Representation Transfer Learning via Multiple Pre-Trained Models for Linear Regression

Navjot Singh and Suhas Diggavi, *Fellow, IEEE*

***Abstract*—In this paper, we consider the problem of learning a linear regression model on a data domain of interest (target) given few samples. To aid learning, we are provided with a set of pre-trained regression models that are trained on potentially different data domains (sources). Assuming a representation structure for the data generating linear models at the sources and the target domains, we propose a representation transfer based learning method for constructing the target model. The proposed scheme is comprised of two phases: (i) utilizing the different source representations to construct a representation that is adapted to the target data, and (ii) using the obtained model as an initialization to a fine-tuning procedure that re-trains the entire (over-parameterized) regression model on the target data. For each phase of the training method, we provide excess risk bounds for the learned model compared to the true data generating target model. The derived bounds show a gain in sample complexity for our proposed method compared to the baseline method of not leveraging source representations when achieving the same excess risk, therefore, theoretically demonstrating the effectiveness of transfer learning for linear regression.**

***Index Terms*—Machine learing: Transfer learning, few shot learning, representation learning.**

I. Introduction

A CRITICAL challenge for Deep Learning applications is the scarcity of available labeled data to train large scale models that generalize well to the data distribution. This is captured under the framework of *Few-Shot Learning* where Transfer Learning has emerged as an attractive framework to address this issue [1]. In transfer learning, one typically has access to a model trained on some data domain (hereby called *source domain*) that can be adapted to the data domain of interest (*target domain*). Within this context, a recently proposed strategy is that of *representation transfer learning* [2], [3], where one typically assumes a shared structure between the source and target learning tasks. The idea is to then learn a feature mapping for the underlying model (e.g. Neural Network representations) using the sample rich source domain that can be utilized directly on the target domain, for e.g, by training a few layers on top of the obtained network representation. This adaptation utilizes much fewer samples than what is required for training the entire model from scratch, while achieving good generalization performance which has been empirically observed for various large-scale machine learning and signal processing applications including image, speech and language [1], [4], [5], [6] tasks.

A defining factor in the need for representation transfer methods is that the source and target domains have different distributions. Learning across different domains has been studied extensively in the context of *Domain Adaptation* (see for e.g. [7], [8]) where it is usually assumed that source and target domain data can be accessed simultaneously. However, in many important practical scenarios of interest, the target data samples (labeled or unlabeled) are not available when training the source models. Transferring the source dataset to the target deployment scenario is infeasible for modern large-scale applications and violates data privacy. Thus there has been an increasing interest in transferring pre-trained source models to the target domain for sample efficient learning.

Despite the immense empirical success of representation transfer learning, development of a theory for understanding the generalization of representation learning methods and the sample complexity requirements is still in its infancy. Recent efforts in this direction have been made in understanding generalization for the simpler case of linear regression models [9], [10], [11], [12]. Within these works, [9], [11], [12] consider a common low-dimensional representation in the data generation process for the source and target domains, while [10] allows for the general case of data-generating representations being different. However, the analysis presented in that work requires the number of samples at the target to scale with the dimension of the model (see [10, Theorem 3.1]), which is impractical for few-shot learning scenarios.

A related line of work for understanding generalization of large scale models in the small sample regime is through the lens of *benign overfitting*. This is inspired by the surprising (empirical) observation that many large models, even when they overfit, tend to generalize well on the data distribution [13], [14]. In this context, [15], [16], [17] study this phenomena for linear models and analyze the generalization properties of the *min-norm solution*, where optimization methods like Gradient Descent are known to converge to in this setting [18], [19]. Specifically, these works seek to understand how the data distribution affects the excess population risk of the min-norm solution relative to the true data-generating linear model.

In this paper, we make efforts to understanding the generalization of linear models while leveraging pre-trained models

Received 27 June 2023; revised 21 May 2024; accepted 9 November 2024. Date of publication 27 November 2024; date of current version 28 February 2025. This work was supported in part by Army Research Laboratory under Cooperative Agreement W911NF-17-2-0196 and in part by NSF under Grant 2139304 and Grant 2146838. The guest editor coordinating the review of this article and approving it for publication was Prof. Xiao-Ping Zhang. *(Corresponding author: Navjot Singh.)*

The authors are with the Department of Electrical, Computer Engineering, University of California, Los Angeles, CA 90025 USA (e-mail: navjotsingh@ucla.edu; suhas@ee.ucla.edu).

Digital Object Identifier 10.1109/JSTSP.2024.3507371

inspired by the notions of representation transfer learning and benign-overfitting discussed above. These ideas lend themselves organically to the construction of a sample efficient training method for the target which we describe below briefly, along with our contributions.

Key Contributions: Our work provides a method for leveraging multiple pre-trained models for linear regression objectives (of dimension d) on a target task of interest in the small sample regime (samples $n_T \ll d$). The proposed two-phase approach leverages representation transfer (Phase 1) and over-parameterized training (Phase 2) to construct the target model, and we provide theoretical bounds for the excess risk for each phase of the training process (Theorems 1 and 2). In particular, we show that the learned model after the first phase has an excess risk of $\mathcal{O}(q/n_T) + \epsilon$, where q is dimension of the subspace spanned by learned source representations and ϵ is a constant that captures the approximation error when utilizing source representations for the target model (c.f. Assumption 1). This provides a gain in sample complexity compared to the baseline $\mathcal{O}(d/n_T)$ when learning the target model from scratch when the given source representations span a subspace of dimension much smaller than d (i.e. $q \ll d$). For the case when all representations are the same ($\epsilon = 0$), we recover the result of [9] for a *single* common representation. Similarly, for the overall model obtained after the second phase, we provide conditions on the target data distribution and the source/target representations that lead to an excess risk much smaller than $\mathcal{O}(d/n_T)$. Thus, we theoretically demonstrate the benefit of leveraging pre-trained models for linear regression.

A. Related Work

The problem of learning with few samples has been studied under the framework of *Few-Shot learning*, where *Meta-learning*– using experiences from previously encountered learning tasks to adapt to new tasks quickly [20], and *Transfer-learning*– transferring model parameters and employing pre-training or fine-tuning methods [3], are two major approaches. Theoretical works on Meta-learning algorithms typically assume some relation between the distribution of source and target tasks, for e.g., being sampled from the same task distribution. A more general framework is that of *Out-of-Domain (OOD)* generalization, where the goal is to learn models in a manner that generalize well to unseen data domains [21].

Transfer learning, especially, representation transfer learning has shown empirical success for large-scale machine learning [2], however, theoretical works on understanding generalization in this setting are few; see [12], [22], [23]. A related line of work is representation learning in context of *Domain Adaptation (DA)*, see for e.g. [24], [25], [26], [27], [28]. However, this usually assumes that source and data domains can be accessed simultaneously. There are deviations from this theme in *Multi-Source DA* where the goal is to understand how multiple source models can be combined to generalize well on a target domain of interest, although without changing the learned model based on the target samples [29], [30], [31].

In context of leveraging pre-trained models for linear regression, our work is most closely related to [9], [10] that theoretically analyze representation transfer for linear models. In contrast to [9], we allow for the true target representations to be different among the source models as well as the target, and introduce a notion of closeness between these representations (c.f. Assumption 1). Although a similar setting was considered in [10] where representations are assumed to be close in the ℓ_2 norm, their resulting bound for the fine-tuned model risk shows that the required number of target samples scale with the dimension of the learned model for efficient transfer [10, Theorem 3.1]. In contrast, the proposed method in our work provides analysis relating these bounds to the properties of the target data distribution taking inspiration from works on benign overfitting for linear regression [15], [19], [32]. This enables us to identify conditions on the target data distribution that allow the required target samples to be much smaller than the overall model dimension (see Theorem 2).

B. Paper Organization

We set up the problem and define the notation we use throughout the paper in Section II. Section III describes our training method for the target task model when given access to multiple pre-trained source models. Section IV establishes excess risk bounds of our proposed scheme, which are proved in Sections V and VI. Section VII provides numerical results and some concluding remarks are presented in Section VIII.

II. Problem Setup and Notation

Notation: We use boldface for vectors and matrices, with matrices in uppercase. For a matrix $\mathbf{A}$, we denote the projection matrix onto its column space by $\mathbf{P_A} := \mathbf{A}(\mathbf{A}^\top\mathbf{A})^\dagger\mathbf{A}^\top$ where $\mathbf{W}^\dagger$ denotes the Moore-Penrose pseudo-inverse of the matrix $\mathbf{W}$. We define $\mathbf{P}_\mathbf{A}^\perp := \mathbf{I} - \mathbf{P_A}$, where $\mathbf{I}$ denotes the identity matrix of appropriate dimensions. We denote by $\mathcal{C}(\mathbf{A})$ the column space of a matrix $\mathbf{A}$ and by $\sigma_i(\mathbf{A}), \lambda_i(\mathbf{A})$ its ith largest singular value and eigenvalue, respectively. $\|.\|_F$ denotes the Frobenius norm. For a vector $\mathbf{v}$, $\|\mathbf{v}\|_2$ denotes the ℓ_2 norm, while for a matrix $\mathbf{V}$, $\|\mathbf{V}\|_2$ denotes the spectral norm. Tr[.] denotes the trace operation. $\lesssim$ denotes the inequality sign where we ignore the constant factors. The notation $\mathcal{O}$ is the 'big-O' notation and we define $[m] = \{1, 2, \ldots, m\}$.

Setup: We consider m number of source tasks and a single target task. We denote by $\mathcal{X} \subseteq \mathbb{R}^d$ the space of inputs and $\mathcal{Y} \subseteq \mathbb{R}$ the output space. The source and target tasks are associated with data distributions $p_i, i \in [m]$ and p_T, respectively, over the space $\mathcal{X}$. We assume a linear relationship between the input and output pairs for source task $i \in [m]$ given by:

$$y_i = \mathbf{x}_i^\top \mathbf{B}_i^* \mathbf{w}_i^* + z_i, \quad \boldsymbol{\theta}_i^* := \mathbf{B}_i^* \mathbf{w}_i^* \tag{1}$$

where $\mathbf{x}_i \in \mathcal{X}$ denotes an input feature vector drawn from distribution p_i, $y_i \in \mathcal{Y}$ is the output, and $z_i \sim \mathcal{N}(0, \sigma^2)$ denotes Gaussian noise. The associated true task parameter $\boldsymbol{\theta}_i^* := \mathbf{B}_i^* \mathbf{w}_i^*$ is comprised of the representation matrix $\mathbf{B}_i^* \in \mathbb{R}^{d\times k}$ which maps the input to a lower $k-$dimensional space (where $k \ll d$)

and a head vector $\mathbf{w}_i^* \in \mathbb{R}^k$ mapping the intermediate sample representation to the output[1]. The data generation process for the target task is defined similarly with distribution p_T and associated target task parameter given by $\boldsymbol{\theta}_T^* = \mathbf{B}_T^* \mathbf{w}_T^*$. For sources $i \in [m]$, we define the input covariance matrix $\boldsymbol{\Sigma}_i = \mathbb{E}_{\mathbf{x}_i \sim p_i}[\mathbf{x}_i \mathbf{x}_i^\top]$ and similarly for the target distribution, $\boldsymbol{\Sigma}_T = \mathbb{E}_{\mathbf{x}_T \sim p_T}[\mathbf{x}_T \mathbf{x}_T^\top]$.

In our scenario of interest, the pre-trained models are trained 'offline' on source distributions and are made available to the target task during deployment. That is, for training the target task, we have access to only the models learned by the source tasks and not the source datasets themselves. For learning the pre-trained source models, we assume n_S number of samples for each of the source tasks (thus, mn_S source task samples in total) denoted by the pair $(\mathbf{X}_i, \mathbf{y}_i)$ for source $i \in [m]$ where $\mathbf{X}_i \in \mathbb{R}^{n_S \times d}$ contains row-wise input feature vectors and $\mathbf{y}_i \in \mathbb{R}^{n_S}$ is the vector of corresponding outputs. We similarly have n_T samples $(\mathbf{X}_T, \mathbf{y}_T)$ for the target task where $n_T \ll n_S$. We also assume $n_T \ll d$.

With the data generation process defined above, we now define the expected population risk on the target distribution for $\hat{\boldsymbol{\theta}}$:

$$R(\hat{\boldsymbol{\theta}}) = \mathbb{E}_{\mathbf{x} \sim p_T} \mathbb{E}_{\mathbf{y}|\mathbf{x}^\top \boldsymbol{\theta}_T^*}[(\mathbf{y} - \mathbf{x}^\top \hat{\boldsymbol{\theta}})^2]$$

Our goal is to learn a model $\hat{\boldsymbol{\theta}}$ for the target task that generalizes well to the target data distribution. Thus, we want $\hat{\boldsymbol{\theta}}$ that minimizes the *Expected Excess Risk* defined by:

$$\text{EER}(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}_T^*) := R(\hat{\boldsymbol{\theta}}) - R(\boldsymbol{\theta}_T^*) \tag{2}$$

Since we are given access to only $n_T \ll d$ target samples, it is infeasible to learn a predictor from scratch that performs well for the excess risk defined in (2).

To aid learning on the target, we have access to models learned on the source tasks. Specifically, the target has access to the trained source models representations $\{\widehat{\mathbf{B}}_i\}_{i=1}^m$ that are solutions of the following empirical minimization problem:

$$\{\widehat{\mathbf{B}}_i, \widehat{\mathbf{w}}_i\}_{i=1}^m \leftarrow \min_{\{\mathbf{B}_i\}, \{\mathbf{w}_i\}} \frac{1}{mn_S} \sum_{i=1}^m \|\mathbf{y}_i - \mathbf{X}_i \mathbf{B}_i \mathbf{w}_i\|_2^2 \tag{3}$$

Since we have data rich source domains ($n_S \gg d$), we expect the obtained source models $\widehat{\mathbf{B}}_i \widehat{\mathbf{w}}_i$ to be close to $\boldsymbol{\theta}_i^*$ for $i \in [m]$ (c.f. (1)). For effective representation transfer, we also want the learned representations $\{\widehat{\mathbf{B}}_i\}$ to be close to the true representations $\{\mathbf{B}_i^*\}$, in the sense that they approximately span the same subspace[2]. We make this notion precise in Lemma 1 stated with our main results in Section IV. Given access to the source model representations, our proposed method for training the target model leverages the representation maps $\{\widehat{\mathbf{B}}_i\}_{i=1}^m$ to drastically reduce the sample complexity. We describe our training method in Section III and provide the excess risk bounds for the resulting target model in Section IV.

III. Learning With Multiple Pre-Trained Models

To leverage source representations for training the target model, it is instinctive that there should be a notion of closeness between the true source and target model representation that can be exploited for target task training. We now make this notion precise. We first define as $\mathbf{V}^* \in \mathbb{R}^{d \times l}$ the matrix whose columns are an orthonormal basis of the set of columns of all the source representation matrices $\{\mathbf{B}_i^*\}_{i=1}^m$. The individual source models can thus be represented by $\boldsymbol{\theta}_i^* = \mathbf{V}^* \widetilde{\mathbf{w}}_i^*$ for all $i \in [m]$. The target model $\boldsymbol{\theta}_T^* = \mathbf{B}_T^* \mathbf{w}_T^*$ governing the target data generation is assumed to satisfy the following:

Assumption 1: Consider the projection of the target model $\mathbf{B}_T^* \mathbf{w}_T^*$ to space $\mathcal{C}(\mathbf{V}^*)$ given by $\mathbf{V}^* \widetilde{\mathbf{w}}_T^*$ for some $\widetilde{\mathbf{w}}_T^* \in \mathbb{R}^l$. Then for some $\epsilon > 0$, we have:

$$\mathbb{E}_{\mathbf{x} \sim p_T} \left[\mathbf{x}^\top \mathbf{V}^* \widetilde{\mathbf{w}}_T^* - \mathbf{x}^\top \mathbf{B}_T^* \mathbf{w}_T^*\right]^2 \leq \epsilon^2$$

The value of ϵ in Assumption 1 above captures how far away the output of the true target model is to a model learned using the true source representations. Note that if the columns of $\mathbf{B}_T^*$ can be constructed by the vectors in $\mathbf{V}^*$, the above is satisfied for $\epsilon = 0$. Assumption 1 can also be re-written as:

$$\left\|\Sigma_T^{1/2} \left(\mathbf{V}^* \widetilde{\mathbf{w}}_T^* - \mathbf{B}_T^* \mathbf{w}_T^*\right)\right\|_2^2 \leq \epsilon^2 \tag{4}$$

We are given access to n_T samples for the target machine given by $(\mathbf{X}_T, \mathbf{y}_T)$ and pre-trained models representations from the sources $\{\widehat{\mathbf{B}}_i\}_{i=1}^m$ (c.f. (3)). Our proposed training scheme consists of two phases, which we will now describe independently in the following. We split the available n_T target samples into n_{T_1}, n_{T_2} for the two respective phases. At a high level, in *Phase 1*, we make use of the available source representations to construct a target representation and adapt it to the target task using n_{T_1} samples. The obtained model is then used as an *initialization* for *Phase 2* where we train the entire (over-parameterized) model, including the representation matrix, using n_{T_2} samples. We provide the resulting excess risk bounds for the model obtained after *Phase 1* and the final target model after *Phase 2* in Section IV.

A. Phase 1: Transferring Source Representation to Target

In the context of utilizing pre-trained models, we will make use of the empirical source representations $\{\widehat{\mathbf{B}}_i\}_{i=1}^m$ to learn the target model. We first construct a matrix $\widehat{\mathbf{V}} \in \mathbb{R}^{d \times q}$ whose columns are the orthonormal basis of the columns of $\{\widehat{\mathbf{B}}_i\}_{i=1}^m$ which denotes a dictionary of the learned source representation matrices[3]. Note that we have $q \leq mk$. Having constructed the representation, we train a head vector $\widehat{\mathbf{w}}_{T_1} \in \mathbb{R}^q$ minimizing the

[1] Our formulation is more general than the one in [9], which considers the same representation matrices for the source and the target tasks, that is $\{\mathbf{B}_i^*\}_{i=1}^m = \mathbf{B}_T^* = \mathbf{B}^*$. The formulation considered in [10] allows for different values of the $\mathbf{B}_i^*$ among the source domains, imposing a structure on the representation matrices: $\mathbf{B}_i^* = \mathbf{B} + \Delta_i$ with $||\Delta_i||_F \leq \delta_0$ for $i \in [m]$.

[2] Due to the bi-linear nature of the problem, multiple $\hat{\mathbf{B}}_i$ and $\hat{\mathbf{w}}_i$ may satisfy the product $\hat{\boldsymbol{\theta}}_i = \hat{\mathbf{B}}_i \hat{\mathbf{w}}_i$. We provide a note on this technical aspect of the problem in Section V.

[3] The construction of $\widehat{\mathbf{V}}$ from $\{\widehat{\mathbf{B}}_i\}$ can be done by the Gram-Schmidt process. This can be done in the pre-deployment phase after training the source models, and $\widehat{\mathbf{V}}$ can be made available directly to the target task.

empirical risk on n_{T_1} samples:

$$\widehat{\mathbf{w}}_{T_1} \leftarrow \min_{\mathbf{w}_T \in \mathbb{R}^q} \frac{1}{n_{T_1}} \left\| \mathbf{y}_{T_1} - \mathbf{X}_{T_1} \widehat{\mathbf{V}} \mathbf{w}_T \right\|_2^2 \tag{5}$$

Here, $\mathbf{y}_{T_1} \in \mathbb{R}^{n_{T_1}}$ denotes the first n_{T_1} values of $\mathbf{y}_T$ and $\mathbf{X}_{T_1} \in \mathbb{R}^{n_{T_1} \times d}$ the first n_{T_1} rows of $\mathbf{X}_T$. We denote the resulting model at the end of this phase by $\boldsymbol{\theta}_{\text{Phase}_1} := \widehat{\mathbf{V}} \widehat{\mathbf{w}}_{T_1}$. Since we only have to learn the head vector using the available representation $\widehat{\mathbf{V}}$, the sample complexity requirement is greatly reduced, which is also evident from our bound for $\text{EER}(\boldsymbol{\theta}_{\text{Phase}_1}, \boldsymbol{\theta}_T^*)$ provided in Theorem 1.

B. Phase 2: Fine-Tuning With Initialization

The obtained model $\boldsymbol{\theta}_{\text{Phase}_1}$ from the previous phase utilizes empirical source representation for its construction. However, the true target model $\boldsymbol{\theta}_T^*$ may not lie in the space spanned by the source representation and thus $\boldsymbol{\theta}_{\text{Phase}_1}$ lies in a ball centered $\boldsymbol{\theta}_T^*$ whose radius scales with ϵ (c.f. Assumption 5). To move towards the true model $\boldsymbol{\theta}_T^*$, we utilize n_{T_2} number of target samples (independent from the n_{T_1} samples in the previous phase) to train the entire linear model using Gradient Descent (GD) with $\boldsymbol{\theta}_{\text{Phase}_1}$ as the initialization. In particular, the GD procedure minimizes the following starting from $\boldsymbol{\theta}_{\text{Phase}_1}$:

$$f(\boldsymbol{\theta}) = \frac{1}{n_{T_2}} \|\mathbf{y}_{T_2} - \mathbf{X}_{T_2} \boldsymbol{\theta}\|_2^2 \tag{6}$$

Here, $\mathbf{y}_{T_2} \in \mathbb{R}^{n_{T_2}}$ and $\mathbf{X}_{T_2} \in \mathbb{R}^{n_{T_2} \times d}$ are the remaining sample values from Phase 1. Since $n_{T_2} \ll d$, we are in an over-parameterized regime, for which it is known that GD procedure optimizing the objective in (6) converges, under appropriate choice of learning rate, to a solution closest in norm to the initialization [18], [19], [33], [34]; mathematically:

$$\begin{aligned} &\min_{\boldsymbol{\theta}} \|\boldsymbol{\theta} - \boldsymbol{\theta}_{\text{Phase}_1}\|_2 \\ &\text{s.t. } \|\mathbf{y}_{T_2} - \mathbf{X}_{T_2} \boldsymbol{\theta}\|_2 = \min_{\mathbf{b}} \|\mathbf{y}_{T_2} - \mathbf{X}_{T_2} \mathbf{b}\|_2 \end{aligned} \tag{7}$$

We denote the solution of the above optimization problem as $\boldsymbol{\theta}_{\text{Phase}_2}$, which forms our final target task model. We provide bounds for $\text{EER}(\boldsymbol{\theta}_{\text{Phase}_2}, \boldsymbol{\theta}_T^*)$ in Theorem 2.

IV. Main Results

We now provide theoretical bounds on the excess risk for the target (c.f. (2)) when leveraging pre-trained source models. In Section IV-A, we first state excess risk bounds for the model obtained after Phase 1 (see Section III-A), denoted by $\boldsymbol{\theta}_{\text{Phase}_1} := \widehat{\mathbf{V}} \widehat{\mathbf{w}}_T$, where target representation $\widehat{\mathbf{V}}$ is constructed as a combination of source representations and adapted to the target data using n_{T_1} amount of target samples by training a target-specific head vector $\widehat{\mathbf{w}}_T$. In Section IV-B, we provide our overall excess risk for the model $\boldsymbol{\theta}_{\text{Phase}_2}$ (c.f. (7)) obtained by re-training the entire (over-parameterized) model via Gradient Descent with n_{T_2} number of target samples (independent form the previously utilized n_{T_1} samples) using $\boldsymbol{\theta}_{\text{Phase}_1}$ as the initialization.

A. Theoretical Results for Representation Transfer: Phase 1

We work with the following assumptions:

Assumption 2 (Subgaussian features): We assume that $\mathbb{E}_{\mathbf{x} \sim p_j}[\mathbf{x}] = 0$ for all $j \in [m] \cup \{T\}$. We consider $\bar{p}_j$ to be the *whitening* of p_j (for $j \in [m] \cup \{T\}$) such that $\mathbb{E}_{\bar{\mathbf{x}} \sim \bar{p}_j}[\mathbf{x}] = 0$ and $\mathbb{E}_{\bar{\mathbf{x}} \sim \bar{p}_j}[\bar{\mathbf{x}} \bar{\mathbf{x}}^\top] = \mathbf{I}$. We assume there exists $\rho > 0$ such that the random vector $\bar{\mathbf{x}} \sim \bar{p}_j$ is ρ^2-subgaussian.

Assumption 3 (Covariance Dominance): There exists $r > 0$ such that $\Sigma_i \succeq r \Sigma_T$ for all $i \in [m]$.

Assumption 4 (Diverse source tasks): Consider the source models $\boldsymbol{\theta}_i^* = \mathbf{V}^* \widetilde{\mathbf{w}}_i^*$ for $i \in [m]$. We assume that the matrix $\widetilde{\mathbf{W}}^* := [\widetilde{\mathbf{w}}_1^*, \ldots, \widetilde{\mathbf{w}}_m^*] \in \mathbb{R}^{l \times m}$ satisfies $\sigma_l^2(\widetilde{\mathbf{W}}^*) \geq \Omega(\frac{m}{l})$

Assumption 5 (Distribution of target task): We assume that $\widetilde{\mathbf{w}}_T^*$ follows a distribution ν such that $\|\mathbb{E}_{\widetilde{\mathbf{w}} \sim \nu}[\widetilde{\mathbf{w}} \widetilde{\mathbf{w}}^\top]\|_2$ is $\mathcal{O}(\frac{1}{l})$. We denote $\Sigma_{\widetilde{\mathbf{w}}_T^*} = \mathbb{E}_{\widetilde{\mathbf{w}} \sim \nu}[\widetilde{\mathbf{w}} \widetilde{\mathbf{w}}^\top]$.

Note on Assumptions: Assumption 2 on sub-Gaussian features is commonly used in literature to obtain probabilistic tail bounds [9], [10], [15], [32]. Following [9], Assumption 3 states the target data covariance matrix is covered by the covariance matrices of the source data distributions. We remark that this assumption allows the covariance matrices to be different, in contrast to works [10], [11] that assume a common covariance matrix for all the distributions. Assumption 4 (also made in related works [9], [10], [35]) says that the head vectors corresponding to the matrix $\mathbf{V}^*$ for each source model should span $\mathbb{R}^l$. This effectively allows us to recover the representation $\mathbf{V}^*$ provided enough source machines ($m > l$) that individually capture one or more features of $\mathbf{V}^*$. This assumption is also central to proving our result in Lemma 1 provided below which show that the matrices $\widehat{\mathbf{V}}$ and $\mathbf{V}^*$, whose columns form an orthonormal basis for the span of $\{\widehat{\mathbf{B}}_i\}$ and $\{\mathbf{B}_i^*\}$, respectively, span the same subspace for constructing the target model.

Lemma 1: Let matrix $\widehat{\mathbf{V}} \in \mathbb{R}^{d \times q}$ be formed by empirical source representations $\{\widehat{\mathbf{B}}_i\}$ obtained from solving (3) and the matrix $\mathbf{V}^* \in \mathbb{R}^{d \times l}$ formed from the true representations $\{\mathbf{B}_i^*\}$. Under Assumptions 2–4, for any $\mathbf{b} \in \mathbb{R}^l$ such that $\|\mathbf{b}\|_2 = 1$, $n_s \gg \rho^4(d + \log(m/\delta))$ and $n_{T_1} \gg \rho^4(\max\{l, q\} + \log(1/\delta))$, with probability at-least $1 - \delta_1$, we have:

$$\begin{aligned} &\min_{\mathbf{u} \in \mathbb{R}^q} \|\mathbf{X}_{T_1} \widehat{\mathbf{V}} \mathbf{u} - \mathbf{X}_{T_1} \mathbf{V}^* \mathbf{b}\|_2 \\ &\quad \leq \frac{\sigma^2 n_{T_1}}{r n_S} \left(km + kdm \log(\kappa n_s) + \log\left(\frac{1}{\delta_1}\right) \right) \end{aligned}$$

A proof of the lemma above is provided in Section V. We now state our main result for the excess risk on after Phase 1.

Theorem 1 (Phase 1 training result): Fix a failure probability $\delta \in (0, 1)$ and further assume $2k \leq \min\{d, m\}$ and the number of samples in the sources and target satisfy $n_s \gg \rho^4(d + \log(m/\delta))$ and $n_{T_1} \gg \rho^4(\max\{l, q\} + \log(1/\delta))$, respectively. Define $\kappa = \frac{\max_{i \in [m]} \lambda_{\max}(\Sigma_i)}{\min_{i \in [m]} \lambda_{\min}(\Sigma_i)}$ where $\lambda_{\max}(\Sigma_i)$ denotes the maximum eigenvalue of Σ_i. Then with probability at least $1 - \delta$ over the samples, under Assumptions 1–5, the expected excess risk

of $\boldsymbol{\theta}_{\text{Phase}_1} := \widehat{\mathbf{V}}\widehat{\mathbf{w}}$ satisfies:

$$\mathbb{E}[\text{EER}(\boldsymbol{\theta}_{\text{Phase}_1}, \boldsymbol{\theta}_T^*)] \lesssim \frac{\sigma^2}{n_{T_1}}(q + \log(1/\delta)) + \epsilon^2 + \sigma^2\left[\frac{1}{rn_s m}\log\left(\frac{1}{\delta}\right) + \left(\frac{kd\log(\kappa n_s) + k}{rn_s}\right)\right]$$

where expectation is taken over $\widetilde{\mathbf{w}}_T^*$ for the target task (c.f. Assumption 5). We provide proof for Theorem 1 in Section V.

Discussion: The bound in Theorem 1 shows the population risk of the learned model $\boldsymbol{\theta}_{\text{Phase}_1}$ lies in a ball centered at the true target model risk $R(\boldsymbol{\theta}_T^*)$ with radius ϵ^2, which represents the *approximation error* for using source representations for the target task (see Assumption 1). Note that the expected excess risk scales as $\mathcal{O}(q/n_{T_1})$ with respect to the number of target samples when the representation is learned from the source representations. This demonstrates a sample gain compared to the baseline of $\mathcal{O}(d/n_{T_1})$ for learning the entire model (including representation) with the target data when $q \ll d$, that is, when the empirical source representations together span a subspace of dimension much smaller than d. For the case when source and target representations are all the same, $\mathbf{B}_T^* = \mathbf{B}_i^* = \mathbf{B}^* \in \mathbb{R}^{d\times k}$ for all $i \in [m]$, the excess risk scales as $\mathcal{O}(k/n_{T_1})$, which recovers the result of [9].

B. Theoretical Results for Overall Scheme: Phase1 + Phase2

We require the following additional assumptions:

Assumption 6: The rows of the target data matrix $\mathbf{X}_T$ are linearly independent.

Assumption 7: The Gradient Descent procedure to optimize (6) converges to $\boldsymbol{\theta}_{\text{Phase}_2}$ with $f(\boldsymbol{\theta}_{\text{Phase}_2}) = 0$.

Assumption 6 is typically made in literature for analysis in the over-parameterized regime for linear regression, see [15], and can also be relaxed to hold with high probability instead and incorporated in the analysis [32]. Assumption 7 holds in our setting as the objective in (6) is strongly convex and smooth for which GD can converge to the optimum [36].

Theorem 2 (Phase 1 + Phase 2 training result): Consider obtaining the final target model by using n_{T_1} samples during Phase 1 for representation transfer and then fine-tuning in Phase 2 with n_{T_2} samples (independently drawn from Phase 1). Denote the eigenvalues of the covariance matrix of the underlying data Σ_T by $\{\lambda_i\}_{i=1}^d$. Then under Assumptions 1–7, the excess risk of the final parameter $\hat{\boldsymbol{\theta}}_T := \boldsymbol{\theta}_{\text{Phase}_2}$ is bounded as follows with probability at least $1-\delta$:

$$\mathbb{E}[\text{EER}(\hat{\boldsymbol{\theta}}_T, \boldsymbol{\theta}_T^*)] \lesssim \frac{\lambda_1}{\lambda_d}\frac{r_0(\Sigma_T)}{n_{T_2}}\left(\frac{\sigma^2}{n_{T_1}}(q + \log(1/\delta)) + \epsilon^2\right) + r\sigma^2\log\left(\frac{1}{\delta}\right)\left(\frac{k^*}{n_{T_2}} + \frac{n_{T_2}}{R_{k^*}(\Sigma_T)}\right) + \frac{\lambda_1\sigma^2}{\lambda_d}\frac{r_0(\Sigma_T)}{n_{T_2}}\left(\frac{1}{rn_s m}\log\left(\frac{1}{\delta}\right) + \left(\frac{kd\log(\kappa n_s) + k}{rn_s}\right)\right)$$

where $r_k(\Sigma_T) = \frac{\Sigma_{i>k}\lambda_i}{\lambda_{k+1}}$, $R_{k^*}(\Sigma_T) = \frac{(\Sigma_{i>k}\lambda_i)^2}{\Sigma_{i>k}\lambda_i^2}$. Here, constant $b > 1$ and $k^* = \min\{k \geq 0 : r_k(\Sigma) \geq bn_{T_2}\}$ with $k^* \leq \frac{n_{T_2}}{c_1}$ for some universal constant $c_1 > 1$.

We provide a proof for Theorem 2 in Section VI.

Discussion: Theorem 2 shows the excess risk of our overall target model ($\hat{\boldsymbol{\theta}}_T = \boldsymbol{\theta}_{\text{Phase}_2}$) as a function of the number of samples n_S, n_{T_1}, n_{T_2} and parameters depending on the target data covariance matrix, Σ_T. Since we re-train the entire model (including the representation) with n_{T_2} target samples, the population risk of the learned model $R(\boldsymbol{\theta}_{\text{Phase}_2})$ can be made closer to the true risk $R(\boldsymbol{\theta}_T^*)$ by increasing n_{T_2}, which is in contrast to the result of Theorem 1 which shows closeness only in an ϵ^2 radius ball due to using source representation directly to construct the target model.

We now provide a baseline comparison to the standard linear regression scenario where we do no utilize any source models and instead learn the target task model from scratch using the available $n_T = n_{T_1} + n_{T_2}$ samples. The excess risk in this setting is $\mathcal{O}(\frac{\sigma^2 d}{n_T})$. If the number of source samples are large enough ($n_S \gg d$) to get a good empirical performance on the source models (c.f. (3)), the bound from Theorem 2 demonstrates a sample gain compared to the baseline when:

$$\frac{\lambda_1}{\lambda_d}\frac{r_0(\Sigma_T)}{n_{T_2}}\left(\frac{\sigma^2}{n_{T_1}}(q + \log(1/\delta)) + \epsilon^2\right) + c\sigma^2\log\left(\frac{1}{\delta}\right)\left(\frac{k^*}{n_{T_2}} + \frac{n_{T_2}}{R_{k^*}(\Sigma_T)}\right) \ll \frac{\sigma^2 d}{n_{T_1} + n_{T_2}} \tag{8}$$

It can be seen that for the above relation to hold, we require:

- The target data covariance matrix Σ_T should be such that the term $R_{k^*}(\Sigma_T)$ is much larger than n_{T_2}, and $k^* \ll n_{T_2}$. This is satisfied, for e.g., in the case when eigenvalues of Σ_T decay slowly from largest to smallest, and are all larger than a small constant [15].
- Using the definition of $r_0(\Sigma_T) = \sum_{i=1}^d \lambda_i/\lambda_1$, the following provides a sufficient condition the first term on the L.H.S. of (8):

$$\frac{q\sum_{i=1}^d \lambda_i}{n_{T_1}n_{T_2}\lambda_d} \ll \frac{d}{n_{T_1} + n_{T_2}}$$

  This, is turn, imposes the following restriction on q, which is the dimension of the subspace formed by the source representations $\{\widehat{\mathbf{B}}_i\}$:

$$q \ll \frac{d\lambda_d n_{T_1} n_{T_2}}{(\sum_{i=1}^d \lambda_i)(n_{T_1} + n_{T_2})} \tag{9}$$

  Since $n_{T_1} + n_{T_2} = n_T$, it is easy to check that the R.H.S. of (9) is maximized when $n_{T_1} = n_{T_2} = n/2$. With this optimal splitting of the target samples for each of the phases, we require $q \ll \frac{d\lambda_d n_T}{2\sum_{i=1}\lambda_i}$ for the inequality in (8).

V. Proof for Phase 1 Training

We now provide the proof for Theorem 1 which establishes a generalization bound for the learned model formed by leveraging the pre-trained source models.

We first comment on the solutions obtained from the optimization problem pertaining ot the source tasks and comment on their efficacy to the representation transfer setup.

Note on the solutions for Source Training (3):

The solution for the bi-linear optimization problem for the source training in (3) can allow for any solution of the form $\hat{\mathbf{B}}_{\mathbf{i}}\hat{\mathbf{w}}_{\mathbf{i}} = \hat{\boldsymbol{\theta}}_i$ for $i \in [m]$ where multiple $\hat{\mathbf{B}}_i$ and $\hat{\mathbf{w}}_i$ may satisfy the product. Crucial to our transfer process is Assumption 4 which considers that we have enough source domains to reconstruct the underlying representation matrix $\mathbf{V}^*$ from the individual source models.

For the sake of argument, consider the scenario where we have a large number of source samples for each source domain. Then, solving the optimization problem leads to solutions $\boldsymbol{\theta}_i^* = \mathbf{B}_i^*\mathbf{w}_i^*$. Assume that from solving (3), we obtain $\mathbf{B_o}^*$ and $\mathbf{w_o}^*$ such that $\mathbf{B_o}^*\mathbf{w_o}^* = \mathbf{B}_i^*\mathbf{w}_i^* = \boldsymbol{\theta}_i^*$. First consider the matrix $\mathbf{V}_o^* \in \mathbb{R}^{d\times p}$ as the matrix formed by the basis of spans of the columns of $\{\mathbf{B}_{\mathbf{o}i}^*\}_{i=1}^m$. Then, as per the definition of $\boldsymbol{\theta}_i^*$ there exists $\widetilde{\mathbf{W_o}}^* \in \mathbb{R}^{p\times m}$ with i^{th} column as $\widetilde{\mathbf{w}}_{\mathbf{o}i}^*$ satisfying $\mathbf{V}_o^*\widetilde{\mathbf{w}}_{\mathbf{o}i}^* = \theta_i^*$, and thus, $\mathbf{V}_o^*\widetilde{\mathbf{W}_o}^* = [\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*, \ldots, \boldsymbol{\theta}_m^*] = \mathbf{V}^*\widetilde{\mathbf{W}}^*$. From Assumption 4, it follows that $\mathcal{C}(\mathbf{V}^*\widetilde{\mathbf{W}}^*) = \mathcal{C}(\mathbf{V}^*)$ and thus $\mathcal{C}(\mathbf{V}_\mathbf{o}^*) \supseteq \mathcal{C}(\mathbf{V}^*)$, which implies that we are transferring the column space of the true underlying representation matrix to the target domain.

Thus, as long as we solve the problem in (3) yielding the source representations matrices $\{\hat{\mathbf{B}}_i\}$ that contain, in their span, a good estimate of the true underlying source model $\boldsymbol{\theta}_i^*$, we can leverage the transferred representation matrix formed from the matrices $\{\hat{\mathbf{B}}_i\}$ to get a good estimate of the target model (c.f. Assumption 1).

We now note the following results from [9] that will enable us to prove Theorem 1 later in Section V-B (and Theorem 2 in Section VI).

Claim 1 (Covariance of Source distribution, Claim A.1 of [9]): Suppose $n_s \gg \rho^4(d + \log(m/\delta))$ for $\delta \in (0, 1)$. Then with probability at least $1 - \frac{\delta}{10}$ over the inputs $\mathbf{X}_1, \ldots, \mathbf{X}_m$ in the source tasks, for all $i \in [m]$ we have

$$0.9\boldsymbol{\Sigma}_i \preceq \frac{1}{n_s}\mathbf{X}_i^\top\mathbf{X}_i \preceq 1.1\boldsymbol{\Sigma}_i$$

Claim 2 (Covariance Target distribution, Claim A.2 of [9]): Suppose $n_T \gg \rho^4(k + \log(1/\delta))$ for $\delta \in (0, 1)$. The for any given matrix $\mathbf{B} \in \mathbb{R}^{d\times 2k}$ that is independent of $\mathbf{X}_T$, with probability at least $1 - \frac{\delta}{20}$ over target data X_T, we have

$$0.9\mathbf{B}^\top\boldsymbol{\Sigma}_T\mathbf{B} \preceq \frac{1}{n_T}\mathbf{B}^\top\mathbf{X}_T^\top\mathbf{X}_T\mathbf{B} \preceq 1.1\mathbf{B}^\top\boldsymbol{\Sigma}_T\mathbf{B}$$

Proposition 1 (Lemma A.7 from [9]): For matrices $\mathbf{A}_1, \mathbf{A}_2$ (with same number of columns) such that $\mathbf{A}_1^\top\mathbf{A}_1 \succeq \mathbf{A}_2^\top\mathbf{A}_2$ and for matrices $\mathbf{B}_1, \mathbf{B}_2$ of compatible dimensions, we have:

$$\left\|\mathbf{P}^\perp_{\mathbf{A}_1\mathbf{B}_1}\mathbf{A}_1\mathbf{B}_2\right\|_F^2 \geq \left\|\mathbf{P}^\perp_{\mathbf{A}_2\mathbf{B}_1}\mathbf{A}_2\mathbf{B}_2\right\|_F^2$$

Proposition 2: Consider matrices $\mathbf{A} \in \mathbb{R}^{a\times b}$ and $\mathbf{B} \in \mathbb{R}^{b\times c}$. Then for any $\mathbf{u} \in \mathbb{R}^a$, we have:

$$\|\mathbf{P}^\perp_{\mathbf{A}}\mathbf{u}\|_2^2 \leq \|\mathbf{P}^\perp_{\mathbf{AB}}\mathbf{u}\|_2^2$$

Proof: For given $\mathbf{A} \in \mathbb{R}^{a\times b}$ and $\mathbf{B} \in \mathbb{R}^{b\times c}$ and $\mathbf{u} \in \mathbb{R}^a$:

$$\begin{aligned}\|\mathbf{P}^\perp_{\mathbf{A}}\mathbf{u}\|_2^2 &= \min_{\mathbf{r}\in\mathbb{R}^b}\|\mathbf{A}\mathbf{r} - \mathbf{u}\|_2^2\\ &\leq \min_{\mathbf{r}\in\mathcal{C}(\mathbf{B})}\|\mathbf{A}\mathbf{r} - \mathbf{u}\|_2^2\\ &= \min_{\mathbf{s}\in\mathbb{R}^c}\|\mathbf{A}\mathbf{B}\mathbf{s} - \mathbf{u}\|_2^2\\ &= \|\mathbf{P}^\perp_{\mathbf{AB}}\mathbf{u}\|_2^2\end{aligned}$$

A. Some Important Results

We now provide proof of results used to establish the resulting bound for Phase 1 training provided in Theorem 1, which is proved later in Section V-B. These results provide guarantees on empirical training of the source models (c.f. Lemma 2) as well as the performance of empirically learned source representations on the target data (c.f. Lemma 1).

These results would also be useful for the proof of Theorem 2 presented later in Section VI.

1) Proof of Lemma 1: We first prove Lemma 1 which establishes a bound on using the learned empirical representation $\widehat{\mathbf{V}}$ on the target data.

[Restating Lemma 1] Consider the matrix $\widehat{\mathbf{V}} \in \mathbb{R}^{d\times q}$ formed by empirical source representations $\{\widehat{\mathbf{B}}_i\}$ obtained from solving (3) and the matrix $\mathbf{V}^* \in \mathbb{R}^{d\times l}$ formed from the true representations $\{\mathbf{B}_i^*\}$. For any $\mathbf{b} \in \mathbb{R}^l$ such that $\|\mathbf{b}\|_2 = 1$, with probability at-least $1 - \delta_1$, we have:

$$\begin{aligned}&\min_{\mathbf{u}\in\mathbb{R}^q}\|\mathbf{X}_T\widehat{\mathbf{V}}\mathbf{u} - \mathbf{X}_T\mathbf{V}^*\mathbf{b}\|_2\\ &\quad\leq \frac{\sigma^2 n_T}{r n_S}\left(km + kdm\log(\kappa n_s) + \log\left(\frac{1}{\delta_1}\right)\right)\end{aligned}$$

Proof: We first note that:

$$\|\mathbf{P}^\perp_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{X}_T\mathbf{V}^*\mathbf{b}\|_2 := \min_{\mathbf{u}\in\mathbb{R}^q}\|\mathbf{X}_T\widehat{\mathbf{V}}\mathbf{u} - \mathbf{X}_T\mathbf{V}^*\mathbf{b}\|_2$$

Using the fact that $\{\widetilde{\mathbf{w}}_i\}$ span the space $\mathbb{R}^l$, we can write $\mathbf{b} = \widetilde{\mathbf{W}}^*\boldsymbol{\alpha}$ for some $\boldsymbol{\alpha} \in \mathbb{R}^m$ where $\boldsymbol{\alpha}$ is $\mathcal{O}(1)$. We have:

$$\begin{aligned}\|\mathbf{P}^\perp_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{X}_T\mathbf{V}^*\mathbf{b}\|_2^2 &= \|\mathbf{P}^\perp_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{W}}^*\boldsymbol{\alpha}\|_2^2\\ &\lesssim \|\mathbf{P}^\perp_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{W}}^*\|_F^2\\ &= \sum_{i=1}^m\|\mathbf{P}^\perp_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{w}}_i^*\|_2^2\\ &\overset{(a)}{\lesssim} n_T\sum_{i=1}^m\|\mathbf{P}^\perp_{\boldsymbol{\Sigma}_T^{1/2}\widehat{\mathbf{V}}}\boldsymbol{\Sigma}_T^{1/2}\mathbf{V}^*\widetilde{\mathbf{w}}_i^*\|_2^2\\ &\overset{(b)}{\lesssim} \frac{n_T}{r}\sum_{i=1}^m\|\mathbf{P}^\perp_{\boldsymbol{\Sigma}_i^{1/2}\widehat{\mathbf{V}}}\boldsymbol{\Sigma}_i^{1/2}\mathbf{V}^*\widetilde{\mathbf{w}}_i^*\|_2^2\\ &\overset{(c)}{\lesssim} \frac{n_T}{r n_S}\sum_{i=1}^m\|\mathbf{P}^\perp_{\mathbf{X}_i\widehat{\mathbf{V}}}\mathbf{X}_i\mathbf{V}^*\widetilde{\mathbf{w}}_i^*\|_2^2 \qquad (10)\end{aligned}$$

where (a) follows from Claim 2 (with $\mathbf{B} = [\widehat{\mathbf{V}} \quad -\mathbf{V}^*]$), (b) follows from Assumption 3 and (c) from Claim 1. We now note that $\mathbf{V}^*\widetilde{\mathbf{w}}_i^* = \mathbf{B}_i^*\mathbf{w}_i^*$. We now note that $\widehat{\mathbf{V}}$ is the matrix whose

columns are an orthonormal basis of the set of columns of the matrices $\{\widehat{\mathbf{B}}_i\}$. Thus for each $i \in [m]$, there exists a matrix $\mathbf{C}_i$ such that $\widehat{\mathbf{B}}_i = \widehat{\mathbf{V}}\mathbf{C}_i$. Now using the result of Proposition 2 we have:

$$\|\mathbf{P}^{\perp}_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{X}_T\mathbf{V}^*\mathbf{b}\|_2^2 \overset{Prop.\tilde{}2}{\lesssim} \frac{n_T}{rn_s}\sum_{i=1}^{m}\left\|\mathbf{P}^{\perp}_{\mathbf{X}_i\widehat{\mathbf{B}}_i}\mathbf{X}_i\mathbf{B}_i^*\mathbf{w}_i^*\right\|_2^2$$

$$\leq \frac{n_T}{rn_s}\sum_{i=1}^{m}\left\|\mathbf{P}_{\mathbf{X}_i\widehat{\mathbf{B}}_i}(\mathbf{X}_i\mathbf{B}_i^*\mathbf{w}_i^* + \mathbf{z}_i) - \mathbf{X}_i\mathbf{B}_i^*\mathbf{w}_i^*\right\|_2^2$$

$$= \frac{n_T}{rn_s}\sum_{i=1}^{m}\left\|\mathbf{P}_{\mathbf{X}_i\widehat{\mathbf{B}}_i}\mathbf{y}_i - \mathbf{X}_i\mathbf{B}_i^*\mathbf{w}_i^*\right\|_2^2$$

$$= \frac{n_T}{rn_s}\sum_{i=1}^{m}\left\|\mathbf{X}_i\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_i - \mathbf{X}_i\mathbf{B}_i^*\mathbf{w}_i^*\right\|_2^2$$

The proof can then be concluded by using the result from Lemma 2 stated below that provides a bound for the trained empirical source models in (3).

We now provide Lemma 2 which establishes guarantees on the learned empirical source models.

Lemma 2 (Multi-Source training guarantee): With probability at least $1 - \frac{\delta}{5}$, we have:

$$\sum_{i=1}^{m}\left\|\mathbf{X}_i(\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_i - \mathbf{B}_i^*\mathbf{w}_i^*)\right\|_2^2 \leq \sigma^2\left(km + kdm\log(\kappa n_s) + \log\left(\frac{1}{\delta}\right)\right)$$

Proof: First we instantiate Claim 1, which happens with probability atleast $1 - \frac{\delta}{10}$. For the given source datasets and matrix $\mathbf{A} \in \mathbb{R}^{d\times m}$ with columns $\{\mathbf{a}_i\}$, we define the map $\mathcal{X}(\mathbf{A})$ as $\mathcal{X}(\mathbf{A}) = [\mathbf{X}_1\mathbf{a}_1 \quad \mathbf{X}_2\mathbf{a}_2 \ldots \quad \mathbf{X}_m\mathbf{a}_m]$. Now consider the matrix $\mathbf{\Delta} \in \mathbb{R}^{d\times m}$ whose columns are given by $\{\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_i - \mathbf{B}_i^*\mathbf{w}_i^*\}_{i=1}^m$. For convenience of notation, we define $\mathcal{X}(\mathbf{\Delta}) := [\mathbf{X}_1(\widehat{\mathbf{B}}_1\widehat{\mathbf{w}}_1 - \mathbf{B}_1^*\mathbf{w}_1^*) \quad \ldots \quad \mathbf{X}_m(\widehat{\mathbf{B}}_m\widehat{\mathbf{w}}_m - \mathbf{B}_m^*\mathbf{w}_m^*)]$. We are interested in providing a bound for the quantity $\|\mathcal{X}(\mathbf{\Delta})\|_F^2$. The ith column of the matrix $\mathbf{\Delta}$ can be decomposed as $\mathbf{R}_i\mathbf{r}_i$ where $\mathbf{R}_i \in \mathcal{O}_{d\times 2k}$ (set of tall orthonormal matrices in $d \times 2k$) and $\mathbf{r}_i \in \mathbb{R}^{2k}$.

$$\mathbf{\Delta} = [\mathbf{R}_1\mathbf{r}_1 \quad \mathbf{R}_2\mathbf{r}_2 \ldots \quad \mathbf{R}_m\mathbf{r}_m]$$

For each $i \in [m]$, we now decompose $\mathbf{X}_i\mathbf{R}_i = \mathbf{U}_i\mathbf{Q}_i$ (where $\mathbf{U}_i \in \mathcal{O}_{n_s\times 2k}$ and $\mathbf{Q} \in \mathbb{R}^{2k\times 2k}$). Since $\{\widehat{\mathbf{B}}_i, \widehat{\mathbf{w}}_i\}_{i=1}^m$ are the optimal solutions for the source regression problems, we have $\sum_{i=1}^m \|\mathbf{y}_i - \mathbf{X}_i\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_i\|_2^2 \leq \sum_{i=1}^m \|\mathbf{y}_i - \mathbf{X}_i\mathbf{B}_i^*\mathbf{w}_i^*\|_2^2$. Substituting $\mathbf{y}_i = \mathbf{X}_i\mathbf{B}_i^*\mathbf{w}_i^* + \mathbf{z}_i$ for $i \in [m]$, we get $\|\mathcal{X}(\mathbf{\Delta})\|_F^2 \leq 2\langle \mathbf{Z}, \mathcal{X}(\mathbf{\Delta})\rangle$ (where the inner product of matrices is trace of their product) and we denote the matrix of noise vectors as $\mathbf{Z} := [\mathbf{z}_1 \quad \mathbf{z}_2 \quad \ldots \quad \mathbf{z}_m] \in \mathbb{R}^{n_s\times m}$. Now:

$$\langle \mathbf{Z}, \mathcal{X}(\mathbf{\Delta})\rangle = \sum_{i=1}^{m}\mathbf{z}_i^\top\mathbf{X}_i\mathbf{R}_i\mathbf{r}_i = \sum_{i=1}^{m}\mathbf{z}_i^\top\mathbf{U}_i\mathbf{Q}_i\mathbf{r}_i \tag{11}$$

$$\leq \sum_{i=1}^{m}\left\|\mathbf{U}_i^\top\mathbf{z}_i\right\|_2\|\mathbf{Q}_i\mathbf{r}_i\|_2$$

$$\leq \sqrt{\sum\nolimits_{i=1}^{m}\left\|\mathbf{U}_i^\top\mathbf{z}_i\right\|_2}\sqrt{\sum\nolimits_{i=1}^{m}\|\mathbf{U}_i\mathbf{Q}_i\mathbf{r}_i\|_2}$$

$$= \sqrt{\sum\nolimits_{i=1}^{m}\left\|\mathbf{U}_i^\top\mathbf{z}_i\right\|_2}\,\|\mathcal{X}(\mathbf{\Delta})\|_F \tag{12}$$

We will now provide a bound for the first term in the product on the R.H.S. of (12). Since $\mathbf{U}_i$ depends on $\mathbf{R}_i$, it also depends on the value of $\mathbf{Z}$. To provide a bound, we use an ϵ-net argument to cover all possible values of $\{\mathbf{R}_i\}_{i=1}^m$. We first consider a fixed set of matrices $\{\bar{\mathbf{R}}_i\}_{i=1}^m \subset \mathcal{O}_{d\times 2k}^m$. For these given matrices, we can find decompose $\mathbf{X}_i\bar{\mathbf{R}}_i = \bar{\mathbf{U}}_i\mathbf{Q}_i$ for $i \in [m]$, where $\{\bar{\mathbf{U}}_i\}_{i=1}^m \subset \mathcal{O}_{n_T\times 2k}^m$ do not depend on $\mathbf{Z}$. Thus we have $\frac{1}{\sigma^2}\sum_{i=1}^m \|\mathbf{U}_i^\top\mathbf{z}_i\|_2 \sim \chi^2(2\text{ km})$. Thus w.p at least $1 - \delta'$, we have:

$$\sum_{i=1}^{m}\left\|\mathbf{U}_i^\top\mathbf{z}_i\right\|_2 \lesssim \sigma^2\left(km + \log\left(\frac{1}{\delta'}\right)\right) \tag{13}$$

Hence for the given $\{\bar{\mathbf{R}}_i\}_{i=1}^m$, using the result from (13) in (12), we have:

$$\langle \mathbf{Z}, \mathcal{X}(\bar{\mathbf{\Delta}})\rangle \lesssim \sigma^2\left(km + \log\left(\frac{1}{\delta'}\right)\right)\|\mathcal{X}(\bar{\mathbf{\Delta}})\|_F$$

where $\bar{\mathbf{\Delta}} = [\bar{\mathbf{R}}_1\mathbf{r}_1 \quad \bar{\mathbf{R}}_2\mathbf{r}_2 \ldots \quad \bar{\mathbf{R}}_m\mathbf{r}_m]$. Now we consider an $\frac{\epsilon}{m}$-net of $\mathcal{O}_{d\times 2k}^m$ denoted by $\mathcal{N}$ of size $|\mathcal{N}| \leq (\frac{6m\sqrt{2k}}{\epsilon})^{2kdm}$. Using the union bound, with probability at least $1 - |\mathcal{N}|\delta'$:

$$\langle \mathbf{Z}, \mathcal{X}(\bar{\mathbf{\Delta}})\rangle \lesssim \sigma^2\left(km + \log\left(\frac{1}{\delta'}\right)\right)\|\mathcal{X}(\bar{\mathbf{\Delta}})\|_F,\ \forall\{\bar{\mathbf{R}}_i\}_{i=1}^m \subset \mathcal{N} \tag{14}$$

Choose $\delta' = \frac{\delta}{20(\frac{6m\sqrt{2k}}{\epsilon})^{2kdm}}$, then the above holds with probability at least $1 - \frac{\delta}{20}$. We will now use the results from the following claim, which is proved in Section V-A2 below.

Claim 3: Under the assumptions of Theorem 1, the following hold:

1) W.p at least $1 - \frac{\delta}{20}$, $\|\mathbf{Z}\|_F^2 \lesssim \sigma^2(n_s m + \log(\frac{1}{\delta}))$
2) If the result in 1) holds and Claim 1 holds, then $\|\mathbf{\Delta}\|_F^2 \lesssim \frac{\sigma^2(n_s m + \log(\frac{1}{\delta}))}{n_s\lambda_{low}}$ where $\lambda_{low} = \min_{i\in[m]}\lambda_{\min}(\mathbf{\Sigma}_i)$
3) If the results in 1), 2) above hold and Claim 1 holds, then $\|\mathcal{X}([\mathbf{R}_1\mathbf{r}_1 \quad \ldots \mathbf{R}_m\mathbf{r}_m] - [\bar{\mathbf{R}}_1\mathbf{r}_1 \quad \ldots \bar{\mathbf{R}}_m\mathbf{r}_m])\| \lesssim \frac{\kappa\epsilon^2}{m^2}\sigma^2(n_s m + \log(\frac{1}{\delta}))$ for some $\{\bar{\mathbf{R}}_i\} \subset \mathcal{N}$ where $\kappa = \frac{\max_{i\in[m]}\lambda_{\min}(\mathbf{\Sigma}_i)}{\min_{i\in[m]}\lambda_{\min}(\mathbf{\Sigma}_i)}$

We now use the results of the Claim 3 to complete the proof of the lemma. We note that there exists some $\bar{\mathbf{\Delta}}$ with $\{\bar{\mathbf{R}}_i\}_{i=1}^m \subset \mathcal{N}$ such that:

$$\frac{1}{2}\|\mathcal{X}(\mathbf{\Delta})\|_F^2 \leq \langle \mathbf{Z}, \mathcal{X}(\mathbf{\Delta})\rangle$$

$$= \langle \mathbf{Z}, \mathcal{X}(\bar{\mathbf{\Delta}})\rangle + \langle \mathbf{Z}, \mathcal{X}(\mathbf{\Delta} - \bar{\mathbf{\Delta}})\rangle$$

$$\overset{(a)}{\lesssim} \sigma\sqrt{\left(km + \log\left(\frac{1}{\delta'}\right)\right)}.\|\mathcal{X}(\bar{\mathbf{\Delta}})\|_F^2 + \|\mathbf{Z}\|_F\|\mathcal{X}(\mathbf{\Delta} - \bar{\mathbf{\Delta}})\|_F$$

$$\overset{(b)}{\leq} \sigma\sqrt{\left(km + \log\left(\frac{1}{\delta'}\right)\right)}.\left(\|\mathcal{X}(\mathbf{\Delta})\|_F + \|\mathcal{X}(\mathbf{\Delta} - \bar{\mathbf{\Delta}})\|_F\right)$$

$$+\sigma\sqrt{\left(n_s m+\log\left(\frac{1}{\delta}\right)\right)}\|\mathcal{X}(\mathbf{\Delta}-\bar{\mathbf{\Delta}})\|_F$$

$$\overset{(c)}{\lesssim}\sigma\|\mathcal{X}(\mathbf{\Delta})\|_F\sqrt{\left(km+\log\left(\frac{1}{\delta'}\right)\right)}$$

$$+\frac{\sqrt{\kappa}\epsilon}{m}\sigma^2\left(n_s m+\log\left(\frac{1}{\delta'}\right)\right)$$

where (a) follows from (14) w.p $\geq 1-\frac{\delta}{20}$, (b) from 1) in Claim 3); w.p. $\geq 1-\frac{\delta}{20}$ and (c) uses the fact that $\delta'<\delta, k\leq n_s$, and 3) in Claim 3. Since the above result gives an inequality in terms of $\|\mathcal{X}(\mathbf{\Delta})\|_F^2$ and $\|\mathcal{X}(\mathbf{\Delta})\|_F$, we can conclude the following:

$$\|\mathcal{X}(\mathbf{\Delta})\|_F\lesssim\max\left\{\sigma\sqrt{\left(km+\log\left(\frac{1}{\delta'}\right)\right)},\right.$$
$$\left.\sigma\sqrt{\frac{\sqrt{\kappa}\epsilon}{m}\left(n_s m+\log\left(\frac{1}{\delta'}\right)\right)}\right\}$$

We choose $\epsilon=\frac{km}{n_s\sqrt{\kappa}}$, and note that $n_S\gg k$, which gives:

$$\|\mathcal{X}(\mathbf{\Delta})\|_F\leq\sigma\sqrt{\left(km+\log\left(\frac{1}{\delta'}\right)\right)}$$

Substituting the value of $\delta'=\frac{\delta}{20(\frac{6m\sqrt{2k}}{\epsilon})^{2kdm}}$ and $\epsilon=\frac{km}{n_s\sqrt{\kappa}}$:

$$\|\mathcal{X}(\mathbf{\Delta})\|_F\lesssim\sigma\sqrt{km+kdm\log\left(\frac{mk}{\epsilon}\right)+\log\left(\frac{1}{\delta}\right)}$$
$$\leq\sigma\sqrt{km+kdm\log\left(n_s\kappa\right)+\log\left(\frac{1}{\delta}\right)}$$

Hence the following holds with probability at least 1 - $(\frac{\delta}{10}+\frac{\delta}{20}+\frac{\delta}{20})$:

$$\|\mathcal{X}(\mathbf{\Delta})\|_F^2\lesssim\sigma^2\left[km+kdm\log\left(n_s\kappa\right)+\log\left(\frac{1}{\delta}\right)\right]$$

2) *Proof of Claim 3:*

1) This follows from the fact that $\frac{1}{\sigma^2}\|\mathbf{Z}\|_F^2\sim\chi(n_s m)$

2)

$$\|\mathcal{X}(\mathbf{\Delta})\|_F^2=\sum_{i=1}^m\|\mathbf{X}_i(\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_{i_0}-\mathbf{B}_i^*\mathbf{w}_i^*)\|_2^2$$
$$=\sum_{i=1}^m(\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_{i_0}-\mathbf{B}_i^*\mathbf{w}_i^*)^\top\mathbf{X}_i^\top\mathbf{X}_i(\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_{i_0}-\mathbf{B}_i^*\mathbf{w}_i^*)$$
$$\gtrsim n_s\sum_{i=1}^m(\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_{i_0}-\mathbf{B}_i^*\mathbf{w}_i^*)^\top\mathbf{\Sigma}_i(\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_{i_0}-\mathbf{B}_i^*\mathbf{w}_i^*)$$
$$\geq n_s\sum_{i=1}^m\lambda_{\min}(\mathbf{\Sigma}_i)\|(\widehat{\mathbf{B}}_i\widehat{\mathbf{w}}_{i_0}-\mathbf{B}_i^*\mathbf{w}_i^*)\|_2^2$$
$$\geq n_s\lambda_{low}\|\mathbf{\Delta}\|_F^2$$

where $\lambda_{low}:=\min_{i\in[m]}\lambda_{\min}(\mathbf{\Sigma}_i)$. Since $\|\mathcal{X}(\mathbf{\Delta})\|_F^2\leq 2\langle\mathbf{Z},\mathcal{X}(\mathbf{\Delta})\rangle\leq 2\|\mathbf{Z}\|_F\|\mathcal{X}(\mathbf{\Delta})\|_F$, we have $\|\mathcal{X}(\mathbf{\Delta})\|_F\leq 2\|\mathbf{Z}\|_F$. Using the result from part 1. of the claim statement combined with the upper bound derived above, we have:

$$\|\mathbf{\Delta}\|_F^2\lesssim\frac{\sigma^2}{n_s\lambda_{low}}\left(n_s m+\log\left(\frac{1}{\delta}\right)\right)$$

3) For some $\{\bar{\mathbf{R}}_i\}\subset\mathcal{N}$ we have $\sum_{i=1}^m\|\mathbf{R}_i-\bar{\mathbf{R}}_i\|_F\leq\sum_{i=1}^m\frac{\epsilon}{m}=\epsilon$. Therefore:

$$\|\mathcal{X}(\mathbf{\Delta}-\bar{\mathbf{\Delta}})\|_F^2=\sum_{i=1}^m\|\mathbf{X}_i(\mathbf{R}_i-\bar{\mathbf{R}}_i)\mathbf{r}_i\|_2^2$$
$$\leq\sum_{i=1}^m\|\mathbf{X}_i\|_2^2\|\mathbf{R}_i-\bar{\mathbf{R}}_i\|_F^2\|\mathbf{r}_i\|_2^2$$
$$\lesssim\frac{n_s\epsilon^2}{m^2}\sum_{i=1}^m\|\mathbf{\Sigma}_i\|_2^2\|\mathbf{r}_i\|_2^2$$
$$\lesssim\frac{n_s\epsilon^2\lambda_{high}}{m^2}\sum_{i=1}^m\|\mathbf{r}_i\|_2^2$$
$$\overset{(a)}{=}\frac{n_s\epsilon^2\lambda_{high}}{m^2}\|\mathbf{\Delta}\|_F^2$$
$$\overset{(b)}{\lesssim}\frac{n_s\epsilon^2\lambda_{high}}{m^2}\frac{\sigma^2(n_s m+\log(\frac{1}{\delta}))}{n_s\lambda_{low}}$$
$$=\frac{\kappa\epsilon^2\lambda_{high}\sigma^2}{m^2\lambda_{low}}\left(n_s m+\log\left(\frac{1}{\delta}\right)\right)$$

Here, $\lambda_{high}:=\max_{i\in[m]}\lambda_{\max}(\mathbf{\Sigma}_i)$ where to arrive at (a), we have used the fact that $\{\mathbf{R}_i\}$ have orthonormal columns and used the definition of $\mathbf{\Delta}$, and (b) follows from using 2) from the claim statement.

B. *Proof of Theorem 1*

Having established the helper results above, we now provide a proof for Theorem 1.

[Restating Theorem 1] Fix a failure probability $\delta\in(0,1)$ and further assume $2k\leq\min\{d,m\}$ and the number of samples in the sources and target satisfy $n_s\gg\rho^4(d+\log(m/\delta))$ and $n_{T_1}\gg\rho^4(\max\{l,q\}+\log(1/\delta))$, respectively. Define $\kappa=\frac{\max_{i\in[m]}\lambda_{\max}(\mathbf{\Sigma}_i)}{\min_{i\in[m]}\lambda_{\min}(\mathbf{\Sigma}_i)}$ where $\lambda_{\max}(\mathbf{\Sigma}_i)$ denotes the maximum eigenvalue of $\mathbf{\Sigma}_i$. Then with probability at least $1-\delta$ over the samples, under Assumptions 1–5, the expected excess risk of the learned predictor $\widehat{\mathbf{w}}_T$ on the target $(\mathbf{x}\to\mathbf{x}^\top\widehat{\mathbf{V}}\widehat{\mathbf{w}}_T)$ for Phase 1 satisfies:

$$\mathbb{E}[\mathrm{EER}(\boldsymbol{\theta}_{\mathrm{Phase}_1},\boldsymbol{\theta}_T^*)]\lesssim\frac{\sigma^2}{n_{T_1}}(q+\log(1/\delta))+\epsilon^2$$
$$+\sigma^2\left[\frac{1}{rn_s m}\log\left(\frac{1}{\delta}\right)+\left(\frac{kd\log(\kappa n_s)+k}{rn_s}\right)\right]$$

Proof: We will first instantiate Lemma 1. We then instantiate Lemma 2 twice, once with $[\widehat{\mathbf{V}}\ \ -\mathbf{V}^*]$ and the other time with $[\mathbf{B}_T^*\ \ -\mathbf{V}^*]$. Then we assume that the result from Lemma 2 holds. All these events happen together with probability at least

$1-\frac{2\delta}{5}$. The expected error for the target distribution is given by:

$$\begin{aligned}
&\mathrm{EER}(\boldsymbol{\theta}_{\text{Phase}_1}, \boldsymbol{\theta}_T^*) = \mathbb{E}_{\mathbf{x}\sim p_T}\left[\mathbf{x}^\top \mathbf{B}_T^* \mathbf{w}_T^* - \mathbf{x}^\top \widehat{\mathbf{V}}\widehat{\mathbf{w}}_T\right]^2 \\
&= \left\|\boldsymbol{\Sigma}_T^{1/2}\left(\widehat{\mathbf{V}}\widehat{\mathbf{w}}_T - \mathbf{B}_T^*\mathbf{w}_T^*\right)\right\|_2^2 \\
&\lesssim \left\|\boldsymbol{\Sigma}_T^{1/2}\left(\widehat{\mathbf{V}}\widehat{\mathbf{w}}_T - \mathbf{V}^*\widetilde{\mathbf{w}}_T^*\right)\right\|_2^2 + \left\|\boldsymbol{\Sigma}_T^{1/2}\left(\mathbf{B}_T^*\mathbf{w}_T^* - \mathbf{V}^*\widetilde{\mathbf{w}}_T^*\right)\right\|_2^2 \\
&\overset{(a)}{\lesssim} \left\|\boldsymbol{\Sigma}_T^{1/2}\left(\widehat{\mathbf{V}}\widehat{\mathbf{w}}_T - \mathbf{V}^*\widetilde{\mathbf{w}}_T^*\right)\right\|_2^2 + \epsilon^2 \\
&\overset{(b)}{\lesssim} \frac{1}{n_T}\left\|\mathbf{X}_T\left(\widehat{\mathbf{V}}\widehat{\mathbf{w}}_T - \mathbf{V}^*\widetilde{\mathbf{w}}_T^*\right)\right\|_2^2 + \epsilon^2 \\
&= \frac{1}{n_T}\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{y}_T - \mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{w}}_T^*\right\|_2^2 + \epsilon^2 \\
&\lesssim \frac{1}{n_T}\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}(\mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{w}}_T^* + \mathbf{z}_T) - \mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{w}}_T^*)\right\|_2^2 \\
&\quad + \frac{1}{n_T}\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}(\mathbf{X}_T\mathbf{B}_T^*\mathbf{w}_T^* - \mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{w}}_T^*)\right\|_2^2 + \epsilon^2
\end{aligned}$$

where (a) follows from Assumption 1 and (b) uses Claim 2. Using the fact that $\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}\|_2 \le 1$ (since $\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}$ is a projection matrix) and using Claim 2, we have:

$$\begin{aligned}
&\lesssim \frac{1}{n_T}\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}(\mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{w}}_T^* + \mathbf{z}_T) - \mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{w}}_T^*)\right\|_2^2 + \epsilon^2 \\
&\lesssim \frac{1}{n_T}\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}^\perp\mathbf{X}_T\mathbf{V}^*\widetilde{\mathbf{w}}_T^*\right\|_2^2 + \epsilon^2 + \frac{1}{n_T}\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{z}_T\right\|_2^2
\end{aligned}$$

where the first inequality follows from Assumption 1 and Claim 2. We can take the expectation over the distribution of $\mathbf{w}_T^*$ and use Assumption 5 to yield:

$$\begin{aligned}
\mathbb{E}_{\widetilde{\mathbf{w}}_T^*}[\mathrm{EER}(\boldsymbol{\theta}_{\text{Phase}_1}, \boldsymbol{\theta}_T^*)] &\lesssim \frac{1}{n_T l}\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}^\perp\mathbf{X}_T\mathbf{V}^*\right\|_F^2 \\
&\quad + \epsilon^2 + \frac{1}{n_T}\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{z}_T\right\|_2^2 \qquad (15)
\end{aligned}$$

We now make use of the following lemma, which is proved below in Section V-B1 that provides a bound for the first term in (15).

Lemma 3 (Target Training Guarantee): Assuming the results in Claim 1, Claim 2 (with $\mathbf{B} = [\widehat{\mathbf{V}} \quad -\mathbf{V}^*]$) and Lemma 2 hold, we then have:

$$\begin{aligned}
\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}^\perp\mathbf{X}_T\mathbf{V}^*\right\|_F^2 &\lesssim \frac{n_T\sigma^2}{rn_s\sigma_l^2(\widetilde{\mathbf{W}}^*)} \\
&\quad \times \left(km + kdm\log(\kappa n_s) + \log\left(\frac{1}{\delta}\right)\right)
\end{aligned}$$

Substituting the result from Lemma 3 in (15) and using $\sigma_l^2(\widetilde{\mathbf{W}}^*) \ge \frac{m}{l}$, the following bound holds with probability at least $1-\frac{2\delta}{5}$:

$$\begin{aligned}
\mathbb{E}_{\widetilde{\mathbf{w}}_T^*}[\mathrm{EER}(\boldsymbol{\theta}_{\text{Phase}_1}, \boldsymbol{\theta}_T^*)] &\lesssim \frac{1}{n_T}\left\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{z}_T\right\|_2^2 + \epsilon^2 \\
&\quad + \frac{1}{rn_s m}\sigma^2\left(km + kdm\log(\kappa n_s) + \log\left(\frac{1}{\delta}\right)\right)
\end{aligned}$$

Finally, the last term in above can be bounded by using a concentration inequality for χ^2-squared distribution. In particular, with probability at least $1-\frac{3\delta}{5}$ we have $\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}\mathbf{z}_T\|_2^2 \lesssim \sigma^2(q+\log(1/\delta))$. Thus the following bound holds on $\mathbb{E}_{\mathbf{w}_T^*}[\mathrm{Err}(\widehat{\mathbf{B}}_S, \mathbf{w}_T^*)]$ with probability at least $1-\delta$:

$$\begin{aligned}
\mathbb{E}_{\mathbf{w}_T^*}[\mathrm{EER}(\boldsymbol{\theta}_{\text{Phase}_1}, \boldsymbol{\theta}_T^*)] &\lesssim \epsilon^2 + \frac{1}{n_T}\sigma^2(q+\log(1/\delta)) \\
&\quad + \frac{1}{rn_s m}\sigma^2\left(km + kdm\log(\kappa n_s) + \log\left(\frac{1}{\delta}\right)\right) \\
&= \sigma^2\left[\frac{1}{rn_s m}\log\left(\frac{1}{\delta}\right) + \left(\frac{kd\log(\kappa n_s) + k}{rn_s}\right)\right] \\
&\quad + \frac{\sigma^2}{n_T}(q+\log(1/\delta)) + \epsilon^2
\end{aligned}$$

1) Proof of Lemma 3: We start the proof by using Proposition 1 and Claim 2:

$$\begin{aligned}
\sigma_l^2(\widetilde{\mathbf{W}}^*)\|\mathbf{P}_{\mathbf{X}_T\widehat{\mathbf{V}}}^\perp\mathbf{X}_T\mathbf{V}^*\|_F^2 &\lesssim \sigma_l^2(\widetilde{\mathbf{W}}^*)n_T\left\|\mathbf{P}_{\boldsymbol{\Sigma}_T^{1/2}\widehat{\mathbf{V}}}^\perp\boldsymbol{\Sigma}_T^{1/2}\mathbf{V}^*\right\|_F^2 \\
&\le n_T\sum_{i=1}^m\left\|\mathbf{P}_{\boldsymbol{\Sigma}_T^{1/2}\widehat{\mathbf{V}}}^\perp\boldsymbol{\Sigma}_T^{1/2}\mathbf{V}^*\widetilde{\mathbf{w}}_i^*\right\|_2^2
\end{aligned}$$

The proof now follows the same procedure as in Proof of Lemma 1 starting from (10) and following it up with Lemma 2.

VI. Proof for Phase 2 Training

We now provide a proof for our bound in Theorem 2 which establishes excess generalization risk for the model obtained after combined Phase 1 and Phase 2 training.

The data for Phase 2 training is given by $(\mathbf{X}_{T_2}, \mathbf{y}_{T_2})$ where $\mathbf{y}_{T_2} = \mathbf{X}_{T_2}\mathbf{B}_T^*\mathbf{w}_T^* + \mathbf{z}_{T_2}$. Here $\boldsymbol{\theta}_T^* := \mathbf{B}_T^*\mathbf{w}^*$ denotes the true data generating target model. We define $\mathbf{P}_\parallel = \mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{X}_{T_2}$ as projection matrix on the row-space of matrix $\mathbf{X}_{T_2}$ and $\mathbf{P}_\perp = \mathbf{I} - \mathbf{P}_\parallel$.

We first note the following result that establishes where the Gradient Descent solution converges to, the proof of which is given in Section VI-A1 below.

Lemma 4: Under the assumptions of Theorem 2, performing gradient descent on the objective (6) with the initialization $\boldsymbol{\theta}^{(0)} := \boldsymbol{\theta}_{\text{Phase}_1}$ and learning rate η, yields the solution:

$$\boldsymbol{\theta}_{GD} := \boldsymbol{\theta}^{(\infty)} = \mathbf{P}_\parallel\boldsymbol{\theta}_T^* + \mathbf{P}_\perp\boldsymbol{\theta}_{\text{Phase}_1} + \mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{z}$$

where $\mathbf{P}_\parallel = \mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{X}_{T_2}$ is the projection matrix on the row-space of matrix $\mathbf{X}_{T_2}$ and $\mathbf{P}_\perp = \mathbf{I} - \mathbf{P}_\parallel$.

A. Proof of Theorem 2

We now use the solution of the gradient descent $\hat{\boldsymbol{\theta}}_T := \boldsymbol{\theta}_{GD}$ derived in Lemma 4 and find the excess population risk. The excess risk is given by:

$$\begin{aligned}
\mathrm{EER}(\hat{\boldsymbol{\theta}}_T, \boldsymbol{\theta}_T^*) &= \mathbb{E}_{\mathbf{X}_{T_2}\sim p_T}(\mathbf{X}_{T_2}^\top\boldsymbol{\theta}_T^* - \mathbf{X}_{T_2}^\top\boldsymbol{\theta}_{GD})^2 \\
&= \mathbb{E}_{\mathbf{X}_{T_2}\sim p_T}\mathrm{Tr}[(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{GD})^\top\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{GD})] \\
&= \|\boldsymbol{\Sigma}_T^{1/2}(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{GD})\|_2^2
\end{aligned}$$

Substituting the value of $\boldsymbol{\theta}_{GD}$ from Lemma 4 we get:

$$\mathrm{EER}(\hat{\boldsymbol{\theta}}_T, \boldsymbol{\theta}_T^*) \leq 2\|\boldsymbol{\Sigma}_T^{1/2}\mathbf{P}_\perp(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\|_2^2 + 2\|\boldsymbol{\Sigma}_T^{1/2}\mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{z}\|_2^2 \tag{16}$$

We focus on the first term for now. We have:

$$\begin{aligned}\|\boldsymbol{\Sigma}_T^{1/2}\mathbf{P}_\perp(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\|_2^2 &= (\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})^\top\mathbf{P}_\perp^\top\boldsymbol{\Sigma}_T\mathbf{P}_\perp(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\\ &= (\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})^\top\mathbf{P}_\perp^\top\left(\boldsymbol{\Sigma}_T - \frac{1}{n_T}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right)\mathbf{P}_\perp(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\\ &= \left\|\left(\boldsymbol{\Sigma}_T - \frac{1}{n_T}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right)^{1/2}\mathbf{P}_\perp(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\right\|_2^2\\ &\leq \left\|\boldsymbol{\Sigma}_T - \frac{1}{n_T}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right\|_2\|\mathbf{P}_\perp(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\|_2^2 \end{aligned}\tag{17}$$

The second term in (17) can be bounded by $\|\mathbf{P}_\perp(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\|_2^2 \leq \|(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\|_2^2$ by noting that $\|\mathbf{P}_\perp\|_2 \leq 1$. We thus finally get the bound:

$$\mathrm{EER}(\hat{\boldsymbol{\theta}}_T, \boldsymbol{\theta}_T^*) \leq 2\left\|\boldsymbol{\Sigma}_T - \frac{1}{n_T}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right\|_2\|(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\|_2^2 + 2\|\boldsymbol{\Sigma}_T^{1/2}\mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{z}\|_2^2 \tag{18}$$

We now provide a bound for the second term in (18). Note:

$$\|\boldsymbol{\Sigma}_T^{1/2}\mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{z}\|_2^2 = \mathbf{z}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{X}_{T_2}\boldsymbol{\Sigma}_T\mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{z}$$

From [15, Lemma 9], we can get a high probability bound (probability $> 1 - e^{-t}$) on this term for some $t > 0$ as

$$\|\boldsymbol{\Sigma}_T^{1/2}\mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{z}\|_2^2 \leq (4t+2)\sigma^2\mathrm{Tr}\left((\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{X}_{T_2}\boldsymbol{\Sigma}_T\mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\right)$$

To bound the trace term, we use [15, Lemma 13, 18]: For universal constant $b, c \geq 1$ and $k^* := \min\{k \geq 0 : r_k(\boldsymbol{\Sigma}_T) \geq bn\}$, we have

$$\mathbf{Tr}\left((\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{X}_{T_2}\boldsymbol{\Sigma}_T\mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\right) \leq c\left(\frac{k^*}{bn} + \frac{bn}{R_{k^*}(\boldsymbol{\Sigma}_T)}\right)$$

where $r_k(\boldsymbol{\Sigma}_T) = \frac{\boldsymbol{\Sigma}_{i>k}\lambda_i}{\lambda_{k+1}}$, $R_{k^*}(\boldsymbol{\Sigma}_T) = \frac{(\boldsymbol{\Sigma}_{i>k}\lambda_i)^2}{\boldsymbol{\Sigma}_{i>k}\lambda_i^2}$. Substituting the value of $t = \log(\frac{2}{\delta})$ Plugging the resulting bound in (18), we can finally claim that the following holds with probability at least $1 - \frac{\delta}{2}$:

$$\mathrm{EER}(\hat{\boldsymbol{\theta}}_T, \boldsymbol{\theta}_T^*) \lesssim \left\|\boldsymbol{\Sigma}_T - \frac{1}{n_T}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right\|_2\|(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\|_2^2 + c\sigma^2\log\left(\frac{1}{\delta}\right)\left(\frac{k^*}{bn} + \frac{bn}{R_{k^*}(\boldsymbol{\Sigma}_T)}\right)$$

The covariance approximation error in the first term of above can be bounded by the result in [37, Theorem 9]. This yields the following bound on the approximation with probability at least $1 - e^{-\delta_1}$ over the choice of data matrix $\mathbf{X}_{T_2}$ for some constant $u > 0$ and $\delta_1 > 1$

$$\left\|\boldsymbol{\Sigma}_T - \frac{1}{n_T}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right\|_2 \leq u\lambda_1\max\left\{\sqrt{\frac{\sum_{i=1}^d\lambda_i}{n_T\lambda_1}}, \frac{\sum_{i=1}^d\lambda_i}{n_T\lambda_1}, \sqrt{\frac{\delta_1}{n_T}}, \frac{\delta_1}{n_T}\right\}$$

Substituting $\delta_1 = \log(\frac{2}{\delta})$, with probability at least $1 - \frac{\delta}{2}$,

$$\left\|\boldsymbol{\Sigma}_T - \frac{1}{n_T}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right\|_2 \leq u\lambda_1\max\left\{\sqrt{\frac{\sum_{i=1}^d\lambda_i}{n_T\lambda_1}}, \frac{\sum_{i=1}^d\lambda_i}{n_T\lambda_1}, \sqrt{\frac{1}{n_T}\log\left(\frac{1}{\delta}\right)}, \frac{1}{n_T}\log\left(\frac{1}{\delta}\right)\right\} \tag{19}$$

Denote the eigenvalues of covariance matrix of the target data as $\{\lambda_i\}_{i=1}^d$, with $\lambda_1 \geq \ldots \lambda_d$, we then have:

$$\mathrm{EER}(\hat{\boldsymbol{\theta}}_T, \boldsymbol{\theta}_T^*) \lesssim \left\|\boldsymbol{\Sigma}_T - \frac{1}{n_{T_2}}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right\|_2\frac{1}{\lambda_d}\|\boldsymbol{\Sigma}_T(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\|_2^2 + c\sigma^2\log\left(\frac{1}{\delta}\right)\left(\frac{k^*}{bn_{T_2}} + \frac{bn_{T_2}}{R_{k^*}(\boldsymbol{\Sigma}_T)}\right)$$

where $r_k(\boldsymbol{\Sigma}_T) = \frac{\boldsymbol{\Sigma}_{i>k}\lambda_i}{\lambda_{k+1}}$, $R_{k^*}(\boldsymbol{\Sigma}_T) = \frac{(\boldsymbol{\Sigma}_{i>k}\lambda_i)^2}{\boldsymbol{\Sigma}_{i>k}\lambda_i^2}$. Here, constant $b > 1$ and $k^* = \min\{k \geq 0 : r_k(\boldsymbol{\Sigma}) \geq bn\}$ and the covariance estimation term can be bounded by $\|\boldsymbol{\Sigma}_T - \frac{1}{n_{T_2}}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\|_2 \leq u\lambda_1\max\{\sqrt{\frac{\sum_{i=1}^d\lambda_i}{n_{T_2}\lambda_1}}, \frac{\sum_{i=1}^d\lambda_i}{n_{T_2}\lambda_1}, \sqrt{\frac{1}{n_{T_2}}\log(\frac{1}{\delta})}, \frac{1}{n_{T_2}}\log(\frac{1}{\delta})\}$ with probability at least $1 - \frac{\delta}{2}$. We now substitute the value of $\|\boldsymbol{\Sigma}_T(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0})\|_2^2$ from Theorem 1 (as $\boldsymbol{\theta}_{T_0} = \widehat{\mathbf{V}}\widehat{\mathbf{w}}_T$ which was obtained by using n_{T_1} target samples). Thus the final bound after Phase 1 and Phase 2 training, after taking the expectation w.r.t the target model $\boldsymbol{\theta}_T^*$, is given by:

$$\begin{aligned}&\mathbb{E}[\mathrm{EER}(\hat{\boldsymbol{\theta}}_T, \boldsymbol{\theta}_T^*)]\\ &\lesssim \left\|\boldsymbol{\Sigma}_T - \frac{1}{n_{T_2}}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right\|_2\frac{\sigma^2}{\lambda_d}\frac{1}{rn_sm}\log\left(\frac{1}{\delta}\right)\\ &+ \left\|\boldsymbol{\Sigma}_T - \frac{1}{n_{T_2}}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right\|_2\frac{\sigma^2}{\lambda_d}\left(\frac{kd\log(\kappa n_s) + k}{rn_s}\right)\\ &+ \left\|\boldsymbol{\Sigma}_T - \frac{1}{n_{T_2}}\mathbf{X}_{T_2}^\top\mathbf{X}_{T_2}\right\|_2\frac{1}{\lambda_d}\left(\sigma^2\left[\frac{1}{n_{T_1}}(q + \log(1/\delta))\right] + \epsilon^2\right)\\ &+ c\sigma^2\log\left(\frac{1}{\delta}\right)\left(\frac{k^*}{bn_{T_2}} + \frac{bn_{T_2}}{R_{k^*}(\boldsymbol{\Sigma}_T)}\right)\end{aligned}$$

where u, c are universal constants. We now substitute the bound for the covariance estimate from (19) and simplfy the expression by assuming $\frac{r_0(\boldsymbol{\Sigma}_T)}{n_{T_2}} \geq \frac{1}{n_{T_2}}\log(\frac{1}{\delta}) \geq 1$ since we have a few target samples:

$$\mathbb{E}[\mathrm{EER}(\hat{\boldsymbol{\theta}}_T, \boldsymbol{\theta}_T^*)] \leq \frac{u\lambda_1}{\lambda_d}\frac{r_0(\boldsymbol{\Sigma}_T)}{n_{T_2}}\left(\frac{\sigma^2}{n_{T_1}}(q + \log(1/\delta)) + \epsilon^2\right)$$

$$+ \frac{u\lambda_1\sigma^2}{\lambda_d}\frac{r_0(\mathbf{\Sigma}_T)}{n_{T_2}}\left(\frac{1}{rn_s m}\log\left(\frac{1}{\delta}\right) + \left(\frac{kd\log(\kappa n_s)+k}{rn_s}\right)\right)$$
$$+ c\sigma^2\log\left(\frac{1}{\delta}\right)\left(\frac{k^*}{bn_{T_2}} + \frac{bn_{T_2}}{R_{k^*}(\mathbf{\Sigma}_T)}\right)$$

1) Proof of Lemma 4: For any time step t of the gradient descent process, the gradient of the objective in (6) evaluated at $\boldsymbol{\theta}^{(t)}$ is given by:

$$\nabla f(\boldsymbol{\theta}^{(t)}) = \frac{2}{n_T}\mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\boldsymbol{\theta}^{(t)} - \mathbf{y}_{T_2})$$

The gradient update step using step size η is given by:

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \frac{2\eta}{n_T}\mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\boldsymbol{\theta}^{(t)} - \mathbf{y}_{T_2})$$
$$= \boldsymbol{\theta}_{T_0} + \mathbf{X}_{T_2}^\top\mathbf{a}^{(t)}$$

where $\mathbf{a}$ is some vector in $\mathbb{R}^{n_T}$. In the limit, the gradient descent convergence to a solution of the form:

$$\boldsymbol{\theta}^{(\infty)} = \boldsymbol{\theta}_{T_0} + \mathbf{X}_{T_2}^\top\mathbf{a} \tag{20}$$

Since the problem in (6) is over-parameterized, there exists a value $\boldsymbol{\theta}^*$ such that $f(\boldsymbol{\theta}^*) = 0$. This follows from the fact that $\mathbf{X}_{T_2}$ has full row rank. The gradient descent solution, under an appropriate choice of the learning rate, thus converges to this value while yield a zero loss, implying:

$$\mathbf{X}_{T_2}\boldsymbol{\theta}^{(\infty)} = \mathbf{y}_{T_2} = \mathbf{X}_{T_2}\boldsymbol{\theta}_T^* + \mathbf{z}_{T_2}$$
$$\Rightarrow \mathbf{X}_{T_2}(\boldsymbol{\theta}_{T_0} + \mathbf{X}_{T_2}^\top\mathbf{a}) = \mathbf{X}_{T_2}\boldsymbol{\theta}_T^* + \mathbf{z}_{T_2}$$
$$\Rightarrow \mathbf{a} = (\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}(\mathbf{X}_{T_2}(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0}) + \mathbf{z}_{T_2})$$

Substituting this in (20), we get

$$\boldsymbol{\theta}_{GD} := \boldsymbol{\theta}^{(\infty)}$$
$$= \boldsymbol{\theta}_{T_0} + \mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}(\mathbf{X}_{T_2}(\boldsymbol{\theta}_T^* - \boldsymbol{\theta}_{T_0}) + \mathbf{z}_{T_2})$$
$$= \mathbf{P}_\parallel\boldsymbol{\theta}_T^* + \mathbf{P}_\perp\boldsymbol{\theta}_{T_0} + \mathbf{X}_{T_2}^\top(\mathbf{X}_{T_2}\mathbf{X}_{T_2}^\top)^{-1}\mathbf{z}_{T_2}$$

VII. Numerical Results

We now provide numerical simulations for our proposed scheme for optimizing linear regression objectives in a data scarce regime. To demonstrate the effectiveness of leveraging pre-trained representations and fine-tuning, we consider the case where we have access to the true representation matrix $\mathbf{V}^*$ formed by the source representations. We compare the performance of models obtained after Phase 1 and Phase 2 training for different parameters of interest and discuss their sample complexity requirements.

A. Setup

We generate the $d \times q$ matrix $\mathbf{V}^*$ matrix with entries sampled from the standard normal distribution, with $d = 1000$ and $q = 50$. We generate $n_{T_1} \in \{100, 200, 300, 1000\}$ number of samples for the target data for Phase 1 to form the matrix $\mathbf{X}_{T_1}$, which is generated with i.i.d Gaussian entries with mean 0 and covariance matrix Σ_T. To simulate slowly decaying eigenvalues of Σ_T, we set them as $\lambda_j = e^{\frac{-j}{\tau}} + \varepsilon$ for $j \in [d]$, where $\tau = 1$ is the decay factor and $\varepsilon = 0.0001$. The true target model $\boldsymbol{\theta}_T^*$ is generated as $\mathbf{u} + \mathbf{v}$ where $\mathbf{u} \in \mathbb{R}^d$ lies in the span of $\mathbf{V}^*$ and $\mathbf{v}$ is Gaussian vector with covariance matrix $\mathrm{I}\sigma_T^2$ and zero mean. We call the expected ratio of $\mathbf{u}$ to $\mathbf{v}$ as the in-out mixture representation ratio.[4] The target output vector $\mathbf{y}_T$ is generated as $\mathbf{y}_{T_1} = \mathbf{X}_T\boldsymbol{\theta}_T^* + \mathbf{z}_{T_1}$, with $\mathbf{z}_{T_1}$ being a Gaussian noise vector. Phase 1 training thus seeks to minimize the objective in (5) (with $\widehat{\mathbf{V}}$ replaced by $\mathbf{V}^*$, since we assume access to complete source representations) and we denote the output model $\boldsymbol{\theta}_{\text{Phase}_1}$. For Phase 2 training, we optimize the objective in (6) using new $n_{T_2} \in \{100, 200, 300, 1000\}$ number of target samples with $\boldsymbol{\theta}_{\text{Phase}_1}$ as initialization to obtain the final model $\boldsymbol{\theta}_{\text{Phase}_2}$. We compare the performance of $\boldsymbol{\theta}_{\text{Phase}_1}$ and $\boldsymbol{\theta}_{\text{Phase}_2}$ on 500 test samples generated from the target data. As a baseline, we also consider the performance of the scheme which takes $n_{T_1} + n_{T_2}$ number of target samples and trains the model from scratch, i.e., without leveraging the source representations $\mathbf{V}^*$. We denote the model obtained form this scheme as $\boldsymbol{\theta}_0$.

B. Results

The results from Phase 1 and Phase 2 training for different splits for the number of phase target samples and ϵ values[5] are shown in Table I. The numerical values, which are averaged over 10 independent runs, denote the ratio of the error obtained by the learned model after the respective phase and the error of the underlying true data generation model $\boldsymbol{\theta}_T^*$ for the target data on a test dataset. In a data-scarce regime, $(n_{T_1}, n_{T_2}) \in \{(100, 100), (200, 200), (300, 300)\}$, the performance of the model learned from scratch (without leveraging source representations; denoted by column *Scratch*) can be unsatisfactory. As expected, even pre-training (Phase 1) and fine-tuning (Phase 2) in low data regimes does not yield good performance if the source models are not useful for the target data, which is the case when the in-representation mixture to out-mixture ratio for $\boldsymbol{\theta}_T^*$ is small, as shown by the performance for values 5dB, 1dB in Table I. However, utilizing source representations gives significantly better performance relative to training from scratch in cases when $\boldsymbol{\theta}_T^*$ doesn't lie far off from $\mathbf{V}^*$ as can be seen by comparing the values of Phase 2 and Scratch training results for in-out signal mixture ratio of 50 dB, 20 dB, 10 dB. Thus leveraging source representations for target training can be beneficial in scare-data regimes when source representation are useful for the target task and thus representation transfer is practical. It can be seen that training from scratch could perform well for a data-rich regime, $(n_{T_1}, n_{T_2}) = (1000, 1000)$. Here, the performance of the learned model after Phase 1 degrades

[4]The parameter σ_T^2 indirectly enables us simulate the value of ϵ in Assumption 1, with larger values of σ_T^2 implying that $\boldsymbol{\theta}_T^*$ lies farther away from the subspace spanned by the columns of $\mathbf{V}^*$. A higher σ_T^2 values thus yields a small value for the in-out representation mixture ratio.

[5]The values in the 'In-Out Representation Mixture Ratio' column in Table I correspond to the ratio of the signal in the subspace spanned by columns of $\mathbf{V}^*$ and the added out of subspace signal (of variance σ_T^2) added to it to generate the true target model $\boldsymbol{\theta}_T^*$. Lower values of this ratio correspond to higher values of σ_T^2.

TABLE I
PERFORMANCE COMPARISON FOR LEARNED MODELS AFTER PHASE 1 (PRE-TRAINING), PHASE 2 (FINE-TUNING) AND LEARNING FROM SCRATCH

Sample Configuration	In-Out Representation Mixture Ratio for θ_T^* (in dB)	Phase 1	Phase 2	Scratch
$n_{T_1} = 100, n_{T_2} = 100$	50	1.01	1.01	10.17
	20	3.21	3.05	10.41
	10	9.97	9.17	10.98
	5	16.13	15.84	12.74
	1	28.01	26.61	15.96
$n_{T_1} = 200, n_{T_2} = 200$	50	1.01	1.00	9.71
	20	2.03	1.88	9.33
	10	5.71	5.14	10.33
	5	10.09	9.10	11.79
	1	15.30	13.62	13.68
$n_{T_1} = 300, n_{T_2} = 300$	50	1.01	1.00	8.13
	20	1.90	1.69	8.35
	10	5.30	4.46	9.07
	5	9.12	7.60	9.95
	1	14.55	12.12	12.13
$n_{T_1} = 1000, n_{T_2} = 1000$	50	1.01	1.01	1.01
	20	1.74	1.01	1.01
	10	4.72	1.09	1.01
	5	7.92	1.12	1.01
	1	12.47	1.21	1.01

with decreasing in-out representation mixture ratio as the source representations become less useful to learn $\boldsymbol{\theta}_T^*$. Meanwhile, performing fine-tuning in addition to utilizing source representations, as in Phase 2, yields much better performance of the overall learned model with relative errors much less than after Phase 1. Thus, fine-tuning on target data (Phase 2) can be essential in addition to leveraging source models directly by pre-training (Phase 1) when the true model $\boldsymbol{\theta}_T^*$ lies farther away from the subspace spanned by the source representations.

VIII. CONCLUSION

In this work, we proposed a method for training linear regression models via representation transfer learning in the limited sample regime, when given access to multiple pre-trained linear models trained on data domains (sources) different form the target of interest. We established excess risk bounds for the learned target model when (i) source representations are used directly to construct a target representation and adapted to the target task, and (ii) when the entire resulting model is fine-tuned in the over-parameterized regime using target task samples. Our bounds showed a gain in target sample complexity compared to the baseline case of learning without access to the pre-trained models, thus demonstrating the benefit of transfer learning for better generalization in the limited sample regime. Our provided numerical results corroborated this fact and showed superior performance of our proposed scheme compared to learning from scratch in data-scare regimes.

As future extensions to this work, it is of interest to see how non-linear activation functions can be introduced in the model to analyze more complicated architectures like Neural Networks (NNs). Analyzing representation transfer learning with multiple NNs and utilizing recently developed results in benign over-fitting for this setting [38] is an interesting next step. In many scenarios of interest, for training the source task models, *unlabeled* data from the target distribution might be available. While there are empirical works utilizing unlabeled samples in the context of semi-supervised adaptation [27], [39], [40], theoretical results on understanding generalization of representation transfer learning methods (with pre-training/fine-tuning) and their sample complexity requirements are missing and would be an interesting direction to pursue.

REFERENCES

[1] Q. S. Y. Liu, T. Chua, and B. Schiele, "Meta-transfer learning for few-shot learning," in *Proc. 2018 IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 403–412.
[2] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[3] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning," in *Proc. ICML Workshop Unsupervised Transfer Learn. JMLR Workshop Conf. Proc.*, 2012, pp. 17–36.
[4] J. Zhao and W.-Q. Zhang, "Improving automatic speech recognition performance for low-resource languages with self-supervised models," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 6, pp. 1227–1241, Oct. 2022.
[5] Y. Jia et al., "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Proc. Annu. Conf. Neural Inf. Process. Syst.* 2018,, vol. 31.
[6] Z. Liu, Y. Lin, and M. Sun, *Representation Learning for Natural Language Processing*. Berlin, Germany: Springer, 2020.
[7] J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman, "Learning bounds for domain adaptation," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2007, vol. 20 .
[8] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation: Learning bounds and algorithms," in *Proc. 22nd Conf. Learn. Theory*, Montreal, QC, Canada,, 2009. [Online]. Available: http://www.cs.mcgill.ca/%7Ecolt2009/papers/003.pdf#page=1
[9] S. S. Du, W. Hu, S. M. Kakade, J. D. Lee, and Q. Lei, "Few-shot learning via learning the representation, provably," in *Proc. 9th Int. Conf. Learn. Representations,* Austria, 2021. [Online]. Available: https://openreview.net/forum?id=pW2Q2xLwIMD
[10] K. Chua, Q. Lei, and J. D. Lee, "How fine-tuning allows for effective meta-learning," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 8871–8884.
[11] N. Tripuraneni, C. Jin, and M. Jordan, "Provable meta-learning of linear representations," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 10434–10443.

[12] A. Maurer, M. Pontil, and B. Romera-Paredes, "The benefit of multitask representation learning," *J. Mach. Learn. Res.*, vol. 17, no. 81, pp. 1–32, 2016.

[13] M. Belkin, D. J. Hsu, and P. Mitra, "Overfitting or perfect fitting? Risk bounds for classification and regression rules that interpolate," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2018, vol. 31.

[14] S. Arora, S. Du, W. Hu, Z. Li, and R. Wang, "Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 322–332.

[15] P. L. Bartlett, P. M. Long, G. Lugosi, and A. Tsigler, "Benign overfitting in linear regression," *Proc. Nat. Acad. Sci.*, vol. 117, no. 48, pp. 30063–30070, 2020.

[16] T. Hastie, A. Montanari, S. Rosset, and R. J. Tibshirani, "Surprises in high-dimensional ridgeless least squares interpolation," *Ann. Statist.*, vol. 50, no. 2, pp. 949–986, 2022.

[17] V. Shah, S. Basu, A. Kyrillidis, and S. Sanghavi, "On generalization of adaptive methods for over-parameterized linear regression," 2020, *arXiv:2011.14066*.

[18] J. Wu, D. Zou, V. Braverman, and Q. Gu, "Direction matters: On the implicit bias of stochastic gradient descent with moderate learning rate," in *Proc. 9th Int. Conf. Learn. Representations, 2021, Virtual Event*, Austria, 2021. [Online]. Available: https://openreview.net/forum?id=3X64RLgzY6O

[19] S. Gunasekar, J. Lee, D. Soudry, and N. Srebro, "Characterizing implicit bias in terms of optimization geometry," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1832–1841.

[20] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1126–1135.

[21] J. Wang et al., "Generalizing to unseen domains: A survey on domain generalization," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 8, pp. 8052–8072, Aug. 2023.

[22] D. McNamara and M.-F. Balcan, "Risk bounds for transferring representations with and without fine-tuning," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2373–2381.

[23] T. Galanti, L. Wolf, and T. Hazan, "A theoretical framework for deep transfer learning," *Inf. Inference: A J. IMA*, vol. 5, no. 2, pp. 159–209, 2016.

[24] H. Zhao, R. T. D. Combes, K. Zhang, and G. Gordon, "On learning invariant representations for domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7523–7532.

[25] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2007, vol. 20.

[26] P. Stojanov, Z. Li, M. Gong, R. Cai, J. Carbonell, and K. Zhang, "Domain adaptation with invariant representation learning: What transformations to learn?," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 24791–24803.

[27] Y. Zhang, T. Liu, M. Long, and M. Jordan, "Bridging theory and algorithm for domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7404–7413.

[28] J. Tian, J. Zhang, W. Li, and D. Xu, "VDM-DA: Virtual domain modeling for source data-free domain adaptation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 6, pp. 3749–3760, Jun. 2022 .

[29] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation with multiple sources," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2008, vol. 21.

[30] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 6028–6039.

[31] S. M. Ahmed, D. S. Raychaudhuri, S. Paul, S. Oymak, and A. K. Roy-Chowdhury, "Unsupervised multi-source domain adaptation without access to source data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10098–10107.

[32] G. Shachaf, A. Brutzkus, and A. Globerson, "A theoretical analysis of fine-tuning with linear teachers," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 15382–15394.

[33] M. Belkin, D. Hsu, and J. Xu, "Two models of double descent for weak features," *SIAM J. Math. Data Sci.*, vol. 2, no. 4, pp. 1167–1180, 2020.

[34] I. Evron, E. Moroshko, R. Ward, N. Srebro, and D. Soudry, "How catastrophic can catastrophic forgetting be in linear regression?," in *Proc. Conf. Learn. Theory*, 2022, pp. 4028–4079.

[35] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "Exploiting shared representations for personalized federated learning," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 2089–2099.

[36] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge Univ. Press, 2004.

[37] V. Koltchinskii and K. Lounici, "Concentration inequalities and moment bounds for sample covariance operators," *Bernoulli*, vol. 23, no. 1, pp. 110–133, 2017.

[38] N. S. Chatterji, P. M. Long, and P. L. Bartlett, "The interplay between implicit bias and benign overfitting in two-layer linear networks," *J. Mach. Learn. Res.*, vol. 23, no. 263, pp. 1–48, 2022.

[39] A. Singh, "CLDA: Contrastive learning for semi-supervised domain adaptation," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2021, pp. 5089–5101.

[40] S. Mishra, K. Saenko, and V. Saligrama, "Surprisingly simple semi-supervised domain adaptation with pretraining and consistency," in *Proc. NeurIPS 2021 Workshop Distrib. Shifts: Connecting Methods Appl.*, 2021.