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; Li 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 $\ell_2$ 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 $A \subset \mathbb{R}^d$. At each round $t$, the agent chooses an action $x_t \in A$ 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 $T := \{\omega_1, \omega_2, \ldots, \omega_K\}$ (which can consist of infinite tasks). We assume that each bandit is played
for $N$ rounds, after which a new task is sampled from $T$. 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 $S := \{\theta_1, \theta_2, \ldots, \theta_S\}$ for a total of $T := NS$ rounds. To measure the agent’s performance, we utilize the (pseudo-)regret, which is defined as: $R_T = \sum_{t=1}^{T} (g(\theta_{s(t)}) - x_t^\top \theta_{s(t)})$, where $g(\theta) := \arg\max_{x \in A} x^\top \theta$ is the optimal arm given a task $\theta$.

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 $A$ is an ellipsoid of the form $A = \{x \in \mathbb{R}^d : x^\top Q^{-1} x \leq 1\}$, where $Q$ is a symmetric positive definite matrix. Second, we assume that there exist $\phi_{\text{min}}$ and $\phi_{\text{max}}$ so that $\Theta(1) = \phi_{\text{min}} \leq \|\omega_s\| \leq \phi_{\text{max}} = \Theta(1)$ for any $s \in [K]$. Third, the additive noise $\eta_t \in \mathbb{R}$ is i.i.d. $\delta^2$-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 $S$.

**Assumption 1** (Non-Stationary Representation) For the bandit task sequence $S$, we assume that there exist positive integers $\tau_1, \tau_2, \ldots, \tau_n$, such that for any feature subspace $S_i = \{\theta_{\tau_{i-1}+1}, \theta_{\tau_{i-1}+1}, \ldots, \theta_{\tau_i}\}$, 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 subspace $S_i$. 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 span($B_i$). To infer $B_i$, the agent must play a sufficient number of tasks from $S_i$. 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 $\|B_{i+1}^	op B_i\|/\sqrt{n_i - r_i} \geq \kappa_1 \delta/\phi_{\text{min}}$ 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 $\theta, \theta' \in T$. Finally, there exists a constant $\ell = \Theta(r)$ such that any subsequence of length $\ell$ in $S_i$ (whose representation is $B_i \in \mathbb{R}^{d \times r_i}$) satisfies $\sigma_{\tau_i}(W_{s, i} W_{s, i}^\top) \geq \nu > 0$ for any $s$, where $W_{s, i} = [\theta_{\tau_{i-1}+s+1}, \ldots, \theta_{\tau_i+s+\ell}]$.

The first (resp. second) statement guarantees 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 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-
Non-Stationary Representation Learning in Sequential Multi-Armed Bandits

Algorithm 1 RepL algorithm

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

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
  $x_t = \lambda_0 \alpha_i, i = (t - 1 \mod \hat{r}) + 1$
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
  $x_t = \arg \max_{x \in A} x^\top \hat{\theta}$; end for

Algorithm 3 Outlier Detection algorithm (OD)

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

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 $O(Sd\sqrt{N})$, 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 $\tilde{O}(\sqrt{SN} + d\sqrt{S}N + Sn_{\text{red}} + Sn_{\text{tsd}})$. 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 $\sqrt{T}$ 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 $\hat{B}$, 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 taking $d$ independent actions $\lambda_0 \alpha_1, \ldots, \lambda_0 \alpha_d$. Here, $A = [a_1, \ldots, a_d]$ can be any orthonormal basis of the space $\mathbb{R}^d$, and the scalar $\lambda_0$ ensures that each action is in the action set $A$. After $N_1$ rounds, the agent estimates the task coefficient and takes the greedy action for $N_1$ rounds.

Let $\hat{W}_k = [\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_k]$. We perform singular value decomposition (SVD) of the matrix $\hat{W}_k^\top \hat{W}_k / k$, and let $B$ be the top $\hat{r}$ singular vectors of $\hat{W}_k^\top \hat{W}_k / k$. Note that $\hat{r}$ can be time-varying due to the non-stationary environment. To compute the current $B$, we ignore the singular vectors that are associated with small singular values (in theory, those that are much less than $\nu$ for $\nu$ 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
on every new bandit task by initially taking probing actions using the knowledge of the current representation (i.e., $B \in \mathbb{R}^{d \times T}$). If abnormal rewards are generated by these trials (the ones beyond the confidence interval $C_{\text{new}} = \{y_{\text{new}} \in \mathbb{R}^{\text{new}} : |Y_{\text{new}} - Y_{\text{new}}| \leq \delta_{\text{new}}\}$), 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 $y_{t_0} = \frac{1}{t_0} \sum_{i=1}^{t_0} y_i$. Then, we monitor rewards in a moving window of length $n_{\text{new}}$, i.e., $Y_{t_0+1}, \ldots, Y_{t_0+n_{\text{new}}}$.

Likewise, the average reward in the time window $Y_{n_{\text{new}}} = \frac{1}{n_{\text{new}}} \sum_{i=1}^{n_{\text{new}}} y_i$ belongs to the interval $[g(\theta) \parallel \theta - \xi_2, g(\theta) \parallel \theta + \xi_2]$ with high probability. Therefore, we can construct a confidence interval for the reward as $C_{\text{new}}(t_0) = [y_{t_0} - \xi_1 - \xi_2, y_{t_0} + \xi_1 + \xi_2]$ 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 $\kappa_2$ in Assumption 2 is sufficiently large and one selects large $\xi_1$, $\xi_2$, 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 $L_{\text{RL}}$ bandit tasks (we set $L_{\text{RL}} = \ell$ with $\ell$ 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 $\theta_i$’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., $B_1, B_2, \ldots, B_n$, appear in sequence, where each $B_i \in \mathbb{R}^{d \times r}$, is shared by $D_i$ consecutive bandit tasks. Then, the regret of ORLT satisfies the following upper bound

\[ R_T = \tilde{O}(S \sqrt{N} + \sum_{i=1}^{n_c} D_i \sqrt{N} + S n_{\text{trd}} + S n_{\text{new}}), \]

where $n_{\text{trd}}$ is the number of trial actions for outlier detection, $n_{\text{new}}$ is the number of rounds for task switch detection, $\kappa_1$ and $\kappa_2$ 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 $\Omega(d \sum_{i=1}^{n_c} \sqrt{i \cdot D_i \cdot N} + S \sqrt{N})$. 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 $\{\theta_1, \ldots, \theta_3\}$ 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 $B_1 \in \mathbb{R}^{d \times 2}, B_2 \in \mathbb{R}^{d \times 4}, B_3 \in \mathbb{R}^{d \times 4}, B_4 \in \mathbb{R}^{d \times 4}, B_5 \in \mathbb{R}^{d \times 2}, B_6 \in \mathbb{R}^{d \times 2}$ (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
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


**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 $\theta_s$. 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...
The following lemma will be utilized throughout this Appendix.

**Lemma 1 (Vershynin, 2018)** Let $X_1, X_2, \ldots, X_n$ be $n$ independent random variables such that $X_i$ is $\delta$-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 $\theta_1, \theta_2, \ldots, \theta_k$ drawn from $\mathcal{T}$, suppose $B \in \mathbb{R}^{d \times \hat{r}}$ is their representation. Let $\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_k$ be their respective estimate after playing each of them for $N$ rounds using the RepL algorithm. Suppose $X_i(B) \geq \nu > 0$. Then,

$$
\sin \theta(B, \hat{B}) \leq \hat{O} \left( \frac{\delta \log \frac{1}{\epsilon N \nu}}{\lambda_0 \nu} \right),
$$

(1.1)

with probability at least $1 - \frac{1}{\epsilon N \nu}$, where $\delta$ is the variance parameter of the sub-Gaussian noise $\eta_i$.

Notice that unlike the results in (Tripuraneni et al., 2020a; Yang et al., 2021), (1.1) has no obvious dependence on $\hat{r}$. That is because in those studies it is assumed that $\text{tr}(W_k W_k^\top) = \mu \hat{r} \nu$ where $\mu$ is the average condition number. However, in our case $\text{tr}(W_k W_k^\top) = \Theta(1)$ since $\phi_{\min}^2 \leq \text{tr}(W_k W_k^\top) \leq \phi_{\max}^2$. 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 $X_1, X_2, \ldots, X_k$ 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 $i = 1, 2, \ldots, k$. 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 $\theta_i$, the model is described by $y_i = x_i^\top \theta_i + \eta_i$. Pre-multiplying both sides with $x_i$ yields $x_i y_i = x_i x_i^\top \theta_i + x_i \eta_i$. In the RepL algorithm (see Fig. 1), $x_t$ periodically picks each column in the matrix $\lambda_0 [a_1, a_2, \ldots, a_d]$, with $[a_1, a_2, \ldots, a_d]$ forming an orthonormal basis of the space $\mathbb{R}^d$. Since $[a_1, a_2, \ldots, a_d]$ 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

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

Denote $X := [x_1, x_2, \ldots, x_{N_1}], Y := [y_1, \ldots, y_{N_1}]^\top$, and $\eta := [\eta_1, \eta_2, \ldots, \eta_{N_1}]^\top$. Note that $\hat{\theta}_i$ 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

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

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

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

Since $[a_1, a_2, \ldots, a_d]$ 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 $(XX^\top)^{-1} = \frac{d}{N_1 \lambda_0^2} I_d$. For the purpose of notational simplicity, denote $D := \frac{d}{N_1 \lambda_0^2} I_d$, then it follows that

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

and

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

(2.2)
Define a set of new variables \( z_i = \frac{1}{k} \theta_i \hat{\theta}_i^T - \frac{1}{k} (\theta_i \theta_i^T + D) \).

From (C.2), we have

\[
z_i = \frac{1}{k} (D A + D^2 C - D).
\]

Then, the expectation of \( z_i \) satisfies

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

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

For \( E A^2 \), it holds that

\[
E A^2 = E (\theta_i \eta_i^T X^T)^2 + E (X \eta_i^T)^2 \\
+ E (\theta_i \eta_i^T X^T X \eta_i^T) + E (X \eta_i^T \theta_i \eta_i^T X^T) \\
= 2 \frac{\lambda_0^2 \psi_2 N}{d} \theta_i \theta_i^T + N_1 \frac{\lambda_0^2 \psi_2 \theta_i \theta_i^T + \frac{\lambda_0^2 \psi_2 N_1}{d} I_d.
\]

For \( E C^2 \), we have

\[
E C^2 = E \left[ X \eta_i^T X^T X \eta_i^T X^T \right] \\
= E \left[ \sum_{i=1}^N N_1 \eta_i^T x_t x_t^T \right] \\
= \psi_4 \frac{\lambda_0^2 N_1}{d} x_t x_t^T \\
= \psi_4 \frac{\lambda_0^2 N_1}{d} x_t x_t^T
\]

where \( \psi_4 = \mathbb{E} \eta_i^T \psi_4 \) (\( \psi_4 \) always exists since \( \eta_i \) is a sub-Gaussian random variable).

For \( E (AC + CA) \), it holds that

\[
E (AC + CA) = E \left[ \left( \theta_i \eta_i^T X^T + X \eta_i^T \right) X \eta_i^T X^T \right] \\
+ E \left[ \left( \theta_i \eta_i^T X^T \right) \left( \theta_i \eta_i^T X^T + X \eta_i^T \right) \right] \\
= E \left[ \left( \eta_i^T X^T X \eta_i \cdot \theta_i \eta_i^T + X \eta_i^T \theta_i \eta_i^T X^T \right) \\
+ E \left[ \left( \eta_i^T X^T \theta_i \cdot X \eta_i^T \right) \right] \\
+ E \left[ \left( \eta_i^T X^T \eta_i \cdot \theta_i \eta_i^T \right) \right] \\
= \frac{\lambda_0^3 \psi_3 N}{d} \theta_i \delta \eta_i^T + \frac{\lambda_0^3 \psi_3 N}{d} \delta \eta_i^T + \frac{\lambda_3 \psi_3 N}{d} \eta_i^T \delta \eta_i^T + 2 \frac{\lambda_3 \psi_3 N}{d} \eta_i^T \delta \eta_i^T \\
= \frac{\lambda_0^3 \psi_3 N}{d} \theta_i \delta \eta_i^T + \frac{\lambda_0^3 \psi_3 N}{d} \delta \eta_i^T + 2 \frac{\lambda_3 \psi_3 N}{d} \eta_i^T \delta \eta_i^T
\]

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

\[
E C = E X \eta_i^T X^T = \frac{\lambda_0^2 \psi_2 N}{d} I_d.
\]

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

\[
E z_i^2 = \frac{1}{k^2} \frac{d^2}{\lambda_0^4 N_1^2} \left[ 2 \frac{\lambda_0^2 \psi_2 N_1}{d} \theta_i \theta_i^T + N_1 \frac{\lambda_0^2 \psi_2 \theta_i \theta_i^T + \frac{\lambda_0^2 \psi_2 N_1}{d} I_d + \frac{d}{\lambda_0^4 N_1^2} I_d \right] \\
+ \frac{d}{\lambda_0^4 N_1^2} (\theta_i \delta \eta_i^T + \frac{d}{\lambda_0^4 N_1^2} I_d) \\
\leq \frac{d^2 \psi_2}{k^2 \lambda_0^4 N_1^2} \left[ 2 + 1 \right] \theta_i \theta_i^T + O(\frac{d^2}{k^2 \lambda_0^4 N_1^2}) I_d.
\]

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

\[
\sigma^2 \leq \left\| \frac{d^2 \psi_2}{k^2 \lambda_0^4 N_1^2} \sum_{i=1}^k \theta_i \theta_i^T \right\|_F \leq O \left( \frac{d^2 \psi_2}{k^2 \lambda_0^4 N_1^2} \right).
\]

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

\[
\sigma^2 \leq O \left( \frac{d^2 \psi_2}{k^2 \lambda_0^4 N_1^2} \right).
\]

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

\[
\left\| \sum_{i=1}^k z_i \right\|_F \leq \frac{d^2 \psi_2}{k \lambda_0^4 N_1^2} \sqrt{\log (kd / \delta) + \log (d / \delta)}
\]

with probability at least \( 1 - \delta \). Let \( \delta = \frac{1}{k^2 N_1^2} \), then with probability \( 1 - \frac{1}{k^2 N_1^2} \), the following inequality holds

\[
\left\| \sum_{i=1}^k z_i \right\|_F \leq \frac{d^2 \psi_2}{k \lambda_0^4 N_1} \sqrt{\log (kdN_1) + \log (kd)}.
\]

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

\[
\| \hat{W}_k - W_k \|_F \leq \| \Delta \|_F \leq \frac{d^2 \psi_2}{k \lambda_0^4 N_1} \sqrt{\log (kdN_1) + \log (kdN_1)}.
\]
From the Davis-Kahan sin $\theta$ Theorem (Bhatia, 2013), we have
\[
\sin \theta(B, \hat{B}) \leq \frac{\|\hat{B}^\top (W_k - W'_k) B\|}{\omega} \leq \frac{\|\hat{W}_k - W'_k\|_F}{\omega},
\]
where $\omega = \inf_{1 \leq i \leq r, r \leq j \leq d} |\lambda_i(W'_k) - \lambda_j(\hat{W}_k)|$. From the Weyl's Theorem, $|\lambda_i(W'_k) - \lambda_j(\hat{W}_k)| \leq \|\hat{W}_k - W'_k\|_F = \left|\sum_{j=1}^k z_j\right|_F$ for any $i = 1, \ldots, d$. Since $\lambda_i(W'_k) = 0$ for all $i \geq r+1$, it holds that $|\lambda_i(W'_k)| \leq \|\Delta\|_F$ for all $i \geq r+1$.

Recall that $\sigma_r$ 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
\[
\sin \theta(U_1, \hat{U}_1) \leq \frac{\|\Delta\|_F}{\sigma_r - \|\Delta\|_F} \leq \frac{d \delta}{\lambda_0 \nu} \left(\sqrt{\frac{1}{kN_1}} \sqrt{\log dkN_1} + M \log dkN_1\right) \leq \frac{d \delta}{\lambda_0 \nu} \frac{1}{kN_1} \cdot \text{polylog}(d, N_1, k).
\]
The proof is complete.

**D. Supporting Results for RepT**

**Lemma 3** Given $\theta \in T$, suppose that there exists $B \in \mathbb{R}^{d \times \ell}$ with orthonormal columns such that $\theta = B\alpha$ for some $\alpha \in \mathbb{R}^d$. Assume that an estimate $\hat{B}$ is known by the learner and satisfies $\sin \theta(B, \hat{B}) \leq \varepsilon$. If the learner plays this bandit task $\theta$ 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 $\varepsilon$), 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 $y_i = x_i^\top B\alpha + \eta_i$. Instead of directly estimating $\theta$, we estimate $\alpha$. 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 $x_1$ repeatedly takes actions from $\lambda_0a_1', \ldots, \lambda_0a_{r'}$, it holds that $XX^\top = N_2 AA^\top$ with $A = [\lambda_0a_1', \ldots, \lambda_0a_{r'}]$. 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
\[
\hat{\alpha} = \left(\frac{N_2}{\hat{r}} \hat{B}^\top AA^\top \hat{B}\right)^{-1}\hat{B}^\top X(XX^\top B\alpha + \eta)
\]
\[
= \left(\frac{N_2}{\hat{r}} \hat{B}^\top AA^\top \hat{B}\right)^{-1} \frac{N_2}{\hat{r}} \hat{B}^\top AA^\top B\alpha
\]
\[
= \left(\frac{N_2}{\hat{r}} \hat{B}^\top AA^\top \hat{B}\right)^{-1} \hat{B}^\top X\eta.
\]
Therefore, $\hat{\theta} = \hat{B}\hat{\alpha}$ and $\theta = B\alpha$, it follows that
\[
\hat{B}\hat{\alpha} - B\alpha = \left(\frac{N_2}{\hat{r}} \hat{B}^\top AA^\top \hat{B}\right)^{-1} \left(\frac{N_2}{\hat{r}} \hat{B}^\top AA^\top B\alpha - B\alpha\right)
\]
\[
= \left(\frac{N_2}{\hat{r}} \hat{B}^\top AA^\top \hat{B}\right)^{-1} \hat{B}^\top X\eta.
\]

Then, it holds that $\mathbb{E}\left[\|\hat{\theta} - \theta\|^2\right] \leq \mathbb{E}\|s_1\|^2 + \mathbb{E}\|s_2\|^2$ since $\eta$ 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 $B = (\hat{B}B^\top + \hat{B}_1\hat{B}_1^\top)B$, therefore, it holds that
\[
s_1 = \hat{B}(\frac{N_2}{\hat{r}} \hat{B}^\top AA^\top \hat{B})^{-1} \frac{N_2}{\hat{r}} \hat{B}^\top AA^\top \hat{B} + \hat{B}_1\hat{B}_1^\top)B\alpha - B\alpha
\]
\[
= (\hat{B}B^\top B\alpha - B\alpha)
\]
\[
= \hat{B}(\hat{B}^\top AA^\top \hat{B})^{-1} \hat{B}^\top AA^\top \hat{B}_1\hat{B}_1^\top B\alpha
\]
\[
= \hat{B}_1\hat{B}_1^\top B\alpha + \hat{B}(\hat{B}^\top AA^\top \hat{B})^{-1} \hat{B}^\top AA^\top \hat{B}_1\hat{B}_1^\top B\alpha
\]
\[
= \hat{B}_1\hat{B}_1^\top B\alpha + \hat{B}_1\hat{B}_1^\top B\alpha
\]
\[
= \hat{B}_1\hat{B}_1^\top B\alpha + \hat{B}^\top AA^\top \hat{B}_1\hat{B}_1^\top B\alpha
\]
Then, $\mathbb{E}\|s_1\|^2$ satisfies
\[
\mathbb{E}\|s_1\|^2 \leq 2 \left\|\hat{B}_1\hat{B}_1^\top B\alpha\right\|^2_F
\]
\[
+ 2 \left\|\hat{B}(\hat{B}^\top AA^\top \hat{B})^{-1} \hat{B}^\top AA^\top \hat{B}_1\hat{B}_1^\top B\alpha\right\|^2_F
\]
\[
\leq 2\phi_1^2\|\hat{B}_1\hat{B}_1^\top\|^2_F
\]
\[
+ 2\phi_1^2 \left\|(\hat{B}^\top AA^\top \hat{B})^{-1} \hat{B}^\top AA^\top \hat{B}_1\hat{B}_1^\top\right\|^2_F
\]
\[
\leq \|\hat{B}_1\hat{B}_1^\top\|^2_F.
\]
where $\theta$ is the variance proxy parameter of the noise $\eta_t$, 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 $p_1 = 1 - \exp(-\frac{c_1\kappa_1 \delta^2 n_{\text{rad}}}{4})$ for some $c_1 > 0$. It can be observed that the probability increases as $\kappa_1$ or $n_{\text{rad}}$ 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 $1 - \exp(-c\xi^2m)$, 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 $z = B\theta \in \mathbb{R}^{(d-r) \times 1}$. 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 $1 - \exp(-c\xi^2m)$, we have

$$(1 - \xi)\sqrt{\frac{m}{d-r}}\|z\|_2 \leq \|Pz\|_2 \leq (1 + \xi)\sqrt{\frac{m}{d-r}}\|z\|_2.$$
variable. Then, for any $\xi > 0$ it holds that
\[
\Pr\left[\|X\|_2 - \sqrt{n}\delta \geq \xi\right] \leq 2 \exp\left(-\frac{c\xi^2}{K^4}\right), \quad (E.1)
\]
where $c$ is an absolute constant and $K = \max_i \|X_i\|_2$.

We are now ready to prove Lemma 4.

**Proof of Lemma 4** Recall that the agent takes $n_{\text{rd}}$ probing actions as in Fig. 3, and denote the corresponding rewards as in $Y_{n_{\text{rd}}} := [y_1, \ldots, y_{n_{\text{rd}}}]^T$. Following Lemma 7, we build an confidence interval for $Y_{n_{\text{rd}}}$
\[
C_{n_{\text{rd}}} = Y_{n_{\text{rd}}} \in \mathbb{R}^{n_{\text{rd}}} : \|Y_{n_{\text{rd}}} - \mu_n\|_2 \leq \gamma_1 \sqrt{n_{\text{rd}}}
\]
From Lemma 7, the probability of $y_t \notin C_{n_{\text{rd}}}$ is less than $2 \exp\left(-\frac{c\gamma^2 n_{\text{rd}}}{K^4}\right)$ for some absolute constants $c$ and $K$.

If $\|B^T_{\perp}\| = \rho$, from Lemma 6 we have $\|P_T B^T_{\perp}\| = \|PP^T B^T_{\perp}\| \geq \frac{1}{2} \rho \sqrt{\frac{n_{\text{rd}}}{d-r}}$ with probability at least $1 - \exp(-\frac{c}{c} n_{\text{rd}})$ for some constant $c$ (since $PP^T B^T_{\perp}$ can be taken as projecting $B^T_{\perp}$ onto the random space spanned by $P_T$). From the reward generating function $Y_{n_{\text{rd}}} = \lambda_0 P_T Y_{n_{\text{rd}}} + \eta = \eta$, one can derive that to ensure that $Y_{n_{\text{rd}}} \in C_{n_{\text{rd}}}$, it should hold that
\[
\|\eta\| \geq \frac{1}{2} \rho \sqrt{\frac{n_{\text{rd}}}{d-r}} \geq \frac{1}{4} \sqrt{\gamma_1 + 1)\sqrt{n_{\text{rd}}}.} \quad (E.2)
\]
From Assumption 2, we know that $\rho \geq \kappa_1 \sqrt{\gamma_1 + 1}$ from Lemma 7, the inequality (E.2) holds with probability less than
\[
2 \exp\left(-\frac{c((\kappa_1 - 4(1 + 1))2\delta^2 n_{\text{rd}}}{4K^4}\right).
\]
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 $x_i = g(\hat{\theta}) = \arg\max_{x \in \mathcal{X}} x^\top \hat{\theta}$ is deterministic, one obtain that $y_i$ is sub-Gaussian with 0 mean. Then, it follows from Lemma 1 that
\[
\Pr[\bar{y}_{t_0} - \mathbb{E}y_{t_0} \geq \xi_1] = \Pr[\bar{y}_{t_0} - g(\hat{\theta})^\top \theta \geq \xi_1]
\leq \exp\left(-\frac{t_0 \xi_1^2}{2\delta^2}\right), \quad (F.1a)
\]
\[
\Pr[\bar{y}_{t_0} - \mathbb{E}y_{t_0} \leq -\xi_1] = \Pr[\bar{y}_{t_0} - g(\hat{\theta})^\top \theta \leq -\xi_1]
\leq \exp\left(-\frac{t_0 \xi_1^2}{2\delta^2}\right), \quad (F.1b)
\]
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 $1 - 2 \exp\left(-\frac{c\xi_1^2}{2\delta^2}\right)$ for any $\xi_1 \geq 0$.

Recall that the average reward in the moving window of width $n_{\text{swd}}$, $y_{t_0+1}, \ldots, y_{t_0+n_{\text{swd}}}$, is $Y_{n_{\text{swd}}} = \frac{1}{n_{\text{swd}}} \sum_{t=1}^{n_{\text{swd}}} y_{t_0+t}$. Likewise, it follows from Lemma 1 that
\[
\Pr\left[Y_{n_{\text{swd}}} \leq g(\hat{\theta})^\top \theta - \xi_2\right] \leq \exp\left(-\frac{-t_{0} \xi_2^2}{2\delta^2}\right), \quad (F.2a)
\]
\[
\Pr\left[Y_{n_{\text{swd}}} \geq g(\hat{\theta})^\top \theta + \xi_2\right] \leq \exp\left(-\frac{-t_{0} \xi_2^2}{2\delta^2}\right). \quad (F.2b)
\]

for any $\xi_2 \geq 0$.

Combining the above inequalities (F.1) and (F.2), we have
\[
\Pr[Y_{n_{\text{swd}}} \notin \bar{y}_{t_0} - \xi_2, \bar{y}_{t_0} + \xi_2] \leq \exp\left(-\frac{t_{0} \xi_1^2}{2\delta^2} + \frac{-t_{0} \xi_2^2}{2\delta^2}\right),
\]
\[
\Pr[Y_{n_{\text{swd}}} \geq y_{t_0} + \xi_1, y_{t_0} + \xi_2] \leq \exp\left(-\frac{t_{0} \xi_1^2}{2\delta^2} + \frac{-t_{0} \xi_2^2}{2\delta^2}\right).
\]

In other words, if we define a confidence interval based on the observed average reward $\bar{y}_{t_0}$ as follows
\[
C_{\text{tsd}}(t_0) = [\bar{y}_{t_0} - \xi_1, \bar{y}_{t_0} + \xi_1, \bar{y}_{t_0} + \xi_2],
\]
we have
\[
\Pr[Y_{n_{\text{swd}}} \notin C_{\text{tsd}}(t_0)] \leq 1 - 2 \exp\left(-\frac{t_{0} \xi_1^2}{2\delta^2} + \frac{-t_{0} \xi_2^2}{2\delta^2}\right). \quad (F.3)
\]
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
\[
\Pr[Y_{n_{\text{swd}}} \notin C_{\text{tsd}}(t_0)] \leq 1 - 2 \exp\left(-\frac{-t_{0} \xi_2^2}{2\delta^2}\right). \quad (F.3)
\]
The choice of $\xi_2$ and the length of observation window $n_{\text{swd}}$ 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 $g(\theta) = \arg\max_{x \in \mathcal{X}} x^\top \theta$. Suppose that Assumption 2 is satisfied, and let $\xi_2 = \mu_2 \kappa_0^2$ with $\mu_2 \leq 1$. Then, all the tasks can be detected with probability at least $1 - O(\exp(-c_2^2 n_{\text{swd}}))$ for some positive constant $c_2 > 0$. 

Proof Suppose the underlying task switches from $\theta$ to $\theta'$ 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 $\tilde{y}_{t_0}$ is computed by

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

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

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

Subsequently, one has

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

From Assumption 2, we know that $|g(\hat{\theta})^\top (\theta' - \theta)| \geq \kappa_2 \delta$ for any $\theta, \theta' \in \mathcal{T}$. To ensure $|Y_{n_{\text{rad}}} - \tilde{y}_{t_0}| \leq \xi_2 = \mu_2 \kappa_2 \delta$, given large $t_0$, it needs to hold that

$$\frac{1}{n_{\text{rad}}} \sum_{t=t_0+1}^{t_0+n_{\text{rad}}} \eta_t \geq \mu_2 \kappa_2 \delta.$$ 

From Lemma 1, we deduce that

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

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 $1 - O(\exp(-\frac{n_{\text{rad}} \mu_2^2 \kappa_2^2}{2}))$ for some positive constant $c_2$, which completes the proof.

G. Proof of Theorem 1

The ORL algorithm operates in a cyclic manner. At the $c$-th cycle, $L_{\text{RL}}$ 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

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

where $\delta$ is the variance of the sub-Gaussian noise $\eta_t$, depending on the variance proxy parameter $\delta$. From (Rusmevichientong and Tsitsiklis, 2010), it holds that $\max_{x \in A} x^\top \theta - \max_{x \in A} x^\top \hat{\theta} \leq J \|\theta - \hat{\theta}\|^2 |\theta|$, where $J$ is a constant that exists since the action set $A$ 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

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

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

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

Therefore, the total regret in $N$ steps satisfies

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

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

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

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 $\theta_1, \theta_2, \ldots, \theta_D$. Assume that there is a linear feature extractor matrix $B_i \in \mathbb{R}^{d \times r}$, such that for any $\theta_s$ in this sequence $\theta_s = B_i \alpha_s$ for some $\alpha_s \in \mathbb{R}^r$. 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 $\theta_1, \theta_2, \ldots, \theta_{L_{\text{RL}}}$; at the end of RepL phase,
Then, the regret of playing these \( D_i \) sequential bandits satisfies

\[
\mathbb{E} R_{ND_i} \leq \tilde{O}(d r_i \sqrt{D_i N} + D_i r_i \sqrt{N} + \frac{d}{r_i} \sqrt{D_i N}).
\]

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

\[
\sin \theta(B_i, B_i) \leq \tilde{O}\left(\frac{d \delta}{\lambda_0 \nu} \sqrt{\frac{1}{m L_{RL} d \sqrt{N}}}\right).
\]

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

\[
\mathbb{E} R_{\text{RepT}}(m) \lesssim m r_i \sqrt{N} + m N \frac{d^2 \delta^2}{\lambda_0^2 \nu^2} \frac{1}{m L_{RL} d \sqrt{N}}
\]

\[
= \tilde{O}(m r_i \sqrt{N} + \frac{d}{L_{RL} \sqrt{N}}).
\]

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

\[
\mathbb{E} R_{ND_i} \lesssim \tilde{L} L_{RL} d \sqrt{N} + \sum_{m=1}^{\tilde{L}} \left( m r_i \sqrt{N} + \frac{d}{L_{RL} \sqrt{N}} \right)
\]

\[
\leq \tilde{L} L_{RL} d \sqrt{N} + D_i r_i \sqrt{N} + \frac{d}{L_{RL} \sqrt{N}}.
\]

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

\[
\mathbb{E} R_{ND_i} = \sum_{m=1}^{\tilde{L}} R_{\text{RepL}}(m) + R_{\text{RepT}}(m)
\]

\[
= \tilde{O}(d r_i \sqrt{D_i N} + D_i r_i \sqrt{N} + \frac{d}{r_i} \sqrt{D_i N}),
\]

which completes the proof. \( \Box \)

We are now ready to prove Theorem 1.

**Proof of Theorem 1** Combining 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 \( 1 - O(\max\{\exp(-c_1 n_{\text{tsd}} \kappa_2^2 \delta^2), \exp(-c_2 n_{\text{tsd}} \kappa_2^2)\}) \) for some positive constants \( c_1 \) and \( c_2 \). This probability is large is \( \kappa_1 \) and \( \kappa_2 \) is large. Since \( n_{\text{tsd}} \) rounds are spent on detecting every new task and \( n_{\text{rsd}} \) probing rounds are spent for every new task on detecting representation switches, the regret, denoted as \( R_{\text{detec}} \), satisfies \( R_{\text{detec}} \lesssim S n_{\text{tsd}} + S n_{\text{rsd}} \).

By Assumption 1, there are \( n_c \) representations, described by \( B_1, B_2, \ldots, B_{n_c} \), 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

\[
\mathbb{E} R_T \lesssim \sum_{i=1}^{n_c} \mathbb{E} R_{ND_i}
\]

\[
\lesssim \sum_{i=1}^{n_c} \left( d r_i \sqrt{D_i N} + D_i r_i \sqrt{N} + \frac{d}{r_i} \sqrt{D_i N} \right)
\]

\[
+ S n_{\text{tsd}} + S n_{\text{rsd}}
\]

\[
\lesssim \left( \sum_{i=1}^{n_c} (d r_i \sqrt{D_i N} + D_i r_i \sqrt{N}) \right) + S n_{\text{tsd}} + S n_{\text{rsd}}
\]

\[
\lesssim \sum_{i=1}^{n_c} d r_i \sqrt{D_i N} + S r \sqrt{N} + S n_{\text{tsd}} + S n_{\text{rsd}},
\]

where the last inequality has used the fact that \( r \geq r_i \) for any \( i \). In other words, \( \mathbb{E} R_T = \tilde{O}(\sum_{i=1}^{n_c} d r_i \sqrt{D_i N} + S r \sqrt{N} + S n_{\text{tsd}} + S n_{\text{rsd}}) \), which completes the proof. \( \Box \)

**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 \( S = \{\theta_1, \theta_2, \ldots, \theta_S\} \), where \( \theta_i \in \mathbb{R}^d \) with \( d = 20 \). Each
bandit is played for 1000 rounds, then a subsequent task from $S$ 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, and 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 $A_t$ is a unit ball, i.e., $A_t := \{x \in \mathbb{R}^d : \|x\|_2 \leq 1\}$ and the noise $\eta_t$ in the reward generating function $y_t = x_t^\top + \eta_t$ is Gaussian with zero mean and 0.1 standard deviation.

First, we assume that there is $B_t \in \mathbb{R}^{d \times 2}$ such that $\theta_t = B_t \alpha_t$ 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 $S$ 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.

We model the WCST as a bandit problem. Specifically, each stimulus card is modeled by a $4 \times 3$ 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

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

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 \(x^*_t = A_t B_t = [0, 1, 0] \). 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**


