Feature and Parameter Selection in Stochastic Linear Bandits

Ahmadreza Moradipari [1] **Berkay Turan** [1] **Yasin Abbasi-Yadkori** [2] **Mahnoosh Alizadeh** [1]
Mohammad Ghavamzadeh [3]

Abstract

We study two model selection settings in stochastic linear bandits (LB). In the first setting, which we refer to as *feature selection*, the expected reward of the LB problem is in the linear span of at least one of M feature maps (models). In the second setting, the reward parameter of the LB problem is arbitrarily selected from M models represented as (possibly) overlapping balls in $\mathbb{R}^d$. However, the agent only has access to misspecified models, i.e., estimates of the centers and radii of the balls. We refer to this setting as *parameter selection*. For each setting, we develop and analyze a computationally efficient algorithm that is based on a reduction from bandits to full-information problems. This allows us to obtain regret bounds that are not worse (up to a $\sqrt{\log M}$ factor) than the case where the true model is known. This is the best reported dependence on the number of models M in these settings. Finally, we empirically show the effectiveness of our algorithms using synthetic and real-world experiments.

1. Introduction

Learning under bandit feedback is a class of online learning problems in which an agent interacts with the environment through a set of actions (arms), and receives rewards only from the arms that it has pulled. The goal of the agent is to maximize its expected cumulative reward without knowledge of the reward distributions of the arms. *Multi-armed bandit* (MAB) is the simplest form of this problem (Lai & Robbins, 1985; Auer et al., 2002a; Lattimore & Szepesvari, 2020; Moradipari et al., 2018). *Linear bandit* (Dani et al., 2008; Rusmevichientong & Tsitsiklis, 2010; Abbasi-Yadkori et al., 2011) is a generalization of MAB to (possibly) infinitely many arms, each associated with a feature vector. The mean reward of each arm is assumed to be the dot product of its feature vector and an *unknown* parameter vector. This setting contains *contextual linear bandit* in which action sets and feature vectors change at every round. The main component of bandit algorithms is to balance *exploration* and *exploitation*: to decide when to *explore* and learn about the arms, and when to *exploit* and select the action with the highest estimated reward. The most common exploration strategies are *optimism in the face of uncertainty* (OFU) or upper confidence bound (UCB) (Auer et al., 2002a; Dani et al., 2008; Abbasi-Yadkori et al., 2011; Moradipari et al., 2020b; 2022a), and Thompson sampling (TS) (Thompson, 1933; Agrawal & Goyal, 2013; Russo & Van Roy, 2014; Abeille et al., 2017; Moradipari et al., 2020a; 2021).

In this paper, we study *model selection* in stochastic linear bandits (LB), where the LB problem at hand is selected from a set of M models. The agent has information about the models but does not know the identity of the one(s) that the new LB problem has been selected from. The goal of the agent is to identify the true model(s) and transfer its (their) collected experience to speedup the learning of the task at hand. It is a common scenario in many application domains that the new task belongs to a family of models that are either known accurately or with misspecification. For example, it is reasonable to assume that the customers of an online marketing website, the users of an app, or the patients in a medical trial belong to a certain number of categories based on their shopping and browsing habits or their genetic signatures. It is also common these days that websites, apps, and clinics have a large amount of information from each of these categories that can be used to build a model.

Model selection is particularly challenging with bandit information. A common approach is to consider each model as a black-box that runs a bandit algorithm with its own information, and then a meta algorithm plays a form of bandit-over-bandits strategy with their outcomes. These algorithms often achieve a regret of $\widetilde{\mathcal{O}}(\sqrt{MT})$, and thus, are not desirable when the number of models M is large. In this paper, we consider two bandit model selection settings and show that it is possible to improve this rate so that the regret scales as $\sqrt{\log M}$ with the number of models. The main

[1]Department of Electrical and Computer Engineering, University of California, Santa Barbara, USA [2]DeepMind, London, UK [3]Google Research, Mountain View, USA. Correspondence to: Ahamdreza Moradipari <ahmadreza_moradipari@ucsb.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

innovation in our algorithms is utilizing reductions from bandits to full-information problems, and performing model selection in the full-information setting for which much stronger results exist. The main reason for $\widetilde{\mathcal{O}}(\sqrt{MT})$ regret in bandit-over-bandits algorithms is that no information is shared among the models (bandit algorithms), i.e., when a bandit algorithm is used to take an action in a round, the resulting feedback is not shared with the other models. On the other hand, model selection in the full-information setting allows the model to share information among each other, which makes the superior $\sqrt{\log M}$ regret bound possible.

The two model selection settings we consider in this paper are: *feature selection*, where the mean reward of the LB problem is in the linear span of at least one of M given feature maps (models), and *parameter selection*, where the reward parameter of the LB problem is arbitrarily selected from M models represented as (possibly) overlapping balls in $\mathbb{R}^d$. Here the models can be misspecified, i.e., only estimates of the centers and radii of the balls are given to the algorithm. We derive algorithms in these settings that use reductions from bandits to full-information. Our algorithms are computationally efficient and have regret bounds that are not worse (up to a $\sqrt{\log M}$ factor) than the case where the true model is known. We achieve this by properly instantiating existing algorithmic paradigms: SquareCB (Foster & Rakhlin, 2020) and OFUL (Abbasi-Yadkori et al., 2011). The SquareCB algorithm in its original form uses a set of static experts, but we need adaptive (learning) experts in order to have a computational efficient algorithm with the desired regret in our *feature selection* setting. Working with adaptive (time-varying) experts requires appropriate and non-trivial modifications to the proof of SquareCB.

There are mainly two types of reductions from bandits to full-information problems. The first one is the classical reduction that uses importance weighted estimates. A popular algorithm in this class is EXP3 that uses Exponentially Weighted Average forecaster as the full-information algorithm. The bandit model selection strategy of Agarwal et al. (2017), known as CORRAL, also uses this type of reduction with an online mirror descent method and a carefully selected mirror map as the full-information algorithm. Given that importance weighted estimates are fed to the full-information algorithm, a $\sqrt{M}$ term is in general unavoidable in the regret of the methods that use this type of reduction. In this work, we use a different type of full-information reduction introduced by Foster & Rakhlin (2020) and Abbasi-Yadkori et al. (2012). Here, the full-information algorithm has direct access to its losses without any importance weighted estimates, and thus, allows us to obtain regrets that scales as $\sqrt{\log M}$.

2. Problem Formulation

In this section, we first provide a brief overview of stochastic linear bandits. We then describe the two model selection settings studied in the paper. We conclude by introducing a regression oracle used by our algorithms that is based on sequential prediction with expert advice and square loss.

2.1. Stochastic Linear Bandits

A stochastic linear bandit (LB) problem is defined by a sequence of T interactions of a learning agent with a stochastic environment. At each round $t \in [T]$, the agent is given a decision set $\mathcal{A}_t \subset \mathbb{R}^d$ from which it has to select an action a_t. Upon taking the action $a_t \in \mathcal{A}_t$, it observes a reward $y_t = \langle \phi_t(a_t), \theta_* \rangle + \eta_t$, where $\theta_* \in \mathbb{R}^d$ is the unknown reward parameter, $\phi_t(a) \in \mathbb{R}^d$ is the feature vector of action a at round t, and η_t is a zero-mean R-sub-Gaussian noise. When the features correspond to the canonical basis, this formulation reduces to *multi-armed bandit*. In case the features depend on both an action $a \in \mathcal{A}$ and a context $x \in \mathcal{X}$, i.e., $\phi_t(a_t) = \phi(x_t, a_t)$, this LB formulation is called *contextual linear bandit*. It is also common in practice that the action set is fixed and finite, i.e., $\mathcal{A} = [K]$, in which case we are in the finite K-action setting. The history H_t of a LB algorithm up to round t consists of all the contexts, actions, and rewards that it has observed from the beginning until the end of round $t-1$, i.e., $H_t = \{(x_s, a_s, y_s)\}_{s=1}^{t-1}$, or equivalently $H_t = \{(\phi_s(a_s), y_s)\}_{s=1}^{t-1}$.

The goal of the agent in LB is to maximize its expected cumulative reward in T rounds, or equivalently to minimize its T-round (pseudo) regret, i.e.,

$$\mathcal{R}(T, \theta_*) = \sum_{t=1}^{T} \langle \phi_t(a_t^*), \theta_* \rangle - \langle \phi_t(a_t), \theta_* \rangle, \tag{1}$$

where $a_t^* = \operatorname{argmax}_{a \in \mathcal{A}_t} \langle \phi_t(a), \theta_* \rangle$ is the optimal action in round t.

2.2. Feature Selection Setting

In this setting, the agent is given a set of M feature maps $\{\phi^i\}_{i=1}^M$ with dimension d. We assume that the expected reward of the LB problem belongs to the linear span of at least one of these M models (features), i.e., there exists an $i \in [M]$ and a $\theta_*^i \in \mathbb{R}^d$, such that for all rounds $t \in [T]$, contexts $x \in \mathcal{X}$, and actions $a \in \mathcal{A}$, we may write the mean rewards as $\mathbb{E}[y_t] = \langle \phi^i(x, a), \theta_*^i \rangle$.[1] We refer to such feature maps as *true* models and denote them by i_*. Note that the agent does not know the identity of the true model(s) i_*.

As a motivational example for this setting, we can consider a recommender system that has trained M models (e.g., M neural networks) to predict the score of customer-item pairs.

[1] Note that we use the contextual linear bandit notation for this setting and in the corresponding sections.

Each model corresponds to a particular mood or type of the customer, or any other latent component of the customer's state. Each model provides an embedding for customer-item pairs and the score is linear in this embedding (think of an embedding as the one to the last layer of a trained NN). When a new customer arrives, the recommender system should find out as soon as possible which of the M models (embeddings) is the best match to the current mood/type of this customer in order to recommend her desirable items.

We make the following standard assumption on the boundedness of the reward parameters and features of the M models.

Assumption 2.1. There are constants $L, S, G \geq 0$, such that for all $i \in [M]$, $t \in [T]$, $x \in \mathcal{X}$, and $a \in \mathcal{A}$, we have $\|\theta_*^i\| \leq S$, $\|\phi^i(x,a)\| \leq L$, and $|\langle \phi^i(x,a), \theta_*^i \rangle| \leq G$.

Our goal here is to design an algorithm that minimizes *transfer regret*, which in this setting we define it as

$$\mathcal{R}(T) = \sum_{t=1}^{T} \langle \phi^{i_*}(x_t, a_t^*), \theta_*^{i_*} \rangle - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle, \quad (2)$$

where $a_t^* = \operatorname{argmax}_{a \in \mathcal{A}} \langle \phi^{i_*}(x_t, a), \theta_*^{i_*} \rangle$. In the results we report for this setting in Section 3, we make two assumptions: **1)** the feature maps are all known (no model misspecification), and **2)** the number of actions is finite, i.e., we are in the finite K-action setting described in Section 2.1. However, we believe that our algorithm and analysis can be extended to the case of having misspecified models and convex action sets using the results in Foster et al. (2020).

2.3. Parameter Selection Setting

In this setting, unlike the classical setting in Section 2.1, we no longer assume that the unknown parameter θ_* can be any vector in $\mathbb{R}^d$. Rather, θ_* can be generated from M possible reward models, each defined as a ball $B(\mu_i, b_i) = \{\theta \in \mathbb{R}^d : \|\theta - \mu_i\| \leq b_i\}$, with center $\mu_i \in \mathbb{R}^d$ and radius $b_i \geq 0$. Note that the models (balls) may overlap and do not have to be disjoint. The M models can be thought of the responses of M types (or clusters) of customers to different items in a recommender system or the reactions of patients with M genotypes to a set of drugs. The radii $\{b_i\}_{i=1}^M$ represent the variation within each cluster. The reward parameter θ_* of the new task (LB problem) is arbitrarily selected from the M models. For example, it can be adversarially selected from the union of the models, i.e., $\theta_* \in \bigcup_{i=1}^M B(\mu_i, b_i)$. In this case, we denote by $\mathcal{I}_*$, the set of indices of the balls that contain θ_*. Since the models are often computed from (finite) historical data, it is reasonable to assume that only *estimates* of their centers $\{\widehat{\mu}_i\}_{i=1}^M$ are available, together with upper-bounds on the error of these estimates $\{c_i\}_{i=1}^M$, such that $\|\mu_i - \widehat{\mu}_i\| \leq c_i$, for all $i \in [M]$.

The agent has no knowledge either about θ_* or the process according to which it has been selected. The only information given to the agent are: **1)** estimates $\widehat{\mu}_i$ of the center of the models, **2)** upper-bounds c_i on the errors of these estimates, and **3)** the exact radii b_i of the models, for all $i \in [M]$. This means that although θ_* is selected from the *actual* models $B(\mu_i, b_i)$, the agent has only access to *estimated* models $B(\widehat{\mu}_i, b_i + c_i)$ that have more uncertainty (their corresponding balls are larger). For simplicity, we assume that the exact values of radii $\{b_i\}_{i=1}^M$ are known. However, our results can be easily extended to the case that instead of b_i's, their estimates $\widehat{b}_i$ and upper-bounds on their errors c_i', i.e., $\|b_i - \widehat{b}_i\| \leq c_i'$, for all $i \in [M]$, are given to the agent. In this case, the agent has to use even more uncertain estimates of the models $B(\widehat{\mu}_i, b_i + c_i + c_i')$.

Our goal is to design an algorithm that can transfer knowledge from these estimated models and learn the new task with parameter θ_* more efficiently than when it is independently learned. This goal can be quantitatively stated as minimizing the *transfer regret*,

$$\mathcal{R}(T) = \sup_{\theta_* \in \bigcup_{i=1}^M B(\mu_i, b_i)} \mathcal{R}(T, \theta_*), \quad (3)$$

where $\mathcal{R}(T, \theta_*)$ is the regret defined by (1). We make the following standard assumption on the boundedness of the features and expected rewards.

Assumption 2.2. There exist constants $L, G \geq 0$, such that $\forall t \in [T]$ and $\forall a \in \bigcup_{t=1}^T \mathcal{A}_t$, we have $\|\phi_t(a)\| \leq L$, and $\forall \theta \in \bigcup_{i=1}^M B(\mu_i, b_i)$, we have $|\langle \phi_t(a), \theta \rangle| \leq G$.

2.4. Regression Oracle

In both model selection settings studied in the paper, our proposed algorithms use a regression oracle that is based on sequential prediction with expert advice and square loss. Following Foster & Rakhlin (2020) and Foster et al. (2020), we refer to this regression oracle as `SqAlg`. We can consider `SqAlg` as a meta algorithm that consists of M learning algorithms (or experts), each corresponding to one of our M models, and returns a prediction by aggregating the predictions of its experts. More precisely, in each round $t \in [T]$, `SqAlg` takes the current context-action pair (x_t, a_t), or equivalently $\phi_t(a_t)$, as input, and gives them to its M experts to predict their reward, i.e., $f_t^i(H_t) = f^i(\phi_t(a_t); H_t)$, $\forall i \in [M]$, given the current history H_t. Then, the meta algorithm `SqAlg` aggregates its experts' predictions, $\{f_t^i(H_t)\}_{i=1}^M$, given their current weights, and returns its own prediction $\widehat{y}_t = \texttt{SqAlg}_t(\phi_t(a_t); H_t)$. Upon observing the actual reward y_t, `SqAlg` updates the weights of its experts according to the difference between their predictions $f^i(\phi_t(a_t); H_t)$ and the actual reward y_t.

The regression oracles (`SqAlg`) used by our model selection algorithms differ in the prediction algorithm used by their experts. However, in both cases, `SqAlg` aggregates its experts' predictions using an algorithm by Haussler et al. (1998) (see Algorithm 3 in Appendix A). The performance of `SqAlg` is evaluated in terms of its regret $\mathcal{R}_{\texttt{Sq}}(T)$, which

is defined as its accuracy (in terms of square loss) w.r.t. the accuracy of the best expert in the set, i.e.,

$$\sum_{t=1}^{T}(\widehat{y}_t - y_t)^2 - \min_{i\in[M]}\sum_{t=1}^{T}(f_t^i(H_t) - y_t)^2 \le \mathcal{R}_{\text{Sq}}(T). \quad (4)$$

In each round t, we define the oracle prediction for a context x and an action a as $\widehat{y}_t(x,a) := \texttt{SqAlg}_t(x,a;H_t)$. As shown in Haussler et al. (1998), in case all observations and experts' predictions are bounded in an interval of size ℓ, this regret can be bounded as $\mathcal{R}_{\text{Sq}}(T) \le \ell^2 \log M$ (see Appendix A for more details). We use this regret bound in the analysis of our proposed algorithms.

3. Feature Selection Algorithm

In this section, we derive an algorithm for the feature selection setting described in Section 2.2 that is based on the SquareCB algorithm (Foster & Rakhlin, 2020). We refer to our algorithm as *feature selection SquareCB* (FS-SCB). We prove an upper-bound on the transfer regret of FS-SCB in Section 3.1, and provide an overview of the related work and a discussion on our results in Section 3.2.

Algorithm 1 contains the pseudo-code of FS-SCB. In each round $t \in [T]$, the algorithm observes a context $x_t \in \mathcal{X}$ and passes it to its regression oracle SqAlg to produce its reward predictions $\widehat{y}_t(x_t,a), \forall a \in [K]$. Each expert in SqAlg corresponds to one of the M models and is a *ridge regression* algorithm with the feature map of that model. Expert $i \in [M]$ predicts the reward of the context x_t, for each action $a \in [K]$, as $f^i(x_t,a;H_t) = \langle \phi^i(x_t,a), \widehat{\theta}_t^i \rangle$, where $\widehat{\theta}_t^i = \text{argmin}_\theta \|\Phi_t^{i\top}\theta - Y_t\|^2 + \lambda_i\|\theta\|^2$. We may write $\widehat{\theta}_t^i$ in closed-form as $\widehat{\theta}_t^i = (V_t^{\lambda_i})^{-1}\Phi_t^{i\top}Y_t$. In these equations, $Y_t = (y_1,\dots,y_{t-1})^\top$ is the reward vector; Φ_t^i is the feature matrix of the i^{th} model, whose rows are $\phi^i(x_1,a_1),\dots,\phi^i(x_{t-1},a_{t-1})$; λ_i is the regularization parameter of model i, which our analysis shows that it only needs to be larger than one, i.e., $\lambda_i \ge 1$; and finally $V_t^{\lambda_i} = \lambda_i I + \Phi_t^\top \Phi_t$. The meta algorithm SqAlg aggregates the experts' predictions $\{f^i(x_t,a;H_t)\}_{i=1}^M$ and produces its own predictions $\widehat{y}_t(x_t,a)$, $\forall a \in [K]$, using Algorithm 3 in Appendix A (see Remark 3.1).

The next step in FS-SCB is computing the action with the highest predicted reward, i.e., $a_t' = \text{argmax}_{a\in[K]}\widehat{y}_t(x_t,a)$, and using it to define a distribution $p_t \in \Delta_K$ over the actions (see Eq. 5). The distribution p_t in (5) is defined similarly to the probability selection scheme of Abe & Long (1999), and assigns a probability to every action inversely proportional to the gap between its prediction and that of a_t'. The algorithm then samples its action a_t from p_t, observes reward y_t, and feeds the tuple (x_t,a_t,y_t) to the oracle to update its weights over the experts. Our analysis in Section 3.1 and Appendix B suggest to set the exploration parameter to $\kappa = K$ and the learning rate to $\alpha = \sqrt{KT/D_T(\delta)}$, where we define $D_T(\delta)$ in Lemma 3.3 and give its exact expression in Eq. 40 in Appendix B.2.

Algorithm 1 Feature Selection Square-CB (FS-SCB)

Input: Models $\{\phi^i\}_{i=1}^M$, Confidence Parameter δ, Learning Rate α, Exploration Parameter κ
for $t = 1$ **to** T **do**
  Observe context x_t
  Oracle predicts:
  $\widehat{y}_t(x_t,a) = \texttt{SqAlg}_t(x_t,a;H_t), \ \forall a \in [K]$
  Define a distribution p_t over the actions:

$$p_t(a) = \begin{cases} \frac{1}{\kappa + \alpha\left(\widehat{y}_t(x_t,a) - \widehat{y}_t(x_t,a_t')\right)}, & a \ne a_t', \\ 1 - \sum_{a\ne a_t'} p_t(a), & a = a_t', \end{cases} \quad (5)$$

  where $a_t' = \text{argmax}_{a\in[K]}\widehat{y}_t(x_t,a)$;
  Sample action $a_t \sim p_t(\cdot)$ and play it;
  Observe reward $y_t = \langle \phi^{i_*}(x_t,a_t), \theta_*^{i_*}\rangle + \eta_t$;
  Update SqAlg with (x_t,a_t,y_t);
end for

Remark 3.1 (Admissible Experts)**.** It is important to note that in each round $t \in [T]$, FS-SCB only uses predictions by admissible experts, i.e., experts i that belong to the set

$$\mathcal{S}_t := \Big\{ i \in \mathcal{S}_{t-1} : \langle \phi^i(x_t,a), \widehat{\theta}_t^i\rangle \le G + RL\sqrt{d\log\left(\frac{1+\frac{tL^2}{\lambda_i d}}{\delta}\right)} + L\sqrt{\lambda_i}S, \quad \forall a \in [K] \Big\}, \quad (6)$$

with $\mathcal{S}_0 = [M]$. This is the set of experts i whose predictions $f^i(x_t,a;H_t) = \langle \phi^i(x_t,a), \widehat{\theta}_t^i\rangle$, $\forall a \in [K]$ are within a bound defined by (6). When an expert was removed from the admissible set in a round t, it will remain out for the rest of the game. We discuss the technical reasons for defining this set in the proof of Lemma 3.5 in Appendix B.2.

3.1. Regret Analysis of FS-SCB

We state a regret bound for FS-SCB followed by a proof sketch. The detailed proofs are all reported in Appendix B.

Theorem 3.2. *Let Assumption 2.1 hold and the regularization parameters λ_i, exploration parameter κ, and learning rate α set to the values described above. Then, for any $\delta \in [0,1/4)$, w.p. at least $1-\delta$, the regret defined by (2) for FS-SCB is bounded as*

$$\mathcal{R}_{\text{FS-SCB}}(T) \le \mathcal{O}\Bigg(\sqrt{2T\log(2/\delta)} + RLG \times \sqrt{KT(1+\log(M)) \max_{i\in[M]}\left\{ \lambda_i S^2 + 4d\log\left(\frac{1+\frac{TL^2}{\lambda_i d}}{\delta}\right)\right\}}\Bigg).$$

Proof Sketch. The proof consists of two main steps:

Step 1. We first need to bound the prediction error of the online regression oracle.

Lemma 3.3. *For any $\delta \in (0, 1/4]$, w.p. at least $1-\delta$, we can bound the prediction error of the regression oracle as*

$$\sum_{s=1}^{t-1} \left(\widehat{y}_s(x_s, a_s) - \langle \phi^{i_*}(x_s, a_s), \theta_*^{i_*}\rangle\right)^2 \le D_t(\delta) :=$$
$$\mathcal{O}\Big(\big(1 + R^2L^2G^2\log(M)\big) \max_{i\in[M]} \big\{\lambda_i S^2 + 4d\log\big(\frac{1+\frac{TL^2}{\lambda_i d}}{\delta}\big)\big\}\Big).$$

The exact definition of $D_t(\delta)$ (see Eq. 40 in Appendix B.3) shows its dependence on the following two terms: **1)** an upper-bound Q_t on the prediction error of the true models,

$$\max_{i_*} \sum_{s=1}^{t-1} \left(\langle \phi^{i_*}(x_s, a_s), \widehat{\theta}_s^{i_*}\rangle - \langle \phi^{i_*}(x_s, a_s), \theta_*^{i_*}\rangle\right)^2 \le Q_t, \quad (7)$$

and **2)** the regret $\mathcal{R}_{\text{Sq}}(t)$ of the regression oracle. Thus, the proof of Lemma 3.3 requires finding expressions for these quantities, which we derive them in the following lemmas.

Lemma 3.4. *For any $\delta \in (0, 1)$, with probability at least $1-\delta$, we may write Q_t defined in (7) as (see Eq. 24 in Appendix B.1 for the exact expression)*

$$Q_t = \mathcal{O}\Big(\max_{i\in[M]} \big\{\lambda_i S^2 + 4d\log\big(1 + \frac{tL^2}{\lambda_i d}\big)\big\} + R^2\log(1/\delta)\Big).$$

Lemma 3.5. *For any $\delta \in (0, 1)$, with probability at least $1-\delta$, we may write the regret of the regression oracle as (see Eq. 34 in Appendix B.2 for the exact expression)*

$$\mathcal{R}_{\text{Sq}}(t) = \mathcal{O}\Big(R^2L^2\log(M)\times$$
$$\Big(G^2 + \max_{i\in[M]} \big\{\lambda_i S^2 + d\log\big(1 + \frac{tL^2}{\lambda_i d}\big)\big\} + \log(1/\delta)\Big)\Big).$$

Step 2. We then show how the overall regret of FS-SCB is related to the prediction error of the online regression oracle, $D_t(\delta)$, using the following lemma:

Lemma 3.6. *Under the same assumptions as Theorem 3.2, for any $\delta \in (0, 1/4]$, with probability at least $1-\delta$, the regret of the FS-SCB algorithm is bounded as*

$$\mathcal{R}_{\text{FS-SCB}}(T) \le \sqrt{2T\log(2/\delta)} + \frac{\alpha}{4} D_T(\delta) +$$
$$\sum_{t=1}^{T} \sum_{a\in[K]} p_t(a) \Big(\langle \phi^{i_*}(x_t, a), \theta_*^{i_*}\rangle - \langle \phi^{i_*}(x_t, a_t^*), \theta_*^{i_*}\rangle$$
$$- \frac{\alpha}{4}\left(\widehat{y}_t(x_t, a) - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*}\rangle\right)^2\Big). \quad (8)$$

Finally, we conclude the proof of Theorem 3.2 by bounding the last term on the RHS of (8) using Lemma B.1 (see Appendix B.5 for details).

3.2. Related Work (Feature Selection)

The most straightforward solution to the feature selection problem described in Section 2.2 is to concatenate all models (feature maps) and build a $(M \times d)$-dimensional feature, and then search for the sparse reward parameter $\theta_* \in \mathbb{R}^{Md}$ with only d non-zero elements. We may then solve the resulting LB problem using a sparse LB algorithm (e.g., Abbasi-Yadkori et al. 2012). This approach would result in a regret bound of $\widetilde{\mathcal{O}}(d\sqrt{MT})$, which may not be desirable when the number of models M is large.

Another approach is to use the EXP4 (or SquareCB) algorithm (Auer et al., 2002b) to obtain a regret that scales only logarithmically with M. If we partition the linear space of each model into $\mathcal{O}(2^d)$ predictors, we will have the total number of $\mathcal{O}(M2^d)$ predictors. Predictor $(i, j) \in ([M], [2^d])$ is associated with a linear map $\theta^{ij} \in \mathbb{R}^d$ and recommends the action $\text{argmax}_{a\in A}\langle \phi^i(a), \theta^{ij}\rangle$. The regret of EXP4 with this set of experts is of $\widetilde{\mathcal{O}}(\sqrt{dKT\log M})$. Although this solution has logarithmic dependence on M, it is still not desirable, since it is not computationally efficient (requires handling $M2^d$ predictors).

To have computational efficiency, we can use the approach of Maillard & Munos (2011), but this results in a $O(T^{2/3})$ regret. They designed a model selection strategy using an EXP4 algorithm with a set of experts that are instances of the S-EXP3 algorithm of Auer et al. (2002c). The interesting fact is that each S-EXP3 expert is a learning algorithm and competes against a set of mappings. The overall regret of this algorithm is of $\widetilde{\mathcal{O}}\big(T^{2/3}(|S|K\log K)^{1/3}\sqrt{\log M}\big)$ (see Bubeck & Cesa-Bianchi 2012, Chapter 4.2). If we apply this algorithm to our setting, the resulting regret bound is of $\widetilde{\mathcal{O}}(T^{2/3}d^{1/3}K^{1/3}\sqrt{\log M})$. Although the algorithm is computationally more efficient than EXP4 and its regret has logarithmic dependence on M, it is still not desirable as its dependence on T is of $\widetilde{\mathcal{O}}(T^{2/3})$, which is not optimal.

The novelty of our results is that we propose a computationally efficient algorithm, whose regret has better dependence on M and T, i.e., $\widetilde{\mathcal{O}}(\sqrt{KT\log M})$, than all the existing methods. Our FS-SCB algorithm achieves this by **1)** using a novel instantiation of SquareCB, or more precisely by constructing a proper full information algorithm (expert), and **2)** using SquareCB with a set of *adaptive (learning)*, and not static, least-squares experts. Note that SquareCB is a reduction that turns any online regression oracle into an algorithm for contextual bandits (Foster & Rakhlin, 2020).

More recently, Papini et al. (2021) studied a feature selection problem where the reward function is linear in *all* M feature maps (all models are *realizable*). Under this *stronger* assumption (than ours), they prove a regret bound that is competitive (up to a $\log M$ factor) with that of a linear bandit algorithm that uses the best feature map. More

specifically, if one of the feature maps is such that a constant regret is achievable, the overall model selection strategy also achieves a constant regret. Although our focus is not on constant regret, we are able to achieve our results without requiring all models to be *realizable*.

4. Parameter Selection Algorithm

We propose a UCB-style algorithm for the parameter selection setting described in Section 2.3, which we refer to as *parameter selection OFUL* (PS-OFUL). We then provide an upper-bound on its transfer regret and conclude with a discussion on the existing results related to this setting.

Algorithm 2 contains the pseudo-code of PS-OFUL. The novel idea in PS-OFUL is the construction of its confidence set $\mathcal{C}_t$ (Eq. 10), which is based on the predictions $\{\widehat{y}_s\}_{s=1}^{t-1}$ by a regression oracle `SqAlg`. As described in Section 2.4, `SqAlg` is a meta algorithm that consists of M learning algorithms (or experts), and its predictions $\widehat{y}_t$ are aggregates of its experts' predictions $f^i(\phi_t(a_t); H_t),\ \forall i \in [M]$. In PS-OFUL, each expert $i \in [M]$ is a *biased regularized least-squares* algorithm with bias $\widehat{\mu}_i$, i.e., our estimate of the center of the i^{th} ball (model). Expert i predicts the reward of the context-action $\phi_t(a_t)$ as $f^i(\phi_t(a_t); H_t) = \langle \phi_t(a_t), \widehat{\theta}_t^i \rangle$, where $\widehat{\theta}_t^i = \text{argmin}_\theta \|\Phi_t^\top \theta - Y_t\|^2 + \lambda_i \|\theta - \widehat{\mu}_i\|^2$. We may write $\widehat{\theta}_t^i$ in closed-form as $\widehat{\theta}_t^i = (V_t^{\lambda_i})^{-1} \Phi_t^\top (Y_t - \Phi_t \widehat{\mu}_i) + \widehat{\mu}_i$. In these equations, the reward vector Y_t and $V_t^{\lambda_i}$ are defined as in Section 3; Φ_t is the feature matrix, whose rows are $\phi_1(a_1), \dots, \phi_{t-1}(a_{t-1})$; and λ_i is the regularization parameter of expert i. Our analysis in Section 4.1 and Appendix C suggests to set them to $\lambda_i = \frac{1}{(b_i + c_i)^2}$.

The PS-OFUL algorithm takes the feature map ϕ and models $\{B(\widehat{\mu}_i, b_i + c_i)\}_{i=1}^M$ as input. At each round $t \in [T]$, it first constructs a confidence set $\mathcal{C}_{t-1}$ using the predictions of the regression oracle $\{\widehat{y}_s\}_{s=1}^{t-1}$. The radius $\gamma_t(\delta)$ of the confidence set $\mathcal{C}_t$ is defined by two terms: **1)** the regret $\mathcal{R}_{\texttt{Sq}}(t)$ of the regression oracle `SqAlg`, defined by (4), and **2)** an upper-bound U_t on the prediction error of the true models (i.e., models that contain θ_*), i.e.,

$$\max_{i \in \mathcal{I}_*} \sum_{s=1}^{t-1} \left(\langle \phi_s(a_s), \widehat{\theta}_t^i \rangle - \langle \phi_s(a_s), \theta_* \rangle \right)^2 \leq U_t. \tag{9}$$

The exact values of U_t, $\mathcal{R}_{\texttt{Sq}}(t)$, and $\gamma_t(\delta)$ come from our analysis and have been stated in Eq. 65 in Appendix C.3. PS-OFUL then computes action a_t as the one that attains the maximum optimistic reward w.r.t. the confidence set $\mathcal{C}_{t-1}$. Using a_t, it calculates $\widehat{y}_t = \texttt{SqAlg}_t(\phi_t(a_t); H_t))$. As described in Section 2.4, `SqAlg` makes use of Algorithm 3 in Appendix A to return its prediction $\widehat{y}_t$ as an aggregate of its experts' predictions (see Remark 4.1). Finally, PS-OFUL takes action a_t, observes reward y_t, and pass the sample $(\phi_t(a_t), y_t)$ to `SqAlg`. This sample is then used within `SqAlg` to evaluate its experts and to update their weights.

Algorithm 2 Parameter Selection OFUL (PS-OFUL)

Input: Feature Map ϕ, Confidence Parameter δ, Models $\{B(\widehat{\mu}_i, b_i + c_i)\}_{i=1}^M$
for $t = 1$ **to** T **do**
  Construct the confidence set:

$$\mathcal{C}_{t-1} = \left\{ \theta : \sum_{s=1}^{t-1} \left(\widehat{y}_s - \langle \phi_s(a_s), \theta \rangle \right)^2 \leq \gamma_{t-1}(\delta) \right\} \tag{10}$$

  Take action: $a_t = \arg\max_{a \in \mathcal{A}_t} \max_{\theta \in \mathcal{C}_{t-1}} \langle \phi_t(a), \theta \rangle$
  Oracle predicts: $\widehat{y}_t = \texttt{SqAlg}_t(\phi_t(a_t); H_t)$
  Observe reward: $y_t = \langle \phi_t(a_t), \theta_* \rangle + \eta_t$
  Update `SqAlg` with $(\phi_t(a_t), y_t)$;
end for

Remark 4.1 (Admissible Experts)**.** Similar to FS-SCB, in each round $t \in [T]$, PS-OFUL only uses predictions by admissible experts, i.e., experts i that belong to the set

$$\mathcal{S}_t := \{ i \in \mathcal{S}_{t-1} : \langle \phi_t(a_t), \widehat{\theta}_t^i \rangle \leq G + RL \sqrt{d \log \left(\frac{1 + \frac{tL^2}{\lambda_i d}}{\delta} \right)} + L\sqrt{\lambda_i}(b_i + c_i) \}, \tag{11}$$

with $\mathcal{S}_0 = [M]$. This is the set of experts i whose prediction $f^i(\phi_t(a_t); H_t) = \langle \phi_t(a_t), \widehat{\theta}_t^i \rangle$ is within a bound defined by (11). When an expert was removed from the admissible set in a round t, it will remain out for the rest of the game. We discuss the technical reasons for defining this set in the proof of Lemma 4.5 in Appendix C.2.

4.1. Regret Analysis of PS-OFUL

We state a regret bound for PS-OFUL followed by a proof sketch. The detailed proofs are all reported in Appendix C.

Theorem 4.2. *Let Assumption 2.2 hold and* $\lambda_i = \frac{1}{(b_i + c_i)^2} \geq 1,\ \forall i \in [M]$. *Then, for any* $\delta \in (0, 1/4]$, *with probability at least* $1 - \delta$, *the transfer-regret defined by* (3) *of PS-OFUL is bounded as*

$$\mathcal{R}(T) = \mathcal{O}\Bigg(dRL \max\{1, G\} \sqrt{1 + \log(M)} \times \sqrt{T \log\left(1 + \frac{T}{d}\right) \log\left(\frac{1 + \frac{TL^2 \max_{i \in [M]} (b_i + c_i)^2}{d}}{\delta} \right)} \Bigg). \tag{12}$$

Proof Sketch. The proof consists of two main steps.

Step 1. We first fully specify the confidence set $\mathcal{C}_t$ and prove its validity i.e., $\mathbb{P}(\theta_* \in \mathcal{C}_t) \geq 1 - \delta,\ \forall t \in [T]$.

Theorem 4.3. *Under the same assumptions as Theorem 4.2, the radius* $\gamma_t(\delta)$ *of the confidence set* $\mathcal{C}_t$ *is fully specified*

by Eq. 65 in Appendix C.3. Moreover, for any $\delta \in (0, 1/4]$, with probability at least $1-\delta$, the true reward parameter θ_* lies in $\mathcal{C}_t$, i.e., $\mathbb{P}(\theta_* \in \mathcal{C}_t) \geq 1-\delta$.

The definition of $\gamma_t(\delta)$ in Eq. 65 shows its dependence on U_t and $\mathcal{R}_{\text{Sq}}(t)$, defined by (9) and (4), respectively. Thus, the proof of Thm. 4.3 requires finding expressions for these quantities, which we derive them in the following lemmas.

Lemma 4.4. *Setting $\lambda_i = \frac{1}{(b_i+c_i)^2}$, $\forall i \in [M]$, with probability $1-\delta$, we may write U_t, defined by (9), as (see Eq. 52 in Appendix C.1 for the exact expression)*

$$U_t = \mathcal{O}\Big(dR^2 \log\big(\frac{1 + \frac{tL^2 \max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\big)\Big).$$

Lemma 4.5. *Setting $\lambda_i = \frac{1}{(b_i+c_i)^2}$, $\forall i \in [M]$, with probability $1-\delta$, we may write $\mathcal{R}_{\text{Sq}}(t)$, defined by (4), as (see Eq. 58 in Appendix C.2 for the exact expression)*

$$\mathcal{R}_{\text{Sq}}(t) = \mathcal{O}\Big(dR^2L^2 \log(M) \log\big(\frac{1 + \frac{tL^2 \max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\big)\Big).$$

Step 2. We then show how the regret is related to the confidence sets using the following lemma:

Lemma 4.6. *Under the same assumptions as Theorem 4.2, for any $\delta \in (0, 1/4]$, with probability at least $1-\delta$, the regret of PS-OFUL is bounded as*

$$\mathcal{R}_{\text{PS-OFUL}}(T) \leq 2Gd + 2\max\{1, G\}\sqrt{2dT \log\big(1 + \frac{T}{d}\big) \max_{d<t\leq T} \gamma_t(\delta)}. \tag{13}$$

We conclude the proof of Theorem 4.2 by plugging the confidence radius $\gamma_t(\delta)$ computed in Theorem 4.3 (Eq. 65 in Appendix C.3) into the regret bound (13).

4.2. Related Work (Parameter Selection)

Cella et al. (2020) and Moradipari et al. (2022b) studied *meta learning* in stochastic linear bandit (LB), where the agent solves a sequence of LB problems, whose reward parameters θ_* are drawn from an unknown distribution ρ of bounded support in $\mathbb{R}^d$. For each LB task, the agent is given an estimate of the mean of the distribution ρ and an upper-bound of its error, and its goal is to minimize the transfer regret $\mathcal{R}(T, \rho) = \mathbb{E}_{\theta_*\sim\rho}\big[\mathbb{E}[\mathcal{R}(T, \theta_*)]\big]$. Their proposed algorithms assume knowing the variance term $\text{Var}_h = \mathbb{E}_{\theta_*\sim\rho}\big[\|\theta_* - h\|^2\big]$, for any $h \in \mathbb{R}^d$, in order to properly set their regularization parameter λ. Thus, the parameter selection setting studied in our paper can be seen as an extension of their transfer learning setting to multiple (M) models. Moreover, we allow the reward parameter of the new LB problem θ_* to be selected arbitrarily from the M models, and consider a worst-case transfer regret (see Eq. 3) for our algorithm (instead of a regret in expectation w.r.t. ρ). Despite these differences, our setting is similar to theirs as we are also given an estimate of the center of each model $\widehat{\mu}_i$, together with an upper-bound on its error c_i, plus the radius b_i of each model. Also similar to their results, our analysis clearly shows the importance of the choice of the regularization parameters, $\lambda_i = 1/(b_i + c_i)^2$, for obtaining a regret bound that only logarithmically depends on the maximum model uncertainty, i.e., $\max_{i\in[M]}(b_i + c_i)^2$.

Our parameter selection setting is also related to *latent* bandits (Maillard & Mannor, 2014; Hong et al., 2020) in which identifying the true latent variable is analogous to finding the correct model. The latest work in this area is by Hong et al. (2020) in which the agent faces a K-armed LB problem selected from a set of M known K-dimensional reward vectors. They proposed UCB and TS algorithms for this setting and showed that their regret (Bayes regret in case of TS) are bounded as $3M + 2T\varepsilon + 2R\sqrt{6MT \log T}$, where the reward vectors are known up to an error of ε. Comparing to their results, the regret of PS-OFUL in (12) has a better dependence on the number of models, $\sqrt{\log M}$ vs. M, and the model uncertainty, $\sqrt{\log(\max_{i\in[M]}(b_i + c_i)^2)}$ vs. ε. However, the number of actions K does not appear in their bound, while the bound of PS-OFUL will have a $\sqrt{K}$ factor when applied to K-armed bandit problems. If the objective is to have a better scaling in K, we can use a different bandit model selection strategy, called *regret balancing* (Abbasi-Yadkori et al., 2020; Pacchiano et al., 2020a), to obtain an improved regret that scales as $\min\{\varepsilon T + \sqrt{MT}, \sqrt{KMT}\}$ (see Appendix E for details).

In another closely related work, Hong et al. (2022) approach a similar problem by initializing TS with a prior that is a mixture of M distributions. They prove a Bayes regret bound for their algorithm in case of Gaussian mixtures that has $\sqrt{M}$ dependence on the number of models and $\sqrt{\max_{i\in[M]} \sigma_{0,i}^2}$ dependence on the maximum variance of the Gaussian priors. Both these dependences are logarithmic $\sqrt{\log M}$ and $\sqrt{\log(\max_{i\in[M]}(b_i + c_i)^2)}$ in the regret of PS-OFUL.

5. Experiments

We evaluate the performances of our FS-SCB and PS-OFUL algorithms using a synthetic LB problem and image classification problems: MNIST (LeCun et al., 1998) and CIFAR-10 (Krizhevsky et al., 2009). We report the details of our experimental setup and additional results in Appendix F.

Feature Selection (Synthetic): We first sample the parameter of the linear bandit problem from a $d = 50$ dimensional Gaussian with variance 0.01: $\theta_* \sim \mathcal{N}(0, 0.01 I_d)$. We generate all feature maps, $\{\phi^i(a)\}_{i=1}^M$, by sampling $10,000$ vectors from the Gaussian with mean θ_* and covariance

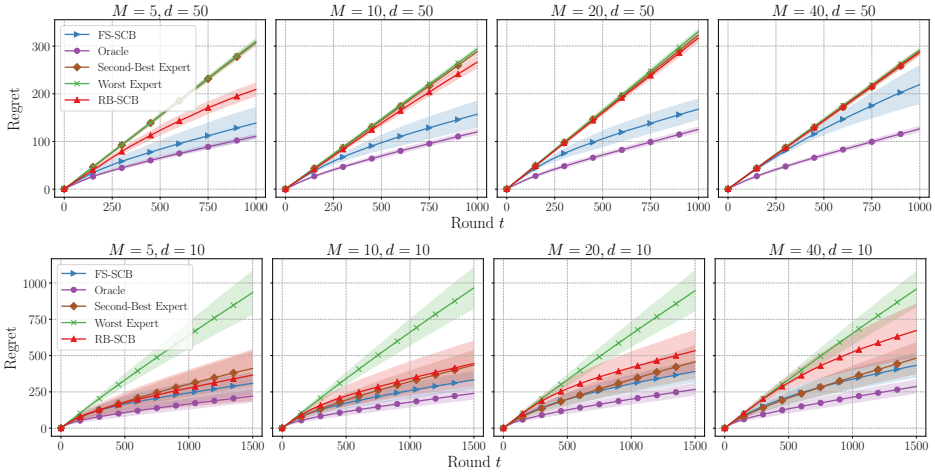

Figure 1. Feature selection in the synthetic LB problem *(top)* and MNIST *(bottom)*. The regrets are averaged over 100 LB problems.

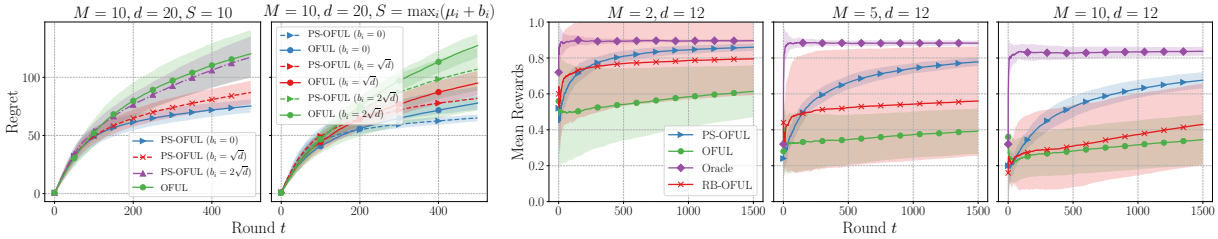

Figure 2. Parameter selection in the synthetic LB problem *(left)* and CIFAR-10 *(right)*. Results are averaged over 50 runs.

$0.1I_d$, i.e., $\phi^i(a) \sim \mathcal{N}(\theta_*, 0.1I_d)$, for $a = 1, \dots, 10,000$. This implies that all M feature maps have the same bias. We set $\phi^1(\cdot)$ to be the *true* feature map. At each round $t \in [T]$, the learner is given an action set consist of 10 numbers from $\mathcal{A} = \{1, 2, \dots, 10,000\}$. The reward of each action a is $\langle \phi^1(a), \theta_\star \rangle + \eta_t$, where $\eta_t \sim \mathcal{U}[-0.5, 0.5]$.

Feature Selection (MNIST): We train a convolutional neural network (CNN) with M different number of epochs on MNIST data, and use their second layer to the last as our $d = 10$-dimensional feature maps $\{\phi^i\}_{i=1}^M$. These feature maps have test accuracy between 20% (worst model) and 97% (best model). We set the best one as true model ϕ^{i_*}. For each class $s \in \mathcal{S} = \{0, \dots, 9\}$, we fit a linear model, given the feature map ϕ^{i_*}, and obtain parameters $\{\theta_s^{i_*}\}_{s=0}^9$. At the beginning of each LB task, we select a class $s_* \in \mathcal{S}$ uniformly at random and set its parameter to $\theta_{s_*}^{i_*}$. At each round $t \in [T]$, the learner is given an action set consists of 10 images, one from class s_* and the rest randomly selected from the other classes. The reward of each action a is defined as $\langle \phi^{i_*}(a), \theta_{s_*}^{i_*} \rangle + \eta_t \in [0, 1]$, where $\phi^{i_*}(a)$ is the application of the feature map ϕ^{i_*} to the image corresponding to action a and $\eta_t \sim \mathcal{U}[-0.5, 0.5]$ is the noise.

In Figure 1, we compare the regret of our FS-SCB algorithm for different number of models M with a regret balancing algorithm that uses SquareCB baselines (RB-SCB), and three SquareCB algorithms that use the best (Oracle), second-best (with test accuracy 84% for MNIST), and worst feature maps. The results in Figure 1 show that **1)** FS-SCB always performs between the best and second-best experts, **2)** the regret of FS-SCB that scales as $\sqrt{\log M}$ is close to RB-SCB (scales as $\sqrt{M}$) for small M, but gets much better as M grows, and **3)** RB-SCB has much higher variance than the other algorithms in MNIST.

Parameter Selection (Synthetic): We first sample the center of $M = 10$ balls from a $d = 20$-dimensional Gaussian, i.e., $\{\mu_i\}_{i=1}^M \sim \mathcal{N}(0, I_d)$, and set their radii to $b_i = b$, $\forall i \in [M]$. At the beginning of each LB task, we select a model $i_* \in [M]$ uniformly at random, and then sample the problem's parameter from its ball, i.e., $\theta_* \sim B(\mu_{i_*}, b_{i_*})$. The action set in each round $t \in [T]$ consists of 10 vectors $\{\phi_t(a_j)\}_{j=1}^{10} \sim \mathcal{N}(0, 0.01I_d)$, and the reward of the selected action a_t is defined as $\langle \phi_t(a_t), \theta_* \rangle + \eta_t$, $\eta_t \sim \mathcal{U}[-0.5, 0.5]$. Figure 2 *(left)* compares the regret of our PS-OFUL algorithm with OFUL (Abbasi-Yadkori et al., 2011) for different sizes of the balls $b \in \{0, \sqrt{d}, 2\sqrt{d}\}$. We run OFUL with the upper-bounds $\|\theta_*\|_2 \le S = 10$ and $S = \max_i(\mu_i + b_i)$ on the reward parameter. Note that the second bound is tighter and shows the best performance of

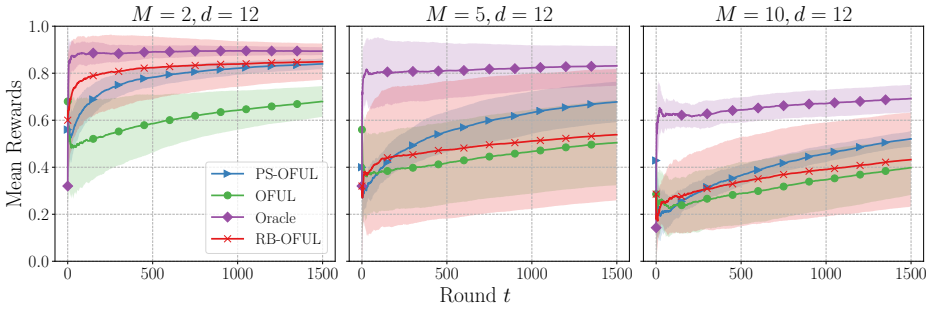

Figure 3. Parameter selection in CIFAR-10 with models less accurate than those in Figure 2 *(right)*. The results are averaged over 50 runs.

OFUL. Our results indicate that the regret of PS-OFUL is better than OFUL, and gets closer to it as we increase the size of the balls from $b = 0$ to $b = 2\sqrt{d} \approx 9$. This clearly shows the potential advantage of transfer (PS-OFUL) over individual (OFUL) learning.

Parameter Selection (CIFAR-10): We modify the EfficientNetV2-S network (Tan & Le, 2021) by adding a layer of $d = 12$ neurons before the last layer and fine-tuning it on CIFAR-10 dataset. We then select this d-dimensional layer as our feature map ϕ. To define our M models (balls), we sample $100M$ datasets of size 500. For each dataset, we randomly select a class $s_* \in [M]$, assign reward 1 to images from s_* and 0 to other images, and fit a linear model to it to obtain a parameter vector. Finally, we fit a Gaussian mixture model with M components to these $100M$ parameter vectors and use the means and covariances of the resulting clusters as the center and radii of our M models (balls). At the beginning of each LB task, we select a class $s_* \in [M]$ uniformly at random. In each round $t \in [T]$, the learner is given an action set consists of 10 images, one from class s_* and the rest randomly selected from the other classes. The learner receives a reward from $\text{Ber}(0.9)$, if it selects the image from class s_*, and from $\text{Ber}(0.1)$, otherwise.

In Figure 2 *(right)*, we compare the mean reward of PS-OFUL for different values of M with a regret balancing algorithm that uses OFUL baselines (RB-OFUL) (Abbasi-Yadkori et al., 2020), OFUL (individual learning), and BIAS-OFUL (Cella et al., 2020) with bias being the center of the true model (Oracle). The results show **1)** the good performance of PS-OFUL, **2)** the performance of PS-OFUL gets better than RB-OFUL as M grows ($\sqrt{\log M}$ vs. $\sqrt{M}$ scaling), **3)** the large variance of RB-OFUL, especially in comparison to PS-OFUL, and finally **4)** the advantage of transfer (PS-OFUL) over individual (OFUL) learning.

In order to show the impact of the model accuracy (the accuracy of the center of the balls and their radii) on the performance of the algorithms, we defined a less accurate set of M models (balls) using $10M$ datasets of size 50 (as opposed to $100M$ datasets of size 500 used in the results reported in Figure 2 *(right)*). In Figure 3, we compare the mean reward of PS-OFUL for different number of models M with RB-OFUL, OFUL, and BIAS-OFUL. The results indicate that with decreasing in the accuracy of the models, the performance of PS-OFUL and RB-OFUL get closer to that for OFUL.

6. Conclusions

We studied two model selection settings in LB, where the mean reward is linear in at least one of M models *(feature selection)*, and where the reward parameter is arbitrarily selected from M misspecified models *(parameter selection)*. We derived computationally efficient algorithms in these settings that are based on reductions from bandits to full-information problems, and proved regret bounds with desirable dependence on the horizon and number of models. An interesting future direction is to extend our results to the meta learning and learning-to-learn setting, where the agent starts with M models, and instead of solving a single LB problem, has to solve N of them one after another.

7. Acknowledgement

This work is partially supported by NSF grant 1847096.

References

Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pp. 2312–2320, 2011.

Abbasi-Yadkori, Y., Pal, D., and Szepesvari, C. Online-to-confidence-set conversions and application to sparse stochastic bandits. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, 2012.

Abbasi-Yadkori, Y., Pacchiano, A., and Phan, M. Regret balancing for bandit and RL model selection. *arXiv:2006.05491*, 2020.

Abe, N. and Long, P. Associative reinforcement learning using linear probabilistic concepts. In *ICML*, 1999.

Abeille, M., Lazaric, A., et al. Linear thompson sampling revisited. *Electronic Journal of Statistics*, 11(2):5165–5197, 2017.

Agarwal, A., Luo, H., Neyshabur, B., and Schapire, R. E. Corralling a band of bandit algorithms. In *COLT*, 2017.

Agrawal, S. and Goyal, N. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pp. 127–135, 2013.

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

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. The non-stochastic multi-armed bandit problem. *SIAM Journal of Computing*, 2002b.

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002c.

Bubeck, S. and Cesa-Bianchi, N. Regret analysis of stochastic and nonstochastic multiarmed bandit problems. *Foundations and Trends in Machine Learning*, 2012.

Cella, L., Lazaric, A., and Pontil, M. Meta-learning with stochastic linear bandits. In *International Conference on Machine Learning*, pp. 1360–1370, 2020.

Cesa-Bianchi, N. and Lugosi, G. *Prediction, learning, and games*. Cambridge University Press, 2006.

Cutkosky, A., Dann, C., Das, A., Gentile, C., Pacchiano, A., and Purohit, M. Dynamic balancing for model selection in bandits and rl. In *International Conference on Machine Learning*, pp. 2276–2285. PMLR, 2021.

Dani, V., Hayes, T., and Kakade, S. M. Stochastic linear optimization under bandit feedback. *21st Annual Conference on Learning Theory*, 2008.

Foster, D. and Rakhlin, A. Beyond ucb: Optimal and efficient contextual bandits with regression oracles. In *International Conference on Machine Learning*, pp. 3199–3210, 2020.

Foster, D., Gentile, C., Mohri, M., and Zimmert, J. Adapting to misspecification in contextual bandits. In *Advances in Neural Information Processing Systems 35*, pp. 11478–11489, 2020.

Haussler, D., Kivinen, J., and Warmuth, M. K. Sequential prediction of individual sequences under general loss functions. *IEEE Trans. Inform. Theory*, pp. 1906–1925, 1998.

Hong, J., Kveton, B., Zaheer, M., Chow, Y., Ahmed, A., and Boutilier, C. Latent bandits revisited. In *NeurIPS*, 2020.

Hong, J., Kveton, B., Zaheer, M., Ghavamzadeh, M., and Boutilier, C. Thompson sampling with a mixture prior. In *AISTATS*, 2022.

Krizhevsky, A. et al. Learning multiple layers of features from tiny images. 2009.

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

Lattimore, T. and Szepesvari, C. *Bandit Algorithms*. Cambridge University Press, 2020.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Maillard, O. and Mannor, S. Latent bandits. In *ICML*, 2014.

Maillard, O. and Munos, R. Adaptive bandits: Towards the best history-dependent strategy. In *AISTATS*, 2011.

Moradipari, A., Silva, C., and Alizadeh, M. Learning to dynamically price electricity demand based on multi-armed bandits. In *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 917–921. IEEE, 2018.

Moradipari, A., Alizadeh, M., and Thrampoulidis, C. Linear thompson sampling under unknown linear constraints. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3392–3396. IEEE, 2020a.

Moradipari, A., Thrampoulidis, C., and Alizadeh, M. Stage-wise conservative linear bandits. *Advances in neural information processing systems*, 33:11191–11201, 2020b.

Moradipari, A., Amani, S., Alizadeh, M., and Thrampoulidis, C. Safe linear thompson sampling with side information. *IEEE Transactions on Signal Processing*, 69:3755–3767, 2021.

Moradipari, A., Ghavamzadeh, M., and Alizadeh, M. Collaborative multi-agent stochastic linear bandits. *arXiv preprint arXiv:2205.06331*, 2022a.

Moradipari, A., Ghavamzadeh, M., Rajabzadeh, T., Thrampoulidis, C., and Alizadeh, M. Multi-environment meta-learning in stochastic linear bandits. *arXiv preprint arXiv:2205.06326*, 2022b.

Pacchiano, A., Dann, C., Gentile, C., and Bartlett, P. Regret bound balancing and elimination for model selection in bandits and rl. *arXiv:2012.13045*, 2020a.

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

Papini, M., Tirinzoni, A., Restelli, M., Lazaric, A., and Pirotta, M. Leveraging good representations in linear contextual bandits. In *ICML*, 2021.

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

Russo, D. and Van Roy, B. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39 (4):1221–1243, 2014. doi: 10.1287/moor.2014.0650.

Tan, M. and Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pp. 6105–6114. PMLR, 2019.

Tan, M. and Le, Q. V. Efficientnetv2: Smaller models and faster training. *arXiv preprint arXiv:2104.00298*, 2021.

Thompson, W. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

A. Sequential Prediction Algorithm

The sequential prediction algorithm SqAlg uses the following algorithm from (Haussler et al., 1998) (also see Cesa-Bianchi & Lugosi 2006, Chapter 3) to aggregate its experts' predictions. Algorithm 3 takes the observations y_t and experts' predictions $f_t^i(H_t)$ that are bounded in the known range $[\beta, \beta+\ell]$ as input. It first scales these input to the range $[0,1]$ and uses its current weights for the experts to generate its own prediction $\widehat{y}_t$.

The performance of SqAlg is evaluated as the accuracy (in terms of square loss) of its prediction w.r.t. the accuracy of the prediction by the best expert in the set, i.e.,

$$\sum_{t=1}^{T}(\widehat{y}_t - y_t)^2 - \min_{i\in[M]}\sum_{t=1}^{T}(f_t^i(H_t) - y_t)^2 \leq \mathcal{R}_{\texttt{Sq}}(T). \tag{14}$$

We call this the regret of SqAlg and denote it by $\mathcal{R}_{\texttt{Sq}}(T)$. Haussler et al. (1998) prove the following bound for $\mathcal{R}_{\texttt{Sq}}(T)$, which we use in the analysis of our algorithms.

Algorithm 3 Sequential Prediction with Expert Advice

Input: ℓ and β *(experts' predictions $f_t^i(H_t)$ are bounded in the known range $[\beta, \beta+\ell]$)*
Initialization: Set the weight $w_{1,i} = 1$ for all experts $i \in [M]$
for $t = 1$ **to** T **do**
Receive predictions $f_t^i(H_t)$ by experts $i \in \mathcal{S}_{t-1}$
Remove experts whose predictions are out of bound and construct the new set of admissible experts $\mathcal{S}_t$ *(see Remarks 3.1 and 4.1)*
Scale experts' predictions $h_{i,t} = \frac{f_t^i(H_t)-\beta}{\ell},\ \forall i \in \mathcal{S}_t$
Set $v_{t,i} = \frac{w_{t,i}}{W},\ \forall i \in \mathcal{S}_t$, where $W = \sum_{i\in\mathcal{S}_t} w_{t,i}$
Prediction: Compute:

$$\Delta(0) = \frac{-1}{2}\log\left(\sum_{i\in\mathcal{S}_t} v_{t,i}e^{-2h_{i,t}^2}\right) \quad , \quad \Delta(1) = \frac{-1}{2}\log\left(\sum_{i\in\mathcal{S}_t} v_{t,i}e^{-2(1-h_{i,t})^2}\right)$$

Predict a value $\widehat{y}'_t$ that satisfies the following conditions:

$$(\widehat{y}'_t)^2 \leq \Delta(0) \quad , \quad (1-\widehat{y}'_t)^2 \leq \Delta(1).$$

Update: Observing reward y_t, scale it as $y'_t = \frac{y_t - a}{\ell}$, and update the experts' weights

$$w_{t+1,i} = w_{t,i}e^{-2(y'_t - h_{i,t})^2} \tag{15}$$

Return prediction $\widehat{y}_t = \beta + \ell\widehat{y}'_t$
end for

Proposition A.1 (Theorem 4.2 in Haussler et al. 1998)**.** *For any arbitrary sequence $\left\{\left(\{f_t^i(H_t)\}_{i=1}^M, \widehat{y}_t, y_t\right)\right\}_{t=1}^T$ in which the experts' predictions $\left\{\{f_t^i(H_t)\}_{i=1}^M\right\}_{t=1}^T$ and observations $\{y_t\}_{t=1}^T$ are all bounded in $[\beta, \beta+\ell]$, the regret defined by (14) of Algorithm 3 is bounded as*

$$\mathcal{R}_{\texttt{Sq}}(T) \leq 2\ell^2 \log M.$$

Here we use the fact that $|\mathcal{S}_t| \leq M,\ \forall t \in [T]$.

B. Proofs of Section 3

In this section, we first provide a brief overview for the steps of our proof. Then, we provide the proofs of lemmas used in Section 3.

The performance analysis of the FS-SCB algorithm requires two steps. First, we control the sum of the prediction error of the agent. Second, we show how the regret is related to the prediction error of the agent, and then we bound the regret.

Step 1. To control the sum of the prediction error of the agent D_t, we need to find two upper bounds: 1) an upper bound on the prediction error of the true model i_* whose identity is unknown to the agent Q_t; 2) an upper bound on the regret caused by the online regression oracle $\mathcal{R}_{\texttt{Sq}}$

First, in Lemma 3.4, we bound the sum of the prediction error of the true model as

$$\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\widehat{\theta}_s^{i_*}\rangle-\langle\phi^{i_*}(x_s,a_s),\theta_*^{i_*}\rangle\right)^2\leq Q_t \tag{16}$$

where

$$Q_t=1+2\left(\max_{i\in[M]}\left\{\lambda_i S^2+4d\log\left(1+\frac{tL^2}{\lambda_i d}\right)\right\}\right)+32R^2\log\left(\frac{R\sqrt{8}+\sqrt{1+\max_{i\in[M]}\left\{\lambda_i S^2+4d\log\left(1+\frac{tL^2}{\lambda_i d}\right)\right\}}}{\delta}\right).$$

Next, in Lemma 3.5, we provide a high probability upper-bound on the regret caused by the online regression oracle as

$$\mathcal{R}_{\texttt{Sq}}(t)\leq 8(\log M)R^2L^2\left(G^2+\max_{i\in[M]}\{\lambda_i S^2+d\log\left(1+\frac{tL^2}{\lambda_i d}\right)\}+\log(1/\delta)\right).$$

Then, in Lemma 3.3, we show the following upper bound on the sum of prediction error of the agent:

$$\begin{aligned}D_t(\delta)\leq{}&1+2\mathcal{R}_{\texttt{Sq}}(t)+2Q_t+4R\sqrt{2(1+Q_t)\log\left(\frac{\sqrt{1+Q_t}}{\delta}\right)}\\&+32R^2\log\left(\frac{R\sqrt{8}+\sqrt{1+\mathcal{R}_{\texttt{Sq}}(t)+Q_t+2R\sqrt{2(1+Q_t)\log\left(\frac{\sqrt{1+Q_t}}{\delta}\right)}}}{\delta}\right).\end{aligned} \tag{17}$$

Step 2. First in Lemma 3.6, we show how the regret is related to the prediction error of the agent using the Azuma's inequality, i.e.,

$$\begin{aligned}\mathcal{R}_{\text{FS-SCB}}(T)\leq{}&\sqrt{2T\log(2/\delta)}+\alpha D_T(\delta)\\&+\sum_{t=1}^{T}\sum_{a\in[K]}p_t(x_t,a)\left(\langle\phi^{i_*}(x_t,a_t^*),\theta_*^{i_*}\rangle-\langle\phi^{i_*}(x_t,a),\theta_*^{i_*}\rangle-\frac{\alpha}{4}\left(\widehat{y}_t(x_t,a)-\langle\phi^{i_*}(x_t,a),\theta_*^{i_*}\rangle\right)^2\right).\end{aligned} \tag{18}$$

Then in Appendix B.5, we put everything together and complete the proof.

B.1. Proof of Lemma 3.4

At each round t, each expert $i_*\in\mathcal{I}_*$ estimates its reward parameter as

$$\widehat{\theta}_t^{i_*}=\arg\min_{\theta}\left\|(\Phi_t^{i_*})^\top\theta-Y_t\right\|_2^2+\lambda_{i_*}\left\|\theta\right\|_2^2. \tag{19}$$

Let $V_t^{\lambda_{i_*}}=\lambda_{i_*}I+\sum_{s=1}^{t-1}\phi^{i_*}(x_s,a_s)\phi^{i_*}(x_s,a_s)^\top$. From the standard least-squares analysis, we have

$$\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\widehat{\theta}_s^{i_*}\rangle-y_s\right)^2-\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\theta_*^{i_*}\rangle-y_s\right)^2\leq\lambda_{i_*}\left\|\theta_*^{i_*}\right\|_2^2+2\sum_{s=1}^{t-1}\left\langle\phi^{i_*}(x_s,a_s)^\top,(V_s^{\lambda_{i_*}})^{-1}\phi^{i_*}(x_s,a_s)\right\rangle.$$

Therefore, we can write:

$$\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\widehat{\theta}_s^{i_*}-\theta_*^{i_*}\rangle\right)^2 \le \lambda_{i_*}\left\|\theta_*^{i_*}\right\|_2^2 + 2\log\left(\frac{\det(V_t^{\lambda_{i_*}})}{\det(\lambda_{i_*}I)}\right) + 2\sum_{s=1}^{t-1}\eta_s\left(\langle\phi^{i_*}(x_s,a_s),\widehat{\theta}_s^{i_*}-\theta_*^{i_*}\rangle\right). \tag{20}$$

The last term on the RHS of (20) can be bounded using Proposition D.1 in Appendix D as

$$\left|\sum_{s=1}^{t-1}\eta_s\left(\langle\phi^{i_*}(x_s,a_s),\widehat{\theta}_s^{i_*}-\theta_*^{i_*}\rangle\right)\right| \le R\sqrt{2\left(1+\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\widehat{\theta}_s^{i_*}-\theta_*^{i_*}\rangle\right)^2\right)\log\left(\frac{1+\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\widehat{\theta}_s^{i_*}-\theta_*^{i_*}\rangle\right)^2}{\delta}\right)}. \tag{21}$$

Define $u=\sqrt{1+\sum_{k=1}^{t-1}\left(\langle\phi^{i_*}(x_k,a_k),\widehat{\theta}_k^{i_*}-\theta_*^{i_*}\rangle\right)^2}$, $v=1+\lambda_{i_*}\left\|\theta_*^{\lambda_{i_*}}\right\|_2^2+2\log\left(\frac{\det(V_t^{i_*})}{\det(\lambda_{i_*}I)}\right)$, and $w=2R\sqrt{2\log(s/\delta)}$. It is easy to see that (21) can be written in the form of $u^2\le v+uq$. Then, by applying Lemma D.5 in Appendix D, we may write $u\le\sqrt{v}+w$. Substituting for w, we can get $u\le\sqrt{v}+2R\sqrt{2\log(u/\delta)}$. Then, by Lemma D.6 in Appendix D, for $\delta\in(0,1/4]$, we have

$$u\le\sqrt{v}+4R\sqrt{\log\left(\frac{2\sqrt{2}R+\sqrt{v}}{\delta}\right)},$$

which using the inequality $(a+b)^2\le 2a^2+2b^2$, for any a and b, we can write it as

$$u^2\le 2v+32R^2\log\left(\frac{2\sqrt{2}R+\sqrt{v}}{\delta}\right).$$

Finally, we substitute u and v, and subtract 1 from both sides, and for $\delta\in(0,1/4]$, we obtain

$$\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\widehat{\theta}_s^{i_*}-\theta_*^{i_*}\rangle\right)^2 \le 1+2\lambda_{i_*}\left\|\theta_*^{i_*}\right\|_2^2+4\log\left(\frac{\det(V_t^{\lambda_{i_*}})}{\det(\lambda_{i_*}I)}\right) + 32R^2\log\left(\frac{R\sqrt{8}+\sqrt{1+\lambda\left\|\theta_*^{i_*}\right\|_2^2+2\log\left(\frac{\det(V_t^{\lambda_{i_*}})}{\det(\lambda_{i_*}I)}\right)}}{\delta}\right). \tag{22}$$

We know that $\left\|\theta_*^{i_*}\right\|_2^2\le S^2$. Moreover, by Lemma D.3 in Appendix D, we can bound the term $\log\left(\frac{\det(V_t^{\lambda_{i_*}})}{\det(\lambda_{i_*}I)}\right)$. Replacing these in (22), we may write

$$\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\widehat{\theta}_s^{i_*}-\theta_*^{i_*}\rangle\right)^2 \le 1+2\lambda_{i_*}S^2+8d\log\left(1+\frac{tL^2}{\lambda_{i_*}d}\right) + 32R^2\log\left(\frac{R\sqrt{8}+\sqrt{1+\lambda_{i_*}S^2+4d\log\left(1+\frac{tL^2}{\lambda_{i_*}d}\right)}}{\delta}\right). \tag{23}$$

Since the algorithm does not know the identity of i_*, we derive an expression for Q_t, and conclude the proof by replacing i_* with the maximum over all $i \in [M]$ in (23) as

$$\sum_{s=1}^{t-1} \left(\langle \phi^{i_*}(x_s, a_s), \widehat{\theta}_s^{i_*} - \theta_*^{i_*} \rangle\right)^2 \leq 1 + 2\left(\max_{i\in[M]} \left\{\lambda_i S^2 + 4d \log\left(1 + \frac{tL^2}{\lambda_i d}\right)\right\}\right) + 32R^2 \log\left(\frac{R\sqrt{8} + \sqrt{1 + \max_{i\in[M]}\left\{\lambda_i S^2 + 4d\log\left(1+\frac{tL^2}{\lambda_i d}\right)\right\}}}{\delta}\right) := Q_t. \tag{24}$$

B.2. Proof of Lemma 3.5

To bound the regret $\mathcal{R}_{\texttt{Sq}}(t)$ of the regression oracle `SqAlg`, similar to the proof of Lemma 4.5 in Appendix C.2, we show the reward signals and the experts' predictions are bounded with high probability. Then, we use Proposition A.1 in Appendix A to complete the proof.

From (53), according to Assumption 2.1, we have $\langle \phi^{i_*}(x, a), \theta_*^{i_*} \rangle \leq LS$, for all $x \in \mathcal{X}$, $a \in [k]$, and $i \in [M]$. Hence, with probability at least $1 - \delta$, we have

$$y_t \in \left[-\left(G + R\sqrt{2\log(2/\delta)}\right), \left(G + R\sqrt{2\log(2/\delta)}\right)\right]. \tag{25}$$

Next we bound the predictions of the experts that FS-SCB considers in its prediction. To do so, we first show an upper bound on the prediction of the any true model i_*. In particular, we can write for $t \in [T]$ and $\forall a \in [K]$:

$$\begin{aligned}
\left|\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} \rangle\right| &= \left|\langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle + \langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} - \theta_*^{i_*} \rangle\right| \\
&\overset{(a)}{\leq} \left|\langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle\right| + \left|\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} - \theta_*^{i_*} \rangle\right| \\
&\overset{(b)}{\leq} G + \left\|\phi^{i_*}(x_t, a_t)\right\|_{(V_t^{\lambda_{i_*}})^{-1}} \left(\left\|\Phi_t^{i_*}\eta_t\right\|_{(V_t^{\lambda_{i_*}})^{-1}} + \sqrt{\lambda_{i_*}} S\right),
\end{aligned} \tag{26}$$

(a) It results from a triangular inequality. **(b)** This is because of the Assumption 2.1, and the fact that the true model is linearly realizable, we can apply Theorem 2 in Abbasi-Yadkori et al. (2011). Then, we use Theorem 1 in Abbasi-Yadkori et al. (2011) and standard matrix analysis together with our assumption that $\left\|\phi^{i_*}(x_t, a_t)\right\| \leq L$, and bound the terms on the RHS of (26) with high probability as

$$\left\|\Phi_t^{i_*}\eta_t\right\|_{(V_t^{\lambda_{i_*}})^{-1}} \leq R\sqrt{2\log\left(\frac{\sqrt{\det(V_t^{\lambda_{i_*}})}}{\delta\sqrt{\det(\lambda_{i_*} I)}}\right)}, \tag{27}$$

and

$$\left\|\phi^{i_*}(x_t, a_t)\right\|_{(V_t^{\lambda_{i_*}})^{-1}} \leq \frac{\left\|\phi^{i_*}(x_t, a_t)\right\|}{\sqrt{\lambda_{\min}(V_t^{\lambda_{i_*}})}} \leq \frac{L}{\sqrt{\lambda_{i_*}}} \leq L, \tag{28}$$

where $\lambda_{\min}(V_t^{\lambda_{i_*}})$ is the smallest eigenvalue of the matrix $V_t^{\lambda_{i_*}}$. In the last step of (28), we use the fact that $\lambda_i \geq 1,\ \forall i \in [M]$. Putting Eqs. 26, 27, and 28 together, with probability at least $1 - \delta$, we have

$$\left|\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} \rangle\right| \leq G + RL\sqrt{2\log\left(\frac{\sqrt{\det(V_t^{\lambda_{i_*}})}}{\delta\sqrt{\det(\lambda_{i_*} I)}}\right)} + L\sqrt{\lambda_i} S. \tag{29}$$

Using Lemma D.3 in Appendix D, we may write (29) as

$$\left|\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} \rangle\right| \leq G + RL\sqrt{d\log\left(\frac{1 + \frac{tL^2}{\lambda_{i_*} d}}{\delta}\right)} + L\sqrt{\lambda_{i_*}} S. \tag{30}$$

FS-SCB employees this idea that at any time step $t \in [T]$, any potentially true model (i.e., linearly realizable) should have a similar bound on its prediction. To do so, the set of admissible experts, $\mathcal{S}_t$, only considers experts that have the following bound on their prediction at each time $t \in [T]$ and $\forall a \in [K]$ as:

$$\left|\langle \phi^i(x_t, a_t), \widehat{\theta}_t^i \rangle\right| \leq G + RL\sqrt{d \log\left(\frac{1 + \frac{tL^2}{\lambda_i d}}{\delta}\right)} + L\sqrt{\lambda_i} S. \tag{31}$$

If at some time step t, this bound does not hold for any expert i, then the algorithm simply eliminates that expert from the set of admissible experts, since that model is not a true model (i.e., the reward is not in the linear span of the prediction of that expert), and that expert will remain out for the rest of the game. Then, we may bound the range of the prediction of each expert $i \in \mathcal{S}_t$ at round $t \in [T]$ as

$$\langle \phi^i(x_t, a_t), \widehat{\theta}_t^i \rangle \in \left[-\left(G + RL\sqrt{d \log\left(\frac{1 + \frac{tL^2}{\lambda_i d}}{\delta}\right)} + L\sqrt{\lambda_i} S\right), \left(G + RL\sqrt{d \log\left(\frac{1 + \frac{tL^2}{\lambda_i d}}{\delta}\right)} + L\sqrt{\lambda_i} S\right)\right]. \tag{32}$$

Putting together (25) and (32), we conclude that for all rounds $t \in [T]$ and experts $i \in \mathcal{S}_t$, with probability at least $1 - \delta$, the reward y_t and the expert's predictions $f_t^i(H_t)$ are in the range $[\beta, \beta + \ell]$ for

$$\beta = -\left(G + RL\sqrt{d \log\left(\frac{1 + \frac{tL^2}{\lambda_i d}}{\delta}\right)} + L\sqrt{\lambda_i} S\right), \qquad \ell = 2\left(G + RL\sqrt{d \log\left(\frac{1 + \frac{tL^2}{\lambda_i d}}{\delta}\right)} + L\sqrt{\lambda_i} S\right). \tag{33}$$

Using Proposition A.1 in Appendix A with the bound on the observations and predictions in (33), with probability at least $1 - \delta$, we obtain the following regret bound for `SqAlg`:

$$\mathcal{R}_{\texttt{Sq}}(t) = 8R^2L^2 \log(M)\left(G^2 + \max_{i \in [M]}\left\{\lambda_i S^2 + d \log\left(1 + \frac{tL^2}{\lambda_i d}\right)\right\} + \log(1/\delta)\right), \tag{34}$$

in which we use the fact that for $a, b > 0,\ (a+b)^2 \leq 2a^2 + 2b^2$. This concludes our proof.

B.3. Proof of Lemma 3.3

Here, we bound the sum of the square loss of the oracle predictions, i.e.,

$$\sum_{s=1}^{t-1} \left(\widehat{y}_s(x_s, a_s) - \langle \phi^{i_*}(x_s, a_s), \theta_*^{i_*} \rangle\right)^2 \leq D_t(\delta). \tag{35}$$

We know that $y_t = \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle + \eta_t$. Hence we can write

$$\begin{aligned}
&\left(\widehat{y}_t(x_t, a_t) - y_t\right)^2 - \left(\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} \rangle - y_t\right)^2 = \\
&\qquad \left(\widehat{y}_t(x_t, a_t) - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle - \eta_t\right)^2 - \left(\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} \rangle - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle - \eta_t\right)^2 \\
&= \left(\widehat{y}_t(x_t, a_t) - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle\right)^2 - \left(\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} \rangle - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle\right)^2 \\
&\qquad + 2\eta_t\left(\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} \rangle - \widehat{y}_t(x_t, a_t)\right) \\
&= \left(\widehat{y}_t(x_t, a_t) - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle\right)^2 - \left(\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} \rangle - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle\right)^2 \\
&\qquad + 2\eta_t\left(\langle \phi^{i_*}(x_t, a_t), \widehat{\theta}_t^{i_*} \rangle - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle\right) + 2\eta_t\left(\langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle - \widehat{y}_t(x_t, a_t)\right).
\end{aligned} \tag{36}$$

Then, from Proposition D.1 in Appendix D, with probability at least $1 - \delta$, we have

$$\begin{aligned}
&\left|\sum_{s=1}^{t-1} \eta_s\left(\langle \phi^{i_*}(x_s, a_s), \widehat{\theta}_s^{i_*} - \theta_*^{i_*} \rangle\right)\right| \leq \\
&\qquad R\sqrt{2\left(1 + \sum_{s=1}^{t-1}\left(\langle \phi^{i_*}(x_s, a_s), \widehat{\theta}_s^{i_*} - \theta_*^{i_*} \rangle\right)^2\right) \log\left(\frac{\sqrt{1 + \sum_{s=1}^{t-1}\left(\langle \phi^{i_*}(x_s, a_s), \widehat{\theta}_s^{i_*} - \theta_*^{i_*} \rangle\right)^2}}{\delta}\right)},
\end{aligned} \tag{37}$$

and

$$\left|\sum_{s=1}^{t-1}\eta_s\big(\langle\phi^{i_*}(x_s,a_s),\theta_*^{i_*}\rangle-\widehat{y}_s(x_s,a_s)\big)\right|\leq$$
$$R\sqrt{2\left(1+\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\theta_*^{i_*}\rangle-\widehat{y}_s\right)^2\right)\log\left(\frac{\sqrt{1+\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\theta_*^{i_*}\rangle-\widehat{y}_s\right)^2}}{\delta}\right)}. \tag{38}$$

Using (37) and (38), the upper-bound $\mathcal{R}_{\text{Sq}}(t)$ from (34) in Appendix B.2, and the upper-bound Q_t on the square error of the prediction of the true model in (24) in Appendix B.1, we may write (36) as

$$\sum_{s=1}^{t-1}\left(\widehat{y}_s(x_s,a_s)-\langle\phi^{i_*}(x_s,a_s),\theta_*^{i_*}\rangle\right)^2\leq\mathcal{R}_{\text{Sq}}(t)+Q_t+2R\sqrt{2(1+Q_t)\log\left(\frac{\sqrt{1+Q_t}}{\delta}\right)}$$
$$+2R\sqrt{2\Big(1+\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(x_s,a_s),\theta_*^{i_*}\rangle-\widehat{y}_s(x_s,a_s)\right)^2\Big)\log\left(\frac{\sqrt{1+\sum_{s=1}^{t-1}\left(\langle\phi^{i_*}(a_s),\theta_*^{i_*}\rangle-\widehat{y}_s(a_s)\right)^2}}{\delta}\right)}. \tag{39}$$

Let $u=\sqrt{1+\sum_{k=1}^{t-1}\left(\widehat{y}_k(x_k,a_k)-\langle\phi^{i_*}(x_k,a_k),\theta_*^{i_*}\rangle\right)^2}$, $v=1+\mathcal{R}_{\text{Sq}}(t)+Q_t+2R\sqrt{2(1+Q_t)\log(\frac{\sqrt{1+Q_t}}{\delta})}$, and $q=2R\sqrt{2\log(s/\delta)}$. Then, following the same machinery as the one in the proof of Lemma 3.4 in Section B.1, and with the use of Lemmas D.5 and D.6, for $\delta\in(0,1/4]$, with probability at least $1-\delta$, we have

$$\sum_{s=1}^{t-1}\left(\widehat{y}_s(x_s,a_s)-\langle\phi^{i_*}(x_s,a_s),\theta_*^{i_*}\rangle\right)^2\leq 1+2\mathcal{R}_{\text{Sq}}(t)+2Q_t+4R\sqrt{2(1+Q_t)\log\left(\frac{\sqrt{1+Q_t}}{\delta}\right)}$$
$$+32R^2\log\left(\frac{R\sqrt{8}+\sqrt{1+\mathcal{R}_{\text{Sq}}(t)+Q_t+2R\sqrt{2(1+Q_t)\log\left(\frac{\sqrt{1+Q_t}}{\delta}\right)}}}{\delta}\right):=D_t(\delta), \tag{40}$$

where

$$Q_t=1+2\left(\max_{i\in[M]}\left\{\lambda_iS^2+4d\log\left(1+\frac{tL^2}{\lambda_id}\right)\right\}\right)+32R^2\log\left(\frac{R\sqrt{8}+\sqrt{1+\max_{i\in[M]}\left\{\lambda_iS^2+4d\log\left(1+\frac{tL^2}{\lambda_id}\right)\right\}}}{\delta}\right),$$

and

$$\mathcal{R}_{\text{Sq}}(t)\leq 8R^2L^2\log(M)\left(G^2+\max_{i\in[M]}\left\{\lambda_iS^2+d\log\left(1+\frac{tL^2}{\lambda_id}\right)\right\}+\log(1/\delta)\right).$$

B.4. Proof of Lemma 3.6

The inequality can be obtained using Azuma's inequality and following similar steps as in Lemma 2 of (Foster & Rakhlin, 2020). We may write the regret as

$$\mathcal{R}_{\text{FS-SCB}}(T)=\sum_{t=1}^{T}\left(\langle\phi^{i_*}(x_t,a_t^*),\theta_*^{i_*}\rangle-\langle\phi^{i_*}(x_t,a_t),\theta_*^{i_*}\rangle-\frac{\alpha}{4}\left(\widehat{y}_t(x_t,a_t)-\langle\phi^{i_*}(x_t,a_t),\theta_*^{i_*}\rangle\right)^2\right)$$
$$+\frac{\alpha}{4}\sum_{t=1}^{T}\left(\widehat{y}_t(x_t,a_t)-\langle\phi^{i_*}(x_t,a_t),\theta_*^{i_*}\rangle\right)^2. \tag{41}$$

The last term on the RHS of (41) is bounded with $D_t(\delta)$ in (40) from the result of Lemma 3.3 in Appendix B.3. Define filtration $F_{t-1} = \sigma\big((x_1, a_1, y_1), \ldots, (x_{t-1}, a_{t-1}, y_{t-1})\big)$. On the RHS of (41), the action a_t is random. We can use the Azuma's inequality and with probability at least $1-\delta$, upper-bound the first term on the RHS of (41) with its expectation counterparts using the probability distribution p_t as

$$\mathcal{R}_{\text{FS-SCB}}(T) \leq \sqrt{2T\log(2/\delta)} + \frac{\alpha}{4} D_T \tag{42}$$
$$+ \sum_{t=1}^{T} \sum_{a\in[K]} p_t(a) \left(\langle \phi^{i_*}(x_t, a_t^*), \theta_*^{i_*} \rangle - \langle \phi^{i_*}(x_t, a), \theta_*^{i_*} \rangle - \frac{\alpha}{4} \left(\widehat{y}_t(x_t, a) - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle \right)^2 \right).$$

B.5. Proof of Theorem 3.2

We first state the following lemma from (Foster & Rakhlin, 2020) to bound the first term on the RHS of (43).

Lemma B.1 (Lemma 3 in (Foster & Rakhlin, 2020))**.** *Under Assumption 2.1, for the probability distribution $p_t \in \Delta_K$ defined in the FS-SCB algorithm, we may write*

$$\sum_{a\in[K]} p_t(a) \left(\langle \phi^{i_*}(x_t, a_t^*), \theta_*^{i_*} \rangle - \langle \phi^{i_*}(x_t, a), \theta_*^{i_*} \rangle - \frac{\alpha}{4} \left(\widehat{y}_t(x_t, a) - \langle \phi^{i_*}(x_t, a_t), \theta_*^{i_*} \rangle \right)^2 \right) \leq \frac{2K}{\alpha}.$$

Putting everything together, with the choice of $\alpha = \sqrt{KT/D_T(\delta)}$, with probability at least $1-\delta$, we can show the following upper-bound on the regret of the FS-SCB algorithm:

$$\mathcal{R}_{\textit{FS-SCB}}(T) \leq 3\sqrt{KTD_T(\delta)} + \sqrt{2T\log(2/\delta)} \tag{43}$$

Here the upper-bound is of order

$$\mathcal{R}_{\textit{FS-SCB}}(T) \leq \mathcal{O}\left(\sqrt{2T\log(2/\delta)} + RLG\sqrt{KT(1+\log(M)) \max_{i\in[M]} \left\{ \lambda_i S^2 + 4d\log\left(\frac{1+\frac{TL^2}{\lambda_i d}}{\delta} \right) \right\}} \right).$$

C. Proofs of Section 4

The regret analysis of the PS-OFUL algorithms requires two steps. First, in Theorem 4.3, we show that the confidence set $\mathcal{C}_t$ is valid at each round t, i.e., for any $t, \delta > 0$, it includes the reward parameter θ_* with probability at least $1-\delta$. Second, we show how the regret is related to the valid confidence set, and then using Lemmas D.2 and D.3 complete the proof.

Step 1. The key idea for showing the validity of the confidence set $\mathcal{C}_t$ requires controlling the square prediction error of the online regression oracle $\widehat{y}_t$, i.e., upper-bounding γ_t. In Appendix C.3, we show that we can relate this distance to the sum of two terms: $\gamma_t \leq \mathcal{O}\left(U_t + \mathcal{R}_{\texttt{Sq}}(t)\right)$, and then show how we can bound each of them.

1) Bounding U_t: Lemma 4.4 shows the worst-case upper-bound on the square error of the prediction of true model i_*, given that the agent does not know the identity of the true model:

$$\sum_{s=1}^{t} \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle^2 \leq U_t \tag{44}$$

where

$$U_t \leq 3 + 8d \log\left(1 + \frac{tL^2 \max_{i\in[M]}(b_i + c_i)^2}{d}\right) + 32R^2 \log(1/\delta) \tag{45}$$

Proof. The proof is provided in Appendix C.1. □

2) Bounding $\mathcal{R}_{\texttt{Sq}}(t)$: In Lemma 4.5, we prove an upper-bound on the regret caused by the prediction oracle `SqAlg`, given our proposed expert predictions as (see Appendix C.2 for details).

$$\mathcal{R}_{\texttt{Sq}}(t) \leq 8(G+L)^2 \log(M) + 8R^2L^2 d \log(M)\log(1/\delta) + 8R^2L^2 d \log(M) \log\left(1 + \frac{tL^2 \max_{i\in[M]}(b_i + c_i)^2}{d}\right)$$

Putting these together, in Appendix C.3, we prove Theorem 4.3 that shows the validity of the confidence set $\mathcal{C}_t$.

Step 2. In Appendix C.4, we first show how regret is related to the confidence set. In particular, we show that given the validity of of the confidence set $\mathcal{C}_t$, i.e., for any $\delta \in (0, 1/4]$, with probability at least $1-\delta$, $\theta_* \in \mathcal{C}_t$, we can bound the regret as

$$\mathcal{R}_{\text{PS-OFUL}}(T) \leq 2Gd + 2\max\{1, G\}\sqrt{2dT \log\left(1 + \frac{T}{d}\right) \max_{d<t\leq T} \gamma_t(\delta)}.$$

Then, in Appendix C.5, we set $\lambda_i = \frac{1}{(b_i+c_i)^2}$, for each $i \in [M]$, and use Lemmas D.2 and D.3 to complete the proof of Theorem 4.2. Here we prove a regret upper-bound of order

$$\mathcal{O}\left(dRL\max\{1,G\}\sqrt{1+\log(M)} \times \sqrt{T\log\left(1+\frac{T}{d}\right)\log\left(\frac{1+\frac{TL^2\max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\right)}\right).$$

C.1. Proof of Lemma 4.4

At each round $s \in [T]$, each expert $i_* \in \mathcal{I}_*$ estimates its reward parameter as

$$\widehat{\theta}_s^{i_*} = \arg\min_{\theta} \left\|\Phi_s^\top \theta - Y_s\right\|^2 + \lambda_{i_*} \left\|\theta - \widehat{\mu}_{i_*}\right\|^2,$$

which is the output of a Follow-The-Regularized-Leader (FTRL) algorithm with quadratic regularizer $\left\|\theta - \widehat{\mu}_{i_*}\right\|^2$. Following the standard FTRL analysis of online regression (see e.g., Cesa-Bianchi & Lugosi 2006, Chapter 11), we have

$$\sum_{s=1}^{t}(\langle \phi_s(a_s), \widehat{\theta}_s^{i_*}\rangle - y_s)^2 - \sum_{s=1}^{t}(\langle \phi_s(a_s), \theta_*\rangle - y_s)^2 \leq \lambda_{i_*}\left\|\theta_* - \widehat{\mu}_{i_*}\right\|^2 + 2\sum_{s=1}^{t}\langle \phi_s(a_s), (V_s^{\lambda_{i_*}})^{-1}\phi_s(a_s)\rangle, \tag{46}$$

where $V_t^{\lambda_{i_*}} = \lambda_{i_*} I + \sum_{s=1}^{t-1} \phi_s(a_s)\phi_s(a_s)^\top$. We may write (46) as

$$\sum_{s=1}^{t} \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle^2 \leq \lambda_{i_*} \left\| \theta_* - \widehat{\mu}_{i_*} \right\|^2 + 2 \log \left(\frac{\det(V_t^{\lambda_{i_*}})}{\det(\lambda_{i_*} I)} \right) + 2 \sum_{s=1}^{t} \eta_s \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle. \tag{47}$$

Using Proposition D.1 in Appendix D, we may bound the last term on the RHS of (47) as

$$\left| \sum_{s=1}^{t} \eta_s \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle \right| \leq R \sqrt{2 \left(1 + \sum_{s=1}^{t} \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle^2 \right) \log \left(\frac{1 + \sum_{s=1}^{t} \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle^2}{\delta} \right)}. \tag{48}$$

It is easy to see that (47) can be written in the form $u^2 \leq v + uw$, where $u = \sqrt{1 + \sum_{s=1}^{t} \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle^2}$, $v = 1 + \lambda_i \|\theta_* - \widehat{\mu}_{i_*}\|^2 + 2 \log \left(\frac{\det(V_t^{\lambda_{i_*}})}{\det(\lambda_{i_*} I)} \right)$, and $w = 2R\sqrt{2 \log(u/\delta)}$. Then, by applying Lemma D.5 in Appendix D, we may write $u \leq \sqrt{v} + w$. Substituting for w, we can get $u \leq \sqrt{v} + 2R\sqrt{2 \log(u/\delta)}$. Then, by Lemma D.6 in Appendix D, for $\delta \in (0, 1/4]$, we have

$$u \leq \sqrt{v} + 4R \sqrt{\log \left(\frac{2\sqrt{2}R + \sqrt{v}}{\delta} \right)},$$

which using the inequality $(a+b)^2 \leq 2a^2 + 2b^2$, for any a and b, we can write it as

$$u^2 \leq 2v + 32R^2 \log \left(\frac{2\sqrt{2}R + \sqrt{v}}{\delta} \right).$$

Finally, we substitute u and v, and subtract 1 from both sides, and for $\delta \in (0, 1/4]$, we obtain

$$\begin{aligned} \sum_{s=1}^{t} \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle^2 \leq 1 + 2\lambda_{i_*} \left\| \theta_* - \widehat{\mu}_{i_*} \right\|^2 + 4 \log \left(\frac{\det(V_t^{\lambda_{i_*}})}{\det(\lambda_{i_*} I)} \right) \\ + 32R^2 \log \left(\frac{2\sqrt{2}R + \sqrt{1 + \lambda_{i_*} \left\| \theta_* - \widehat{\mu}_{i_*} \right\|^2 + 2 \log \left(\frac{\det(V_t^{\lambda_{i_*}})}{\det(\lambda_{i_*} I)} \right)}}{\delta} \right). \end{aligned} \tag{49}$$

We know $\|\theta_* - \widehat{\mu}_{i_*}\|^2 \leq (b_{i_*} + c_{i_*})^2$. Moreover, by Lemma D.3 in Appendix D, we can bound the term $\log \left(\frac{\det(V_t^{\lambda_{i_*}})}{\det(\lambda_{i_*} I)} \right)$. Replacing these terms in (49), we have

$$\begin{aligned} \sum_{s=1}^{t} \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle^2 \leq 1 + 2\lambda_{i_*} (b_{i_*} + c_{i_*})^2 + 8d \log \left(1 + \frac{tL^2}{d\lambda_{i_*}} \right) \\ + 32R^2 \log \left(\frac{2\sqrt{2}R + \sqrt{1 + \lambda_{i_*} (b_{i_*} + c_{i_*})^2 + 4d \log(1 + \frac{tL^2}{d\lambda_{i_*}})}}{\delta} \right). \end{aligned} \tag{50}$$

Setting $\lambda_{i_*} = \frac{1}{(b_{i_*} + c_{i_*})^2}$, as used by the PS-OFUL algorithm, we obtain

$$\begin{aligned} \sum_{s=1}^{t} \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle^2 \leq 3 + 8d \log \left(1 + \frac{tL^2 (b_{i_*} + c_{i_*})^2}{d} \right) \\ + 32R^2 \log \left(\frac{2\sqrt{2}R + \sqrt{2 + 4d \log \left(1 + \frac{tL^2 (b_{i_*} + c_{i_*})^2}{d} \right)}}{\delta} \right). \end{aligned} \tag{51}$$

Since the algorithm does not know the identity of i_*, we derive an expression for U_t and conclude the proof by replacing i_* with the maximum over all $i \in [M]$ in (51), as

$$\sum_{s=1}^{t} \langle \phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_* \rangle^2 \leq 3 + 8d \log \left(1 + \frac{tL^2 \max_{i\in[M]}(b_i + c_i)^2}{d}\right) + 32R^2 \log \left(\frac{2\sqrt{2}R + \sqrt{2 + 4d \log\left(1 + \frac{tL^2 \max_{i\in[M]}(b_i+c_i)^2}{d}\right)}}{\delta} \right) := U_t. \tag{52}$$

C.2. Proof of Lemma 4.5

To obtain a high probability bound on the regret $\mathcal{R}_{\texttt{Sq}}(t)$ of the regression oracle `SqAlg`, we first show that the inputs to the regression oracle, i.e., reward signals $y_t = \phi_t(a_t) + \eta_t$ and the experts' predictions $f_t^i(H_t) = \langle \phi_t(a_t), \widehat{\theta}_t^i \rangle$ are all bounded with high probability. We then use Proposition A.1 in Appendix A to complete the proof.

Since each noise η_t is R-sub-Gaussian, from Lemma D.4 in Appendix D, with probability at least $1-\delta$, we have that $|\eta_t| \leq R\sqrt{2\log(2/\delta)}$. We also have from Assumption 2.2 that for each context and each action $a \in \bigcup_{t=1}^{T} \mathcal{A}_t$, their mean reward $|\langle \phi_t(a), \theta_* \rangle| \leq G$. Thus, by the triangular inequality, with probability at least $1-\delta$, we obtain

$$y_t \in \left[-\left(G + R\sqrt{2\log(2/\delta)}\right), \left(G + R\sqrt{2\log(2/\delta)}\right) \right]. \tag{53}$$

Next we bound the prediction of the experts that PS-OFUL considers in its prediction. To do so, we employ the same idea as we mentioned in the proof of Lemma 3.5 in Appendix B.2, where we first show an upper bound on the prediction of the any true model i_*. In particular, we can write for any time $t \in [T]$:

$$\begin{aligned}
\left| \langle \phi_t(a_t), \widehat{\theta}_t^{i_*} \rangle \right| &= \left| \langle \phi_t(a_t), \theta_* \rangle + \langle \phi_t(a_t), \widehat{\theta}_t^{i_*} - \theta_* \rangle \right| \\
&\overset{(a)}{\leq} |\langle \phi_t(a_t), \theta_* \rangle| + \left| \langle \phi_t(a_t), \widehat{\theta}_t^{i_*} - \theta_* \rangle \right| \\
&\overset{(b)}{\leq} G + \|\phi_t(a_t)\|_{(V_t^{\lambda_i})^{-1}} \left(\|\Phi_t \eta_t\|_{(V_t^{\lambda_i})^{-1}} + \sqrt{\lambda_{i_*}} \|\widehat{\mu}_{i_*} - \theta_*\| \right) \\
&\overset{(c)}{\leq} G + RL\sqrt{d \log\left(\frac{1 + \frac{tL^2}{\lambda_i d}}{\delta}\right)} + L\sqrt{\lambda_i}(b_i + c_i)
\end{aligned} \tag{54}$$

(a) It results from triangular inequality. **(b)** This comes from the Assumption 2.2 as well as Theorem 1 in Abbasi-Yadkori et al. (2011). **(c)** This is because of the Theorem 2 in Abbasi-Yadkori et al. (2011) and the fact that i_* is the true model and hence $\theta_* \in B(\widehat{\mu}_{i_*}, b_{i_*})$. Thus, we can have $\|\widehat{\mu}_{i_*} - \theta_*\| \leq (b_i + c_i)$. PS-OFUL employees this idea that at any time step, any potentially true model should have a similar bound on its prediction. This is being enforced by the set of admissible expert, $\mathcal{S}_t$, where it only considers experts that have the following bound on their prediction at each time $t \in [T]$ as:

$$\left| \langle \phi_t(a_t), \widehat{\theta}_t^{i} \rangle \right| \leq G + RL\sqrt{d \log\left(\frac{1 + \frac{tL^2}{\lambda_i d}}{\delta}\right)} + L\sqrt{\lambda_i}(b_i + c_i). \tag{55}$$

If at some time step t, this bound does not hold for any expert i, then the algorithm simply eliminates that expert from the set of admissible experts, since that model is not a true model (i.e., the reward does not belong to the ball of that model), and that expert will remain out for the rest of the game.

Setting $\lambda_i = \frac{1}{(b_i+c_i)^2}$ in (55), we can bound the prediction of each expert $i \in \mathcal{S}_t$ at round $t \in [T]$ as

$$\langle \phi_t(a_t), \widehat{\theta}_t^{i} \rangle \in \left[-\left(G + L + RL\sqrt{d \log\left(\frac{1 + \frac{tL^2 \max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\right)}\right), \left(G + L + RL\sqrt{d \log\left(\frac{1 + \frac{tL^2 \max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\right)}\right) \right]. \tag{56}$$

Putting together (53) and (56), we conclude that for all rounds $t \in [T]$ and experts $i \in \mathcal{S}_T$, with probability at least $1-\delta$, the rewards y_t and the experts' predictions $f_t^i(H_t)$ are in the range $[\beta, \beta+\ell]$ for

$$\begin{aligned}\beta &= -\left(G + L + RL\sqrt{d\log\left(\frac{1+\frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\right)}\right),\\ \ell &= 2\left(G + L + RL\sqrt{d\log\left(\frac{1+\frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\right)}\right).\end{aligned} \tag{57}$$

Using Proposition A.1 in Appendix A with the bound on the observations and predictions in (57), with probability at least $1-\delta$, we obtain the following regret bound for `SqAlg`:

$$\mathcal{R}_{\text{Sq}}(t) \le 8(\log M)\left((G+L)^2 + R^2L^2 d\log\left(\frac{1+\frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\right)\right), \tag{58}$$

in which we use the fact that for $a, b > 0,\ (a+b)^2 \le 2a^2 + 2b^2$. This concludes our proof.

C.3. Proof of Theorem 4.3

In order to fully specify the confidence set $\mathcal{C}_t$ and prove its validity, i.e., $\theta_* \in \mathbb{P}(\theta_* \in \mathcal{C}_t) \ge 1-\delta$, we should find a high probability upper-bound $\gamma_t(\delta)$ for the sum of the square loss of the oracle predictions, i.e.,

$$\sum_{s=1}^{t}\left(\widehat{y}_s - \langle\phi_s(a_s), \theta_*\rangle\right)^2 \le \gamma_t(\delta).$$

Let $z_s = (\widehat{y}_s - y_s)^2 - (\langle\phi_s(a_s), \widehat{\theta}_s^{i_*}\rangle - y_s)^2$, where $i_* \in \mathcal{I}_*$ is the index of a ball that contains θ_*. Since $y_s = \langle\phi_s(a_s), \theta_*\rangle + \eta_s$, we may write

$$\begin{aligned}z_s &= (\widehat{y}_s - \langle\phi_s(a_s), \theta_*\rangle - \eta_s)^2 - (\langle\phi_s(a_s), \widehat{\theta}_s^{i_*}\rangle - \langle\phi_s(a_s), \theta_*\rangle - \eta_s)^2\\ &= (\widehat{y}_s - \langle\phi_s(a_s), \theta_*\rangle)^2 - (\langle\phi_s(a_s), \widehat{\theta}_s^{i_*}\rangle - \langle\phi_s(a_s), \theta_*\rangle)^2 + 2\eta_s(\langle\phi_s(a_s), \widehat{\theta}_s^{i_*}\rangle - \widehat{y}_s).\end{aligned}$$

Since $\sum_{s=1}^t z_s \le \mathcal{R}_{\text{Sq}}(t)$, where $\mathcal{R}_{\text{Sq}}(t)$ is the regret of the regression oracle at round t, we have

$$\sum_{s=1}^{t}(\widehat{y}_s - \langle\phi_s(a_s), \theta_*\rangle)^2 \le \mathcal{R}_{\text{Sq}}(t) + \sum_{s=1}^{t}(\langle\phi_s(a_s), \widehat{\theta}_s^{i_*}\rangle - \langle\phi_s(a_s), \theta_*\rangle)^2 + 2\sum_{s=1}^{t}\eta_s(\langle\phi_s(a_s), \widehat{\theta}_s^{i_*}\rangle - \widehat{y}_s). \tag{59}$$

From the definition of U_t in (9), we may upper-bound $\sum_{s=1}^t(\langle\phi_s(a_s), \widehat{\theta}_s^{i_*}\rangle - \langle\phi_s(a_s), \theta_*\rangle)^2$ with U_t and write (59) as

$$\begin{aligned}\sum_{s=1}^{t}(\widehat{y}_s - \langle\phi_s(a_s), \theta_*\rangle)^2 &\le \mathcal{R}_{\text{Sq}}(t) + U_t + 2\sum_{s=1}^{t}\eta_s(\langle\phi_s(a_s), \widehat{\theta}_s^{i_*}\rangle - \widehat{y}_s)\\ &\le \mathcal{R}_{\text{Sq}}(t) + U_t + 2\sum_{s=1}^{t}\eta_s\langle\phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_*\rangle + 2\sum_{s=1}^{t}\eta_s(\langle\phi_s(a_s), \theta_*\rangle - \widehat{y}_s).\end{aligned} \tag{60}$$

Then, from Proposition D.1 in Appendix D, with probability at least $1-\delta$, we have

$$\begin{aligned}&\left|\sum_{s=1}^{t}\eta_s\langle\phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_*\rangle\right| \le\\ &R\sqrt{2\left(1+\sum_{s=1}^{t}\langle\phi_s(a_s), \widehat{\theta}_s^{i_*} - \theta_*\rangle^2\right)\log\left(\frac{\sqrt{1+\sum_{s=1}^{t}\langle\phi_s(a_k), \widehat{\theta}_s^{i_*} - \theta_*\rangle^2}}{\delta}\right)},\end{aligned} \tag{61}$$

and

$$\left|\sum_{s=1}^{t}\eta_s(\langle\phi_s(a_s),\theta_*\rangle-\widehat{y}_s)\right|\leq \tag{62}$$
$$R\sqrt{2\left(1+\sum_{s=1}^{t}\left(\langle\phi_s(a_s),\theta_*\rangle-\widehat{y}_s\right)^2\right)\log\left(\frac{\sqrt{1+\sum_{s=1}^{t}\left(\langle\phi_s(a_s),\theta_*\rangle-\widehat{y}_s\right)^2}}{\delta}\right)}.$$

Using (61) and (62), we may write (60) as

$$\sum_{s=1}^{t}\left(\widehat{y}_s-\langle\phi_s(a_s),\theta_*\rangle\right)^2\leq\mathcal{R}_{\text{Sq}}(t)+U_t+2R\sqrt{2(1+U_t)\log\left(\sqrt{1+U_t}/\delta\right)}$$
$$+R\sqrt{8\left(1+\sum_{s=1}^{t}\left(\widehat{y}_s-\langle\phi^s(a_s),\theta_*\rangle\right)^2\right)\log\left(\frac{\sqrt{1+\sum_{s=1}^{t}\left(\widehat{y}_s-\langle\phi_s(a_s),\theta_*\rangle\right)^2}}{\delta}\right)}. \tag{63}$$

It is easy to see that (63) can be written in the form $u^2\leq v+uw$, where $u=\sqrt{1+\sum_{s=1}^{t}(\langle\phi_s(a_s),\theta_*\rangle-\widehat{y}_s)^2}$, $v=1+\mathcal{R}_{\text{Sq}}(t)+U_t+2R\sqrt{2(1+U_t)\log(\frac{\sqrt{1+U_t}}{\delta})}$, and $w=R\sqrt{8\log(u/\delta)}$. Then, by applying Lemma D.5 in Appendix D, we may write $u\leq w+\sqrt{v}$. Substituting for w, we can get $u\leq\sqrt{v}+R\sqrt{8\log(u/\delta)}$. Then, by Lemma D.6 in Appendix D, for $\delta\in(0,1/4]$, we have

$$u\leq\sqrt{v}+4R\sqrt{\log\left(\frac{R\sqrt{8}+\sqrt{v}}{\delta}\right)},$$

which using the inequality $(a+b)^2\leq 2a^2+2b^2$, for any a and b, we can write it as

$$u^2\leq 2v^2+32R^2\log\left(\frac{R\sqrt{8}+\sqrt{v}}{\delta}\right).$$

Finally, we substitute u and v, and subtract 1 from both sides, and for $\delta\in(0,1/4]$, we obtain

$$\sum_{s=1}^{t}\left(\widehat{y}_s-\langle\phi_s(a_s),\theta_*\rangle\right)^2\leq 1+2\mathcal{R}_{\text{Sq}}(t)+2U_t+4R\sqrt{2(1+U_t)\log\left(\sqrt{1+U_t}/\delta\right)}$$
$$+32R^2\log\left(\frac{R\sqrt{8}+\sqrt{1+\mathcal{R}_{\text{Sq}}(t)+U_t+2R\sqrt{2(1+U_t)\log\left(\sqrt{1+U_t}/\delta\right)}}}{\delta}\right):=\gamma_t(\delta). \tag{64}$$

Eq. 64 shows that for $\delta\in(0,1/4]$, with probability at least $1-\delta$, we have $\theta^*\in C_t$, which completes the proof of the validity of the confidence set $\mathcal{C}_t$.

We can now fully specify $\mathcal{C}_t$ by plugging U_t from (52) (see Appendix C.1) and $\mathcal{R}_{\text{Sq}}(t)$ from (58) (see Appendix C.2)

into (64), and write $\gamma_t(\delta)$ as

$$\gamma_t(\delta) := 1 + 2\mathcal{R}_{\text{Sq}}(t) + 2U_t + 4R\sqrt{2(1+U_t)\log\left(\sqrt{1+U_t}/\delta\right)} + 32R^2\log\left(\frac{R\sqrt{8} + \sqrt{1+\mathcal{R}_{\text{Sq}}(t) + U_t + 2R\sqrt{2(1+U_t)\log\left(\sqrt{1+U_t}/\delta\right)}}}{\delta}\right),$$

where (65)

$$U_t = 3 + 8d\log\left(1 + \frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}\right) + 32R^2\log\left(\frac{2\sqrt{2}R + \sqrt{2 + 4d\log\left(1+\frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}\right)}}{\delta}\right),$$

$$\mathcal{R}_{\text{Sq}}(t) = 8\log(M)\left(G^2 + L^2 + 2GL + R^2L^2d\log\left(\frac{1+\frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\right)\right),$$

which concludes the proof.

A closer look at U_t and $\mathcal{R}_{\text{Sq}}(t)$, the two main terms in the definition of $\gamma_t(\delta)$, we may write them in terms of the dominant terms as

$$\begin{aligned} U_t &\approx \overbrace{3 + 16R^2\log(2)}^{C_1} + 8d\log\left(1 + \frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}\right) + 32R^2\log(1/\delta) \\ &\quad + 32R^2\log\left(1 + 2R + d\log\left(1 + \frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}\right)\right) \\ &\approx C_1 + 32R^2\log(1/\delta) + 8d\log\left(1 + \frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}\right), \end{aligned} \tag{66}$$

and

$$\begin{aligned} \mathcal{R}_{\text{Sq}}(t) &= \overbrace{8(G+L)^2\log(M)}^{C_2} + 8R^2L^2d\log(M)\log(1/\delta) + 8R^2L^2d\log(M)\log\left(1 + \frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}\right) \\ &= C_2 + 8R^2L^2d\log(M)\log(1/\delta) + 8R^2L^2d\log(M)\log\left(1 + \frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}\right). \end{aligned} \tag{67}$$

Using (66) and (67), we may write $\gamma_t(\delta)$ in terms of the dominant terms as

$$\begin{aligned} \gamma_t(\delta) &\approx 1 + 2C_1 + 2C_2 + 16R^2\left(4 + L^2d\log(M)\right)\log(1/\delta) \\ &\quad + 16d\left(1 + R^2L^2\log(M)\right)\log\left(1 + \frac{tL^2\max_{i\in[M]}(b_i+c_i)^2}{d}\right). \end{aligned} \tag{68}$$

C.4. Proof of Lemma 4.6

In Theorem 4.3, we proved that at each round, with probability at least $1-\delta$, the true reward parameter θ_* belongs to the confidence set $\mathcal{C}_t$ of the PS-OFUL algorithm. Here, we show how the regret of PS-OFUL is related to the radius $\gamma_t(\delta)$ of this confidence set.

Here we assume that at the first d rounds, the algorithm plays actions whose features are of the form $\phi_i(a_i) = Le_i,\ \forall i \in [d]$, where $e_i = [0, \dots, 1, \dots, 0]$ is a d-dimensional vector whose elements are all 0, except a 1 at the i^{th} position. In this case,

we can define a matrix V_t as

$$V_t = \sum_{s=1}^{t-1} \phi_s(a_s)^\top \phi_s(a_s) = L^2 I + \sum_{s=d+1}^{t-1} \phi_t(a_t)^\top \phi_t(a_t), \tag{69}$$

and use it to rewrite the confidence set as

$$\mathcal{C}_{t-1} = \big\{\theta \in \mathbb{R}^d : (\theta - \widehat{\theta}_t) V_t (\theta - \widehat{\theta}_t) + \sum_{s=1}^{t-1} \big(\widehat{y}_s - \langle \phi_s(a_s), \widehat{\theta}_t \rangle\big)^2 \le \gamma_t(\delta)\big\}, \tag{70}$$

where $\widehat{\theta}_t = \operatorname{argmin}_{\theta \in \mathbb{R}^d} \sum_{s=1}^{t-1} \big(\widehat{y}_s - \langle \phi_s(a_s), \theta \rangle\big)^2$. The confidence set $\mathcal{C}_t$ in (70) is contained in a larger ellipsoid

$$\mathcal{C}_{t-1} \subseteq \big\{\theta \in \mathbb{R}^d : (\theta - \widehat{\theta}_t) V_t (\theta - \widehat{\theta}_t) \le \gamma_t(\delta)\big\} = \big\{\theta \in \mathbb{R}^d : \|\theta - \widehat{\theta}_t\|_{V_t}^2 \le \gamma_t(\delta)\big\}. \tag{71}$$

Given $(a_t, \widetilde{\theta}_t) = \arg\max_{a \in \mathcal{A}_t} \max_{\theta \in \mathcal{C}_{t-1}} \langle \phi_t(a), \theta \rangle$ are the action and parameter resulted from solving the optimization problem at round t of the PS-OFUL algorithm, we may write

$$\begin{aligned}
\langle \phi_t(a_t^*), \theta_* \rangle - \langle \phi_t(a_t), \theta_* \rangle &\le \langle \phi_t(a_t), \widetilde{\theta}_t \rangle - \langle \phi_t(a_t), \theta_* \rangle \\
&= \langle \phi_t(a_t), \widetilde{\theta}_t - \widehat{\theta}_t \rangle + \langle \phi_t(a_t), \widehat{\theta}_t - \theta_* \rangle \\
&\le \|\phi_t(a_t)\|_{V_t^{-1}} \|\widetilde{\theta}_t - \widehat{\theta}_t\|_{V_t} + \|\phi_t(a_t)\|_{V_t^{-1}} \|\widehat{\theta}_t - \theta_*\|_{V_t} \\
&\le 2\sqrt{\gamma_t(\delta)}\, \|\phi_t(a_t)\|_{V_t^{-1}} \qquad \text{(because } \theta_*, \widetilde{\theta}_t \in \mathcal{C}_{t-1}\text{)}.
\end{aligned} \tag{72}$$

Since $\forall a \in \bigcup_{t=1}^T \mathcal{A}_t$, we assume that $|\langle \phi(a), \theta_* \rangle| \le G$, we can upper-bound the instantaneous regret in (72) as

$$\langle \phi_t(a_t^*), \theta_* \rangle - \langle \phi_t(a_t), \theta_* \rangle \le 2 \min\big\{G, \sqrt{\gamma_t(\delta)}\, \|\phi_t(a_t)\|_{V_t^{-1}}\big\}. \tag{73}$$

Using (73), we can bound the transfer-regret of PS-OFUL as

$$\begin{aligned}
\mathcal{R}_{\text{PS-OFUL}}(T) = \sum_{t=1}^T \langle \phi_t(a_t^*) - \phi_t(a_t), \theta_* \rangle &\le 2Gd + \sum_{t=d+1}^T \langle \phi_t(a_t^*) - \phi_t(a_t), \theta_* \rangle \\
&\le 2Gd + 2 \sum_{t=d+1}^T \min\{G, \sqrt{\gamma_t(\delta)}\, \|\phi_t(a_t)\|_{V_t^{-1}}\} \\
&\le 2Gd + 2 \sum_{t=d+1}^T \sqrt{\gamma_t(\delta)}\, \min\{G, \|\phi_t(a_t)\|_{V_t^{-1}}\} \qquad \text{(since } \gamma_t(\delta) \ge 1\text{)} \\
&\le 2Gd + 2 \Big(\max_{d<t\le T} \sqrt{\gamma_t(\delta)}\Big) \sum_{t=d+1}^T \min\{G, \|\phi_t(a_t)\|_{V_t^{-1}}\} \\
&\le 2Gd + 2 \Big(\max_{d<t\le T} \sqrt{\gamma_t(\delta)}\Big) (\max\{1, G\}) \sum_{t=d+1}^T \min\{1, \|\phi_t(a_t)\|_{V_t^{-1}}\} \\
&\le 2Gd + 2 \Big(\max_{d<t\le T} \sqrt{\gamma_t(\delta)}\Big) (\max\{1, G\}) \sqrt{T \sum_{t=d+1}^T \min\{1, \|\phi_t(a_t)\|_{V_t^{-1}}^2\}} \\
&\le 2Gd + 2 \Big(\max_{d<t\le T} \sqrt{\gamma_t(\delta)}\Big) (\max\{1, G\}) \sqrt{2T \log\Big(\frac{\det(V_T)}{\det(V_d)}\Big)},
\end{aligned} \tag{74}$$

where the last inequality follows from Lemma D.2 in Appendix D. Then, using Lemma D.3 in Appendix D, we can bound $\det(V_T) \le \Big(L^2 + \frac{TL^2}{d}\Big)^d$ and $\det(V_d) = L^{2d}$. Hence, we may write (74) as

$$\mathcal{R}_{\text{PS-OFUL}}(T) \le 2Gd + 2\max\{1, G\} \sqrt{2dT \log\Big(1 + \frac{T}{d}\Big) \max_{d<t\le T} \gamma_t(\delta)}. \tag{75}$$

C.5. Proof of Theorem 4.2

If we substitute $\gamma_t(\delta)$ from (68) in the regret bound (75), we may write it (in terms of the dominant terms) as

$$\begin{aligned}\mathcal{R}_{\text{PS-OFUL}}(T) &\leq 2Gd + 2\sqrt{2}\max\{1,G\}\sqrt{dT\log\left(1+\frac{T}{d}\right)} \\ &\times \sqrt{C_3 + 16R^2\big(4+L^2 d\log(M)\big)\log(1/\delta) + 16d\big(1+R^2L^2\log(M)\big)\log\left(1+\frac{TL^2\max_{i\in[M]}(b_i+c_i)^2}{d}\right)} \\ &= \mathcal{O}\left(dRL\max\{1,G\}\sqrt{1+\log(M)} \times \sqrt{T\log\left(1+\frac{T}{d}\right)\log\left(\frac{1+\frac{TL^2\max_{i\in[M]}(b_i+c_i)^2}{d}}{\delta}\right)}\right), \qquad (76)\end{aligned}$$

where $C_3 = 1 + 2C_1 + 2C_2$, and hence $C_3 = 7 + 32R^2\log(2) + 16(G+L)^2\log(M)$.

D. Auxiliary Tools

Here we report auxiliary results that we use in our proofs in other appendices.

We start with stating Theorem 7 in (Abbasi-Yadkori et al., 2012), which is the self-normalized martingale tail inequality for the scalar random variables.

Proposition D.1 (Self-normalized bound for martingales). *Let $\{F_t\}_{t=1}^{\infty}$ be a filtration. Let τ be a stopping time w.r.t to the filtration $\{F_t\}_{t=1}^{\infty}$, i.e., the event $\{\tau \leq t\}$ belongs to F_{t+1}. Let $\{Z_t\}_{t=1}^{\infty}$ be a sequence of real-valued variables such that Z_t is F_t-measurable. Let $\{\eta_t\}_{t=1}^{\infty}$ be a sequence of real-valued random variables such that η_t is F_{t+1} measurable and is conditionally R-sub-Gaussian. Then, for any $\delta > 0$, with probability at least $1-\delta$,*

$$\left\|\sum_{t=1}^{\tau} \eta_t Z_t\right\| \leq R\sqrt{2\left(1+\sum_{t=1}^{\tau} Z_t^2\right)\log\left(\frac{\sqrt{1+\sum_{t=1}^{\tau} Z_t^2}}{\delta}\right)}.$$

Next, we state a direct application of Lemma 11 in (Abbasi-Yadkori et al., 2011) that bounds the cumulative sum of $\sum_{s=1}^{t-1} \|\phi_s(a_s)\|^2_{V_s^{-1}}$ which plays an important role in most of the proofs for linear bandits problems.

Lemma D.2. *Let $\lambda > 0$ and $V_t = \lambda I + \sum_{s=1}^{t-1} \phi_s(a_s)\phi_s^{\top}(a_s)$. If for all $a \in \cup_{s=1}^{t-1}\mathcal{A}_s$, we have $\|\phi_s(a)\|_2 \leq L$, then we may write*

$$\sum_{s=1}^{t-1} \min\{1, \|\phi_s(a_s)\|^2_{V_s^{-1}}\} \leq 2\log\left(\frac{\det(V_t)}{\det(\lambda I)}\right).$$

Next, we present a determinant-trace inequality matrix result.

Lemma D.3 (Determinant-Trace Inequality). *Suppose $X_1, \ldots, X_{t-1} \in \mathbb{R}^d$, and for any $1 \leq s \leq t-1$, we have $\|X_s\|_2 \leq L$. Let $V_t = \lambda I + \sum_{s=1}^{t-1} X_s X_s^{\top}$, for some $\lambda > 0$. Then we have*

$$\det(V_t) \leq \left(\lambda + \frac{tL^2}{d}\right)^d.$$

Proof. Let $\alpha_1, \ldots, \alpha_d$ be the eigenvalues of V_t. Since V_t is positive definite, its eigenvalues are positive. Also not that $\det(V_t) = \Pi_{s=1}^d \alpha_s$ and $trace(V_t) = \sum_{s=1}^d \alpha_s$. By arithmetic-geometric means inequality we have

$$\sqrt[d]{\alpha_1 \ldots \alpha_d} \leq \frac{\alpha_1 + \cdots + \alpha_d}{d}.$$

Therefore, $\det(V_t) \leq \left(\frac{trace(V_t)}{d}\right)^d$. It suffices to upper-bound the trace of V_t as

$$trace(V_t) = trace(\lambda I) + \sum_{s=1}^{t-1} trace(X_s X_s^{\top}) = d\lambda + \sum_{s=1}^{t-1} \|X_s\|_2^2 \leq d\lambda + tL^2,$$

and the result follows. □

Next, we state a bound on the absolute value of the R-sub-Gaussian random variable.

Lemma D.4. *Let $\{F_t\}_{t=1}^{\infty}$ be a filtration. Let $\{\eta\}_{t=1}^{\infty}$ be a real-valued stochastic process such that η_t is F_t-measurable and η_t is conditionally R-sub-Gaussian for some $R > 0$, i.e.,*

$$\forall \lambda \in \mathbb{R}, \;\; \mathbb{E}\left[\eta_t | F_t\right] = 0, \;\; \mathbb{E}\left[e^{\lambda \eta_t} | F_t\right] \leq \exp\left(\frac{\lambda^2 R^2}{2}\right).$$

Then, condition on filtration F_t, with probability at least $1-\delta$, we have $|\eta_t| \leq R\sqrt{2\log(2/\delta)}$.

Proof. Let $\lambda > 0$. Then,

$$\begin{aligned}\mathbb{P}(\eta_t \geq k|F_t) &= \mathbb{P}(e^{\lambda\eta_t} \geq e^{\lambda k}|F_t) \leq e^{-\lambda k}\,\mathbb{E}[e^{\lambda\eta_t}|F_t] \qquad \text{(by Markov's inequality)} \\ &\leq e^{-\lambda k} e^{\frac{\lambda^2 R^2}{2}} = \exp\left(-\lambda k + \frac{\lambda^2 R^2}{2}\right). \end{aligned} \tag{77}$$

Optimizing for λ, and thus, selecting $\lambda = \frac{k}{R^2}$, we conclude that

$$\mathbb{P}(\eta_t \geq k|F_t) \leq e^{-\frac{k^2}{2R^2}}.$$

Repeating this argument for $-\eta_t$, we also obtain $\mathbb{P}(\eta_t \leq -k|F_t) \leq e^{-\frac{k^2}{2R^2}}$. Combining these two bounds, we can conclude that

$$\mathbb{P}(|\eta_t| \geq k|F_t) \leq 2e^{-\frac{k^2}{2R^2}}. \tag{78}$$

From (78), with the choice of $\delta = 2e^{-\frac{k^2}{2R^2}}$, and thus $k = R\sqrt{2\log(2/\delta)}$, completes the proof. □

Then, we state a square-root trick for positive numbers.

Lemma D.5. *Let $a, b > 0$. If $z^2 \leq a + bz$, then $z \leq \sqrt{a} + b$.*

Proof. Let $q(z) = z^2 - bz - a$. We can rewrite the condition $z^2 \leq a + bz$ as $q(z) \leq 0$. Then we know that the quadratic polynomial $q(z)$ has the following two roots

$$z_1^* = \frac{b + \sqrt{b^2 + 4a}}{2} \quad z_2^* = \frac{b - \sqrt{b^2 + 4a}}{2}.$$

Then, we know that the condition $q(z) \leq 0$, implies that $\min\{z_1^*, z_2^*\} \leq z \leq \max\{z_1^*, z_2^*\}$. Therefore, for positive numbers a, b, we get

$$z \leq \max\{z_1^*, z_2^*\} = \frac{b + \sqrt{b^2 + 4a}}{2} \leq b + \sqrt{a},$$

where for the last inequality, we use the fact that for $u, v > 0$, $\sqrt{u + v} \leq \sqrt{u} + \sqrt{v}$. □

Next, we restate a simple logarithmic trick from (Abbasi-Yadkori et al., 2012).

Lemma D.6 (Proposition 10 in Abbasi-Yadkori et al. 2012)**.** *Let $c \geq 1$, $q > 0$, $\delta \in (0, 1/4]$. If $s \geq 1$ and $s \leq c + q\sqrt{\log(s/\delta)}$, then we have $s \leq c + q\sqrt{2\log(\frac{c+q}{\delta})}$.*

E. Relation to Latent Bandits

In this section, we informally show that if the goal in latent bandits is to have a better scaling with the number of actions K (e.g., the number of actions K is much larger than the number of latent states M), we can use a different bandit model selection strategy, called *regret balancing* (Abbasi-Yadkori et al., 2020; Pacchiano et al., 2020a;b; Cutkosky et al., 2021) to obtain an improved regret that scales as $\min\{\varepsilon T + \sqrt{MT}, \sqrt{KMT}\}$. This rate is the best of the regret of PS-OFUL, which scales as $\sqrt{KT}$, and the regret of the latent bandit algorithm of Hong et al. (2020), which scales as $\varepsilon T + \sqrt{MT}$.

In regret balancing, in each round, the model selection strategy chooses one of M base algorithms. We denote by $N_{i,t}$, the number of times that the base algorithm i has been selected up to round t, and by $R_{i,t}$, the cumulative rewards of this base algorithm during these $N_{i,t}$ rounds. Given a reference regret bound $U : [T] \to \mathbb{R}$, in each round $t \in [T]$, the algorithm first finds the optimistic base algorithm I_t and its value b_t, i.e.,

$$I_t = \operatorname*{argmax}_{i \in [M]} \frac{R_{i,t}}{N_{i,t}} + \frac{U(N_{i,t})}{N_{i,t}} \qquad , \qquad b_t = \frac{R_{I_t,t}}{N_{I_t,t}} + \frac{U(N_{I_t,t})}{N_{I_t,t}} \ , \tag{79}$$

and then takes the action recommended by I_t and uses its observed reward to update the base algorithm I_t.

We can apply regret balancing to the problem of latent bandits in the following way. We consider $M+1$ base algorithms: one that plays UCB, and M, each corresponds to a latent value and always plays the greedy action of that latent model (which is guaranteed to be ε-accurate by assumption). If the regret balancing strategy selects the UCB base algorithm in all rounds, it would suffer the regret $\sqrt{Kt} + \sqrt{t}$, and if it selects the optimal base algorithm, i.e., the base algorithm corresponding to the correct latent model, it would suffer the regret $\varepsilon t + \sqrt{t}$. Note that by regret, we mean the actual regret and not pseudo-regret, and thus, $\sqrt{t}$ is the consequence of noise in the reward signal. Thus, we select the reference regret bound of our regret balancing strategy as $U(t) = \min\{\varepsilon t + \sqrt{t}, \sqrt{Kt} + \sqrt{t}\}$. We may write the regret of the resulting regret balancing strategy as follows:

$$\begin{aligned}
\mathcal{R}(T) &\overset{\text{(a)}}{=} \sum_{i=1}^{M+1} N_{i,T}\mu_* - R_{i,T} \overset{\text{(b)}}{\leq} \sum_{i=1}^{M+1} N_{i,T} b_T - R_{i,T} \overset{\text{(c)}}{=} \sum_{i=1}^{M+1} U(N_{i,T}) \\
&\leq \sum_{i=1}^{M+1} \min\left\{\varepsilon N_{i,T} + \sqrt{N_{i,T}}, \sqrt{KN_{i,T}} + \sqrt{N_{i,T}}\right\} \\
&\leq \min\left\{\sum_{i=1}^{M+1} \left(\varepsilon N_{i,T} + \sqrt{N_{i,T}}\right), \sum_{i=1}^{M+1} \left(\sqrt{KN_{i,T}} + \sqrt{N_{i,T}}\right)\right\} \\
&\overset{\text{(d)}}{=} \min\left\{\varepsilon T + \sum_{i=1}^{M+1} \sqrt{N_{i,T}}, \sum_{i=1}^{M+1} \left(\sqrt{KN_{i,T}} + \sqrt{N_{i,T}}\right)\right\} \\
&\overset{\text{(e)}}{\leq} \min\left\{\varepsilon T + \sum_{i=1}^{M+1} \sqrt{\frac{T}{M+1}}, \sum_{i=1}^{M+1} \left(\sqrt{K\frac{T}{M+1}} + \sqrt{\frac{T}{M+1}}\right)\right\} \\
&= \min\left\{\varepsilon T + \sqrt{(M+1)T}, \sqrt{K(M+1)T} + \sqrt{(M+1)T}\right\} \\
&= \mathcal{O}\left(\min\left\{\varepsilon T + \sqrt{MT}, \sqrt{KMT}\right\}\right),
\end{aligned}$$

which concludes our claim. Note that we used the following steps in our above derivations: **(a)** μ_* is the mean of the optimal arm. **(b)** This is because with high probability we have $\mu_* \leq b_t, \ \forall t \in [T]$. **(c)** This is from the definition b_t in (79). **(d)** This is due to the fact that $\sum_{i=1}^{M+1} N_{i,T} = T$. **(e)** The maximizer of $\sum_{i=1}^{M+1} \sqrt{N_{i,T}}$, subject to $\sum_{i=1}^{M+1} N_{i,T} = T$, is when all $\{N_{i,T}\}_{i=1}^{M+1}$ are equal.

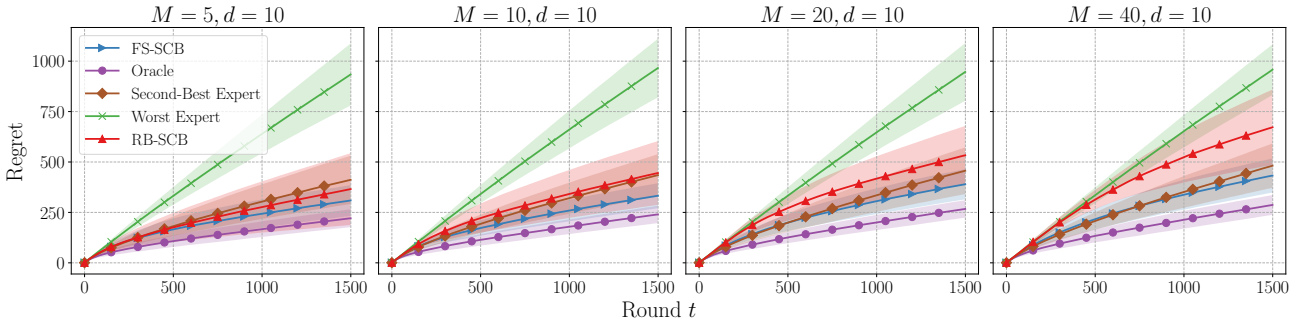

Figure 4. Feature selection on MNIST dataset. The regrets are averaged over 100 LB problems.

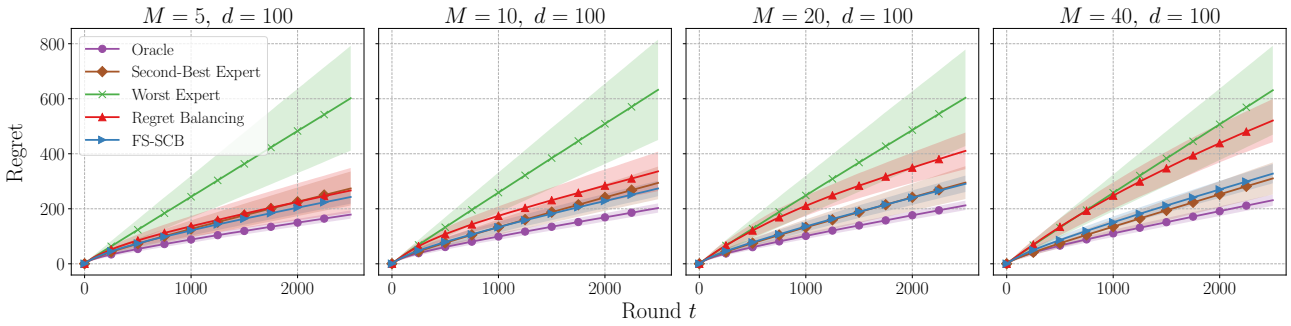

Figure 5. Feature selection on CIFAR-100 dataset. The regrets are averaged over 100 LB problems.

F. More on Experimental Results

We evaluate the performances of FS-SCB and PS-OFUL algorithms in a synthetic linear bandit problem and real-world image classification problems on CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009), and MNIST datasets (LeCun et al., 1998).

F.1. Feature Selection

F.1.1. MNIST Dataset

MNIST dataset consists of 60000 training and 10000 test images of size 28×28, each belonging to one of 10 classes. We train a convolutional neural network (CNN) with M different number of epochs on MNIST data, and use their second layer to the last as our $d = 10$-dimensional feature maps $\{\phi^i\}_{i=1}^M$. These feature maps have test accuracy between 20% (worst model) and 97% (best model). We set the best one as true model ϕ^{i_*}. For each class $s \in \mathcal{S} = \{0, \dots, 9\}$, we fit a linear model, given the feature map ϕ^{i_*}, and obtain parameters $\{\theta_s^{i_*}\}_{s=0}^9$. At the beginning of each LB task, we select a class $s_* \in \mathcal{S}$ uniformly at random and set its parameter to $\theta_{s_*}^{i_*}$. At each round $t \in [T]$, the learner is given an action set consists of 10 images, one from class s_* and the rest randomly selected from the other classes. The reward of each action a is defined as $\langle \phi^{i_*}(a), \theta_{s_*}^{i_*} \rangle + \eta_t \in [0, 1]$, where $\phi^{i_*}(a)$ is the application of the feature map ϕ^{i_*} to the image corresponding to action a and $\eta_t \sim \mathcal{U}[-0.5, 0.5]$ is the noise.

In Figure 4, we compare the regret of our FS-SCB algorithm for different number of models M with a regret balancing algorithm that uses SquareCB baselines (RB-SCB), and three SquareCB algorithms that use the best (Oracle), second-best (with test accuracy 84%), and worst feature maps (experts). Each plot is averaged over 100 LB problems. Figure 4 shows that **1)** FS-SCB always performs between the best and second-best experts, **2)** the regret of FS-SCB that scales as $\sqrt{\log M}$ is close to RB-SCB (scales as $\sqrt{M}$) for small M, but gets much better as M grows, and **3)** RB-SCB has much higher variance than the other algorithms.

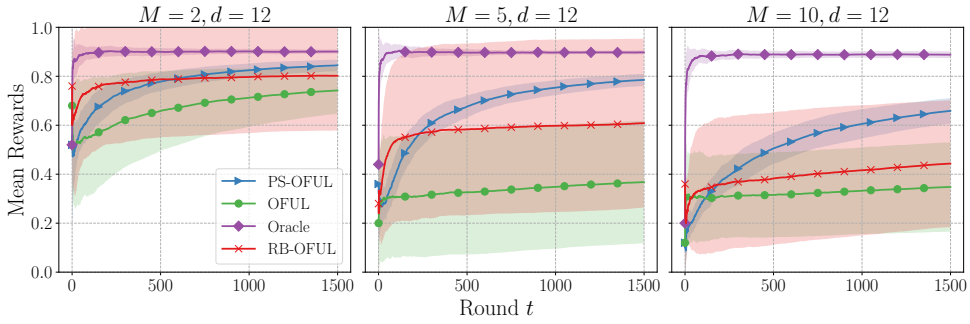

Figure 6. Parameter selection on MNIST dataset, where $100M$ datasets of size 500 are used to define the balls. The results are averaged over 50 runs.

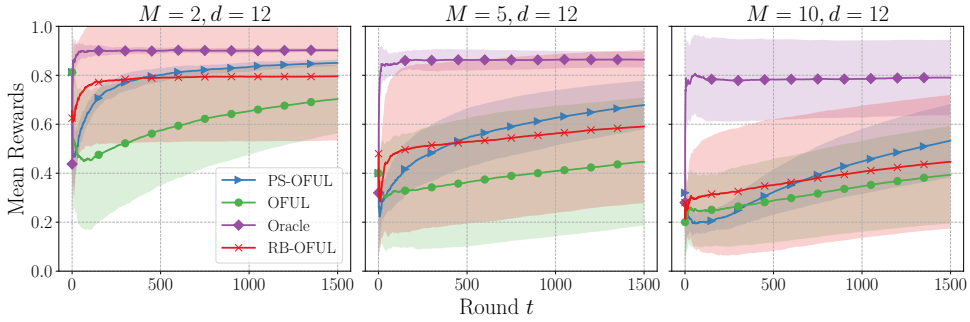

Figure 7. Parameter selection on MNIST dataset, where $10M$ datasets of size 50 are used to define the balls. The results are averaged over 50 runs.

F.1.2. CIFAR-100 Dataset

CIFAR-100 dataset consists of 50000 training and 10000 test images of size 32×32, each belonging to one of 100 classes. We extracted the features of the images by fine tuning and taking the output of the second-to-last layer of the EfficientNet-B0 Network (Tan & Le, 2019) and got the feature matrix of dimension 50000×1280. For all experts $i \in [M]$, we multiply this feature matrix with a Gaussian random matrix of dimension $1280 \times d_i$ for $d_i \in [2, 128]$ to get the d_i dimensional feature maps ϕ^i. These feature maps have accuracy between 5% (worst model) and 78% (best model). We set the best one as true model ϕ^{i_*}. For each class $s \in \mathcal{S} = \{0, \dots, 99\}$, we fit a linear model, given the feature map ϕ^{i_*} and obtain parameters $\{\theta_s^{i_*}\}_{s=0}^{99}$. At the beginning of each LB task, we select a class $s_* \in \mathcal{S}$ uniformly at random and set its parameter to $\theta_{s_*}^{i_*}$. At each round $t \in [T]$, the learner is given an action set consists of 10 images, one from class s_* and the rest randomly selected from the other classes. The reward of each action a is defined as $\langle \phi^{i_*}(a), \theta_{s_*}^{i_*} \rangle + \eta_t \in [0, 1]$, where $\phi^{i_*}(a)$ is the application of the feature map ϕ^{i_*} to the image corresponding to action a and $\eta_t \sim \mathcal{U}[-0.5, 0.5]$ is the noise.

In Figure 5, we compare the regret of our FS-SCB algorithm for different number of models M with a regret balancing algorithm that uses SquareCB baselines (RB-SCB) and aggregate them according to (79), and three SquareCB algorithms that use the best (Oracle), second-best (with test accuracy 55%), and worst feature maps (experts). Each plot is averaged over 100 LB problems. Figure 5 shows that **1)** FS-SCB always performs close to the best and second-best experts, **2)** the regret of FS-SCB that scales as $\sqrt{\log M}$ is close to RB-SCB (scales as $\sqrt{M}$) for small M, but gets much better as M grows, and **3)** RB-SCB has much higher variance than the other algorithms.

F.2. Parameter Selection

F.2.1. Image Classification on MNIST Dataset

MNIST dataset consists of 60000 test and 10000 training images of size 28×28, each belonging to one of 10 classes. We train a CNN with $d = 12$ neurons on second-to-last layer on MNIST dataset with 98% accuracy. We then select this d-dimensional layer as our feature map ϕ. To define our M models (balls), we sample $100M$ datasets of size 500. For each

dataset, we randomly select a class $s_* \in [M]$, assign reward 1 to images from s_* and 0 to other images, and fit a linear model to it to obtain a parameter vector. Finally, we fit a Gaussian mixture model (GMM) with M components to these $100M$ parameter vectors and use the means and covariances of the resulting clusters as the center and radii of our M models (balls). At the beginning of each LB task, we select a class $s_* \in [M]$ uniformly at random. At each round $t \in [T]$, the learner is given an action set consists of 10 images, one from class s_* and the rest randomly selected from the other classes. The learner receives a reward from $\text{Ber}(0.9)$ if it selects the image from class s_*, and from $\text{Ber}(0.1)$, otherwise.

In Figure 6, we compare the mean reward of PS-OFUL for different number of models M with a regret balancing algorithm that uses OFUL baselines (RB-OFUL) (Abbasi-Yadkori et al., 2020), OFUL (individual learning), and BIAS-OFUL (Cella et al., 2020) with bias being the center of the true model (Oracle). Figure 6 shows **1)** the good performance of PS-OFUL, **2)** the performance of PS-OFUL gets better than RB-OFUL as M grows ($\sqrt{\log M}$ vs. $\sqrt{M}$ scaling), **3)** the large variance of RB-OFUL, especially in comparison to PS-OFUL, and finally **4)** the advantage of transfer (PS-OFUL) over individual (OFUL) learning.

Impact of the model estimates: In order to demonstrate the impact of the accuracy of the model center estimates as well as the radii of the balls, we defined a less accurate set of M models (balls) using $10M$ datasets of size 50 (as opposed to $100M$ datasets of size 500). In Figure 7, we compare the mean reward of PS-OFUL for different number of models M with RB-OFUL, OFUL, and BIAS-OFUL.