mathbb{E}C$, which satisfies

$$\mathbb{E}C = \mathbb{E}X\eta\eta^\top X^\top = \frac{\lambda_0^2\delta^2N}{d}I_d.$$

Overall, substituting all the above terms into Eq. (C.3), we have

$$\begin{aligned}\mathbb{E}z_i^2 =& \frac{1}{k^2}\frac{d^2}{\lambda_0^4N_1^2}\Big[2\frac{\lambda_0^2\delta^2N_1}{d}\theta_i\theta_i^\top + N_1\lambda_0^2\delta^2\theta_i\theta_i^\top \\ &+ \theta_i^\top\theta_i\frac{\lambda_0^2\delta^2N_1}{d}I_d + D^2\frac{\psi_4\lambda_0^4N_1}{d}I_d + I_d \\ &+ D(\frac{\lambda_0^3\psi_3N}{d}(\theta_i\mathbf{1}_d^\top + \mathbf{1}_d\theta_i^\top) \\ &+ \frac{2\lambda_0^3\psi_3N}{d}\mathrm{diag}(\theta_i)) - 2D\frac{\lambda_0^2\delta^2N}{d}I_d\Big] \\ \le& \frac{d^2\delta^2}{k^2\lambda_0^2N_1}(\frac{2}{d} + 1)\theta_i\theta_i^\top + O(\frac{d^2}{k^2\lambda_0^4N_1^2}I_d).\end{aligned}$$

Let $\sigma^2 = \|\sum_{i=1}^k \mathbb{E}z_i^2\|_F$, which satisfies

$$\sigma^2 \lesssim \left\|\frac{d^2\delta^2}{k\lambda_0^2N_1}\frac{1}{k}\sum_{i=1}^k\theta_i\theta_i^\top\right\|_F \le O\left(\frac{d^2\delta^2}{k\lambda_0^2N_1}\mathrm{tr}(W_k)\right).$$

Since for any $\theta \in \mathcal{T}$, it holds that $\phi_{\min} \le \|\theta\| \le \phi_{\max}$ with $\phi_{\min} = \Theta(1)$ and $\phi_{\max} = \Theta(1)$, it follows that $\mathrm{tr}(W_k) = \Theta(1)$. Therefore, it holds that

$$\sigma^2 \lesssim O\left(\frac{d^2\delta^2}{k\lambda_0^2N_1}\right).$$

Applying Lemma 2 with $t = 2c_1\log(2d/\delta) + c_2\sqrt{4\sigma^2\log(2d/\delta)}$ for sufficiently large $c_1, c_2 > 0$, we have

$$\left\|\sum_{i=1}^k z_i\right\|_F \lesssim \frac{d\delta}{\lambda_0}\sqrt{\frac{1}{kN_1}}\sqrt{\log(d/\delta)} + \log(d/\delta))$$

with probability at least $1 - \delta$. Let $\delta = \frac{1}{kN_1}$, then with probability $1 - \frac{1}{kN_1}$, the following inequality holds

$$\left\|\sum_{i=1}^k z_i\right\|_F \lesssim \frac{d\delta}{\lambda_0}\sqrt{\frac{1}{kN_1}}\sqrt{\log(kdN_1)} + \log(kdN_1).$$

Notice that $\sum_{i=1}^k z_i = \hat{W}_k - (W_k + \frac{d}{\lambda_0^2N_1}I_d)$. If we let $W_k' = W_k + \frac{d}{\lambda_0^2N_1}I_d$, one can observe that W_k' share the same left singular vectors as W_k. Observe that

$$\begin{aligned}\left\|\hat{W}_k - W_k'\right\|_F &:= \|\Delta\|_F \\ &\lesssim \frac{d\delta}{\lambda_0}\sqrt{\frac{1}{kN_1}}\sqrt{\log(kdN_1)} + \log(kdN_1).\end{aligned}$$

From the Davis-Kahan $\sin\boldsymbol{\theta}$ Theorem (Bhatia, 2013), we have

$$\sin\boldsymbol{\theta}(B,\hat{B}) \leq \frac{\|\hat{B}_\perp^\top (W_k - W_k')B\|}{\omega} \leq \frac{\|\hat{W}_k - W_k'\|_F}{\omega}, \tag{C.4}$$

where $\omega = \inf_{1\leq i\leq r, r<j\leq d} |\lambda_i(W_k') - \lambda_j(\hat{W}_k)|$. From the Weyl's Theorem, $|\lambda_i(W_k') - \lambda_i(\hat{W}_k)| \leq \|\hat{W}_k - W_k'\|_F = \left\|\sum_{j=1}^k z_j\right\|_F$ for any $i = 1,\ldots,d$. Since $\lambda_i(W_k') = 0$ for all $i \geq r+1$, it holds that $|\lambda_i(\hat{W})| \leq \|\Delta\|_F$ for all $i \geq r+1$. Recall that σ_r is the r-th largest eigenvalue of W, therefore $\omega \geq \sigma_k - \|\Delta\|_F$. From the Assumption 2, we know $\sigma_r \geq \nu$, therefore we obtain

$$\begin{aligned}
\sin\boldsymbol{\theta}(U_1,\hat{U}_1) &\lesssim \frac{\|\Delta\|_F}{\sigma_r - \|\Delta\|_F} \\
&\lesssim \frac{d\delta}{\lambda_0\nu}\left(\sqrt{\frac{1}{kN_1}}\sqrt{\log dkN_1}\right. \\
&\quad \left. + M\log dkN_1\right) \\
&\leq \frac{d\delta}{\lambda_0\nu}\sqrt{\frac{1}{kN_1}}\cdot \mathrm{polylog}(d,N_1,k).
\end{aligned}$$

The proof is complete. ■

D. Supporting Results for RepT

Lemma 3 *Given $\theta \in \mathcal{T}$, suppose that there exists $B \in \mathbb{R}^{d\times\hat{r}}$ with orthonormal columns such that $\theta = B\alpha$ for some $\alpha \in \mathbb{R}^{\hat{k}}$. Assume that an estimate $\hat{B}$ is known by the learner and satisfies $\sin\boldsymbol{\theta}(\hat{B},B) \leq \varepsilon$. If the learner plays this bandit task θ for N rounds using the RepT algorithm with the input $\hat{B}$, then the regret satisfies $R_N = O(\hat{r}\sqrt{N} + N\varepsilon^2)$.*

Given a bandit task $\theta \in \mathbb{R}^d$, earlier studies ((Dani et al., 2008; Rusmevichientong and Tsitsiklis, 2010; Li et al., 2021)) have shown that the optimal regret bound is $\Theta(d\sqrt{N})$ in the same setting. By contrast, Lemma 3 states that with the knowledge of an estimated low-dimensional representation $\hat{B}$ (i.e., $\hat{r} \ll d$) the regret can be significantly reduced provided the estimate $\hat{B}$ is sufficiently accurate (i.e., small ε), indicating the advantages of learning and transferring the representation.

Proof In the RepT algorithm shown in Fig 2, $N_2 = O(\hat{r}\sqrt{N})$ rounds are spent in the exploration phase and the other $N - N_2$ rounds are in the commitment phase. Without loss of generality, we assume that N_2 is a multiple of $\hat{r}$.

Since $\theta = B\alpha$, then the model becomes $y_t = x_t^\top B\alpha + \eta_t$. Instead of directly estimating θ, we estimate α. Denote $X = [x_1, x_2, \ldots, x_{N_2}]$ and $Y = [y_1, y_2, \ldots, y_{N_2}]^\top$. At the end of the exploration phase, it holds that

$$\hat{\alpha} = (\hat{B}^\top XX^\top \hat{B})^{-1}\hat{B}^\top XY.$$

Since x_t repeatedly takes actions from $\lambda_0 a_1', \ldots, \lambda_0 a_{\hat{r}}'$, it holds that $XX^\top = \frac{N_2}{\hat{r}} AA^\top$ with $A = [\lambda_0 a_1', \ldots, \lambda_0 a_{\hat{r}}']$. Therefore, we have

$$\hat{\alpha} = \left(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B}\right)^{-1}\hat{B}^\top XY.$$

As $Y = X^\top B\alpha + \eta$ with $\eta = [\eta_1, \ldots, \eta_{N_2}]^\top$, we have

$$\begin{aligned}
\hat{\alpha} &= \left(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B}\right)^{-1}\hat{B}^\top X(X^\top B\alpha + \eta) \\
&= \left(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B}\right)^{-1}\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top B\alpha \\
&\quad + \left(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B}\right)^{-1}\hat{B}^\top X\eta.
\end{aligned}$$

As $\hat{\theta} = \hat{B}\hat{\alpha}$ and $\theta = B\alpha$, it follows that

$$\begin{aligned}
&\hat{B}\hat{\alpha} - B\alpha = \\
&\underbrace{\hat{B}\left(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B}\right)^{-1}\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top B\alpha - B\alpha}_{s_1} \\
&+ \underbrace{\hat{B}\left(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B}\right)^{-1}\hat{B}^\top X\eta}_{s_2}.
\end{aligned}$$

Then, it holds that $\mathbb{E}\left[\|\hat{\theta}(c) - \theta\|^2\right] \leq \mathbb{E}\|s_1\|^2 + \mathbb{E}\|s_2\|^2$ since η_t is independent random variable with zero mean. Next, we evaluate the two terms on the right-hand side of this inequality separately.

First, we evaluate $\mathbb{E}\|s_1\|^2$. Observe that $B = (\hat{B}\hat{B}^\top + \hat{B}_\perp\hat{B}_\perp^\top)B$, therefore, it holds that

$$\begin{aligned}
s_1 &= \hat{B}(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B})^{-1}\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top(\hat{B}\hat{B}^\top \\
&\quad + \hat{B}_\perp\hat{B}_\perp^\top)B\alpha - B\alpha \\
&= (\hat{B}\hat{B}^\top B\alpha - B\alpha) \\
&\quad + \hat{B}(\hat{B}^\top AA^\top \hat{B})^{-1}\hat{B}^\top AA^\top \hat{B}_\perp\hat{B}_\perp^\top B\alpha \\
&= \hat{B}_\perp\hat{B}_\perp^\top B\alpha + \hat{B}(\hat{B}^\top AA^\top \hat{B})^{-1}\hat{B}^\top AA^\top \hat{B}_\perp\hat{B}_\perp^\top B\alpha
\end{aligned}$$

Then, $\mathbb{E}\|s_1\|^2$ satisfies

$$\begin{aligned}
\mathbb{E}\|s_1\|_F^2 &\leq 2\left\|\hat{B}_\perp\hat{B}_\perp^\top B\alpha\right\|_F^2 \\
&\quad + 2\left\|\hat{B}(\hat{B}^\top AA^\top \hat{B})^{-1}\hat{B}^\top AA^\top \hat{B}_\perp\hat{B}_\perp^\top B\alpha\right\|_F^2 \\
&\leq 2\phi_{\max}^2\|\hat{B}_\perp^\top B\|_F^2 \\
&\quad + 2\phi_{\max}^2\left\|(\hat{B}^\top AA^\top \hat{B})^{-1}\hat{B}^\top AA^\top \hat{B}_\perp\right\|_F^2 \\
&\quad \cdot \|\hat{B}_\perp^\top B\|_F^2.
\end{aligned}$$

It can be derived that there exists $\mu > 0$ such that $\left\|(\hat{B}^\top AA^\top \hat{B})^{-1}\hat{B}^\top AA^\top \hat{B}_\perp\right\|_F^2 \leq \mu$. Therefore, it holds that $\mathbb{E}\|s_1\|^2 \leq 2\phi_{\max}^2(1+\mu)\|\hat{B}_\perp^\top B\|^2$. Since $\sin\boldsymbol{\theta}(\hat{B}, B) \leq \varepsilon$, we arrive at

$$\mathbb{E}\|s_1\|^2 \leq 2\phi_{\max}^2(1+\mu)\varepsilon^2.$$

Second, we evaluate $\mathbb{E}\|s_2\|^2$ as

$$\begin{aligned}
\mathbb{E}\|s_2\|^2 &= \left\|\hat{B}\left(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B}\right)^{-1}\hat{B}^\top X\right\|^2 \mathbb{E}\eta_t^2 \\
&\leq X^\top \hat{B}\left(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B}\right)^{-2}\hat{B}^\top X \cdot \mathbb{E}\eta_t^2 \\
&\leq \left\|\hat{B}\left(\frac{N_2}{\hat{r}}\hat{B}^\top AA^\top \hat{B}\right)^{-2}\hat{B}^\top\right\|_F \\
&\quad \cdot \left\|\frac{N_1}{\hat{r}}A^\top \hat{B}\hat{B}^\top A\right\|_F \cdot \mathbb{E}\eta_t^2(s) \leq \frac{\hat{r}^2\bar{\delta}^2}{N_2},
\end{aligned}$$

where $\bar{\delta}$ is the variance of the sub-Gaussian noise η_t, depending on the variance proxy parameter δ. As $N_2 = O(\hat{r}\sqrt{N})$, then we obtain that $\mathbb{E}\|s_2\|^2 \leq \frac{\hat{r}\bar{\delta}^2}{\sqrt{N}}$. Combining $\mathbb{E}\|s_1\|^2$ and $\mathbb{E}\|s_2\|^2$, we have $\mathbb{E}\left[\|\hat{\theta}-\theta\|^2\right] \leq \frac{\hat{r}\bar{\delta}^2}{\sqrt{N}} + 2\phi_{\max}^2(1+\mu)\varepsilon^2$.

From (Rusmevichientong and Tsitsiklis, 2010), it holds that $\max_{x\in\mathcal{A}} x^\top\theta - \max_{x\in\mathcal{A}} x^\top\hat{\theta} \leq J\frac{\|\theta-\hat{\theta}\|^2}{\|\theta_t\|}$, where J is a constant that exists since the action set $\mathcal{A}$ is an ellipsoid of the form $\{x \in \mathbb{R}^d : x^\top Q^{-1}x \leq 1\}$. Consequently,

$$\begin{aligned}
\mathbb{E}[\max_{x\in\mathcal{A}} x^\top\theta - \max_{x\in\mathcal{A}} x^\top\hat{\theta}] &\leq J\frac{\hat{r}\bar{\delta}^2}{\phi_{\min}\sqrt{N}} \\
&\quad + 2\frac{1}{\phi_{\min}}J\phi_{\max}^2(1+\mu)\varepsilon^2.
\end{aligned}$$

For the commitment phase, there are $N - N_2$ steps. Thus, the overall regret satisfies

$$\begin{aligned}
\mathbb{E}R_N &\leq N_2\phi_{\max} + (N-N_2)\left(\max_{x\in\mathcal{A}} x^\top\theta - \max_{x\in\mathcal{A}} x^\top\hat{\theta}\right) \\
&\leq \hat{r}\sqrt{N}\phi_{\max} \\
&\quad + N\left(J\frac{\hat{r}\bar{\delta}^2}{\phi_{\min}\sqrt{N}} + 2\frac{1}{\phi_{\min}}J\phi_{\max}^2(1+\mu)\varepsilon^2\right) \\
&= O(\hat{r}\sqrt{N} + N\varepsilon^2),
\end{aligned}$$

which completes the proof. ∎

E. Supporting Results for Representation Switch Detection

The main result that supports the representation switch detection in Section 3 is in the following lemma.

Lemma 4 (Probability of Outlier Detection) *Suppose Assumption 2 holds. If the underlying representation switches from B_i to B_{i+1} for any $i = 1, \ldots, n_c - 1$, then any bandit task in B_{i+1} can be detected as an outlier by the agent using the OD algorithm in Fig. 3 with probability at least*

$$p_1 = 1 - \exp\left(-\frac{C((\kappa_1 - 4(\gamma_1+1))^2\delta^2 n_{\text{rsd}}}{4K^4}\right),$$

where κ_1 is given in Assumption 2, δ is the variance proxy parameter of the noise η_t, and C and K are constants.

One can let $\gamma_1 = \mu_1\kappa_1$ for some $\mu_1 > 0$, then the probability that any outlier can be detected becomes at least $p_1 = 1 - \exp(-\frac{c_1\kappa_1^2\delta^2 n_{\text{rsd}}}{4})$ for some $c_1 > 0$. It can be observed that the probability increases as κ_1 or n_{rsd} grow larger. In other words, it is easier to detect a representation switch if two representations are sufficiently different. Meanwhile, more probing actions also increase the probability of detecting representation switches. Before proving this lemma, we present some instrumental lemmas.

Lemma 5 (Random projection) *Let P be a projection from $\mathbb{R}^n$ onto a random m-dimensional subspace uniformly distributed in the Grassmann manifold $G_{n,m}$ (which consists of all m-dimensional subspaces in $\mathbb{R}^n$). Let $x \in \mathbb{R}^n$ be a fixed point and $\xi > 0$. Then, with probability at least $1 - \exp(-c\xi^2 m)$, we have*

$$(1-\xi)\sqrt{\frac{m}{n}}\|x\|_2 \leq \|Px\|_2 \leq (1+\xi)\sqrt{\frac{m}{n}}\|x\|_2.$$

Lemma 6 (Action selection by random projection) *Given a matrix $B \in \mathbb{R}^{d\times r}$ with orthonormal columns and a bandit task $\theta \in \mathbb{R}^d$. Let $z = B_\perp^\top\theta \in \mathbb{R}^{(d-r)\times 1}$. Let P be a projection matrix from $\mathbb{R}^{d-r}$ onto a random m-dimensional subspace uniformly distributed in the Grassmann manifold $G_{(d-r),m}$. Then, with probability at least $1 - \exp(-c\xi^2 m)$, we have*

$$(1-\xi)\sqrt{\frac{m}{d-r}}\|z\|_2 \leq \|Pz\|_2 \leq (1+\xi)\sqrt{\frac{m}{d-r}}\|z\|_2.$$

The proof of Lemma 6 directly follows from Lemma 5. In this paper, we let $\xi = \frac{1}{2}$ without loss of generality. Then, it hold that $\frac{1}{2}\sqrt{\frac{m}{d-r}}\|z\|_2 \leq \|Pz\|_2 \leq \frac{3}{2}\sqrt{\frac{m}{d-r}}\|z\|_2$ with probability at least $1 - \exp(-\frac{c}{4}m)$. Next, we present another lemma, whose proof can be found in Chapter 3.1 of (Vershynin, 2018).

Lemma 7 (Concentration of the norm) *Suppose that $X = [X_1, X_2, \ldots, X_n]^\top$ is a random vector, where $X_1, \ldots, X_n$ are independent δ-sub-Gaussian random*

variable. Then, for any $\xi > 0$ it holds that

$$\Pr\left[\left|\|X\|_2 - \sqrt{n}\delta\right| \geq \xi\right] \leq 2\exp\left(-\frac{c\xi^2}{K^4}\right), \quad \text{(E.1)}$$

where c is an absolute constant and $K = \max_i \|X_i\|_{\psi_2}$.

We are now ready to prove Lemma 4.

Proof of Lemma 4 Recall that the agent takes n_{rsd} probing actions as in Fig. 3, and denote the corresponding rewards as in $Y_{n_{\text{rsd}}} := [y_1, \ldots, y_{n_{\text{rsd}}}]^\top$. Following Lemma 7, we build an confidence interval for $Y_{n_{\text{rsd}}}$

$$\mathcal{C}_{n_{\text{rsd}}} = \left\{Y_{n_{\text{rsd}}} \in \mathbb{R}^{n_{\text{rsd}}} : \left|\|Y_{n_{\text{rsd}}}\|_2 - \delta\sqrt{n_{\text{rsd}}}\right| \leq \gamma_1\delta\sqrt{n_{\text{rsd}}}\right\}.$$

From Lemma 7, the probability of $y_t \notin \mathcal{C}_{n_{\text{rsd}}}$ is less than $2\exp\left(-\frac{c\gamma_1^2\delta^2 n_{\text{rsd}}}{K^4}\right)$ for some absolute constants c and K.

If $\|B_\perp^\top\theta\| = \rho$, from Lemma 6 we have $\|P^\top B_\perp^\top\theta\| = \|PP^\top B_\perp^\top\theta\| \geq \frac{1}{2}\rho\sqrt{\frac{n_{\text{rsd}}}{d-\hat{r}}}$ with probability at least $1 - \exp(-\frac{c}{4}n_{\text{rsd}})$ for some constant c (since $PP^\top B_\perp^\top\theta$ can be taken as projecting $B_\perp^\top\theta$ onto the random subspace spanned by P). From the reward generating function $Y_{n_{\text{rsd}}} = \lambda_0 P^\top \hat{B}_\perp^\top\theta + \eta = \eta$, one can derive that to ensure that $Y_{n_{\text{rsd}}} \in \mathcal{C}_{n_{\text{rsd}}}$, it should hold that

$$\|\eta\| \geq \frac{1}{2}\rho\sqrt{\frac{n_{\text{rsd}}}{d-\hat{r}}} - (\gamma_1+1)\delta\sqrt{n_{\text{rsd}}}. \quad \text{(E.2)}$$

From Assumption 2, we know that $\rho \geq \kappa_1\delta\sqrt{d-r_i}$. From lemma 7, the inequality (E.2) holds with probability less than

$$2\exp\left(-\frac{c((\kappa_1 - 4(\gamma_1+1))^2\delta^2 n_{\text{rsd}}}{4K^4}\right).$$

The proof is complete. ■

F. Supporting Result for Task Switch Detection

Recall that $\bar{y}_{t_0} = \sum_{i=1}^{t_0} = y_i$, where all i's are in the commitment phase. Observing from $y_i = x_i^\top\theta + \eta_i$ that

$$x_i = g(\hat{\theta}) = \arg\max_{\mathrm{x}\in\mathcal{A}} \mathrm{x}^\top\hat{\theta}$$

is deterministic, one obtain that y_i is sub-Gaussian with 0 mean. Then, it follows from Lemma 1 that

$$\Pr[\bar{y}_{t_0} - \mathbb{E}y_{t_0} \geq \xi_1] = \Pr[\bar{y}_{t_0} - g(\hat{\theta})^\top\theta \geq \xi_1] \leq \exp\left(\frac{-t_0\xi_1^2}{2\delta^2}\right), \quad \text{(F.1a)}$$

$$\Pr[\bar{y}_{t_0} - \mathbb{E}y_{t_0} \leq -\xi_1] = \Pr[\bar{y}_{t_0} - g(\hat{\theta})^\top\theta \leq -\xi_1] \leq \exp\left(\frac{-t_0\xi_1^2}{2\delta^2}\right), \quad \text{(F.1b)}$$

where the fact $\mathbb{E}y_{t_0} = g(\hat{\theta})^\top\theta$ has been used. In other words, $g(\hat{\theta})^\top\theta$ lies in the confidence interval

$$[\bar{y}_{t_0} - \xi_1, \bar{y}_{t_0} + \xi_1],$$

with probability at least $1 - 2\exp\left(\frac{-t_0\xi_1^2}{2\delta^2}\right)$ for any $\xi_1 \geq 0$.

Recall that the average reward in the moving window of width n_{tsd}, $y_{t_0+1}, \ldots, y_{t_0+n_{\text{tsd}}}$, is $Y_{n_{\text{tsd}}} = \frac{1}{n_{\text{tsd}}}\sum_{i=1}^{n_{\text{tsd}}} y_{t_0+i}$. Likewise, it follows from Lemma 1 that

$$\Pr\left[Y_{n_{\text{tsd}}} \leq g(\hat{\theta})^\top\theta - \xi_2\right] \leq \exp\left(\frac{-n_{\text{tsd}}\xi_2^2}{2\delta^2}\right), \quad \text{(F.2a)}$$

$$\Pr\left[Y_{n_{\text{tsd}}} \geq g(\hat{\theta})^\top\theta + \xi_2\right] \leq \exp\left(\frac{-n_{\text{tsd}}\xi_2^2}{2\delta^2}\right). \quad \text{(F.2b)}$$

for any $\xi_2 \geq 0$.

Combining the above inequalities (F.1) and (F.2), we have

$$\Pr\left[Y_{n_{\text{tsd}}} \leq \bar{y}_{t_0} - \xi_1 - \xi_2\right] \leq \exp\left(\frac{-t_0\xi_1^2}{2\delta^2} + \frac{-n_{\text{tsd}}\xi_2^2}{2\delta^2}\right),$$

$$\Pr\left[Y_{n_{\text{tsd}}} \geq \bar{y}_{t_0} + \xi_1 + \xi_2\right]\right] \leq \exp\left(\frac{-t_0\xi_1^2}{2\delta^2} + \frac{-n_{\text{tsd}}\xi_2^2}{2\delta^2}\right).$$

In other words, if we define a confidence interval based on the observed average reward $\bar{y}_{t_0}$ as follows

$$\mathcal{C}_{\text{tsd}}(t_0) = \left[\bar{y}_{t_0} - \xi_1 - \xi_2, \bar{y}_{t_0} + \xi_1 + \xi_2\right],$$

we have

$$\Pr\left[Y_{n_{\text{tsd}}} \notin \mathcal{C}_{\text{tsd}}(t_0)\right] \leq 1 - 2\exp\left(\frac{-t_0\xi_1^2}{2\delta^2} + \frac{-n_{\text{tsd}}\xi_2^2}{2\delta^2}\right).$$

If $t_0 \gg 1$, it holds that $\exp(-t_0\xi_1^2/2\delta^2) \to 0$. Therefore, the term $\exp(-t_0\xi_1^2/2\delta^2) \to 0$ can be ignored for large t_0. In this paper, t_0 represents the rounds of commitment phase of RepL or RepT algorithm, and thus satisfied $t_0 \gg 1$. Therefore, we ignore the term $\exp(-t_0\xi_1^2/2\delta^2)$. In other words, we have

$$\Pr\left[Y_{n_{\text{tsd}}} \notin \mathcal{C}_{\text{tsd}}(t_0)\right] \leq 1 - 2\exp\left(\frac{-n_{\text{tsd}}\xi_2^2}{2\delta^2}\right). \quad \text{(F.3)}$$

The choice of ξ_2 and the length of observation window n_{tsd} determines the probability of task switch detection.

Lemma 8 (Probability of Task Switch Detection) *For any $\theta \in \mathcal{T}$, denote the greedy action by $g(\theta) = \arg\max_{\mathrm{x}\in\mathcal{A}} \mathrm{x}^\top\theta$. Suppose that Assumption 2 is satisfied, and let $\xi_2 = \mu_2\kappa_2\delta$ with $\mu_2 \leq 1$. Then, all the tasks can be detected with probability at least $1 - O(\exp(\frac{-c_2\kappa_2^2 n_{\text{tsd}}}{2}))$ for some positive constant $c_2 > 0$.*

Proof Suppose the underlying task switches from θ to θ' at t_0, then the reward generating function satisfies

$$y_t = \begin{cases} x_t^\top \theta' + \eta_t = g(\hat{\theta})^\top \theta' + \eta_t, & \text{if } t \geq t_0, \\ x_t^\top \theta + \eta_t = g(\hat{\theta})^\top \theta + \eta_t, & \text{if } t < t_0. \end{cases}$$

The observed average reward $\bar{y}_{t_0}$ is computed by

$$\bar{y}_{t_0} = g(\hat{\theta})^\top \theta + \frac{1}{t_0} \sum_{t=1}^{t_0} \eta_t.$$

The average reward in the moving window $y_{t_0+1}, \ldots, y_{t_0+n_{\text{tsd}}}$ satisfies

$$Y_{n_{\text{tsd}}} = g(\hat{\theta})^\top \theta' + \frac{1}{t_0} \sum_{t=t_0+1}^{t_0+n_{\text{tsd}}} \eta_t.$$

Subsequently, one has

$$|Y_{n_{\text{tsd}}} - \bar{y}_{t_0}| = \left| g(\hat{\theta})^\top (\theta' - \theta) + \frac{1}{n_{\text{tsd}}} \sum_{t=t_0+1}^{t_0+n_{\text{tsd}}} \eta_t - \frac{1}{t_0} \sum_{t=1}^{t_0} \eta_t \right|.$$

From Assumption 2, we know that $|g(\theta)^\top (\theta' - \theta)| \geq \kappa_2 \delta$ for any $\theta, \theta' \in \mathcal{T}$. To ensure $|Y_{n_{\text{tsd}}} - \bar{y}_{t_0}| \leq \xi_2 = \mu_2 \kappa_2 \delta$, given large t_0, it needs to hold that $\left| \frac{1}{n_{\text{tsd}}} \sum_{t=t_0+1}^{t_0+n_{\text{tsd}}} \eta_t \right| \geq \mu_2 \kappa_2 \delta$. From Lemma 1, we deduce that

$$\Pr\left[\left| \frac{1}{n_{\text{tsd}}} \sum_{t=t_0+1}^{t_0+n_{\text{tsd}}} \eta_t \right| \geq \mu_2 \kappa_2 \delta \right] \leq 2 \exp\left(\frac{-n_{\text{rsd}} \kappa_2^2 \mu_2^2}{2} \right). \tag{F.4}$$

Let $\xi_2 = \mu_2 \kappa_2 \delta$ in (F.3), then one can conclude from (F.3) and (F.4) that any task switch can be detected with probability $1 - O(\exp(\frac{-n_{\text{rsd}} \kappa_2^2 c_2}{2}))$ for some positive constant c_2, which completes the proof. ■

G. Proof of Theorem 1

The ORLT algorithm operates in a cyclic manner. At the c-th cycle, L_{RL} tasks are played utilizing the RepL algorithm, and c tasks are played with the RepT algorithm. Before presenting the proof of Theorem 1, let us provide some intermediate results.

Lemma 9 *Let a bandit task $\theta \in \mathcal{T}$ be played utilizing the RepL algorithm in Fig.1 for N steps. If $N_1 = d\sqrt{N}$, the regret in N steps, denoted as R_N, reaches its optimal $R_N = \Theta(d\sqrt{N})$.*

Proof Without loss of generality, we assume N_1 is a multiple of d. Following similar steps as those in Lemma 3.4 of (Rusmevichientong and Tsitsiklis, 2010), we can obtain that after N_1 steps of exploration

$$\mathbb{E}[\|\hat{\theta} - \theta\|^2] \leq \frac{d^2 \bar{\delta}^2}{N_1},$$

where $\bar{\delta}$ is the variance of the sub-Gaussian noise η_t, depending on the variance proxy parameter δ. From (Rusmevichientong and Tsitsiklis, 2010), it holds that $\max_{x \in \mathcal{A}} x^\top \theta - \max_{x \in \mathcal{A}} x^\top \hat{\theta} \leq J \frac{\|\theta - \hat{\theta}\|^2}{\|\theta\|}$, where J is a constant that exists since the action set $\mathcal{A}$ is an ellipsoid of the form $\{x \in \mathbb{R}^d : x^\top Q^{-1} x \leq 1\}$. Since $\|\theta\| \geq \phi_{\min}$, it follows that

$$\mathbb{E}\left[\max_{x \in \mathcal{A}} x^\top \theta - \max_{x \in \mathcal{A}} x^\top \hat{\theta} \right] \leq J \frac{\|\theta - \hat{\theta}\|^2}{\phi_{\min}} \leq J \frac{d^2 \bar{\delta}^2}{N_1 \phi_{\min}}.$$

Further, at the exploration phase it holds with $g(\theta) := \arg\max_{\mathrm{x} \in \mathcal{A}} \mathrm{x}^\top \theta$ that

$$g(\theta)\theta - x_t^\top \theta \leq 2g(\theta)\theta = \max_{x \in \mathcal{A}} x^\top \theta - \max_{x \in \mathcal{A}} x^\top (-\theta) \leq J \frac{\|\theta + \hat{\theta}\|^2}{\|\theta\|} = 2J.$$

Therefore, the total regret in N steps satisfies

$$\mathbb{E}R_N \leq 2JN_1 + (N - N_1) J \frac{d^2 \bar{\delta}^2}{N_1 \phi_{\min}} \leq 2JN_1 + NJ \frac{d^2 \bar{\delta}^2}{N_1 \phi_{\min}}.$$

Let $N_1 = d\sqrt{N}$, then the regret reaches its optimal as

$$\mathbb{E}R_N \leq 2Jd\sqrt{N} + NJ \frac{d^2 \bar{\delta}^2}{d\sqrt{N} \phi_{\min}} = O(d\sqrt{N}).$$

Previous work (e.g., (Dani et al., 2008; Rusmevichientong and Tsitsiklis, 2010; Li et al., 2021)) has demonstrated that the lower bound of the bandit problem in the same setting is $\Omega(d\sqrt{N})$. Therefore, the upper bound of the RepL algorithm matches the lower bound, proving its optimality. ■

Theorem 3 (Regret Bound for subsequences) *Consider a sequence of bandit $\theta_1, \theta_2, \ldots, \theta_{D_i}$. Assume that there is a linear feature extractor matrix $B_i \in \mathbb{R}^{d \times r_i}$ such that for any θ_s in this sequence $\theta_s = B_i \alpha_s$ for some $\alpha_s \in \mathbb{R}^{r_i}$. Let the agent play these sequential bandits with the cyclic RepL-RepT algorithm (shown in Fig. 2). Specifically, in the m-th cycle there are two phase:*

1. *RepL phase: play $L_{\text{RL}} = \ell$ bandits in sequence utilizing the RepL algorithm in Fig. 1 with $N_1 = d\sqrt{N}$, and collect $\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_{L_{\text{RL}}}$; at the end of RepL phase,*

estimate $\hat{B}_i$ by letting it be the top $\hat{r}_i$ singular vectors of the matrix $W = \sum \hat{\theta}_t \hat{\theta}_t^\top$ with t being the round indices such that θ_t is in the RepL phases of all the m cycles (i.e., cycles $1, 2, \ldots, m$).

2. *RepT phase: play m bandit tasks in sequence utilizing the RepT algorithm with input $\hat{B}_i$ in Fig. 2.*

Then, the regret of playing these D_i sequential bandits satisfies

$$\mathbb{E}R_{ND_i} \leq \tilde{O}(dr_i\sqrt{D_iN} + D_ir_i\sqrt{N} + \frac{d}{r_i}\sqrt{D_iN}).$$

Proof In the m-th cycle, the regret in the RepL phase, denoted as $R_{\mathrm{RepL}}(m)$, satisfies $\mathbb{E}R_{\mathrm{RepL}}(m) = O(L_{\mathrm{RL}}d\sqrt{N})$, which follows from Lemma 9 straightforwardly. At the end of the m-th cycle, mL_{RL} sequential bandits are played using the RepL algorithm. From Theorem 2, we have

$$\sin\boldsymbol{\theta}(\hat{B}_i, B_i) \leq \tilde{O}\left(\frac{d\delta}{\lambda_0\nu}\sqrt{\frac{1}{mL_{\mathrm{RL}}d\sqrt{N}}}\right).$$

Then, m bandit tasks are played in sequence utilizing the RepT algorithm with input $\hat{B}_i$. It follows from Lemma 3 that the regret in the RepT phase, denoted as $R_{\mathrm{RepT}}(m)$, satisfies

$$\begin{aligned}\mathbb{E}R_{\mathrm{RepT}}(m) &\lesssim mr_i\sqrt{N} + mN\frac{d^2\delta^2}{\lambda_0^2\nu^2}\frac{1}{mL_{\mathrm{RL}}d\sqrt{N}} \\ &= \tilde{O}(mr_i\sqrt{N} + \frac{d}{L_{\mathrm{RL}}}\sqrt{N}).\end{aligned}$$

Observe that there are at most $\bar{L} = \lceil\sqrt{2D_i}\rceil$ cycles in the sequence of length D_i since $L_{\mathrm{RL}}\bar{L} + \bar{L}(\bar{L}+1)/2 \geq D_i$. Summing up the regret in the RepL and RepT phases in every cycle, we obtain

$$\begin{aligned}\mathbb{E}R_{ND_i} &\lesssim \bar{L}L_{\mathrm{RL}}d\sqrt{N} + \sum_{m=1}^{\bar{L}}\left(mr_i\sqrt{N} + \frac{d}{L_{\mathrm{RL}}}\sqrt{N}\right) \\ &\leq \bar{L}L_{\mathrm{RL}}d\sqrt{N} + D_ir_i\sqrt{N} + \bar{L}\frac{d}{L_{\mathrm{RL}}}\sqrt{N}.\end{aligned}$$

Since $L_{\mathrm{RL}} = \ell = \Theta(r)$ (see Assumption 1) and $\bar{L} = \lceil\sqrt{2D_i}\rceil$, then

$$\begin{aligned}\mathbb{E}R_{ND_i} &= \sum_{m=1}^{\bar{L}} R_{\mathrm{RepL}}(m) + R_{\mathrm{RepT}}(m) \\ &= \tilde{O}(dr_i\sqrt{D_iN} + D_ir_i\sqrt{N} + \frac{d}{r_i}\sqrt{D_iN}),\end{aligned}$$

which completes the proof. ■

We are now ready to prove Theorem 1.

Proof of Theorem 1 Combing Lemmas 4 and 8 and applying a union bound, one can derive that the probability that every task switch and every representation switch can be detected with probability at least $1 - O(\max\{\exp(\frac{-c_1 n_{\mathrm{rsd}}\kappa_1^2\delta^2}{2}), \exp(\frac{-c_2 n_{\mathrm{tsd}}\kappa_2^2}{2})\})$ for some positive constants c_1 and c_2. This probability is large is κ_1 and κ_2 is large. Since n_{tsd} rounds are spent on detecting every new task and n_{rsd} probing rounds are spent for every new task on detecting representation switches, the regret, denoted as R_{detec}, satisfies $R_{\mathrm{detec}} \lesssim Sn_{\mathrm{tsd}} + Sn_{\mathrm{rsd}}$.

By Assumption 1, there are n_{c} representations, described by $B_1, B_2, \ldots, B_{n_{\mathrm{c}}}$, in the total of S sequential bandits. For each representation B_i, its duration is D_i and satisfies $\sum_{i=1}^{D_i} = S$. Therefore, the total regret of playing all the S sequential bandits can be computed from Theorem 3 as

$$\begin{aligned}\mathbb{E}R_T &\lesssim \sum_{i=1}^{n_{\mathrm{c}}} \mathbb{E}R_{ND_i} \\ &\lesssim \sum_{i=1}^{n_{\mathrm{c}}}\left(dr_i\sqrt{D_iN} + D_ir_i\sqrt{N} + \frac{d}{r_i}\sqrt{D_iN}\right) \\ &\quad + Sn_{\mathrm{tsd}} + Sn_{\mathrm{rsd}} \\ &\lesssim \left(\sum_{i=1}^{n_{\mathrm{c}}}(dr_i\sqrt{D_iN} + D_ir_i\sqrt{N})\right) + Sn_{\mathrm{tsd}} + Sn_{\mathrm{rsd}} \\ &\lesssim \sum_{i=1}^{n_{\mathrm{c}}} dr\sqrt{D_iN} + Sr\sqrt{N} + Sn_{\mathrm{tsd}} + Sn_{\mathrm{rsd}},\end{aligned}$$

where the last inequality has used the fact that $r \geq r_i$ for any i. In other words, $\mathbb{E}R_T = \tilde{O}(\sum_{i=1}^{n_{\mathrm{c}}} dr\sqrt{D_iN} + Sr\sqrt{N} + Sn_{\mathrm{tsd}} + Sn_{\mathrm{rsd}})$, which completes the proof. ■

H. Further Numerical Analysis

In this section, we provide more details of the numerical experiments performed in the main text and also present further experiments to demonstrate our ORLT algorithm.

Synthetic data In addition to the main text, we also analyze the case where the representation does not vary by considering 1000 bandit tasks that share a single representation. We compare our ORLT algorithm with the E^{2}TC algorithm ((Yang et al., 2021)) where the tasks are played simultaneously. Comparing the average reward per task, our algorithm outperforms E^{2}TC (see Fig. 4), validating our earlier analysis.

In the main text and above, we have shown that ORLT is capable of learning and transferring non-stationary representations. In what follows, we showcase the other important feature of ORLT, which is its ability to detect task switches. To do that, we perform some experiments on synthetic data. We consider 2000 bandits in the task sequence $\mathcal{S} = \{\theta_1, \theta_2, \ldots, \theta_S\}$, where $\theta_i \in \mathbb{R}^d$ with $d = 20$. Each

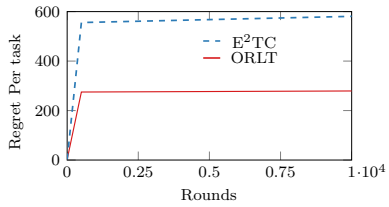

Figure 4. Performance comparison: Even if there is no non-stationarity (only one underlying representation), our online algorithm outperforms the batch algorithm E^{2}TC.

bandit is played for 1000 rounds, then a subsequent task from $\mathcal{S}$ comes into play. However, the agent is not informed of the times of task switches. We let three algorithms to play the sequential bandits: 1) our ORLT, which detects task switches and representation, 2) the classic Explore-Then-Commit (ETC) algorithm, and 3) the phased-Exploration-Greedy-Exploitation (PEGE) algorithm introduced in (Rusmevichientong and Tsitsiklis, 2010). Note that both ETC and PEGE are optimal for individual task, but they are not capable of detecting task switches. We consider that the action set $\mathcal{A}$ is a unit ball, i.e., $\mathcal{A} := \{x \in \mathbb{R}^d : \|x\|_2 \leq 1\}$ and the noise η_t in the reward generating function $y_t = x_t^\top + \eta_t$ is Gaussian with zero mean and 0.1 standard deviation.

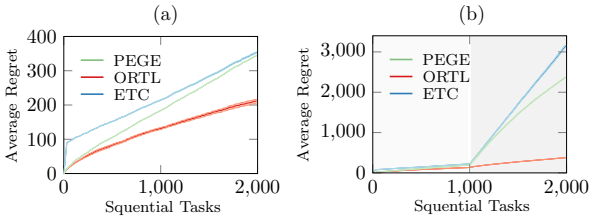

Figure 5. Performance comparison between ORLT and some other algorithms. (a) stationary representation (b) Non-stationary representation. Average regret is obtained by dividing the cumulative reward by the task number S. Experiments are repeated for 10 times, and realizations are contained in the shaded area.

First, we assume that there is $B \in \mathbb{R}^{d\times 2}$ such that $\theta_i = B\alpha_i$ for any i. In other words, there exists a representation, described by B, shared by all the bandit tasks. From Fig. 5 (a), it can be observed that ORLT outperforms ETC and PEGE. Also, the advantage of ORLT increases as the number of tasks in the sequence grows.

Second, we consider the situation where the underlying representation is non-stationary. We assume that the first 1000 tasks in $\mathcal{S}$ share a representation described by $B_1 \in \mathbb{R}^{d\times 2}$, and the remaining 1000 tasks share a representation described by $B_2 \in \mathbb{R}^{d\times 4}$. From Fig. 5 (b), ORLT outperforms the other two algorithms even more significantly. These experiments further demonstrate the important role of attention shift to adapt to new environments (including new tasks and new representations) in facilitating efficient learning.

WCST The WCST is typically utilized to assess "human abstraction and shift of set" (Grant and Berg, 1948). In WCST, a participant is given 4 different cards at the beginning of the test. Then, a number of stimulus cards containing symbols of varying shape, number, and color are presented to the participant in sequence. The participant is asked to associate the stimulus cards to one of the 4 cards on the table according to different rules (i.e., shape, color, number). The underlying rule changes over time, and is not known by the participant. The only feedback the participant receives is whether the classification is correct or not (e.g., receiving reward 1 for correct action, 0 otherwise). By interacting with the sequential tasks, the participant needs to infer which rule dictates the correct association.

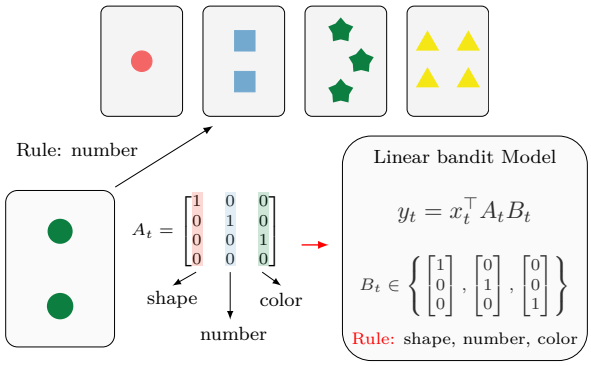

Figure 6. Illustration of WCST and modeling as a linear bandit problem.

We model the WCST as a bandit problem. Specifically, each stimulus card is modeled by a 4×3 matrix A_t. As illustrated in Fig. 6, the columns of A_t stand for shape, number, and color, respectively. Because there are four shapes, four numbers, and four colors, each column can take values from the set

$$\left\{ \begin{bmatrix}1\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\1\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\1\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\1\end{bmatrix} \right\}.$$

There are 3 possible classification rules, i.e., shape, number, and color. We model each of these 3 rules as a standard unit vector B_t, as shown in Fig. 6. Then, the reward is generated by $y_t = x_t^\top A_t B_t$. The agent receives 1 reward if it takes the classification action x_t satisfies $x_t = A_t B_t$, otherwise, it receives 0 reward.

Here the unit vector B_t can be taken as the current representation since the correct classification action can always be computed by $x_t^* = A_t B_t$ if the agent knows the correct B_t, no matter what stimulus card A_t the agent sees. For instance, given classification rule being number (i.e., $B_t = [0,1,0]^\top$), if the agent sees the stimulus card with

two green circles, i.e.,

$$A_t = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix},$$

then correct classification is the second card on the table since it can be computed that $x_t^* = A_t B_t = [0, 1, 0, 0]^\top$. The problem suffices to learn the underlying representation B_t.

Different from the continuous subspaces that we investigated in the theoretical study and in the experiment of synthetic data, the representations here are discrete and can only take values from a set of 3 vectors. Therefore, we adapt our ORLT algorithm to this discrete setting. Inspired by human behaviors, we consider a trial-and-error approach to learn the underlying representation. Initially, we let the agent randomly pick a representation. If the action computed by this representation turns out to be incorrect (generating 0 reward), then the agent switches to another representation until it finds the correct one. After the correct underlying representation is found, the agent can always take the correct actions (i.e., make the correct classification), receiving reward 1 at each round. Once the agent starts to receive reward 0 again, it immediately knows that the underlying representation (classification rule) has changed. Then, it restarts to learn the new representation from scratch.

References for Appendix

Rie Kubota Ando, Tong Zhang, and Peter Bartlett. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6(11), 2005.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy, July 2019.

Jason D Lee, Qi Lei, Nikunj Saunshi, and Jiacheng Zhuo. Predicting what you already know helps: Provable self-supervised learning. *arXiv preprint arXiv:2008.01064*, 2020.

Bharath Ramsundar, Steven Kearnes, Patrick Riley, Dale Webster, David Konerding, and Vijay Pande. Massively multitask networks for drug discovery. *arXiv preprint arXiv:1502.02072*, 2015.

Alexei Baevski, Sergey Edunov, Yinhan Liu, Luke Zettlemoyer, and Michael Auli. Cloze-driven pretraining of self-attention networks. *arXiv preprint arXiv:1903.07785*, 2019.

Yee Whye Teh, Victor Bapst, Wojciech M. Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. Distral: Robust multitask reinforcement learning. In *NIPS*, pages 4499–4509, 2017.

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.

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*, 2020.

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.

Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. In *the 21st Annual Conference on Learning Theory*, pages 355–366, 2008.

Paat Rusmevichientong and John N Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010.

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *NIPS*, volume 11, pages 2312–2320, 2011.

Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214. JMLR Workshop and Conference Proceedings, 2011.

Sahin Lale, Kamyar Azizzadenesheli, Anima Anandkumar, and Babak Hassibi. Stochastic linear bandits with hidden low rank structure. *arXiv preprint arXiv:1901.09490*, 2019.

Kwang-Sung Jun, Rebecca Willett, Stephen Wright, and Robert Nowak. Bilinear bandits with low-rank structure. In *International Conference on Machine Learning*, pages 3163–3172. PMLR, 2019.

Yangyi Lu, Amirhossein Meisami, and Ambuj Tewari. Low-rank generalized linear bandit problems. In *International Conference on Artificial Intelligence and Statistics*, pages 460–468. PMLR, 2021.

Yasin Abbasi-Yadkori, David Pal, and Csaba Szepesvari. Online-to-confidence-set conversions and application to sparse stochastic bandits. In *Artificial Intelligence and Statistics*, pages 1–9. PMLR, 2012.

Botao Hao, Tor Lattimore, and Mengdi Wang. High-dimensional sparse linear bandits. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 10753–10763. Curran Associates, Inc., 2020.

Junzhe Zhang and Elias Bareinboim. Transfer learning in multi-armed bandit: a causal approach. In *Proceedings of the 16th Conference on Autonomous Agents and Multi-Agent Systems*, pages 1778–1780, 2017.

Leonardo Cella, Alessandro Lazaric, and Massimiliano Pontil. Meta-learning with stochastic linear bandits. In *International Conference on Machine Learning*, pages 1360–1370. PMLR, 2020.

Neha Gupta, Ole-Christoffer Granmo, and Ashok Agrawala. Thompson sampling for dynamic multi-armed bandits. In *2011 10th International Conference on Machine Learning and Applications and Workshops*, volume 1, pages 484–489. IEEE, 2011.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Stochastic multi-armed-bandit problem with non-stationary rewards. *Advances in Neural Information Processing Systems*, 27: 199–207, 2014.

Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Hedging the drift: Learning to optimize under non-stationarity. *Available at SSRN 3261050*, 2018.

Haipeng Luo, Chen-Yu Wei, Alekh Agarwal, and John Langford. Efficient contextual bandits in non-stationary worlds. In *Conference On Learning Theory*, pages 1739–1776. PMLR, 2018.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Optimal exploration–exploitation in a multi-armed bandit problem with non-stationary rewards. *Stochastic Systems*, 9(4): 319–337, 2019.

Lilian Besson and Emilie Kaufmann. The generalized likelihood ratio test meets klucb: an improved algorithm for piece-wise non-stationary bandits. *arXiv preprint arXiv:1902.01575*, 2019.

Yoan Russac, Claire Vernade, and Olivier Cappé. Weighted Linear Bandits for Non-Stationary Environments. In *NeurIPS 2019*, Vancouver, Canada, December 2019.

Yoan Russac, Olivier Cappé, and Aurélien Garivier. Algorithms for non-stationary generalized linear bandits. *arXiv preprint arXiv:2003.10113*, 2020.

Yingjie Fei, Zhuoran Yang, Zhaoran Wang, and Qiaomin Xie. Dynamic regret of policy optimization in non-stationary environments. *Advances in Neural Information Processing Systems*, 33, 2020.

Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In *International Conference on Algorithmic Learning Theory*, pages 174–188. Springer, 2011.

Peng Zhao, Lijun Zhang, Yuan Jiang, and Zhi-Hua Zhou. A simple approach for non-stationary linear bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 746–755. PMLR, 2020.

Vishnu Raj and Sheetal Kalyani. Taming non-stationary bandits: A bayesian approach. *arXiv preprint arXiv:1707.09727*, 2017.

Baekjin Kim and Ambuj Tewari. Randomized exploration for non-stationary stochastic linear bandits. In *Conference on Uncertainty in Artificial Intelligence*, pages 71–80. PMLR, 2020.

Yang Cao, Wen Zheng, Branislav Kveton, and Yao Xie. Nearly optimal adaptive procedure for piecewise-stationary bandit: a change-point detection approach. *AISTATS,(Okinawa, Japan)*, 2019.

Peter Auer, Pratik Gajane, and Ronald Ortner. Adaptively tracking the best arm with an unknown number of distribution changes. In *European Workshop on Reinforcement Learning*, volume 14, page 375, 2018.

Qingyun Wu, Naveen Iyer, and Hongning Wang. Learning contextual bandits in a non-stationary environment. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 495–504, 2018.

Giulia Denevi, Massimiliano Pontil, and Carlo Ciliberto. Conditional meta-learning of linear representations. *arXiv preprint arXiv:2103.16277*, 2021.