On the Statistical Benefits of Curriculum Learning

Ziping Xu [1]   **Ambuj Tewari** [1]

Abstract

Curriculum learning (CL) is a commonly used machine learning training strategy. However, we still lack a clear theoretical understanding of CL's benefits. In this paper, we study the benefits of CL in the multitask linear regression problem under both structured and unstructured settings. For both settings, we derive the minimax rates for CL with the oracle that provides the optimal curriculum and without the oracle, where the agent has to adaptively learn a good curriculum. Our results reveal that adaptive learning can be fundamentally harder than the oracle learning in the unstructured setting, but it merely introduces a small extra term in the structured setting. To connect theory with practice, we provide justification for a popular empirical method that selects tasks with highest local prediction gain by comparing its guarantees with the minimax rates mentioned above.

1. Introduction

It has long been realized that we can design more efficient learning algorithms if we can make them learn on multiple tasks. Transfer learning, multitask learning and meta-learning are just few of the sub-areas of machine learning where this idea has been pursued vigorously. Often the goal is to minimize the weighted average losses over a set of tasks that are expected to be similar. While previous literature often assumes a predetermined (and often equal) number of observations for all the tasks, in many applications, we are allowed to decide the *order* in which the tasks are presented and the *number of observations* from each task. Any strategy that tries to improve the performance with a better task scheduling is usually referred to **curriculum learning (CL)** (Bengio et al., 2009). The agent that schedules tasks at each step is often referred as the *task scheduler*.

Though curriculum learning has been extensively used in modern machine learning (Gong et al., 2016; Sachan & Xing, 2016; Tang et al., 2018; Narvekar et al., 2020), there is very little theoretical understanding of the actual benefits of CL. We also do not know whether the heuristic methods used in many empirical studies can be theoretically justified. Even the problem itself has not been rigorously formulated. To address these challenges, we first formulate the curriculum learning problem in the context of the linear regression problem. We analyze the minimax optimal rate of CL in two settings: an unstructured setting where parameters of different tasks are arbitrary and a structured setting where they have a low-rank structure. Finally we discuss the theoretical justification of a popular heuristic task scheduler that greedily selects tasks with highest local prediction gain.

Related literature. Despite the lack of the literature on curriculum learning theory, we found the following problems under a similar setting. Active learning (Settles, 2009; 2011), addresses the problem of actively selecting unlabeled data points to maximize model accuracy. Active learning can be treated as a curriculum learning when each unlabeled point is a task with point mass. Active learning has also been used for domain adaptation or transfer learning setting (Persello & Bruzzone, 2012). Multi-source domain adaptation (Wang & Deng, 2018; Sun et al., 2015) also considers multiple candidate source domains for a given target task. Bhatt et al. (2016) proposed an iterative adaptation methods to integrate the source data from each domain. However, such adaptive procedure has not been theoretically understood.

2. Background

We would like to point out previous work on three crucial aspects of CL: two types of benefits one may expect from CL, task similarities assumptions, and task scheduler used in empirical studies.

Two types of benefits. There are two distinct ways to understand the benefits of CL. From the perspective of optimization, some papers argue that the benefits of curriculum can be interpreted as learning from more convex and more smooth objective functions, which serves as a better initialization point for the non-convex target objective function (Bengio et al., 2009). The order of task scheduling is es-
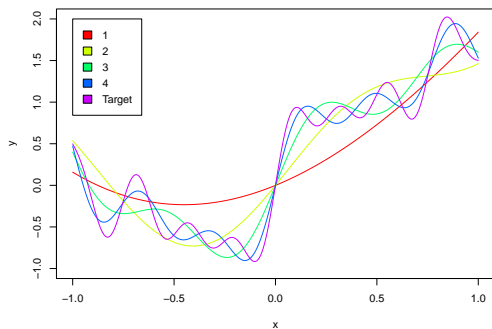
---

[1]Department of Statistics, University of Michigan, Ann Arbor. Correspondence to: Ziping Xu <zipingxu@umich.edu>.

Figure 1. An example of tasks with increasing non-convexity. Solid lines of different colors represent the true object functions of different tasks.

sential here. As an example, Figure 1 shows the objective functions of a problem with four source tasks and one target task with increasing difficulty (non-convexity). Directly minimizing the target task (marked in purple line) using gradient descent can be hard due to the non-convexity. However, the simple gradient descent algorithm can converge to the global optima of the current task if it starts from the global optima of the previous task. We refer to the benefit that involves a faster convergence in optimization as *optimization benefit*. Optimization benefits highly depend on the order of scheduling. Generally speaking, if one directly considers the empirical risk minimizer (ERM) which requires global minimization of empirical risk, there may not be any optimization benefit.

The second type of benefit concerns the benefit brought by carefully choosing the number of observations from each task while independent of the order, which we call *statistical benefit*. For example, we have two linear regression problems that are identical except for the standard deviation of the Gaussian noise on response variables. If we consider the OLS estimator on the joint dataset of the two tasks, there is a reduction in noise level when more samples are allocated to the task with a lower level, and the benefit is independent of the order by the nature of OLS. A *statistical benefit* can be seen as any benefit one can get except for the reduction in the difficulties of optimization. (Weinshall & Amir, 2020) focused on a special curriculum learning task where each sample is considered a task and they analyzed the error rate on the samples of different noise levels. They analyzed the benefits on the error rate of linear models, whose global minimizer can be easily found. Thus, it should also count as the *statistical benefits*.

In general, the two types of benefits can coexist. A good curriculum should account for both the non-convexity and the noise levels. However, due to the significantly different

underlying mechanism in the two learning benefits. it is natural to study them separately. This paper will focus on the analysis of the **statistical benefits**. Thus, we analyze algorithms that map datasets to an estimator for each task that may involve finding global minima of the empirical errors for non-convex functions.

Similarity assumptions. We discussed the problem with two almost identical tasks, where we can achieve perfect transfer and the trivial curriculum that allocates all the samples to the simpler task is optimal. However, tasks are generally not identical. Understanding how much benefits the target task can gain from learning source tasks has been a central problem in transfer learning and multitask learning literature. The key is to propose meaningful similarity assumptions.

Let $\mathcal{X}$ and $\mathcal{Y}$ be the input and output space. Assume we have T tasks with data distributions $D_1 \ldots, D_T$ over $\mathcal{X} \times \mathcal{Y}$. Let (X_t, Y_t) be a sample from task t. Let $f_t = \mathbb{E}[Y_t \mid X_t]$, a mapping $\mathcal{X} \mapsto \mathcal{Y}$, be the mean function. In this paper, we adopt the simple parametric model on the mean function with f_t represented by parameter $\theta_t^* \in \mathbb{R}^d$.

We consider two scenarios: structured and unstructured. In Section 3, we adopt simple linear regression models and do not assume any further internal structure on the true parameters. Two tasks are similar if $\|\theta_{t_1}^* - \theta_{t_2}^*\|_2$ is small. A learned parameter is directly transferred to the target task. This setting has been applied in many previous studies (Yao et al., 2018; Bengio et al., 2009; Xu et al., 2021). In Section 4, we study the multitask representation learning setting (Maurer et al., 2016; Tripuraneni et al., 2020; Xu & Tewari, 2021), where a stronger internal structure is assumed. To be specific, we write $f_t(x) = x^T B^* \beta_t^*$, where $x \in \mathbb{R}^d$, $B^* \in \mathbb{R}^{d \times k}$ is the linear representation mapping and $\beta_t^* \in \mathbb{R}^k$ is the task specific parameter. Generally, the input dimension is much larger than the representation dimension ($d \gg k$).

These two settings, while representative, do not exhaust all of the settings in the literature. We refer the reader to (Teshima et al., 2020) for a brief summary of theoretical assumptions on the task similarity.

Task schedulers. Many empirical methods have been developed to automatically schedule tasks. (Liu et al., 2020) designed various heuristic strategies for task selection for computer vision tasks. (Cioba et al., 2021) discussed several meta-learning scenarios where the optimal data allocations are different, which interestingly aligns with our theoretical results. For a more general use, one major family of task scheduler is based on the intuition that the task scheduler should select the task that leads to the highest local gain on the target loss (Graves et al., 2017). Since the accurate prediction gain is not accessible, online decision-making al-

gorithms (bandit and reinforcement learning) are frequently used to adaptively allocate samples (Narvekar et al., 2020). However, there is no theoretical guarantee that such greedy algorithms can lead to the optimal curriculum.

Notation. For any positive integer n, we let $[n] = \{1, \ldots, n\}$. We use the standard $O(\cdot), \Omega(\cdot)$ and $\Theta(\cdot)$ notation to hide universal constant factors. We also use $a \lesssim b$ and $a \gtrsim b$ to indicate $a = O(b)$ and $b = \Omega(b)$.

3. Unstructured Linear Regression

In this section, we study the problem of learning from T tasks to generate an estimate for a single target task.

3.1. Formulations

We consider linear regression tasks. Let $1 \ldots, T$ denote T tasks. Let $\theta_t^* \in \mathbb{R}^d$ denote the true parameter of task t. The response Y_t of task t is generated in the following manner

$$Y_t = X_t^T \theta_t^* + \epsilon_t,$$

where ϵ_t is assumed to be the Gaussian noise with $\mathbb{E}[\epsilon_t^2] = \sigma_t^2$ and $X_t \sim \mathcal{N}(\mathbf{0}, \Sigma_t)$, where $\Sigma_t \in \mathbb{R}^{d \times d}$ is the covariance matrix that is positive definite. Any task, therefore, can be fully represented by a triple $(\theta_t^*, \sigma_t, \Sigma_t)$.

Throughout the paper, we are more interested in the unknown parameters rather than the covariate distribution or the noise level. We simply denote $\boldsymbol{\theta}^* \in \mathbb{R}^{d \times T}$ the parameters of a *problem* (T tasks) and let $\boldsymbol{\theta}_t^*$ be the t-th column of the matrix.

We make a uniform assumption on the covariance matrix of input variables. The same assumption is also used by (Du et al., 2020).

Assumption 1 (Coverage of covariate distribution). *We assume that all* $C_0 I_d \succ \Sigma_t \succ C_1 I_d$ *for some constant* $C_0, C_1 > 0$ *and any* $t \in [T]$.

Goal. Let S_t^n be random n samples from task t. Let $l : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$ be a loss function and $L_t(\theta) = \mathbb{E}[l(X^T \theta, Y)]$ be the expected loss of a given hypothesis θ evaluated on task t. Moreover, we denote the excess risk by

$$G_t(\theta) = L_t(\theta) - \inf_{\theta'} L_t(\theta').$$

Our goal in this section is to minimize the expected loss of the last task T, which we call the **target task**. Throughout the paper, we use square loss function.

Transfer distance. Algorithms tend to perform better when the tasks are similar to each other, such that any observations collected from non-target task bear less transfer

bias. We define *transfer distance* between tasks t_1, t_2 as $\Delta_{t_1, t_2} = \|\theta_{t_1}^* - \theta_{t_2}^*\|_2$.

It is not fair to compare the performances between problems with different transfer distances. To study a minimax rate, we are interested in the worst performance over a set of problems with similar transfer distance. Let $\boldsymbol{Q} \in \mathbb{R}^T$ be the distance vector encoding the upper bounds on the distance between the target task to any task. We define the hypothesis set with known transfer distance as $\Theta(\boldsymbol{Q}) = \cup_p \{\boldsymbol{\theta}^* : \|\boldsymbol{\theta}_t^* - \boldsymbol{\theta}_T^*\|_2 \le \boldsymbol{Q}_t\} \subset \mathbb{R}^{T \times d}$. The hypothesis set with unknown transfer distance can be defined as $\tilde{\Theta}(\boldsymbol{Q}) = \cup_p \{\boldsymbol{\theta}^* : \|\boldsymbol{\theta}_{p_t}^* - \boldsymbol{\theta}_T^*\|_2 \le \boldsymbol{Q}_{p_t}\}$, where $p \in [T]^T$ is any permutation of $[T]$. We say this hypothesis set has unknown transfer distance because even if there exists some small $\boldsymbol{Q}_t$ such that the transfer distance is low, an agent does not know which task has the low transfer distance.

Curriculum learning and task scheduler. This paper concerns only the statistical learning benefits. Since the order of selecting tasks does not affect the outcome of the algorithm, we denote a curriculum by $\boldsymbol{c} \in [N]^T$, where each $\boldsymbol{c}_t$ is the total number of observations from task t and $\sum_t \boldsymbol{c}_t = N$. Note that $\boldsymbol{c}$ can consist of random variables depending on the task scheduler. The set of all the curriculum with a total number of observations N is denoted by $\mathcal{C}_N = \{\boldsymbol{c} \in [N]^T : \sum_t \boldsymbol{c}_t = N\}$.

Any curriculum learning involves a multitask learning algorithm, which is defined as a mapping $\mathcal{A}$ from a set of datasets $(S_1^{n_1}, \ldots, S_T^{n_T})$ to a hypothesis θ for the target task.

A task scheduler runs the following procedure. At the start of the step $i \in [N]$, we have $n_{i,1}, \ldots, n_{i,T}$ observations from each task. The task scheduler $\mathcal{T}$ at step i is defined as a mapping from the past observations $(S_1^{n_{i,1}}, \ldots, S_T^{n_{i,T}})$ to a task index. Then a new observation from the selected task $\mathcal{T}(S_1^{n_{i,1}}, \ldots, S_T^{n_{i,T}}) \in [T]$ is sampled.

Minimax optimality and adaptivity. One of the goals of this work is to understand the minimax rate of the excess risk on the target task over all the possible combinations of multitask learning algorithms and task schedulers. We first attempt to understand a limit of that rate by considering an oracle scenario that provides the optimal curriculum for any problem.

Rigorously, we denote the loss of a fixed curriculum $\boldsymbol{c} \in \mathcal{C}_N$ with respect to a fixed algorithm $\mathcal{A}$ and problem $\boldsymbol{\theta}$ by

$$R_T^N(\boldsymbol{c}, \mathcal{A} \mid \boldsymbol{\theta}) = \mathbb{E}_{S_1^{\boldsymbol{c}_1}, \ldots, S_T^{\boldsymbol{c}_T}} G_T(\mathcal{A}(S_1^{\boldsymbol{c}_1}, \ldots, S_T^{\boldsymbol{c}_T})).$$

We define the following oracle rate, which takes infimum over all the possible fixed curriculum designs given a fixed

task set with different $\boldsymbol{\theta}$ in a hypothesis set Θ.

$$R_T^N(\Theta) := \inf_{\mathcal{A}} \sup_{\boldsymbol{\theta} \in \Theta} \inf_{c \in \mathcal{C}_N} R_T^N(c, \mathcal{A} \mid \boldsymbol{\theta}). \quad (1)$$

In general, the above oracle rate considers an ideal case, because the optimal curriculum depends on the unknown problem and any learning algorithm has to adaptively learn the problem to decide the optimal curriculum.

We ask the following question: can adaptively learned curriculum perform as well as the optimal one as in Equ. (1)? To answer the question, we define the minimax rate for adaptive learning:

$$\tilde{R}_T^N(\Theta) := \inf_{\mathcal{A}} \inf_{\mathcal{T}} \sup_{\boldsymbol{\theta} \in \Theta} \mathbb{E} G_T(\mathcal{A}(S_1^{\boldsymbol{c}_{\mathcal{T}},1}, \ldots, S_T^{\boldsymbol{c}_{\mathcal{T}},T})), \quad (2)$$

where $\boldsymbol{c}_{\mathcal{T}} \in \mathcal{C}_N$ is the curriculum adaptively selected by the task scheduler $\mathcal{T}$ and the expectation is taken over both datasets $S_1, \ldots, S_T$ and $\boldsymbol{c}_{\mathcal{T}}$.

In this section, we are interested in the oracle rate in (1) compared to some naive strategy that allocates all the samples to one task. This answers how much benefits one can achieve compared to some naive learning schedules. We are also interested the gap between Equ. (1) and Equ. (2).

3.2. Oracle rate

In this section, we analyze the oracle rate defined in Equ. (1). We first give an overview of our results. For any problem instance, there exists a single task t such that the naive curriculum with $\boldsymbol{c}_t = N$ matches a lower bound for the oracle rate defined in Equ. (1).

For any task $t \in [T]$, its direct transfer performance of the OLS estimator on the target task T can be roughly bounded by $\Delta_{t,T}^2 + d\sigma_t^2/N$.

Thus, our result implies that essentially, the goal of curriculum learning is to identify the best task that balance the transfer distance and the noise level.

Theorem 1. *Let $\boldsymbol{Q}$ be a fixed distance vector defined above. The oracle rate within $\Theta(\boldsymbol{Q})$ in Equ. (1) can be lower bounded by*

$$R_T^N(\Theta(\boldsymbol{Q})) \gtrsim C_0 \min_t \{\boldsymbol{Q}_t^2 + \frac{d\sigma_t^2}{N}\}. \quad (3)$$

Proof highlights. Kalan et al. (2020) showed a minimax rate of the transfer learning problem with only one source task. They considered three scenarios, which can be uniformly lower bounded by the right hand side of Equ. (3). Our analysis can be seen as an extension of their results to multiple source tasks. In general, let the rate in (3) be δ and $C > 0$ be a constant. Let t^* be the best task indicated by (3). Any task t with a large distance ($\boldsymbol{Q}_t > C\delta$) is not helpful to

learn the target task. Thus, samples from these tasks can be discarded without reducing the performance. For any task t with $\boldsymbol{Q}_t \leq C\delta$, we will show that any sample from task t gives almost as much as information as the best task t^* gives. Thus, one can replace them with a random sample from the best task t^* without reducing the loss. Then the problem can be converted to a single source task problem, from where we follow the lower bound construction in Kalan et al. (2020).

3.3. Minimax rate for adaptive learning

The problem can be hard when the transfer distance is unknown. We introduce an intuitive example to help understand our theoretical result. Assume we have three tasks including one target task and two source tasks. One of the two source tasks is identical to the target task. We have n samples for both source tasks, while no observations from the target task. In this example, even if one of the source tasks is identical to the target task, no algorithm can decide which source task should be adopted, since we have no information from the target task. In other words, any algorithm can be as bad as the worst out of the two source tasks. This is not an issue when the transfer distance is known to the agent in the oracle scenario. This example implies that to adaptively gain information from source tasks, we will need sufficient information from the target task. Otherwise, there is risk of including information from tasks that contaminates the target task. Similarly, (David et al., 2010) also showed that without any observations from the target task, domain adaptation is impossible.

More generally, even if we have some data from the target task, we will show that one is not able to avoid σ_T^2 term, the learning difficulty of the target task. Now we formally introduce our results.

Theorem 2. *Assume $T \geq 4$. Let $Q_{sub} > 0$. Let $\boldsymbol{Q}$ be a fixed distance vector that satisfies $\boldsymbol{Q}_1 = 0$ and $\boldsymbol{Q}_t = Q_{sub} > 0$ for all $t = 2, \ldots, T - 1$. The minimax rate in Equ. (2) can be lower bounded by*

$$\tilde{R}_T^N(\tilde{\Theta}(\boldsymbol{Q})) \gtrsim \min\{\frac{\sigma_T^2 \log(T)}{N}, Q_{sub}^2\} + \min_t \frac{d\sigma_t^2}{N}. \quad (4)$$

Theorem 2 implies that without knowing the transfer distance, any adaptively learned curriculum of any multitask learning algorithm will suffer an unavoidable loss of $\sigma_T^2 \log(T)/N$, when Q_{sub} is large. Compared to the rate $\sigma_T^2 d/N$ without transfer learning, there is still a potential improvement of a factor of $d/\log(T)$ when Q_{sub} and σ_t^2 are small.

Upper bound. As we showed above, there is a potential improvement of $d/\log(T)$. This is because given the prior information that one of the source tasks is identical to the target task, the problem reduces from estimating a

d-dimensional vector to identifying the best task from a candidate set, whose complexity reduces to $\log(T)$.

In fact, a simple fixed curriculum could achieve the above minimax rate. Assume that any $\|\theta_t^*\|_2 \leq C_2$ for some constant $C_2 > 0$. Let $c_T = N/2$ and for all the other tasks $c_t = N/(2T - 2)$. For each $t = 1, \ldots T - 1$, let $\tilde{\theta}_t$ be the OLS estimator using only its own samples. Let $\hat{\theta}_t$ be the projection of $\tilde{\theta}_t$ onto $\{\theta : \|\theta\|_2 \leq C_2\}$. Then we choose one estimator from $t = 1, \ldots, T - 1$, that minimizes the empirical loss for the target task:

$$t^* = \arg\min_{t \in [T-1]} \sum_{i=1}^{N/2} (Y_{T,i} - X_{T,i}^T \hat{\theta}_t)^2. \tag{5}$$

Theorem 3. *Assume there exists a task t such that $\Delta_{t,T} = 0$ and $\|\theta_t^*\|_2 \leq C_2$. With a probability at least $1 - \delta$, $\hat{\theta}_{t^*}$ satisfies*

$$G_T(\hat{\theta}_{t^*}) \lesssim C_0 C_2^2 \log(Td/\delta) \left(\frac{C_2 \sigma_T^2}{N} + \frac{dT\sigma_{t^*}^2}{C_1 N} \right). \tag{6}$$

Note that t^* is a random value. However, when all t satisfy $dT\sigma_t^2 < \Delta_{max}\sigma_T^2 \log(T)$, the first term is the dominant term and our bound matches the lower bound in (4). This could happen when $\sigma_T^2 \gg \sigma_t^2$ for all $t = 1, \ldots, T - 1$.

General function class. As we mentioned before, though it is difficult to identify the good source tasks, the complexity of doing so is still lower than learning the parameters directly. We remark that this result can be generalized to any function class beyond linear functions. Keeping all the other setup unchanged, we assume that the mean function $f_t^* \in \mathcal{F}_t : \mathcal{X} \mapsto \mathcal{Y}$ for some input space $\mathcal{X}$ and output space $\mathcal{Y}$ shared by all the tasks. For convenience, we assume there is no covariate shift, i.e. the input distributions are the same. We give an analogy of Theorem 3.

Assumption 2 (Assumption B in Jin et al. (2021)). *Assume $l(\cdot, y)$ is L_2-strongly convex and L_1-Lipschitz at any $y \in \mathcal{Y}$. Furthermore, for all $x \in \mathbb{R}^d$ and $t \in [T]$,*

$$\mathbb{E}[\nabla l(f_t^*(X), Y) \mid X = x] = 0.$$

Assume we have $N/(2T - 2)$ observations for all tasks $t = 1, \ldots, T$ and $N/2$ observations for the target task. Let $\hat{f}_t$ be the empirical risk minimizer of the task t. Similarly to (5), let

$$t^* = \arg\min_{t \in [T-1]} l_T^N(\hat{f}_t),$$

where l_T^N is the empirical loss on task T. Let $L^* = \min_{t \in [T-1]} L_T(f_t^*)$ and $t' = \arg\min_{t \in [T-1]} L_T(f_t^*)$. We will use Rademacher complexity to measure the hardness of learning a function class. We refer readers to (Bartlett & Mendelson, 2002) for the detailed definition of Rademacher complexity.

Proposition 1. *Given the above setting and Assumption 2, we have with a probability at least $1 - \delta$,*

$$G_T(\hat{f}_{t^*}) \lesssim L^* + \frac{L_1}{L_2} \left(\mathcal{R}_{N/T}(\mathcal{F}_{t'}) + \sqrt{\frac{\log(1/\delta)}{N/T}} \right),$$

where $\mathcal{R}_N(\mathcal{F})$ is the Rademacher complexity of function space $\mathcal{F}$.

This bound improves the bound for single target task learning, which scales with $\mathcal{R}_N(\mathcal{F}_T)$, when $\mathcal{R}_N(\mathcal{F}_T) \gg \mathcal{R}_{N/T}(\mathcal{F}_{t^*})$. The underlying proof idea is still that identifying good tasks is easier than learning the model itself.

4. Structured Linear Regression

Now we consider a slightly different setting, where we want to learn a shared linear representation that generalizes to any target task within a set of interest.

A lot of recent papers have shown that to achieve a good generalization ability of the learned representation, the algorithm have to choose diverse source tasks (Tripuraneni et al., 2020; Du et al., 2020; Xu & Tewari, 2021). They all study the performance of a given choice of source tasks, while it has been unclear whether an algorithm can adaptively select diverse tasks.

4.1. Problem setup

We adopt the setup in Du et al. (2020). Let $d, k > 0$ be the dimension of input and representation, respectively ($k \ll d$). We also set $T \leq d$. Let $B^* \in \mathbb{R}^{d \times k}$ be the shared representation. Let $\beta_1^*, \ldots, \beta_T^* \in \mathbb{R}^k$ be the linear coefficients for prediction functions. The model setup is essentially the same as the setup in Section 3.1 except for the true parameters being $B^* \beta_t$. We call this setting structured because if one stacks the true parameters as a matrix, the matrix has a low-rank structure. To be specific, the output of task t given by

$$Y_t = X_t^T B^* \beta_t^* + \epsilon_t.$$

We use the same setup for the covariate X_t as in Section 3 and we consider $\sigma_1^2 = \cdots = \sigma_T^2 = \sigma^2$ for some $\sigma^2 > 0$.

Diversity. Let t_i be the task selected by the scheduler at step i. It has been well understood that to learn a representation that could generalize to any target task t' with arbitrary $\beta_{t'}^*$, we will need a lower bound on the following term

$$\lambda_k \left(\sum_{i=1}^N \beta_{t_i}^* \beta_{t_i}^{*T} \right) =: \lambda_{N,k}, \tag{7}$$

where λ_k is the k-th largest eigenvalue of a matrix, i.e. the smallest eigenvalue. Basically, we hope the source tasks

cover all the possible directions such that any new task could be similar to at least some of the source tasks. Equ. (7) serves as an assumption in (Du et al., 2020). When the true β_t^* are known, we can simply diversely pick tasks. When the β_t^* are unknown, the trivial strategy that equally allocates samples will perform badly. For example, let $T \gg k$ and let all the $\beta_t, t = k+1, \ldots, T$ be identical. The trivial strategy will only cover one direction sufficiently, which ruins the generalization ability.

In this section, we will show that it is possible to adaptively schedule tasks to achieve the diversity even in the hard case discussed above.

4.2. Lower bounding diversity

In this section, we introduce an OFU (optimism in face of uncertainty) algorithm that adaptively selects diverse source tasks.

Two-phase estimator. We first introduce an estimator on the unknown parameters. Assume up to step i, we have dataset $S_1^{n_{i,1}}, \ldots, S_T^{n_{i,T}}$ for each task t. We evenly split each dataset $S_t^{n_{i,t}}$ to two datasets $S_{i,t}^{(1)}$ and $S_{i,t}^{(2)}$, both with a sample size of $\lfloor n_{i,t}/2 \rfloor$. We solve the optimization problem below:

$$\hat{B}_i = \arg\min_{B \in \mathbb{R}^{d \times k}} \min_{\beta_t \in \mathbb{R}^k, t \in [T]} \sum_{t \in [T]} \sum_{(x,y) \in S_{i,t}^{(1)}} \|y - x^T B \beta_t\|^2,$$

$$\text{and } \hat{\beta}_{i,t} = \arg\min_{\beta_t \in \mathbb{R}^k} \sum_{(x,y) \in S_{i,t}^{(2)}} \|y_j - x_j B \beta_{t_j}\|^2.$$

Note that we split the dataset such that $\hat{B}_i$ and $\hat{\beta}_{i,t}$ are independent.

Algorithm 1 CL by optimistic scheduling
1: **Input:** T tasks and total number of observations N and constant $\gamma > 0$.
2: Sample $\lceil \gamma(d + \log(N/\delta)) \rceil$ samples for each task.
3: **for** $i = T\lceil \gamma(d + \log(N/\delta)) \rceil + 1, \ldots, N$ **do**
4:     Construct confidence set $\mathcal{B}_{i,t}$ for each $t \in [T]$ according to Equ. (8).
5:     Select t_i according to Equ. (9).
6: **end for**
7: **return:** Curriculum $(t_1, \ldots, t_N)$.

Optimistic task scheduler. Our algorithm runs by keeping a confidence bound for $B^* \beta_t^*$ for each $t \in [T]$ and each step $i \in [N]$. Lemma 1 introduces a suitable upper bound construction. Lemma 1 holds under the following assumptions.

Lemma 1. *Let $\kappa = C_0/C_1$. Assume Assumption 1 hold. There exists universal constants $\gamma > 0, \alpha > 0$ such that, at*

all step $i > T\lceil \gamma(d + \log(N/\delta)) \rceil$, with a probability $1 - \delta$, we have for all $t \in [T]$,

$$\|\hat{B}_i^T \hat{\beta}_{i,t} - B^* \beta_{i,t}^*\|_2^2 \lesssim \frac{\alpha C_5 \sigma^2 dk \log(\kappa N \delta/T)}{C_1^2 n_{i,t}},$$

where $n_{i,t}$ is the number of observations from task t up to step i.

Following the bound in Lemma 1, we construct the confidence set with width

$$\mathcal{W}_{i,t} := \frac{C_5 \sigma^2 dk \log(\kappa N \delta/T)}{C_1^2 n_{i,t}}.$$

At each step i for each task t, we construct a confidence set around $\hat{B}_i \hat{\beta}_{i,t}$,

$$\mathcal{B}_{i,t} = \{\theta \in \mathbb{R}^d : \|\hat{B}_i \hat{\beta}_{i,t} - \theta\|_2^2 \leq \mathcal{W}_{i,t}\}. \quad (8)$$

Then following the principle of optimism in face of uncertainty, we select the task t_i such that

$$t_i \in \arg\max_{t \in [T]} \max_{\theta \in \mathcal{B}_{i,t}} \lambda_k \left(\sum_{j=1}^{i-1} \tilde{\theta}_j \tilde{\theta}_j^T + \theta \theta^T \right) \quad (9)$$

and $\tilde{\theta}_i = \arg\max_{\theta \in \mathcal{B}_{i,t}} \lambda_k (\sum_{j=1}^{i-1} \tilde{\theta}_j \tilde{\theta}_j^T + \theta \theta^T)$. Here $\tilde{\theta}_i$ is our belief for task t at the step i.

Now we are ready to present our lower bound results for diversity. Our results hold under two assumptions. The first assumption require the representation matrix B^* is not degenerated. We also assume boundedness on β_t^*'s.

Assumption 3. *Assume the largest singular value of B^* is smaller than C_4 for some $C_4 > 0$.*

Assumption 4 (Boundedness)**.** *We also assume that $\|\beta_t^*\|^2 \leq C_5$ for all $t \in [T]$.*

Theorem 4. *Suppose Assumption 3 and 4 hold. Assume for all $\nu \in \mathbb{R}^k, \|\nu\|_2 = 1$, there exists some task t such that $\nu^T B^* \beta_t^* \beta_t^{*T} B^{*T} \nu \geq \lambda$ for some $\lambda > 0$. Let $t_i, i = 1, \ldots, N$ be the tasks select by Algorithm 1 for some constant α. Then there exists some $\alpha > 0$, such that with a probability at least $1 - \delta$,*

$$\frac{\lambda_{N,k}}{N} \gtrsim \frac{\lambda}{C_4 k} - \sqrt{\frac{\sigma^2 C_5^2 dkT \log(\kappa N/(T\delta))}{C_1^2 C_4^2 \lambda N}}.$$

If we are provided with the oracle, we will only have the first term above. When N is sufficiently large, the second term in Theorem 4 is negligible and we will achieve diversity asymptotically as long as $dkT \ll N$. Our proof follows the standard framework for OFU algorithms. We first show the correctness of the confidence set implied by Lemma 1. Then the key steps are to show the optimism, i.e. $\lambda_k(\sum_{i=1}^N \tilde{\theta}_i \tilde{\theta}_i^T) = \Omega(\lambda/k)$ and to bound the difference term between the belief $\lambda_k(\sum_{i=1}^N \tilde{\theta}_i \tilde{\theta}_i^T)$ and the actual value $\lambda_{N,k}$. We provide the proof in Appendix E.

4.3. Upper bound results

Though the lower bound in Theorem 4 is already satisfying, we still want to shed some light on whether the dependency on $\sqrt{1/N}$ is avoidable by showing an upper bound result in Theorem 5.

Theorem 5. *For any curriculum learning algorithm, there exists T tasks $(T > k)$ such that for all $\nu \in \mathbb{R}^k, \|\nu\|_2 = 1$, there exists some β_t^*, $\|\beta_t^* \nu\| \geq 1$ and*

$$\mathbb{E}[\frac{\lambda_{N,k}}{N}] \lesssim \frac{\max_{t_1,\ldots,t_N \in [T]} \lambda_k(\sum_{i=1}^N \beta_{t_i}^* \beta_{t_i}^{*T})}{N} - \sqrt{\frac{\sigma^2 T}{Nk^3}}.$$

Theorem 5 states that the $\sqrt{1/N}$ dependency is unavoidable, while there is still a gap of dk^4 between the upper bound and the lower bound. Our hard case construction is inspired by the case where the naive strategy that allocates samples evenly. To be specific, we consider T tasks such that k of them are diversely specified and all the other $T - k$ tasks are identical. Naive strategies will fail by having $\lambda_{N,k} \approx \frac{1}{kT}$. We divide T tasks into $\lfloor T/k \rfloor$ blocks. Then we construct similar problems. Different problems have the diverse tasks in different blocks. The difficulty of the problem becomes identifying the block with diverse tasks, which is analogous to the idea of bandit model in a general sense. From here, we follow a similar proof of stochastic bandits (Lattimore & Szepesvári, 2020). The full proofs can be found in Appendix F.

5. Analysis of Prediction Gain

In this section, we give some theoretical guarantees on prediction-gain driven task scheduler under the unstructured setting discussed in Section 3. We do not consider the structured setting because it is not clear how to apply the prediction-gain driven method to multitask representation learning setting.

Prediction Gain and convergence rate. We define prediction gain in the following way. At the step i, a multitask learning algorithm $\mathcal{A}$ maps any trajectory $\mathcal{H}_i = \{x_{t_i,j}, y_{t_i,j}\}_{j=1}^i$ to a parameter $\theta \in \mathbb{R}^d$ for the target task. Let the estimate at step i be θ_i. The prediction gain is defined as

$$G(\mathcal{A}, \mathcal{H}_{i+1}) := L_T(\theta_i) - L_T(\theta_{i+1}).$$

At the start of the round i, the prediction-gain based task scheduler selects $t_i \in [T]$ such that $G(\mathcal{A}, \mathcal{H}_i)$ is maximized.

Note that in general, prediction gain is not observable to the algorithm before $x_{t_i,i}$ and $y_{t_i,i}$ are actually sampled. There are simple ways to estimate prediction gain, for example, from several random samples from each task.

In a linear model, the prediction gain is equivalent to convergence rate.

$$L_T(\theta_i) - L_T(\theta_{i+1}) = \|\theta_i - \theta_T^*\|_{\Sigma_T}^2 - \|\theta_{i+1} - \theta_T^*\|_{\Sigma_T}^2.$$

Weinshall & Amir (2020) discussed various benefits of curriculum learning by show that their strategy gives higher local convergence rate. It is not clear from the context that the greedy strategy that selects the highest local prediction gain gives the best total prediction gain in long run.

Decomposing prediction gain. Considering a identical covariance matrix $\Sigma_t = I$, the loss over a given parameter θ can be written as $\|\theta - \theta_T^*\|_2^2 + \sigma_T^2$.

Assume the gradient is calculated from a sample from the task t. According to the update of SGD, at the step i, we have

$$\theta_{i+1} - \theta_T^* = (I - \eta_i x_i^{(t)} x_i^{(t)T})(\theta_i - \theta_T^*) + \eta_i x_i^{(t)}(\epsilon_i + x_i^T \theta_{t,T}^\Delta),$$

where $\theta_{t,T}^\Delta = \theta_t^* - \theta_T^*$.

The one-step prediction gain is

$$\|\theta_i - \theta_T^*\|^2 - \|\theta_{i+1} - \theta_T^*\|^2$$
$$= \eta_i \|\theta_i - \theta_T^*\|_{(2-\eta_i \|x_i^{(t)}\|_2^2) x_i x_i^T}^2 - \eta_i^2 \|x_i^{(t)}(\epsilon_i + x_i^T \theta_{t,T}^\Delta)\|_2^2$$
$$- \eta_i(\theta_i - \theta_T^*)^T (I - \eta_i x_i^{(t)} x_i^T) x_i^{(t)}(\epsilon_i + x_i^T \theta_{t,T}^\Delta).$$

The first term on the R.H.S is the absolute gain shared by all the tasks. On expectation, the second term is

$$-\mathbb{E}\eta_i^2 \|x_i(\epsilon_i - x_i^T \theta_{t,T}^\Delta)\|_2^2 = -\mathbb{E}\eta_i^2 \|x_i\|_2^2 (\sigma_t^2 + \|\theta_{t,T}^\Delta\|_{x_i x_i^T}^2). \tag{10}$$

In expectation, the third term is

$$- \mathbb{E}\eta_i(\theta_i - \theta_T^*)^T (I - \eta_i x_i x_i^T) x_i x_i^T \theta_{t,T}^\Delta$$
$$= -\mathbb{E}(1 - \eta_i \|x_t\|_2^2)\eta_i(\theta_i - \theta_T^*)^T x_i x_i^T \theta_{t,T}^\Delta. \tag{11}$$

Now we discuss term (10) and (11), respectively. (11) is independent of σ_t^2 and it is a dynamic effects depending on the current estimate θ_i. That means (11) is independent of the task difficulty and its constantly changes. When $(\theta_i - \theta_T^*)^T x_t x_t^T (\theta_t^* - \theta_T^*)^T < 0$, the task t has a larger prediction gain. This is when the gradient descent direction is consistent in both target and the task t.

For term (10), we notice that task difficulty σ_t^2 and transfer distance $\Delta_{t,T}$ play equal importance in the prediction gain measure regardless of the number of observations.

Optimality of prediction gain. Let t^* be the optimal task defined by

$$t^* = \arg\min_t \Delta_{t,T}^2 + \frac{d\sigma_t^2}{N}.$$

We consider an averaging SGD algorithm with a step size $\eta_i = 1/i$. In general, let $\bar{\theta}_N = \sum_{i=1}^N \theta_i/N$. The following Theorem shows that the performance of the averaging SGD with an accurate prediction-gain based task scheduler matches the minimax lower bound in Theorem 1.

Theorem 6. *Assume* $\Sigma_1 = \cdots = \Sigma_T = I$. *Assume* $\|\theta_t^*\|_2^2 \leq C_5$ *for all* t. *Given* T *tasks with noise levels* $\sigma_1^2, \ldots, \sigma_T^2$ *and transfer distance* $\Delta_{1,T}, \ldots, \Delta_{T,T}$, *let* $\bar{\theta}_N$ *be the averaging SGD estimator with an accurate prediction-gain based task scheduler defined above. We have*

$$G_T(\bar{\theta}_N) \lesssim \Delta_{t^*,T}^2 + \frac{(d\sigma_{t^*}^2 + C_5)\log(N)}{N}. \tag{12}$$

Theorem 6 gives an upper bound on $G_T(\bar{\theta}_N)$ that matches the lower bound in Theorem 1.

5.1. Simulation Studies

To compliment the theoretical analyses, we conduct simulations studies by applying actual SGD with tasks chosen to maximize the local prediction gain. We consider two SGD scenarios: 1) assuming the algorithm has the accurate estimate on the prediction gain as in our analysis; 2) algorithms that have to estimate prediction gain.

For the second scenario, we follow Graves et al. (2017), which regard the task scheduling as a sequential decision-making problem. A popular choice of agent is to use adversarial bandit model. To be specific, we use EXP3 algorithm. See Appendix H for details of the algorithm. Bandit algorithm runs by maximizing rewards. In our experiments, let θ_i be the estimate at the step i. We sample one observation (x_i, y_i) from the target task after each gradient descent, and the reward r_i at the step i is given by

$$r_i = (y_i - \theta_{i-1}^T x_i)^2 - (y_i - \theta_i^T x_i)^2.$$

To evaluate the accurate prediction gain, we directly calculate the distance $\|\theta_i - \theta_t^*\|^2$.

Following the setup throughout the paper, we consider a multitask linear regression problem. We set $T = 5$ and $\sigma_t^2 = 0.001, 0.01, 0.1, 1, 1$ for $t = 1, \ldots, 5$, respectively. Note that the 5-th task is the target task. We test the effects of total number of observations $n = 10, 50, 100, 500, 1000$ and the effects of dimension $d = 5, 10, 50, 100$. By default, we set $n = 1000$ and $d = 5$. The true parameters of all the tasks are sampled from $\mathcal{N}(0, 0.001I_d)$. On expectation, the transfer distance $\Delta_{t,T}^2$ between task t and the target task is about $0.01d$. The input x's are sampled from the same distribution $\mathcal{N}(0, I_d)$ for all the tasks.

Figure 2 shows the L_2 distance of the final estimate and the true parameters of the target task. Our simulation results suggest that 1) prediction-gain based task scheduler can
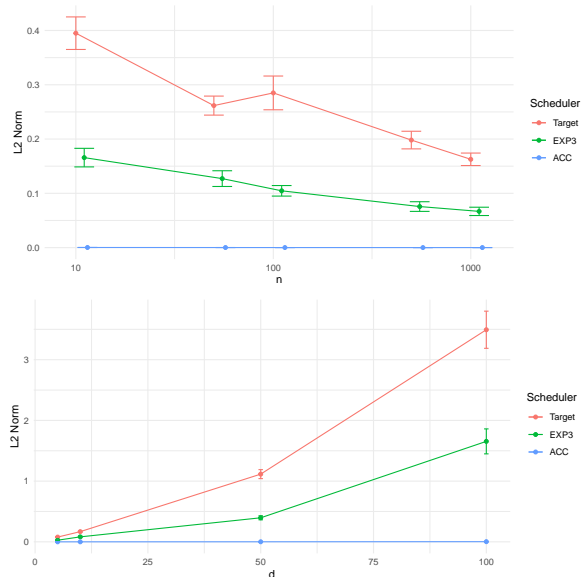

Figure 2. L_2 distance between the final estimate θ_i and θ_T^* under different total numbers of observations n and different numbers of dimensions d. The confidence intervals are the standard deviation of 1000 independent runs.

significantly improve the performance over the target task, when there exists some source task with low transfer distance and low noise; 2) there is still benefits when scheduler has to adaptive select tasks which coincidences our Theorem 3. The results are robust under different choices of n and d.

6. Discussion

In this paper, we discussed the benefits of Curriculum Learning under two special settings: multitask linear regression and multitask representation learning. In the multitask linear regression setting, it is fundamentally hard to adaptively identify the optimal source task to transfer. In the multitask representation learning setting, a good curriculum is the curriculum that diversifies the source tasks. We show that the extra error caused by the adaptive learning is small and it is possible to achieve a near-optimal curriculum. Then we provided theoretical justification for the popular prediction-gain driven task scheduler that has been used in the empirical work.

Our results suggest some natural directions for future work. We show a lower bound (Thm. 5) on the diversity in the multitask representation learning setting, while leaving a gap of d compared to our upper bound (Thm. 4). We believe this gap is because a loose construction of the hard cases that ignores the difficulty of learning the shared representation. Another direction is to show whether prediction-gain methods with no accurate gain estimation could still have performance close to lower bounds for the adaptive learning

setting.

An important direction is to consider the how the order of presenting tasks affects the learning performance. Since the order of tasks are irrelevant for the analysis on empirical risk minimizer, one have to analyze the actual benefits in terms of optimization.

References

Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.

Bhatt, H. S., Rajkumar, A., and Roy, S. Multi-source iterative adaptation for cross-domain classification. In *IJCAI*, pp. 3691–3697, 2016.

Cioba, A., Bromberg, M., Wang, Q., Niyogi, R., Batzolis, G., Shiu, D.-s., and Bernacchia, A. How to distribute data across tasks for meta-learning? *arXiv preprint arXiv:2103.08463*, 2021.

David, S. B., Lu, T., Luu, T., and Pál, D. Impossibility theorems for domain adaptation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 129–136. JMLR Workshop and Conference Proceedings, 2010.

Du, S. S., Hu, W., Kakade, S. M., Lee, J. D., and Lei, Q. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*, 2020.

Gong, C., Tao, D., Maybank, S. J., Liu, W., Kang, G., and Yang, J. Multi-modal curriculum learning for semi-supervised image classification. *IEEE Transactions on Image Processing*, 25(7):3249–3260, 2016.

Graves, A., Bellemare, M. G., Menick, J., Munos, R., and Kavukcuoglu, K. Automated curriculum learning for neural networks. In *international conference on machine learning*, pp. 1311–1320. PMLR, 2017.

Jin, C., Netrapalli, P., Ge, R., Kakade, S. M., and Jordan, M. I. On nonconvex optimization for machine learning: Gradients, stochasticity, and saddle points. *Journal of the ACM (JACM)*, 68(2):1–29, 2021.

Kalan, S. M. M., Fabian, Z., Avestimehr, A. S., and Soltanolkotabi, M. Minimax lower bounds for transfer learning with linear and one-hidden layer neural networks. *Advances in Neural Information Processing Systems*, 33: 1959–1969, 2020.

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

Liu, C., Wang, Z., Sahoo, D., Fang, Y., Zhang, K., and Hoi, S. C. Adaptive task sampling for meta-learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pp. 752–769. Springer, 2020.

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

Narvekar, S., Peng, B., Leonetti, M., Sinapov, J., Taylor, M. E., and Stone, P. Curriculum learning for reinforcement learning domains: A framework and survey. *Journal of Machine Learning Research*, 21:1–50, 2020.

Persello, C. and Bruzzone, L. Active learning for domain adaptation in the supervised classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 50(11):4468–4483, 2012.

Sachan, M. and Xing, E. Easy questions first? a case study on curriculum learning for question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 453–463, 2016.

Settles, B. Active learning literature survey. 2009.

Settles, B. From theories to queries: Active learning in practice. In *Active learning and experimental design workshop in conjunction with AISTATS 2010*, pp. 1–18. JMLR Workshop and Conference Proceedings, 2011.

Sun, S., Shi, H., and Wu, Y. A survey of multi-source domain adaptation. *Information Fusion*, 24:84–92, 2015.

Tang, Y., Wang, X., Harrison, A. P., Lu, L., Xiao, J., and Summers, R. M. Attention-guided curriculum learning for weakly supervised classification and localization of thoracic diseases on chest radiographs. In *International Workshop on Machine Learning in Medical Imaging*, pp. 249–258. Springer, 2018.

Teshima, T., Sato, I., and Sugiyama, M. Few-shot domain adaptation by causal mechanism transfer. In *International Conference on Machine Learning*, pp. 9458–9469. PMLR, 2020.

Tripuraneni, N., Jordan, M. I., and Jin, C. On the theory of transfer learning: The importance of task diversity. *Advances in Neural Information Processing Systems*, 33: 7852–7862, 2020.

Wang, M. and Deng, W. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018.

Weinshall, D. and Amir, D. Theory of curriculum learning, with convex loss functions. *Journal of Machine Learning Research*, 21(222):1–19, 2020.

Xu, Z. and Tewari, A. Representation learning beyond linear prediction functions. *Advances in Neural Information Processing Systems*, 34, 2021.

Xu, Z., Meisami, A., and Tewari, A. Decision making problems with funnel structure: A multi-task learning approach with application to email marketing campaigns. In *International Conference on Artificial Intelligence and Statistics*, pp. 127–135. PMLR, 2021.

Yao, J., Killian, T., Konidaris, G., and Doshi-Velez, F. Direct policy transfer via hidden parameter markov decision processes. In *LLARLA Workshop, FAIM*, volume 2018, 2018.

A. Proof of Theorem 1

Proof. Our proof is inspired by the proof of (Kalan et al., 2020), which gives a lower bound construction for the two-tasks transfer learning problem. Our results can be seen as an extension of their constructions to multiple-source tasks setting.

We define the optimal task

$$t^* = \arg\min_t \{\boldsymbol{Q}_t^2 + \frac{d\sigma_t^2}{N}\}.$$

Let $\delta^2 = (\boldsymbol{Q}_{t^*}^2 + \frac{d\sigma_{t^*}^2}{N})/64$. In general, we construct $T \times M$ parameters $\{\theta_{t,i}\}_{t\in[T],i\in[M]}$ with the t-th row corresponding to the hypothesis set of the t-th task.

We start by constructing the the hypothesis set of the target task and the task t^*. Let $\delta' = \boldsymbol{Q}_{t^*}/16 + \delta$. By definition, we have $\delta' \leq 1.5\delta$.

Consider the set $\Theta = \{\theta : \|\theta\|_2 \leq 2\delta'\}$. Let $\{\theta_{t^*,1}, \ldots, \theta_{t^*,M}\}$ be a δ'-packing of the set in the L_2-norm ($\|\theta_{t^*,i} - \theta_{t^*,j}\|_2 \geq \delta'$). We can find the packing with $\log(M) \leq d\log(2)$. Since $\theta_{t^*,i}, \theta_{t^*,j} \in \Theta$, we also have $\|\theta_{t^*,i} - \theta_{t^*,j}\|_2 \leq 4\delta'$ for any $i, j \in [M]$.

Now we construct hypothesis set for the target task. For all $i \in [M]$, we choose $\theta_{T,i}$ such that $\|\theta_{T,i} - \theta_{t^*,i}\|_2 = \boldsymbol{Q}_{t^*}/16$. So the construction for the target tasks satisfies

$$\|\theta_{T,i} - \theta_{T,j}\|_2 \geq \delta' - \boldsymbol{Q}_{t^*}/16 \geq \delta/2 \quad \text{and} \quad \|\theta_{T,i} - \theta_{T,j}\|_2 \leq 4\delta' + \boldsymbol{Q}_{t^*}/16 \leq 5\delta'.$$

Now we discuss two cases. For any task t with $\boldsymbol{Q}_t \geq 5\delta'$, we randomly pick a parameter in the hypothesis set of the target task which we denote by $\tilde{\theta}_t$ and we set all $\theta_{t,i} = \tilde{\theta}_t$ for all $i \in [M]$. This construction is valid since any $\|\theta_{t,i} - \theta_{T,i}\|_2 \leq 5\delta' \leq \boldsymbol{Q}_t$.

For any task t with $\boldsymbol{Q}_t \leq 5\delta'$, we will use the same construction as we use for t^*.

Let J be a random variable uniformly over $[M]$ representing the true hypothesis. The samples for each task t is i.i.d. generated from the linear model described in Section 3.1 with a parameter $\theta_{t,J}$. Our goal is to show that on expectation, any algorithm will perform badly as in Theorem 1.

Let E_t be a random sample from task t given the true parameter being $\theta_{t,J}$. Similarly to (5.2) in (Kalan et al., 2020), using Fano's inequality, we can conclude that

$$R_T^N(\Theta(\boldsymbol{Q})) \geq \delta^2 \left(1 - \frac{\log(2) + \sum_{t=1}^T n_t I(J; E_t)}{\log(M)} \right). \tag{13}$$

We proceed by giving an uniform bound on the mutual information. We will need the following lemma to upper bound the mutual information term.

Lemma 2 (Lemma 1 in (Kalan et al., 2020)). *The mutual information between J and any sample E_t can be upper bounded by $I(J; E_t) \leq \frac{1}{M^2} \sum_{i,j} D_{KL}\left(\mathbb{P}_{\theta_{t,i}} \| \mathbb{P}_{\theta_{t,j}}\right)$, where $\mathbb{P}_{\theta_{t,i}}$ is the induced distribution by the parameter $\theta_{t,i}$. Furthermore we have*

$$D_{KL}\left(\mathbb{P}_{\theta_{t,i}} \| \mathbb{P}_{\theta_{t,j}}\right) = \|\Sigma_t^{1/2}(\theta_{t,i} - \theta_{t,j})\|_2^2/(2\sigma_t^2) \leq C_0 \|\theta_{t,i} - \theta_{t,j}\|_2^2/(2\sigma_t^2).$$

Using Lemma 2, we bound the mutual information of any task t.

Lemma 3. *Under the constructions introduced above, the mutual information*

$$I(J; E_t) \leq \frac{512 C_0}{7\sigma_{t^*}^2} \delta'^2 \text{ for all } t \in [T].$$

Proof. For any task in the first case discussed above ($Q_t \geq 5\delta'$), the mutual information $I(J; E_t)$ is 0. Thus the statement holds trivially.

Now we discuss the second case above. By definition, we have

$$\boldsymbol{Q}_t^2 + \frac{d\sigma_t^2}{N} \geq \boldsymbol{Q}_{t^*}^2 + \frac{d\sigma_{t^*}^2}{N} = 64\delta^2. \tag{14}$$

Note that

$$\boldsymbol{Q}_t \leq 5\delta' = 5(\boldsymbol{Q}_{t^*,T}/16 + \delta) \leq 7.5\delta.$$

Plugging back into (14), we have $d\sigma_t^2/N \geq 7\delta^2$, and by definition we have $d\sigma_{t^*}^2/N \leq 64\delta^2$. Therefore, we have $\sigma_t^2 \geq 7\sigma_{t^*}^2/(64)$.

Since the constructions are the same for the second case, the mutual information can be uniformly bounded by

$$I(J, E_t) \leq \frac{1}{M^2} \sum_{i,j} \frac{32C_0}{7\sigma_{t^*}^2} \|\theta_{t^*,i} - \theta_{t^*,j}\|_2^2 \leq \frac{512C_0}{7\sigma_{t^*}^2} \delta'^2.$$

$\square$

Finally, we follow the analysis in Section 7.4 of (Kalan et al., 2020). Using Lemma A on Equation (13), we have

$$R_T^N(\Theta(\boldsymbol{Q})) \geq \delta^2 \left(1 - \frac{\log(2) + N\frac{512C_0}{7\sigma_{t^*}^2}\delta'^2}{\log(M)}\right).$$

Plugging in $\delta' = Q_{t^*}/16 + \delta$, we can conclude

$$R_T^N(\Theta(\boldsymbol{Q})) \gtrsim \boldsymbol{Q}_{t^*}^2 + \frac{d\sigma_{t^*}^2}{N}.$$

$\square$

B. Proof of Theorem 2

Proof. We first show the lower bound of the first term within the maximization. We construct the following problem: we have $T-1$ tuples of parameters $\{(\theta_{1,i}, \ldots, \theta_{T,i})\}_{i=1}^{T-1}$, where $\theta_{t,i}$ corresponds to the parameters of the t-th task. Let $\{\tilde{\theta}_i\}_{i=1}^{T-1}$ be a set of parameters that are 2δ-separated for some $\delta > 0$. The parameters of our source and target tasks are chosen in the following manners:

1. $\theta_{t,i} = \tilde{\theta}_t$ for all $t \in [T-1]$.

2. $\theta_{T,i} = \tilde{\theta}_i$ for all $i \in [T-1]$.

Two important properties of this construction is that 1) there is always one source task that is identical to the target task; 2) the information from source tasks can not help learn the target task.

Let J follow the uniform distribution over $[T-1]$. Assume we have $n_1, \ldots, n_M$ and n_T be the number of observations for $T-1$ source tasks and target task from parameter $(\theta_{1,J}, \ldots, \theta_{T-,J})$, respectively.

Proposition 2. *Since J is independent of $(\theta_{1,J}, \ldots, \theta_{T-1,J})$, we have the mutual information $I(J; \theta_{1,J}, \ldots, \theta_{T-1,J}) = 0$.*

Let ψ be any test statistics that maps our dataset to an index. For all ψ, by Fano's Lemma, we can conclude that

$$\tilde{R}_T^N(\tilde{\Theta}(\boldsymbol{Q})) \geq \delta^2 \frac{1}{M} \sum_{i=1}^{M} \mathbb{P}\{\psi(S_1^{n_1}, \ldots, S_M^{n_M}, S_T^{n_T}) \neq j\}$$

$$\geq \delta^2 \left(1 - \frac{I(J; \psi(S_1^{n_1}, \ldots, S_T^{n_T})) + \log(2))}{\log(T-1)}\right) \tag{15}$$

To proceed, we analyze the mutual information

$$
\begin{aligned}
&I(J; \psi(S_1^{n_1}, \ldots, S_T^{n_T})) \\
&\leq I(J; S_1^{n_1}, \ldots, S_T^{n_T}) \\
&\quad \text{(By the independence of } S_1^{n_1}, \ldots, S_{T-1}^{n_{T-1}} \text{ and } S_T^{n_T}) \\
&\leq I(J; S_1^{n_1}, \ldots, S_{T-1}^{n_{T-1}}) + I(J; S_T^{n_T}) \\
&= I(J; S_T^{n_T}).
\end{aligned}
$$

Let E be a random sample from the target task. We follow the analysis from (Kalan et al., 2020), which construct $\tilde{\theta}$ by the 2δ-packing of the set

$$
\{\theta : \theta \in \mathbb{R}^d, \|\theta\|_2 \leq 4\delta\}.
$$

Then we can find such packing as long as $T - 1 \leq d \log(2)$. The mutual information by the above construction gives

$$
I(J; S_T^{n_T}) \leq n_T I(J; E) \leq n_T \frac{32\delta^2}{\sigma_T^2}.
$$

By choosing the optimal $\delta^* = \log((T-1)/2)\sigma_T^2/(64 n_T)$, we have for some $c > 0$,

$$
\tilde{R}_T^N \gtrsim \frac{\sigma_T^2 \log(T-1)}{n_T}.
$$

Note that the lower bound by Theorem 1 still applies here. In total, since $n_T \leq N$, we have

$$
\tilde{R}_T^N \gtrsim \frac{\sigma_T^2 \log(T)}{N} + \min_t \frac{d\sigma_t^2}{N}.
$$

The above analysis only works when $Q_{sub}^2 \geq \delta^* = \log((T-1)/2)\sigma_T^2/(64 n_T)$. Otherwise, one will at least suffer Q_{sub}^2 plus the learning difficulty term $\min_t \frac{d\sigma_t^2}{N}$. □

C. Proof of Theorem 3

Since the number of observations for each source task is $N/(2T - 2)$, we notice that

$$
L_t(\hat{\theta}_{t^*}) - L_t(\theta_{t^*}^*) \lesssim \frac{C_0 d T \sigma_{t^*}^2 \log(T/\delta)}{N}.
$$

Using Assumption 1 and the definition of the loss function, we have

$$
\|\hat{\theta}_{t^*} - \theta_{t^*}^*\|_2^2 \leq \frac{C_0 d T \sigma_{t^*}^2 \log(T/\delta)}{C_1 N}.
$$

The proof of Theorem 3 is similar to many proofs of generalization bound. We let the empirical loss on the target task w.r.t θ be

$$
\hat{L}_T(\theta) = \frac{2}{N} \sum_{i=1}^{N/2} \left(Y_{T,i} - X_{T,i}^T \theta\right)^2.
$$

Write $Y_{T,i} = X_{T,i}^T \theta_T^* + \epsilon_{T,i}$. Let $\hat{\Sigma}_T = \frac{1}{n_T} \sum_{i=1}^{n_T} X_{T,i} X_{T,i}^T$ be the sample covariance matrix. We start with

$$
\begin{aligned}
&L_T(\hat{\theta}_{t^*}) - \min_{t \in [T-1]} L_T(\hat{\theta}_t) \\
&\leq \hat{L}_T(\hat{\theta}_{t^*}) - \min_{t \in [T-1]} \hat{L}_T(\hat{\theta}_t) + 2 \max_{t \in [T-1]} |L_T(\hat{\theta}_t) - \hat{L}_T(\hat{\theta}_t)| \\
&= 2 \max_{t \in [T-1]} |L_T(\hat{\theta}_t) - \hat{L}_T(\hat{\theta}_t)|,
\end{aligned}
$$

where the last equality is based on the definition of t^*.

Now we bound the difference term. Let $n_T = N/2$.

Lemma 4. *With a probability at least $1 - \delta$, we have*

$$|L_T(\hat{\theta}_t) - \hat{L}_T(\hat{\theta}_t)| \lesssim \frac{C_1 C_2^2 d \log(T/\delta) \sigma_T^2}{n_T} + \sqrt{\frac{d + \log(\delta)}{n_T}} \text{ for all } t \in [T-1].$$

Proof. We make the following decomposition.

$$|L_T(\hat{\theta}_t) - \hat{L}_T(\hat{\theta}_t)|$$

$$= |\|\hat{\theta}_t - \theta_T^*\|_{\Sigma_T}^2 + \sigma_T^2 - \frac{1}{n_T}\sum_{i=1}^{n_T}[X_{T,i}^T(\hat{\theta}_t - \theta_T^*) - \epsilon_{T,i}]^2|$$

$$\le \|\hat{\theta}_t - \theta_T^*\|_{\Sigma_T - \hat{\Sigma}_T}^2 + |\sigma_T^2 - \frac{1}{n_T}\sum_{i=1}^{n_T}\epsilon_{T,i}^2| + \frac{1}{n_T}|\sum_{i=1}^{n_T}X_{T,i}^T(\hat{\theta}_t - \theta_T^*)\epsilon_{T,i}|.$$

Now we bound the three terms above separately. The second term is the concentration for $\chi^2(n_T)$ distribution. We have with a probability at least $1 - \delta/(3T)$, $|\sigma_T^2 - \frac{1}{n_T}\sum_{i=1}^{n_T}\epsilon_{T,i}^2| \lesssim \sigma_T^2(\sqrt{\log(3T/\delta)/n_T} + \log(3T/\delta)/n_T)$.

To proceed, we consider the concentration of sample covariance matrix.

Lemma 5 (Matrix Hoeffding's inequality). *Let $X_1, \ldots, X_n$ be centered, independent, symmetric, $d \times d$ random matrices that are sub-Gaussian with parameters $V_1, \ldots, V_n$. Then for all $\delta > 0$ with a probability $1 - \delta$,*

$$\|\frac{1}{n}\sum_{i=1}^{n}X_i\|_{op} \le \sqrt{\log(2d/\delta)\frac{2\sigma^2}{n}}, \text{ where } \sigma^2 = \|\frac{1}{n}\sum_{i=1}^{n}V_i\|.$$

Using the boundedness of both $\hat{\theta}_t$ and θ_T^*, $\|\hat{\theta}_t - \theta_T^*\|_2 \lesssim C_2$. Then applying Lemma 5, we have with a probability at least $1 - \delta/(3T)$,

$$\|\hat{\theta}_t - \theta_T^*\|_{\Sigma_T - \hat{\Sigma}_T}^2 \lesssim C_2^2 C_0 \frac{\log(Td/\delta)}{n_T}.$$

For the third term, we apply the martingale concentration inequality on the sum $\sum_{i=1}^{n_T}X_{T,i}^T(\hat{\theta}_t - \theta_T^*)\epsilon_{T,i}$, we have with a probability at least $1 - \delta/(3T)$, we have for all $t \in [T-1]$

$$\frac{1}{n_T}|\sum_{i=1}^{n_T}X_{T,i}^T(\hat{\theta}_t - \theta_T^*)\epsilon_{T,i}| \lesssim \frac{C_0 C_2 \sigma_T^2 \log(T/\delta)}{n_T}.$$

$\square$

D. Proof of Proposition 1

Proof. The target task is basically minimizing the empirical loss over $T - 1$ estimators. We first apply the standard generalization bound with Radermacher complexity

$$L_T(\hat{f}_{t^*}) \le \min_{t \in [T-1]} L_T(\hat{f}_t) + \sqrt{\frac{2\log(T-1)}{N}} + c\sqrt{\frac{2\log(1/\delta)}{N}},$$

where c is a universal constant.

To proceed, we bound $\min_{t \in [T-1]} L_T(\hat{f}_t)$. By the assumption, we have $L^* = \min_t L_T(f_t^*)$. Let the task that realizes the minimization be t'. Using Assumption 2, we have

$$\min_{t \in [T-1]} L_T(\hat{f}_t) \le L_T(\hat{f}_{t'}) = \mathbb{E}_{(X_T, Y_T)}l(\hat{f}_{t'}, Y_T) \le L^* + L_1\mathbb{E}_{X_T}\|\hat{f}_{t'} - f_{t'}^*\|^2 \le L^* + \frac{L_1}{L_2}(L_{t'}(\hat{f}_{t'}) - L_{t'}^*).$$

We can apply the generalization bound on $L_{t'}(\hat{f}_{t'}) - L_{t'}^*$, which gives us the result.

$\square$

E. Proof of Theorem 4

E.1. Proof of Lemma 1

We will borrow some techniques from (Du et al., 2020) for the proof of Theorem 4. We start with the proof of Lemma 1, which provides a valid confidence set for the unknown parameters $B^*\beta_t^*$. First, we let $X_{i,t}^{(1)}$ be the covariance matrix of the first split $S_{i,t}^{(1)}$.

Claim 1 (Covariance concentration on the first split.). *For $\delta \in (0,1)$, there exists a constant $\gamma_1 > 0$ such that with a probability at least $1 - \delta/10$, we have*

$$0.9\Sigma_t \prec \frac{2}{n_{i,t}}X_{i,t}^{(1)T}X_{i,t}^{(1)} \prec 1.1\Sigma_t \quad \text{for all } i \in \{i' \in [N] : n_{i',t} \geq \gamma_1(d + \log(N/\delta))\}.$$

Claim 2 (Covariance concentration on the second split.). *For $\delta \in (0,1)$, there exists some $\gamma_2 > 0$ such that for any given $B \in \mathbb{R}^{d \times 2k}$ that is independent of $X_{i,t}^{(2)}$, with a probability at least $1 - \delta/10$, we have*

$$0.9B^T\Sigma_t B \prec \frac{2}{n_{i,t}}B^T X_{i,t}^{(2)T}X_{i,t}^{(2)}B \prec 1.1B^T\Sigma_t B \quad \text{for all } i \in \{i' \in [N] : n_{i',t} \geq \gamma_2(d + \log(N/\delta)).$$

Let $\gamma = \max\{\gamma_1, \gamma_2\}$. Recall that $\kappa = C_0/C_1$. Then the good events in Claim 1 and 2 hold for all $i : n_{i,*} \geq \lceil\gamma(d + \log(N/\delta))\rceil$. We first apply the Claim A.3 in (Du et al., 2020), which guarantees the loss on the source training data. We rephrase it here as Lemma 6. Note that the only difference is that we require the good events hold for all $i : n_{i,t}$'s are sufficiently large.

Lemma 6 (Claim A3 in (Du et al., 2020)). *With a probability at least $1 - \delta/5$, we have*

$$\sum_{t=1}^{T}\|X_{i,t}^{(1)}(\hat{B}_i\hat{\beta}_{i,t} - B^*\beta_t^*)\|_2^2 \lesssim \sigma^2\left(kT + dk\log(\kappa i/T) + \log(N/\delta)\right) \quad \text{for all } i : n_{i,*} > \lceil\gamma(d + \log(N/\delta))\rceil. \quad (16)$$

Note that $X_{i,t}^{(1)}\hat{B}_i\hat{\beta}_{i,t} = P_{X_{i,t}^{(1)}\hat{B}_i}Y_t = P_{X_{i,t}^{(1)}\hat{B}_i}(X_{i,t}^{(1)}B^*\beta_t^* + z_t)$. To proceed, for any fixed $t' \in [T]$, we have

$$\sigma^2\left(kT + dk\log(\kappa i/T) + \log(N/\delta)\right)$$

$$\gtrsim \sum_{t=1}^{T}\|X_{i,t}^{(1)}(\hat{B}_i\hat{\beta}_{i,t} - B^*\beta_t^*)\|_2^2$$

$$= \sum_{t=1}^{T}\|P_{X_t\hat{B}_i}(X_{i,t}^{(1)}B^*\beta_t^* + z_t)\|_2^2$$

$$\geq \sum_{t=1}^{T}\|P_{X_t\hat{B}_i}X_{i,t}^{(1)}B^*\beta_t^*\|_2^2$$

$$\geq 0.9\sum_{t=1}^{T}\frac{n_{i,t}}{2}\|P_{\Sigma_t^{1/2}\hat{B}_i}\Sigma_t B^*\beta_t^*\|_2^2 \quad \text{(Using Claim 1)}$$

$$\geq 0.45C_1\sum_{t=1}^{T}n_{i,t}\|P_{\Sigma_{t'}^{1/2}\hat{B}_i}\Sigma_{t'}B^*\beta_t^*\|_2^2$$

$$= 0.45C_1\sum_{j=1}^{i}\|P_{\Sigma_{t'}^{1/2}\hat{B}_i}\Sigma_{t'}B^*\beta_{t_j}^*\|_2^2$$

Then we have

$$\|\hat{B}_i\hat{\beta}_{t'} - B^*\beta_{t'}^*\|_2^2$$

$$\leq \frac{1}{C_1}\|\Sigma_{t'}^2(\hat{B}_i\hat{\beta}_{t'} - B^*\beta_{t'}^*)\|_2^2$$

$$\leq \frac{1}{0.9C_1 n_{i,t'}}\|X_{i,t'}^{(2)}(\hat{B}_i\hat{\beta}_{t'} - B^*\beta_{t'}^*)\|_2^2 \quad \text{(Using Claim 2)}$$

$$= \frac{1}{0.9C_1 n_{i,t'}}\left(\|P_{X_{i,t'}^{(2)}\hat{B}_i}X_{i,t'}^{(2)}B^*\beta_{t'}^*\|_2^2 + \|P_{X_{i,t'}^{(2)}\hat{B}}z_{t'}\|_2^2\right)$$

For the second term above,

$$\frac{1}{\sigma^2}\|P_{X_{i,t'}^{(2)}\hat{B}}z_{t'}\|_2^2 \sim \chi^2(k).$$

Thus with a probability at least $1 - \delta$, $\|P_{X_{i,t'}^{(2)}\hat{B}}z_{t'}\|_2^2 \lesssim k + \log(NT/\delta)$ for all $t' \in [T]$ and $i > T\lceil\gamma(d + \log(N/\delta))\rceil$. Therefore, we obtain the bound: for all $i > T\lceil\gamma(d + \log(N/\delta))\rceil$ and all $t' \in [T]$, it holds that

$$\|\hat{B}_i\hat{\beta}_{t'} - B^*\beta_{t'}^*\|_2^2 \lesssim \frac{\sigma^2}{C_1}\left(\frac{kT + dk\log(\kappa i/T) + \log(N/\delta)}{C_1 n_{i,t'}/C_5} + \frac{k + \log(NT/\delta)}{n_{i,t'}}\right)$$

$$\lesssim \frac{C_5\sigma^2 dk\log(\kappa N\delta/T)}{C_1^2 n_{i,t'}}.$$

E.2. Proof of the full theorem

Now we prove the full theorem. By Assumption 3, we convert our target $\lambda_d(\sum_{i=1}^N \beta_{t_i}^*\beta_{t_i}^{*T})$ to $\lambda_d(\sum_{i=1}^N B^*\beta_{t_i}^*\beta_{t_i}^{*T}B^{*T})$:

$$\frac{1}{N}\lambda_d(\sum_{i=1}^N \beta_{t_i}^*\beta_{t_i}^{*T}) \geq \frac{1}{NC_4}\lambda_d(\sum_{i=1}^N B^*\beta_{t_i}^*\beta_{t_i}^{*T}B^{*T}).$$

Then we follow the standard decomposition framework of UCB analysis:

$$\frac{1}{N}\lambda_d(\sum_{i=1}^N B^*\beta_{t_i}^*\beta_{t_i}^{*T}B^{*T}) = \frac{1}{N}\left(\lambda_k(\sum_{i=1}^N \tilde{\theta}_i\tilde{\theta}_i^T) + \lambda_k(\sum_{i=1}^N B^{*T}\beta_{t_i}^*\beta_{t_i}^{*T}B^*) - \lambda_k(\sum_{i=1}^N \tilde{\theta}_i\tilde{\theta}_i^T)\right). \tag{17}$$

Our proof proceeds by first showing

$$\frac{1}{N}\lambda_k(\sum_{i=1}^N \tilde{\theta}_i\tilde{\theta}_i^T) \geq \lambda/d,$$

which is usually interpreted as optimism. Then we bound the difference term

$$\frac{1}{N}\left(\lambda_k(\sum_{i=1}^N B^*\beta_{t_i}^*\beta_{t_i}^{*T}B^{*T}) - \lambda_k(\sum_{i=1}^N \tilde{\theta}_i\tilde{\theta}_i^T)\right),$$

which is expected to vanish when N becomes large.

Proof of optimism. We apply Lemma 1 and have $\tilde{\theta}_i \in \mathcal{B}_{i,t}^\alpha$. Since $B^*\beta_t^* \in \mathcal{B}_{i,t}^\alpha$ for all $t \in [T], i \in [N]$, it is easy to show that the greedy selection over $\mathcal{B}_{i,t}^\alpha$ will lead to

$$\lambda_k(\sum_{i=1}^N \tilde{\theta}_i\tilde{\theta}_i^T) \geq \lambda(\frac{N}{k} - 1).$$

We prove by induction. Assume at any step n, we have for all $\|\nu\|_2 = 1$,

$$\nu^T \sum_{i=1}^n \tilde{\theta}_i\tilde{\theta}_i^T \nu \geq \lambda(i/k - 1).$$

We will show that at the step $n + k$, we will at least have

$$\nu^T \sum_{i=1}^{n} \tilde{\theta}_i \tilde{\theta}_i^T \nu \geq \lambda(i/k).$$

The proof is simple, if there exists a ν such that the above inequality fails, we will select a task that brings it to $\lambda(i/k)$. This process can be done at most k times.

Upper bounding the differences. We first write the difference of eigenvalues in terms of the difference of the matrices. We will use a trick here.

$$\lambda_k(\sum_{i=1}^{N} B^* \beta_{t_i}^* \beta_{t_i}^{*T} B^{*T}) - \lambda_k(\sum_{i=1}^{N} \tilde{\theta}_i \tilde{\theta}_i^T)$$

$$= \min_{\|\nu\|_2=1} \nu^T \sum_{i=1}^{N} B^* \beta_{t_i}^* \beta_{t_i}^{*T} B^{*T} \nu - \min_{\|\nu\|_2=1} \nu^T \sum_{i=1}^{N} \tilde{\theta}_i \tilde{\theta}_i^T \nu$$

$$\geq \min_{\|\nu\|_2=1} \nu^T \sum_{i=1}^{N} B^* \beta_{t_i}^* \beta_{t_i}^{*T} B^{*T} \nu - \min_{\|\nu\|_2=1} \nu^T \sum_{i=1}^{N} \tilde{\theta}_i \tilde{\theta}_i^T \nu$$

$$\geq \min_{\|\nu\|_2=1} \left(\nu^T \sum_{i=1}^{N} (B^* \beta_{t_i}^* \beta_{t_i}^{*T} B^{*T} - \tilde{\theta}_i \tilde{\theta}_i^T) \nu \right)$$

$$\geq \sum_{i=1}^{N} \min_{\|\nu\|_2=1} \nu^T (B^* \beta_{t_i}^* \beta_{t_i}^{*T} B^{*T} - \tilde{\theta}_i \tilde{\theta}_i^T) \nu$$

$$\geq -\sum_{i=1}^{N} \|B^* \beta_{t_i}^* - \tilde{\theta}_i\|_2 (\|B^* \beta_{t_i}^*\|_2 + \|\tilde{\theta}_i\|_2)$$

$$\geq -2C_5 \sum_{i=1}^{N} \|B^* \beta_{t_i}^* - \tilde{\theta}_i\|_2.$$

Applying Lemma 1 and by the construction of the confidence set $\mathcal{B}_{i,t}^\alpha$, we have

$$\|B^* \beta_{t_i}^* - \tilde{\theta}_i\|_2 \lesssim \sqrt{\frac{C_5 \sigma^2 dk \log(\kappa N \delta/T)}{C_1^2 n_{i,t}}}.$$

Thus,

$$\lambda_k(\sum_{i=1}^{N} B^* \beta_{t_i}^* \beta_{t_i}^{*T} B^{*T}) - \lambda_k(\sum_{i=1}^{N} \tilde{\theta}_i \tilde{\theta}_i^T) \gtrsim -\sum_i \sqrt{\frac{C_5^2 \sigma^2 dk \log(\kappa N \delta/T)}{C_1^2 n_{i,t}}} \gtrsim -\sum_i \sqrt{\frac{C_5^2 \sigma^2 dk T N \log(\kappa N \delta/T)}{C_1^2}}.$$

Plugging this back to the decomposition term (17) we arrive the final bound.

F. Proof of Theorem 5

Assume we have T tasks in total. We pick a set of orthogonal vectors $\{\beta_1, \ldots, \beta_k\} \in \mathbb{R}^k$ with $\|\beta_i\|_2^2 = \lambda$. We first construct a simple instance in the following way: the first k tasks are diverse such that $(\beta_1^*, \ldots, \beta_k^*) = (\beta_1, \ldots, \beta_k)$. Then all the other tasks share the same parameter β_1. We denote the instance by v. This construction is hard for naive task scheduler that evenly allocates samples to all the tasks since the direction for β_1^* will be over-exploited.

We evenly divide T tasks into $M = \lfloor T/k \rfloor$ blocks. Let T_m be the total number of visits in the m-th block. For any task scheduler, there exists $m' \in [M]$ such that $\mathbb{E}[T_m] \leq \frac{N}{M}$ by pigeonhole theorem.

Then we construct another instance denoted by v' such that v is the same as v' except for that the m-th block has the parameters $(2\beta_1, \ldots, 2\beta_k)$ for the k tasks in the block.

Let P_v and $P_{v'}$ be the probability measure on the linear regression model with true parameter defined in Section 3.1 for v and v'.

Define $\Delta_{N,k}(\mathcal{T}, v)$ be the expected difference using task scheduler $\mathcal{T}$ on instance v, i.e.

$$\Delta_{N,k}(\mathcal{T}, v) := \max_{t_1, \ldots, t_N} \lambda_k \left(\sum_{i=1}^{N} \beta_{t_i}^* \beta_{t_i}^{*T} \right) - \lambda_{N,k}.$$

Thus, applying Bretagnolle–Huber inequality (Theorem 14.2 (Lattimore & Szepesvári, 2020)) we have

$$\Delta_{N,k}(\mathcal{T}, v) + \Delta_{N,k}(\mathcal{T}, v') \geq \frac{N\lambda}{2k} (P_v(T_1 \leq N/M) + P_{v'}(T_1 > N/M)) \geq \frac{N\lambda}{2k} \exp(-D(P_v, P_{v'})).$$

where $D(P, Q)$ is the relative entropy between distributions P and Q.

Then we apply Lemma 15.1 (Lattimore & Szepesvári, 2020), which we rephrase here.

Lemma 7. *Let P_t and P_t' be the probability measure of the t-th task using true parameters from v and v', respectively. We also let $\bar{T}_t$ be the number of observations on the t-th task. Then we have*

$$D(P_v, P_{v'}) = \sum_{t=1}^{T} \mathbb{E}_v[\bar{T}_t] D(P_t, P_t') \leq \mathbb{E}_v[T_m] \max_{t=m(k-1)+1, \ldots, mk} D(P_t, P_t').$$

Now since $D(P_t, P_t') = \|\beta_t - \beta_t^*\|^2/(2\sigma^2) = \lambda^2/(2\sigma^2)$, we have

$$\Delta_{N,k}(\mathcal{T}, v) + \Delta_{N,k}(\mathcal{T}, v') \geq \frac{N\lambda}{2k} \exp\left(\frac{N\lambda^2}{2M\sigma^2}\right).$$

Choosing $\lambda = \frac{2M\sigma^2}{N}$, we have

$$\Delta_{N,k}(\mathcal{T}, v) + \Delta_{N,k}(\mathcal{T}, v') \geq \sigma\sqrt{NM}/k = \sigma\sqrt{NT/k^3}.$$

G. Proof of Theorem 6

Proof. We follow the standard procedure of the convergence analysis of SGD. Let t_i be the task that the task scheduler chooses at the step i. Let $v_i^{(t)}$ be the virtual gradient calculated at the step i if task t is scheduled, i.e.

$$v_i^{(t)} = x_i^{(t)T}(\theta_t^* - \theta_i)x_i^{(t)} + \epsilon_i^{(t)} x_i^{(t)},$$

where $\epsilon_i^{(t)}$ and $x_i^{(t)}$ is the random noise and the input sampled at the step i from task t. To start with, let $\theta_{i+1}^{(t)}$ be the virtual next step if task t is scheduled. We have

$$\theta_{i+1}^{(t)} - \theta_T^* = \theta_i - \theta_T^* - \eta_i v_i^{(t)}.$$

By algebra, we derive

$$\begin{aligned}
&\|\theta_{i+1}^{(t)} - \theta_T^*\|^2 - \|\theta_i - \theta_T^*\|^2 \\
&= -2\eta_i(\theta_i - \theta_T^*)^T x_i^{(t)} x_i^{(t)T}(\theta_i - \theta_T^*) + \\
&\quad 2\eta_i(\theta_i - \theta_T^*)^T x_i^{(t)} x_i^{(t)T}(\theta_t^* - \theta_T^*) - \\
&\quad 2\eta_i x_i^{(t)T}(\theta_i - \theta_T^*)\epsilon_i + \eta_i^2 \|v_i^{(t)}\|^2.
\end{aligned}$$

Taking expectations over $x_i^{(t)}$ and $\epsilon_i^{(t)}$ and arrange the equation, we have

$$\|\theta_i - \theta_T^*\|^2 = \frac{\|\theta_i - \theta_T^*\|^2 - \mathbb{E}_{t,i}\|\theta_{i+1}^{(t)} - \theta_T^*\|^2}{2\eta_i} + (\theta_i - \theta_T^*)^T(\theta_t^* - \theta_T^*) + \frac{\eta_i}{2}\|v_i^{(t)}\|^2., \tag{18}$$

where $\mathbb{E}_{t,i}$ takes marginal expectation over the randomness of $x_i^{(t)}$ and $\epsilon_i^{(t)}$. Note that

$$L_T(\theta_i) - \sigma_T^2 = \|\theta_i - \theta_T^*\|^2.$$

Plugging this into Equ. (18), we have

$$L_T(\theta_i) - \sigma_T^2 = \frac{\|\theta_i - \theta_T^*\| - \mathbb{E}_{t,i}\|\theta_{i+1}^{(t)} - \theta_T^*\|^2}{2\eta_i} + (\theta_i - \theta_T^*)^T(\theta_t^* - \theta_T^*) + \frac{\eta_i}{2}\|v_i^{(t)}\|^2.$$

Since this holds for any t, we let $t = t^*$ and note that by the definition of the task scheduler with accurate prediction gain estimate,

$$\mathbb{E}\|\theta_i - \theta_T^*\|^2 \le \mathbb{E}\|\theta_i^{(t)} - \theta_T^*\|^2.$$

We have

$$\begin{aligned}
&\mathbb{E}[L_T(\theta_i)] - \sigma_T^2 \\
&\le \mathbb{E}\frac{\|\theta_i^{(t^*)} - \theta_T^*\|^2 - \|\theta_{i+1}^{(t^*)} - \theta_T^*\|^2}{2\eta_i} + \mathbb{E}(\theta_i - \theta_T^*)^T(\theta_{t^*}^* - \theta_T^*) + \frac{\eta_i}{2}\mathbb{E}\|v_i^{(t^*)}\|^2.
\end{aligned}$$

Summing over all $i = 1, \ldots, N$, we have

$$\begin{aligned}
&\sum_i \mathbb{E}[L_T(\theta_i)] - N\sigma_T^2 \\
&\le \sum_{i=1}^N \mathbb{E}\frac{\|\theta_i^{(t^*)} - \theta_T^*\|^2 - \|\theta_{i+1}^{(t^*)} - \theta_T^*\|^2}{2\eta_i} + \sum_{i=1}^N \mathbb{E}(\theta_i - \theta_T^*)^T(\theta_{t^*}^* - \theta_T^*) + \sum_{i=1}^N \frac{\eta_i}{2}\mathbb{E}\|v_i^{(t^*)}\|^2.
\end{aligned}$$

Note that taking $\eta_i = 1/i$, the first term on the right hand side collapses to $-N\mathbb{E}\|\theta_{N+1}^{(t)} - \theta_T^*\|^2 \le 0$.

To proceed, we have

$$\begin{aligned}
&\sum_{i=1}^N \mathbb{E}(\theta_i - \theta_T^*)^T(\theta_i - \theta_T^*) \\
&\le \sum_{i=1}^N \mathbb{E}(\theta_i - \theta_T^*)^T(\theta_{t^*}^* - \theta_T^*) + \sum_i \frac{\eta_i}{2}\mathbb{E}\|v_i^{(t^*)}\|^2 \\
&\le \sum_{i=1}^N \mathbb{E}(\theta_i - \theta_T^*)^T(\theta_{t^*}^* - \theta_T^*) + \log(N)(\sigma_{t^*}^2 d + C_5) \\
&\le \sqrt{\sum_{i=1}^N \mathbb{E}\|\theta_i - \theta_T^*\|^2}\sqrt{\sum_{i=1}^N \|\theta_{t^*}^* - \theta_T^*\|^2} + \log(N)(\sigma_{t^*}^2 d + C_5).
\end{aligned}$$

By solving the inequality, we have

$$\begin{aligned}
\sum_{i=1}^N \mathbb{E}\|(\theta_i - \theta_T^*)^T\|^2 &\le \sum_{i=1}^N \mathbb{E}\|\theta_{t^*}^* - \theta_T^*\|^2 + \log(N)(\sigma_{t^*}^2 d + C_5) \\
&\le \sum_{i=1}^N \Delta_{t^*,T}^2 + \log(N)(\sigma_{t^*}^2 d + C_5)
\end{aligned}$$

Divided by N on both side, we reach Theorem 6. $\qquad\square$

H. Additional details for simulations

We provide the details for EXP3 algorithm. The EXP3 task scheduler picks tasks from $\{1, \ldots, T\}$, which is the action set for the bandit problem. In our experiment, $\eta = 0.85$.

Algorithm 2 Exponential-weight Algorithm for Exploration and Exploitation (Exp3)

1: **Input:** T, K, η
2: Set $\hat{S}_{0,t} = 0$ for all $t \in [T]$.
3: **for** $i = 1, \ldots, n$ **do**
4:    Calculate $P_{it} \leftarrow \frac{\exp\left(\eta \hat{S}_{i-1,t}\right)}{\sum_{t' \in [T]} \exp\left(\eta \hat{S}_{i-1,t'}\right)}$ for all $t \in [T]$
5:    Sample t_i from P_i and receive reward r_i
6:    Calculate $\hat{S}_{i,t} \leftarrow \hat{S}_{i-1,t} + 1 - \frac{\mathbb{1}_{\{t_i = a\}}(1 - r_i)}{P_{it}}$
7: **end for**