Bridging Domains with Approximately Shared Features

Ziliang Samuel Zhong* NYU Shanghai

Xiang Pan* New York University

Qi Lei New York University

Abstract

Machine learning models can suffer from performance degradation when applied to new tasks due to distribution shifts. Feature representation learning offers a robust solution to this issue. However, a fundamental challenge remains in devising the optimal strategy for feature selection. Existing literature is somewhat paradoxical: some advocate for learning invariant features from source domains, while others favor more diverse features. For better understanding, we propose a statistical framework that evaluates the utilities of the features (*i.e.,* how differently the features are used in each source task) based on the variance of their correlation to y across different domains. Under our framework, we design and analyze a learning procedure consisting of learning content features (comprising both invariant and approximately shared features) from source tasks and fine-tuning them on the target task. Our theoretical analysis highlights the significance of learning approximately shared features—beyond strictly invariant ones—when distribution shifts occur. Our analysis also yields an improved population risk on target tasks compared to previous results. Inspired by our theory, we introduce ProjectionNet, a practical method to distinguish content features from environmental features via *explicit feature space control*, further consolidating our theoretical findings.

1 INTRODUCTION

Machine learning models are often sensitive to *distribution shifts*, *i.e.,* they do not adapt well to the datasets that are related but distribute differently from the training data. Even a small distribution shift can cause substantial performance degradation (Koh et al., 2021). To make machine learning models robust to distributional shifts, feature representation learning has become a pivotal strategy, enabling the rapid transfer of knowledge from source domains to the target domain (Quinonero-Candela et al., 2008; Saenko et al., 2010). Despite its empirical success (Wang and Deng, 2018; Daumé III, 2007), a systematic theoretical understanding is still lacking.

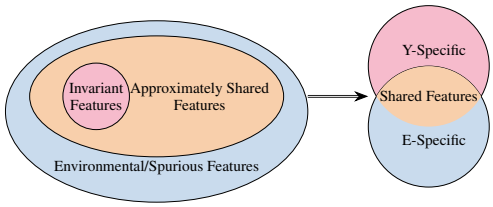
A fundamental challenge in feature representation learning is devising the optimal strategy for feature selection on source domains. Most previous theoretical works pivot around specific transfer learning/domain adaptation algorithms and select features based on causality, which requires strong assumptions on the statistical relationship between the source and target distributions. These assumptions include but are not limited to (1) the source domain covering the target domain (Shimodaira, 2000; Heckman, 1979; Cortes et al., 2010; Zadrozny, 2004); (2) the source and target domain having overlapping subgroups (Wei et al., 2020; Cai et al., 2021); (3) content features being exactly shared across domains (Arjovsky et al., 2019; Ben-David et al., 2010; Ahuja et al., 2020). Besides the limitation of learning causal features, existing literature also has paradoxical opinions on selecting non-causal features. For instance, some works suggest that we should learn invariant features, (Ahuja et al., 2020; Ben-David et al., 2010; Ajakan et al., 2014) while others suggest that we should learn diverse (even possibly spurious) features (Shen et al., 2022b; Wortsman et al., 2022) to get a more distributionally robust model.

To resolve the limitations and paradoxes in previous works, we will answer the following questions: *What features are robust to the natural distributional shifts, and how can we learn them purely from the observational data without causal information?*

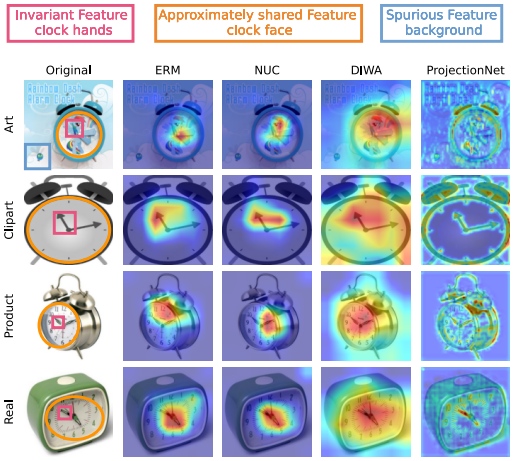
Our contribution. In this work, we propose a feature representation learning framework that distinguishes content (invariant + approximately shared) vs. environmental features solely from observational data (based on how differently the features are utilized across tasks); see illustration diagram in Figure 1a. Under our framework, we design and analyze a learning procedure consisting of (1) learning the

*Equal contribution. Correspondence to: Qi Lei (ql518@nyu.edu).

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.



(a) Diagram for content (invariant+approximately shared) and environmental features, mathematically defined in Equation 4. The left figure shows the invariance of the feature, from the least invariant feature (environmental) to the most invariant feature (invariant feature). The right part shows the feature relationship with the environment (E) or the label (Y), the feature both predictive to the Y and E is the y-e shared feature. Our work indicates that in addition to invariant features, we should utilize approximately shared features to efficiently transfer the knowledge from the source to the target domain. The practical way to learn the **approximately shared features** is learning features that are correlated to both y and the environment e, which is the y-e **shared features**. See detailed illustration following equation 4.



(b) **Feature Space Visualization**: We show the GradCAM++ of the OfficeHome dataset with the feature space of source pre-trained models. We can see that the feature space of proposed ProjectionNet is more semantically meaningful than existing works: ERM and NUC focus more locally and the DiWA(Rame et al., 2022) feature is more globally distributed.

content features via meta-representation learning on source tasks, and (2) fine-tuning the learned representation on the target tasks. When distribution shift occurs, our theoretical analysis yields a smaller and more interpretable population risk bound by removing the irreducible term in the previous works such as (Tripuraneni et al., 2021) and Du et al. (2021). It necessitates learning both invariant and approximately shared features instead of only invariant features for a quick adaption to the target domain. Inspired by our theory, we proposed ProjectionNet, a training method to isolate the content (invariant+approximately shared) from environmental features. As illustrated in Figure 1b, the feature space of ProjectionNet is more semantically meaningful than existing methods. Our findings effectively bridge the gap between different opinions in previous works mentioned above.

1.1 RELATED WORKS

We discuss relevant work in (multi-source) domain adaptation categorized by the types of distributional shifts that cause performance degradation.

Selection bias describes the phenomenon of data collected in separate routines to present as if drawn from different distributions. It can be introduced by the selection of individuals, groups, or data for analysis so that proper randomization is not achieved. For selection bias on individual samples, it can represent general covariate shift, and prior works seek to mitigate such bias via practices like importance reweighting (Shimodaira, 2000; Heckman, 1979; Cortes et al., 2010; Zadrozny, 2004; Sun et al., 2011), hard sample reweighting (Liu et al., 2021a; Nam et al., 2020) or distribution matching/discrepancy minimization (Cortes et al., 2015; Ben-David et al., 2010; Berthelot et al., 2021), and domain-adversarial algorithms (Ajakan et al., 2014; Ganin et al., 2016; Long et al., 2018) between source and target domains in their feature representation space. For selection bias on groups or subpopulation, namely subpopulation shift or dataset imbalance, people have investigated label propagation (Cai et al., 2021; Berthelot et al., 2021) or other consistency regularization (Miyato et al., 2018; Xie et al., 2020a; Yang et al., 2023) that migrate the predictions from source to target. Some more studies go beyond semi-supervised learning setting to contrastive representation learning (HaoChen et al., 2022; Shen et al., 2022a; Liu et al., 2021b) or self-training (Wei et al., 2020; Kumar et al., 2020). Under small covariate shift, learning algorithms have been investigated to achieve near minimax risks (Lei et al., 2021; Pathak et al., 2022).

Spurious correlation corresponds to the dependence between features and labels that is not fundamental or not consistent across domains. It poses a significant challenge in the deployment of machine learning models, as they can lead to reliance on irrelevant or environmental features and poor generalization. Specifically, with the existence of spurious or environmental features accompanying the invariant/content features, studies show fine-tuning can distort pretraining features (Kumar et al., 2022b), hurting out-of-distribution generalization. The existence of spurious features also brings about a trade-off between in-distribution and out-of-distribution performances. Prior work uses model ensemble (Kumar et al., 2022a) or model soups (Wortsman et al., 2022) to learn rich features and balance the effect of spurious features and to trade-off the above effect. More practices include introducing auxiliary information through human annotation (Srivastava et al., 2020) or from multiple sources (Xie et al., 2020b), through invariant feature representation learning (Arjovsky et al., 2019; Chen et al., 2022; Ahuja et al., 2020), through self-training (Chen et al., 2020) or overparametrization (Sagawa et al., 2020), feature manipulation (Shen et al., 2022b), adding regularizations (Park and Lee, 2021; Shi et al., 2023), or through

causal approaches (Lu et al., 2021).

Finally, we discuss some relevant results in **meta-representation learning** that bear some resemblances but are fundamentally distinct in the presenting purposes. Prior works (Du et al., 2021; Chua et al., 2021; Tripuraneni et al., 2020, 2021) focused on when and how one can identify the ground-truth representation from multiple training tasks. All of their results (as well as Chen and Marchand (2023) that is more closely related to our work) suffer from an irreducible term that arises from source representation error. In addition to that, we are more interested in how to handle spurious correlation and how one can adapt to the new task sample efficiently.

1.2 NOTATIONS

Let $[n] = \{1, \ldots, n\}$. We use $\|\cdot\|$ to denote the ℓ_2 norm of a vector or the spectral norm of a matrix. We denote the Frobenius norm of a matrix as $\|\cdot\|_F$. Let $\langle \cdot, \cdot \rangle$ be the Euclidean inner product between two vectors or matrices. The $d \times d$ identity matrix is denoted as I_d. For a vector $\boldsymbol{v} \in \mathbb{R}^m$ v_k is the k-th entry and $\boldsymbol{v}_{[k:\ell]}$ is the vector formed by $\boldsymbol{v}$'s k-th to ℓ-th entries, $1 \le k < \ell \le m$. For a matrix $A \in \mathbb{R}^{m \times n}$, $m \ge n$, A_k is the k-th column and $A_{[k:\ell]}$ is the matrix formed by A's k-th to ℓ-th columns, $1 \le k < \ell \le n$. Denote $\mathcal{P}_A = A(A^\top A)^\dagger A^\top \in \mathbb{R}^{m \times m}$, which is the projection matrix onto $\mathrm{span}(A) = \{A\boldsymbol{v} \mid \boldsymbol{v} \in \mathbb{R}^n\}$. Here $\dagger$ stands for the Moore-Penrose pseudo-inverse. We define $\mathcal{P}_A^\perp = I_m - \mathcal{P}_A$ which is the projection matrix onto the orthogonal complement of $\mathrm{span}(A)$. The k-th smallest eigenvalue and k-th smallest singular value of A are denoted by $\lambda_k(A)$ and $\sigma_k(A)$ respectively. We denote $\mathcal{O}(n)$ as the $n-$dimensional orthogonal group. We use $\gtrsim, \lesssim, \asymp$ to denote greater than, less than, equal to up to some constant. Our use of O, Ω, Θ is standard.

2 METHODOLOGY

This paper develops a feature selection strategy robust to distribution shifts when adapting machine learning models from multiple source domains to a target domain with limited labeled data. We have E source environments indexed by $1, \ldots, E$ and a target environment indexed by $E+1$. For $e \in [E]$, each source environment provides n_1 samples, $\{(\boldsymbol{x}_i^e, y_i^e)\}_{i=1}^{n_1}$, and the target environment provides n_2 samples, $\{(\boldsymbol{x}_i^e, y_i^e)\}_{i=1}^{n_2}$. Each $(\boldsymbol{x}_i^e, y_i^e)$ is an i.i.d. sample from an environment-specific distribution μ_e over the data space $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} \subseteq \mathbb{R}$. We consider the *few-shot model shift* setting, where $n_2 \ll n_1$ and distributions μ_e vary across $e \in [E+1]$.

2.1 DATA GENERATION PROCESS

As illustrated in Figure 2, we categorize features into two types: content (approximately shared + invariant) and environmental (spurious) features, distinguished by their variance of correlation to y across environments. To theoretically model the features of these two types, we introduce the following data generation process.

Let $\phi_D : \mathcal{X} \to \mathcal{Z}$ be a representation function from the input space $\mathcal{X} \subseteq \mathbb{R}^d$ to a latent space $\mathcal{Z} \subseteq \mathbb{R}^D$, with $\boldsymbol{\Phi}_D$ as its function class. Here, d is the input dimension and D is the representation dimension. For $k < D$, we define low-dimensional representation function classes induced by $\boldsymbol{\Phi}_D$:

$$\begin{aligned} \boldsymbol{\Phi}_k &:= \{f : f(\boldsymbol{x}) = \phi_{D[1:k]}(\boldsymbol{x}), \phi_D \in \boldsymbol{\Phi}_D\}, \\ \boldsymbol{\Phi}_{D-k} &:= \{f : f(\boldsymbol{x}) = \phi_{D[k+1:D]}(\boldsymbol{x}), \phi_D \in \boldsymbol{\Phi}_D\}, \end{aligned} \tag{1}$$

where $\phi_{D[1:k]}(\boldsymbol{x})$ and $\phi_{D[k+1:D]}(\boldsymbol{x})$ split $\phi_D(\boldsymbol{x})$ as:

$$\phi_D(\boldsymbol{x}) = [\phi_{D[1:k]}(\boldsymbol{x})^\top, \phi_{D[k+1:D]}(\boldsymbol{x})^\top]^\top.$$

We apply different linear predictors on the shared representation ϕ_D^* to model input-output relations in each environment. For $e \in [E+1]$, let $\phi_D^* \in \boldsymbol{\Phi}_D$ be the ground-truth representation and $\boldsymbol{\theta}^{*1}, \ldots, \boldsymbol{\theta}^{*E+1} \in \mathbb{R}^D$ characterize feature correlations with y. The data distribution $(\boldsymbol{x}, y) \sim \mu_e$ is given by:

$$y = \langle \phi_D^*(\boldsymbol{x}), \boldsymbol{\theta}^{*e} \rangle + z, \quad \boldsymbol{x} \sim p_e, \quad z \sim \mathcal{N}(0, \sigma^2), \tag{2}$$

where $\boldsymbol{x}$ and z are independent, with p_e as the environment-specific input distribution. Environment-specific parameters follow a multivariate Gaussian meta-distribution:

$$\boldsymbol{\theta}^{*e} \overset{\text{i.i.d}}{\sim} \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\theta}^* \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \Lambda_{11} & 0 \\ 0 & \Lambda_{22} \end{bmatrix} \right), \tag{3}$$

where $\boldsymbol{\theta}^* \in \mathbb{R}^k$, and $\Lambda_{11} \in \mathbb{R}^{k \times k}$, $\Lambda_{22} \in \mathbb{R}^{(D-k) \times (D-k)}$ are diagonal matrices with $\|\Lambda_{11}\| < \|\Lambda_{22}\|$. Equation equation 2 can then be split as:

$$y = \langle \phi^*_{D[1:k]}(\boldsymbol{x}), \boldsymbol{\theta}^{*e}_{[1:k]} \rangle + \langle \phi^*_{D[k+1:D]}(\boldsymbol{x}), \boldsymbol{\theta}^{*e}_{[k+1:D]} \rangle + z. \tag{4}$$

For each training sample, we consider the triplet $(\boldsymbol{x}, y, e)$, where e represents the environment to which $(\boldsymbol{x}, y)$ belongs. The first k entries of $\phi_D^*(\boldsymbol{x})$ capture the content features that correlate with both y and e, due to the parameters $\boldsymbol{\theta}^{*e}_{[1:k]}$ being centered around $\boldsymbol{\theta}^*$ with variance Λ_{11}. Specifically, invariant features correspond to smaller values in Λ_{11}, indicating stability across changes in e, while approximately shared features correspond to larger values, reflecting their fluctuating correlation with y as e changes. In contrast, the remaining $D-k$ entries are environmental features, which are uncorrelated with y on average (as $\boldsymbol{\theta}^{*e}$ is zero-mean in these entries) but do correlate with e. Their correlation with y varies significantly across environments, as characterized by $\|\Lambda_{22}\| > \|\Lambda_{11}\|$.

The assumption of Gaussian distribution is for simplicity. The fundamental requirement is for the meta distribution to be light-tail.

In the clock image classification task, each row in the first column of Figure 1b represents a source environment. The source datasets contain labeled clock images with different joint distributions μ^e (i.e., style variation). In this context, the clock hands are invariant features, as they consistently correlate with the clock class across environments. The clock face, however, is an approximately shared feature, as its classification effectiveness depends on the surrounding environment. For example, when the background color closely matches the clock face, identifying the clock becomes challenging based on the face alone. Finally, the background is an environmental feature, as it lacks intrinsic correlation with the clock class and varies significantly across environments.

Importance of Approximately Shared Features. Including approximately shared features in the low-dimensional representation addresses certain paradoxes in prior work. While it may seem intuitive to prioritize learning invariant features, as advocated by Arjovsky et al. (2019), model soups (Wortsman et al., 2022) and model ensembles show that incorporating richer features leads to greater robustness against domain shifts. This robustness arises because the influence of environmental features is mitigated through model averaging, consistent with the zero-mean component in equation 3.

Comparison to Prior Work. Our data generation process is more reflective of real-world data and offers a reasonable alternative to existing meta-learning theory. For instance, some works (Arjovsky et al., 2019; Ahuja et al., 2020) assume that y is generated solely from strictly invariant features, ignoring environmental influences. They model this as $y = \langle \phi_k^*(\boldsymbol{x}), \boldsymbol{\theta}^e \rangle + z$ for $e \in [E+1]$, where $\phi^k \in \boldsymbol{\Phi}_k$ and $k < d$. However, in real-world data, there is often no clear-cut distinction between invariant and environmental features, making it impractical to simply exclude features that seem uncorrelated with y. Furthermore, works on representation learning (Du et al., 2021; Tripuraneni et al., 2020) make no distinction to the utilities of features (variance of correlation); they learn all features correlated to y and have a different focus from us. Their risk bound also has an irreducible term from source tasks without investigating how the representation error can be eliminated through fine-tuning.

2.2 THE META-REPRESENTATION LEARNING ALGORITHM

Given the data generation process described above, a key question is how to learn content features from source environments using only observational data. In theory, one could learn the low-dimensional representation $\phi_{D[1:k]} \in \boldsymbol{\Phi}_k$ from the source environments and use these features for fine-tuning in the target environment. However, this theoretical approach is impractical due to the complexity of optimizing over $\boldsymbol{\Phi}_k$ when the representation function is nonlinear. Therefore, the algorithm presented here mainly serves as a theoretical basis for the analysis in Section 3, showing that accurately capturing content features from source environments can significantly accelerate adaptation to the target task. In Section 2.3, we will present a more practical approach for learning these content features.

Before describing the algorithm, we introduce some auxiliary notations. For each source environment $e \in [E]$, let $X^e \in \mathbb{R}^{n_1 \times d}$ be the data matrix and $\boldsymbol{y}^e \in \mathbb{R}^{n_1}$ the response vector, constructed as:

$$X^e = \left[\boldsymbol{x}_1^e, \ldots, \boldsymbol{x}_{n_1}^e\right]^\top, \quad \boldsymbol{y}^e = \left[y_1^e, \ldots, y_{n_1}^e\right]^\top.$$

Similarly, for the target environment, $X^{E+1} \in \mathbb{R}^{n_2 \times d}$ and $\boldsymbol{y}^{E+1} \in \mathbb{R}^{n_2}$. Given a representation function $\phi_D \in \boldsymbol{\Phi}_D$, we extend the notation so that ϕ_D applies to all samples in a data matrix simultaneously:

$$\phi_D(X^e) = \left[\phi_D(\boldsymbol{x}_1^e), \ldots, \phi_D(\boldsymbol{x}_{n_1}^e)\right]^\top \in \mathbb{R}^{n_1 \times D}.$$

The same applies to $\phi_D(X^{E+1}) \in \mathbb{R}^{n_2 \times D}$ for the target environment.

Our meta-representation learning algorithm proceeds as follows. **Source Pretraining:** For each source environment $e \in [E]$, we set the prediction function as $\boldsymbol{x} \mapsto \langle \phi_k(\boldsymbol{x}), \boldsymbol{\theta}^e \rangle$ ($\boldsymbol{\theta}^e \in \mathbb{R}^k$) and learn the content features $\widehat{\phi}_k^E \in \boldsymbol{\Phi}_k$ by optimizing:

$$\begin{aligned} &\left(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^1, \ldots, \widehat{\boldsymbol{\theta}}^E\right) \\ &= \underset{\phi_k \in \boldsymbol{\Phi}_k, \boldsymbol{\theta}^e \in \mathbb{R}^k}{\operatorname{argmin}} \frac{1}{2n_1 E} \sum_{e=1}^{E} \|\boldsymbol{y}^e - \phi_k(X^e)\boldsymbol{\theta}^e\|^2. \end{aligned} \tag{5}$$

Target Fine-tuning: For the target environment, we solve:

$$\begin{aligned} &\left(\widehat{\phi}_D^{E+1}, \widehat{\boldsymbol{\theta}}^{E+1}\right) \\ &= \underset{\phi_D \in \boldsymbol{\Phi}_D, \boldsymbol{\theta}^{E+1} \in \mathbb{R}^D}{\operatorname{argmin}} \frac{1}{2n_2} \left\|\boldsymbol{y}^{E+1} - \phi_D(X^{E+1})\boldsymbol{\theta}^{E+1}\right\|^2 \\ &+ \frac{\lambda_1}{2n_2} \left\|\mathcal{P}^{\perp}_{\widehat{\phi}_k^E(X^{E+1})} \phi_D(X^{E+1})\boldsymbol{\theta}^{E+1}\right\|^2 + \frac{\lambda_2}{2} \left\|\boldsymbol{\theta}^{E+1}\right\|^2. \end{aligned} \tag{6}$$

In Equation 6, the first term is the empirical risk minimization on the target environment with two regularizing terms. The first one penalizes the prediction on the directions *perpendicular* to the content features learned from source environments, which discourages the learning dynamics from capturing the environmental (spurious) features since we assume they are uncorrelated to y on average. The second one is the standard ℓ_2 regularization.

2.3 ISOLATING CONTENT FEATURES IN PRACTICE: THE PROJECTIONNET

The optimizations in equation 5 and equation 6 are intractable, as searching through the function class to find the optimal representation is generally expensive. To address this, we propose **ProjectionNet**, an efficient training method that naturally distinguishes between invariant features (ϕ_y), approximately shared features (ϕ_s), and environment-specific features (ϕ_e).

Source Pretraining: For each training sample we consider the triplet $(\boldsymbol{x}, y, e)$. Let $\phi_{\boldsymbol{\theta}_{\text{rep}}}$ with trainable weights $\boldsymbol{\theta}_{\text{rep}}$ be some feature map (such as ResNet). Besides the model weights, we would like to train three projections $\mathcal{P}_y$, $\mathcal{P}_s$, $\mathcal{P}_e$ such that $\phi_y = \phi_{\boldsymbol{\theta}_{\text{rep}}}(\boldsymbol{x})\mathcal{P}_y$, $\phi_{y,e} = \phi_{\boldsymbol{\theta}_{\text{rep}}}(\boldsymbol{x})\mathcal{P}_s$, $\phi_e = \phi_{\boldsymbol{\theta}_{\text{rep}}}(\boldsymbol{x})\mathcal{P}_e$. The algorithm is summarized in equation 7.

$$\min_{\boldsymbol{\theta}_{\text{rep}}, \boldsymbol{\theta}_y, \boldsymbol{\theta}_e, \mathcal{P}_e, \mathcal{P}_y, \mathcal{P}_s} \mathcal{L}_y + \mathcal{L}_e + \lambda_1 \cdot \mathcal{L}_{\text{disentangle}} + \lambda_2 \cdot \mathcal{L}_{\text{reg}} \tag{7}$$

where

$$\mathcal{L}_y = \frac{1}{n_1} \sum_{i=1}^{n_1} \mathcal{L}\left(\begin{bmatrix} \phi_{\boldsymbol{\theta}_{\text{rep}}}(\boldsymbol{x}_i)\mathcal{P}_y \\ \phi_{\boldsymbol{\theta}_{\text{rep}}}(\boldsymbol{x}_i)\mathcal{P}_s \end{bmatrix} \boldsymbol{\theta}_y, y_i \right),$$

$$\mathcal{L}_e = \frac{1}{n_1} \sum_{i=1}^{n_1} \text{CE}\left(\begin{bmatrix} \phi_{\boldsymbol{\theta}_{\text{rep}}}(\boldsymbol{x}_i)\mathcal{P}_s \\ \phi_{\boldsymbol{\theta}_{\text{rep}}}(\boldsymbol{x}_i)\mathcal{P}_e \end{bmatrix} \boldsymbol{\theta}_e, e_i \right),$$

$$\mathcal{L}_{\text{disentangle}} = \|\mathcal{P}_y^\top \mathcal{P}_e\|_F^2 + \|\mathcal{P}_y^\top \mathcal{P}_s\|_F^2 + \|\mathcal{P}_e^\top \mathcal{P}_s\|_F^2,$$

$$\mathcal{L}_{\text{reg}} = \left\|\mathcal{P}_e^\top \mathcal{P}_e - I\right\|_F^2 + \left\|\mathcal{P}_y^\top \mathcal{P}_y - I\right\|_F^2 + \left\|\mathcal{P}_s^\top \mathcal{P}_s - I\right\|_F^2,$$

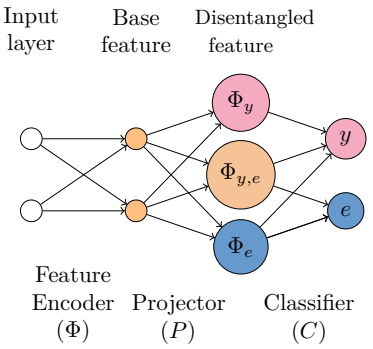


Figure 2: ProjectionNet: We disentangle the base representation ϕ into target-specific feature ϕ_y, approximately shared feature ϕ_s, and environment-specific feature ϕ_e via orthogonal projections $\mathcal{P}_y$, $\mathcal{P}_s$, $\mathcal{P}_e$. The $[\phi_y, \phi_s]$ are used in the target label prediction. $[\phi_s, \phi_e]$ are used in the environment label prediction.

where $\text{CE}(\cdot, \cdot)$ is the standard Cross-Entropy loss. This method is based on the assumption that the approximately shared features are correlated to both y and e. We jointly minimize the four loss functions above. $\mathcal{L}_y$ is the standard empirical risk minimization, it extracts the features that are correlated to the response y, *i.e.,* invariant features and approximately shared features. $\mathcal{L}_e$ extracts the features that determine the environment of a training sample $(\boldsymbol{x}, y)$, *i.e.,* approximately shared features and environmental features. $\mathcal{L}_{\text{disentangle}}$ and $\mathcal{L}_{\text{reg}}$ ensure the learned $\mathcal{P}$'s are approximately orthogonal projections that are mutually perpendicular to each other. The method is visualized in Figure 2. **Target Finetuning:** We can flexibly choose to utilize ϕ_y or $[\phi_y, \phi_{y,e}]$ as the initialization for the target finetuning phase, which depends on whether we want to use invariant features or content features.

One may wonder how closely ProjectionNet approximates the theoretical algorithm. To explore this, we compare the performance of ProjectionNet with its theoretical counterpart within the linear representation function class, where optimizations in equations equation 5 and equation 6 are solvable in polynomial time (see Section B for details). As shown in Figure 5, when the number of few-shot samples is moderate, the feature spaces learned by ProjectionNet (red dashed line) and the theoretical algorithm (purple line) achieve comparable MSE in the target domain during finetuning.

Comparison to Nuclear Norm Regularization: Shi et al. (2023) propose nuclear norm regularization to learn shared features from source environments, which is closely related to our approach. They control the complexity of the source representation $\phi_{\boldsymbol{\theta}_{\text{rep}}}$ by solving

$$\min_{\boldsymbol{\theta}_{\text{rep}}, \boldsymbol{\theta}} \frac{1}{E} \sum_{e=1}^{E} \mathcal{L}(\langle \phi_{\boldsymbol{\theta}_{\text{rep}}}(X^e), \boldsymbol{\theta} \rangle, \boldsymbol{y}^e) + \lambda_{\text{nuc}} \cdot \left\|\phi_{\boldsymbol{\theta}_{\text{rep}}}(X^e)\right\|_*, \tag{8}$$

where the nuclear norm regularization enforces a low-rank structure on the representation, promoting shared features across source environments. The learned representation $\phi_{\boldsymbol{\theta}_{\text{rep}}}$ is then used for fine-tuning on the target environment. While this method shares our objective of learning shared features, it has a drawback: it is sensitive to the regularization strength λ, requiring prior knowledge to select an appropriate value. In contrast, ProjectionNet is more flexible and less sensitive to the choice of regularization strength. A detailed comparison is provided in Table 1.

3 THEORETICAL ANALYSIS

In this section, we present the statistical analysis of the learning framework introduced in Section 2.2. We focus on the model shift setting with homogeneous input distributions, where for $e \in [E+1]$, $p_1 = \cdots = p_{E+1} = p$. Despite this simplification, the joint distributions $(\boldsymbol{x}^e, y^e) \sim \mu^e$ still differ across environments, as each environment e has its own parameter $\boldsymbol{\theta}^{*e}$. This setup allows for a clearer theoretical presentation while preserving the generality of our framework. The primary requirement for the input distribution is that the representation covariance (defined later) of the function class $\boldsymbol{\Phi}_D$ under each p^e is bounded by a universal constant. In cases where the representation function is linear, this can be further reduced to moment boundedness conditions (see Section B).

Table 1: Domain Generalization Results: We use the source pretrained model to directly test the target domain to evaluate the domain generalization ability of the pretrained model. NUC-0.1 and NUC-0.01 stand for different regularization strength λ_{NUC} in Equation 8. DIWA (M=20) means 20 models are used to get the ensembling model. ProjectionNet (PN) can achieve similar or better results than the baseline methods.

	OfficeHome					PACS					TerrainCognita					VLCS				
Method	A	C	P	R	**Mean**	A	C	P	S	**Mean**	L100	L38	L43	L46	**Mean**	C	L	S	V	**Mean**
ERM	69.14	52.40	77.93	82.11	70.39$_{\pm 1.82}$	82.44	79.06	96.41	82.70	85.15$_{\pm 1.59}$	51.90	44.15	56.93	38.61	47.89$_{\pm 1.33}$	97.18	65.03	70.98	81.36	78.64$_{\pm 3.30}$
DANN (Ganin et al., 2016)	60.91	2.97	65.99	74.08	50.99$_{\pm 4.75}$	14.63	10.26	67.07	21.12	28.27$_{\pm 3.84}$	8.90	7.70	20.40	28.06	16.27$_{\pm 1.41}$	94.37	36.36	40.06	44.38	53.79$_{\pm 3.99}$
DIWA$_{\text{(M=20)}}$ (Rame et al., 2022)	69.14	57.67	78.38	79.82	71.25$_{\pm 1.49}$	90.34	81.54	98.80	82.44	**88.28**$_{\pm 1.19}$	57.76	45.17	59.95	43.54	51.61$_{\pm 1.23}$	97.18	60.14	73.50	80.47	77.82$_{\pm 2.25}$
NUC-0.01 (Shi et al., 2023)	69.96	54.46	80.18	81.19	71.45$_{\pm 1.30}$	82.44	75.21	98.20	78.88	83.68$_{\pm 0.89}$	20.25	41.38	55.16	21.77	34.64$_{\pm 1.54}$	31.69	66.92	76.52	13.91	47.26$_{\pm 1.99}$
NUC-0.1 (Shi et al., 2023)	72.84	54.23	79.95	81.88	72.23$_{\pm 1.82}$	81.95	81.20	98.80	79.90	85.46$_{\pm 0.87}$	61.81	49.69	61.21	21.77	48.62$_{\pm 1.54}$	97.89	68.42	72.87	80.77	**79.99**$_{\pm 1.98}$
PN-Y	72.43	49.20	80.63	84.17	71.61$_{\pm 0.72}$	77.07	81.28	96.41	71.50	81.57$_{\pm 1.60}$	55.61	52.77	53.40	47.62	**52.36**$_{\pm 0.40}$	97.89	55.94	73.82	81.66	77.33$_{\pm 1.98}$
PN-YS	72.84	50.57	81.08	84.86	**72.34**$_{\pm 0.78}$	78.05	81.62	97.01	71.50	83.68$_{\pm 1.61}$	56.12	52.77	52.90	47.28	52.27$_{\pm 0.41}$	98.59	55.94	73.82	83.43	77.95$_{\pm 1.98}$

We will use the covariance between two representations to quantify the distance between representation functions. Our notion of representation covariance generalizes from Du et al. (2021) to accommodate different function classes. Detailed properties of representation covariance are provided in Section A.

Definition 3.1 (Covariance between representations) *Let p be a distribution over $\mathbb{R}^d$ and $\mathbf{\Phi}_D$ be some function class with $\mathbf{\Phi}_k/\mathbf{\Phi}_{D-k}$ defined in equation 1. For two representation functions $\phi \in \mathbf{\Phi}_{d_1}$, $\phi' \in \mathbf{\Phi}_{d_2}$, $d_1, d_2 \in \{D, k, D-k\}$, we define the covariance between ϕ and ϕ' with respect to p as $\Sigma_p(\phi, \phi') = \mathbb{E}_{\boldsymbol{x}\sim p}[\phi(\boldsymbol{x})\phi'(\boldsymbol{x})^\top] \in \mathbb{R}^{d_1 \times d_2}$, where d_1 and d_2 are the output dimension of ϕ and ϕ'.We also define the symmetric covariance as*

$$S_p(\phi, \phi') = \begin{bmatrix} \Sigma_p(\phi, \phi) & \Sigma_p(\phi, \phi') \\ \Sigma_p(\phi', \phi) & \Sigma_p(\phi', \phi') \end{bmatrix} \in \mathbb{R}^{(d_1+d_2)\times(d_1+d_2)}.$$

If $\phi := \phi'$, we abbreviate $\Sigma_p(\phi, \phi) := \Sigma_p(\phi)$, $S_p(\phi, \phi) = S_p(\phi)$.

We make Assumption A.1 and A.2 on the ground-truth representation function class $\mathbf{\Phi}_D$ and the input distribution p, which ensure the point-wise and uniform concentration of empirical covariance $S_{\widehat{p}}$ to its population counterpart S_p. In our main theorem, we assume n_1 is large enough to guarantee uniform concentration and n_2 is large enough to guarantee point-wise concentration (still $n_1 \gtrsim n_2$). The following assumption guarantees $\langle \phi^*_{D[1:k]}(\boldsymbol{x}), \boldsymbol{\theta}^{*e}_{[1:k]} \rangle$ dominants in equation 4, that is, the contribution of content features to y is stronger than that of environmental features.

Assumption 3.2 (Dominance of content features) *For some constant $\gamma > 0$, it holds that $\gamma \geq \frac{\left\|\Sigma_p(\phi^*_{D[1:k]})\right\|}{\left\|\Sigma_p(\phi^*_{D[k+1:D]})\right\|} \gtrsim \frac{\mathrm{Tr}(\Lambda_{22})}{\|\boldsymbol{\theta}^*\|^2 + \mathrm{Tr}(\Lambda_{11})}$.*

Assumption 3.3 (Diverse source tasks) *Let $\Theta^* = [\boldsymbol{\theta}^{*1}, \ldots, \boldsymbol{\theta}^{*E}]$. The smallest singular value of Θ^* satisfies $\sigma^2_{\min}(\Theta^*) \gtrsim \frac{E}{\|\boldsymbol{\theta}^*\|^2 + \mathrm{Tr}(\Lambda_{11})}$.*

This assumption requires source tasks to utilize all directions of the representation function, or otherwise the weakest direction will be hard to learn. We present the short versions of our main theorems below. The full versions can be found in Theorem A.3 and A.4.

To evaluate the performance of a predictor $\boldsymbol{x} \mapsto \langle \phi(\boldsymbol{x}), \boldsymbol{\theta} \rangle$ on unseen data $(\boldsymbol{x}, y) \sim \mu^e$ for $e \in [E+1]$, we examine the excess risk, $\mathrm{ER}_e(\phi, \boldsymbol{\theta}) = \frac{1}{2}\mathbb{E}_{\boldsymbol{x}\sim p^e}\left(\langle \phi^*_D(\boldsymbol{x}), \boldsymbol{\theta}^{*e} \rangle - \langle \phi(\boldsymbol{x}), \boldsymbol{\theta} \rangle\right)^2$, and its expectation over the meta-distribution, $\mathbb{E}_{\boldsymbol{\theta}^{*e}}[\mathrm{ER}_e(\phi, \boldsymbol{\theta})]$. For the target environment, we denote ER_{E+1} simply as ER.

Theorem 3.4 (Source guarantee, short version) *Under Assumption A.2, 3.2, and 3.3, if n_1 is large enough, the average excess risk across the source environments with high probability satisfies $\overline{\mathrm{ER}}(\widehat{\phi}^E_k, \widehat{\boldsymbol{\theta}}^1, \ldots, \widehat{\boldsymbol{\theta}}^E) := \frac{1}{E}\sum_{e=1}^E \mathrm{ER}_e(\widehat{\phi}^E_k, \widehat{\boldsymbol{\theta}}^e) \lesssim \underbrace{\sigma\sqrt{\frac{\mathcal{C}_{cont}}{n_1 E}} + \mathcal{C}_{env}}_{:=RE}$ where $\mathcal{C}_{cont} := \left\|\Sigma_p(\phi^*_{D[1:k]})\right\|\left(\|\boldsymbol{\theta}^*\|^2 + \mathrm{Tr}(\Lambda_{11})\right)$ measures the complexity of content features and $\mathcal{C}_{env} := \left\|\Sigma_p(\phi^{*e}_{D[k+1:D]})\right\|\mathrm{Tr}(\Lambda_{22})$ measures the complexity of environmental features.*

In this result, the first term of RE reflects that content features are learned using all $n_1 E$ samples, while the second term represents a non-vanishing error due to the exclusion of environmental features. Although this result is vacuous on its own, it provides a foundation in the analysis of the target environment.

Theorem 3.5 (Target guarantee, short version) *Under Assumption A.1, A.2, 3.2, and 3.3, we further assume that n_1 and n_2 are large enough but still $n_2 \lesssim n_1$. Under proper choice of λ_1 and λ_2, the excess risk of the learned predictor $\boldsymbol{x} \mapsto \langle \widehat{\phi}^{E+1}_D(\boldsymbol{x}), \widehat{\boldsymbol{\theta}}^{E+1} \rangle$ in equation 6 on the target domain with high probability satisfies $\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\mathrm{ER}(\widehat{\phi}^{E+1}_D, \widehat{\boldsymbol{\theta}}^{E+1})\right] \lesssim \underbrace{\sigma\sqrt{\frac{\overline{\mathcal{C}'}_{env}}{n_2}} + \sigma\sqrt{\frac{k}{n_2}}}_{\text{adaptation error}} + \underbrace{\sigma\sqrt{\frac{\mathrm{RE}}{n_2}}}_{\text{representation error}}$, where $\overline{\mathcal{C}'}_{env} := \left(\|\boldsymbol{\theta}^*\|^2 + \mathrm{Tr}(\Lambda_{11})\right)\max_{\phi\in\mathbf{\Phi}_{D-k}}\mathrm{Tr}(\Sigma_p(\phi))$ measures the complexity of the environmental representation function class and* RE *is defined in Theorem 3.4.*

In this excess risk, the first two terms are the adaptation error: it roughly is what the result will look like if we fine-tune the model with the perfect $\phi^*_{D[1:k]}$. The third term is the representation error caused by learning $\phi^*_{D[1:k]}$ with finite samples from the source tasks. It can be further reduced by target samples in the fine-tuning phase. Compared to previous works Chua et al. (2021); Du et al. (2021); Tripuraneni et al. (2020, 2021) that presents irreducible representation error (which does not go to zero as $n_2 \to \infty$) plus adaptation/estimation error, we no longer have the irreducible term from source tasks. We refer interested readers to Section B.2 for results for linear representations.

Remark 3.6 (Benefit of approximately shared features) *Using the notation in Section 2.3, the fine-tuned feature map consists of invariant, approximately shared, and environmental features:*

$$\widehat{\phi}_D^{E+1}(\boldsymbol{x}) = [\phi_y, \phi_{y,e}, \phi_e]\,.$$

Assuming the first r entries correspond to invariant features, we have $\Lambda_{11} = Diag(\Lambda_y, \Lambda_{y,e})$, where $\Lambda_y \approx 0$ and $\mathrm{Tr}(\Lambda_{y,e}) \ll \mathrm{Tr}(\Lambda_{22})$. Our theorem illustrates that incorporating approximately shared features enhances sample efficiency when adapting to the target task with a moderate number (n_2) of labeled samples. If only the invariant feature ϕ_y is learned from the source tasks, an irreducible source error $\left\|\Sigma_p(\phi^*_{D[r+1:k]})\right\| \mathrm{Tr}(\Lambda_{y,e})$ is introduced to Theorem 3.4, increasing the representation error in Theorem 3.5. This also results in a change of

$$\begin{aligned}&\left(\|\boldsymbol{\theta}^*\|^2 + \mathrm{Tr}(\Lambda_{11})\right) \cdot \\ &\left(\max_{\phi\in\boldsymbol{\Phi}_{D-r}} \mathrm{Tr}(\Sigma_p(\phi)) - \max_{\phi\in\boldsymbol{\Phi}_{D-k}} \mathrm{Tr}(\Sigma_p(\phi))\right) \\ &- \left(\left\|\boldsymbol{\theta}^*_{[r+1:k]}\right\|^2 + \mathrm{Tr}(\Lambda_{y,e})\right) \max_{\phi\in\boldsymbol{\Phi}_{D-r}} \mathrm{Tr}(\Sigma_p(\phi)).\end{aligned}$$

to $\overline{C'}$ in the the adaptation error. In realistic settings, where approximately shared features dominate the content features ($r \ll k$), this change in representation covariance can exceed $\Lambda_{y,e}$, thus increasing the adaptation error.

4 EXPERIMENTS

In Section 4.1, we consolidate our theoretical findings via simulations for linear representations (see Section B for detailed settings). In Section 4.2, we control the feature space implicitly by using the **Nuclear Norm (NUC)** regularization and explicitly by learning the *Disentangled Representation* via **ProjectionNet**. We show that controlling the feature space provide better controllability over the feature space than the NUC regularization by learning the disentangled feature space. We evaluate the learned feature space performance in domain generalization (section 4.2), and its adaptivity in domain adaptation (linear probing and finetuning) (section 4.2).

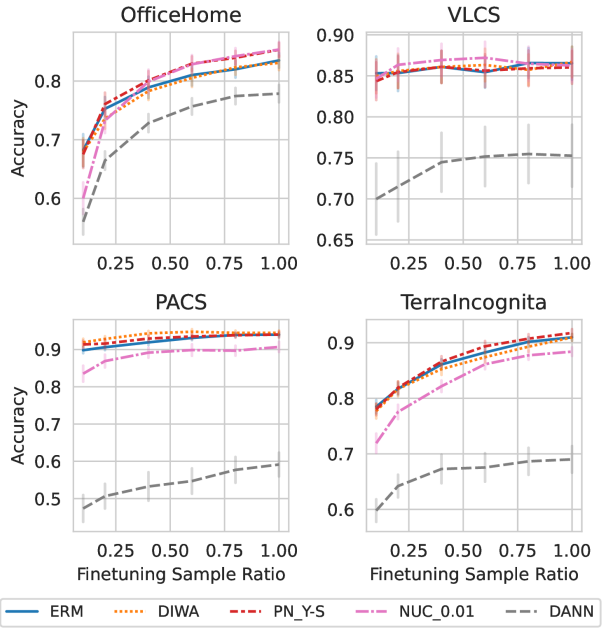


Figure 3: Linear Probing Results: ProjectionNet shares similar adaptive performance as 20 Model ensembled DiWA

4.1 SYNTHETIC DATA

We use synthetic data to assess the performance of our algorithm in the linear representation setting, where the algorithm in Section 2.2 leads to a closed-form solution (Equation 22). The plot of loss versus the number of few-shot samples is shown in Figure 5. For a smaller number of few-shot samples, a larger regularization strength leads to better performance, whereas for a larger number of few-shot samples, a smaller regularization strength yields better results, aligning with our theoretical analysis.

4.2 REAL DATA

We evaluated performance on real-world datasets: OfficeHome (Venkateswara et al., 2017), VLCS (Fang et al., 2013), PACS (Li et al., 2017), TerraIncognita (Beery et al., 2018) subset from DomainBed benchmark (Gulrajani and Lopez-Paz, 2020).

We adopted the training setup from DeepDG (Wang and Lu): using three domains as source environments and one as the target environment. We train the model for 60 epochs on source domains, selecting the best-performing one on the *source data validation set*. This model is directly evaluated on the target domain to assess its **domain generalization** capability. We assessed adaptability via **linear probing** and **target fine-tuning** on the target domain with different sizes of target data. .

All the runs are averaged over 5 runs seeds. We show the figure with the shaded area representing the Standard Error. We describe the dataset statistics, ablation study, full results, and more detailed parameter settings in the appendix. Our code is available at `https://github.com/Xiang-Pan/ProjectionNet`.

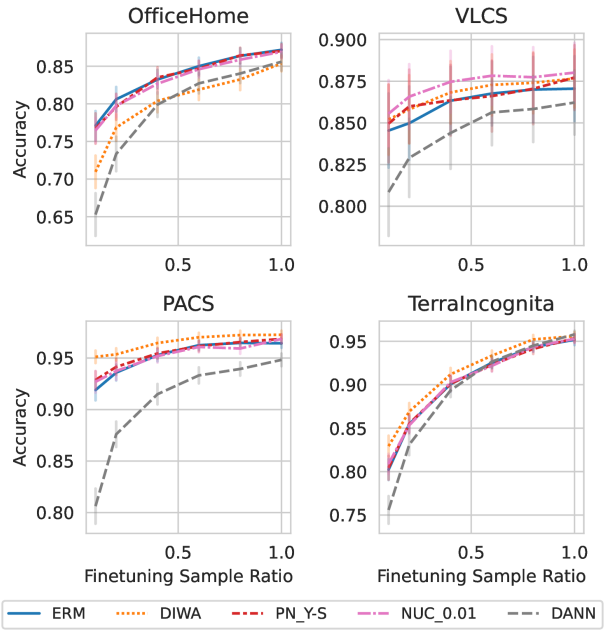


Figure 4: Target Finetuning Results: With different feature parts, we show that PN-YS and PN-YSE can achieve similar or better results than the traditional diverse feature learning method (DiWA) and strict invariant feature learning method (DANN).

Domain Generalization We show the results of Domain Generalization in Table 1 with model only trained on source domains. We show that ProjectionNet can achieve similar or better performance than DiWA with **20 times less training time**. NUC is sensitive to the regularization strength, and we need to train the model **multiple times** to choose a suitable value to achieve good performance. In contrast, ProjectionNet is not sensitive to the hyperparameter for feature disentanglement, as long as it is not too small (well-regularized). ProjectionNet controls the feature space by explicitly selecting the appropriate disentangled feature components, rather than implicitly adjusting the regularization strength.

Domain Adaptation via Finetuning Figure 3 presents the linear probing results, which demonstrate that the performance improves as more target data becomes available. Disentangled feature space is more adaptive to the target domain, which benefits the target feature learning during finetuning. For less complex tasks, the curve is flat, and invariant feature learning methods (e.g., ERM, DANN) servers as strong baselines and ProjNet-Y can achieve good performance; While for more complex tasks (OfficeHome and TerraIncognita), ProjectionNet provides more flexible control over the feature space we can use, which push the frontier of the domain adaptation among different target data size.

Feature Space Visualization To validate what kind of feature space different methods learn, we visualize the feature space using the GradCAM++ (Chattopadhay et al., 2018) with source pretrained models in Figure 1b. We provide more visualization in the Appendix G.1.

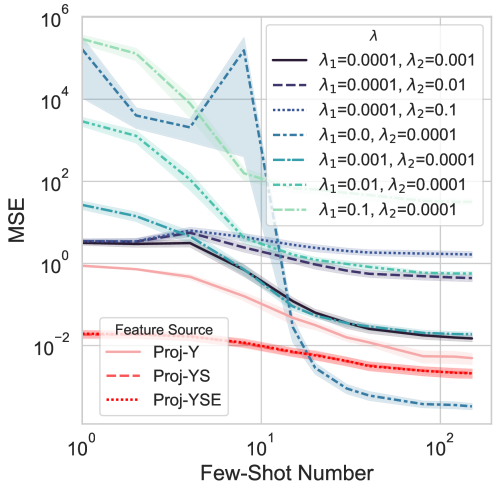


Figure 5: Target loss vs. few-shot sample size for linear synthetic data. Both Reg_1 and Reg_2 in Equation 22 improve domain adaptation performance. ProjectionNet shows similar behavior with an appropriate choice of regularization strength, especially when the number of few-shot samples is moderate.

4.3 EMPIRICAL TAKEAWAYS

Diverse Feature Space benefits with large target dataset size: For unsupervised domain generalization Table 1 or when target samples are limited, methods that emphasize invariant features, such as PN-Y and NUC-0.1, perform well. However, as the number of target samples increases, models that allow more diversity in the feature space generally achieve better performance. As discussed in Remark 3.6, learning approximately shared features can improve the sample efficiency when adapting to the target task with a moderate number (n_2) of labeled samples.

Approximately shared feature is not always helpful but feature space disentanglement is: The benefits of approximately shared feature is task-dependent. For some domain generalization tasks, the invariant feature space is rich and sufficient, e.g., PACS and VLCS, the invariant feature learning methods (DANN, PN-Y, NUC with large λ_{NUC}) will achieve good performance. For complex tasks with limited invariant features, the approximately shared feature space will help the target domain adaptation performance. If we have no such prior knowledge about the task, we have to tune the regularization strength multiple times to get the suitable feature space implicitly for the target domain adaptation, which is time-consuming and infeasible if the source pretraining is costly. If we tune the regularization strength for small target dataset size, imagine we get more target samples in the future, we have to retrain the model again to adjust the feature space. In ProjectionNet, we disentangle the feature space into pieces with different invariance, which

can be optionally used in downstream tasks with different target dataset size.

Disentangled feature is more semantical meaningful: In additional to the performance, we show that the disentangled feature space is ,more semantically meaningful, which is helpful for understanding the learned feature space and monitoring the finetuning process.

5 CONCLUSION

In this work, we propose a feature selection strategy that is robust to distributional shifts. Our theoretical framework naturally distinguishes between content features and environmental features. We explore algorithms that can reduce spurious correlations under this framework and derive a generalization bound for the target distribution. We demonstrate that adding regularization in both the pre-training and finetuning phases reduces the impact of spurious features and allows for rapid adaptation to the target task with limited samples. Inspired by this theory, we introduce ProjectionNet, a practical method that disentangles content features from environmental features through explicit feature space control. This approach alleviates the need for extensive regularization tuning, which is common in implicit feature control methods, and reduces the reliance on multiple training strategies like model ensembling. Our empirical results show a significant advantage of our methods, further consolidating our theoretical findings.

Acknowledgements

This material is based upon work supported by the U.S. Department of Energy, Office of Science Energy Earthshot Initiative as part of the project "Learning reduced models under extreme data conditions for design and rapid decision-making in complex systems" under Award #DE-SC0024721.

References

Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *International Conference on Machine Learning*, pages 145–155. PMLR, 2020.

Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, and Mario Marchand. Domain-adversarial neural networks. *stat*, 1050:15, 2014.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pages 456–473, 2018.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79:151–175, 2010.

David Berthelot, Rebecca Roelofs, Kihyuk Sohn, Nicholas Carlini, and Alexey Kurakin. Adamatch: A unified approach to semi-supervised learning and domain adaptation. In *International Conference on Learning Representations*, 2021.

Tianle Cai, Ruiqi Gao, Jason Lee, and Qi Lei. A theory of label propagation for subpopulation shift. In *International Conference on Machine Learning*, pages 1170–1182. PMLR, 2021.

Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE, 2018.

Qi Chen and Mario Marchand. Algorithm-dependent bounds for representation learning of multi-source domain adaptation. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent, editors, *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 10368–10394. PMLR, 25–27 Apr 2023. URL https://proceedings.mlr.press/v206/chen23h.html.

Yining Chen, Colin Wei, Ananya Kumar, and Tengyu Ma. Self-training avoids using spurious features under domain shift. *Advances in Neural Information Processing Systems*, 33:21061–21071, 2020.

Yining Chen, Elan Rosenfeld, Mark Sellke, Tengyu Ma, and Andrej Risteski. Iterative feature matching: Toward provable domain generalization with logarithmic environments. *Advances in Neural Information Processing Systems*, 35:1725–1736, 2022.

Kurtland Chua, Qi Lei, and Jason D. Lee. How fine-tuning allows for effective meta-learning. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=-KGLlWv6kIc.

Corinna Cortes, Yishay Mansour, and Mehryar Mohri. Learning bounds for importance weighting. *Advances in neural information processing systems*, 23, 2010.

Corinna Cortes, Mehryar Mohri, and Andrés Muñoz Medina. Adaptation algorithm and theory based on generalized discrepancy. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 169–178, 2015.

Hal Daumé III. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, 2007.

Chandler Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation III. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970. ISSN 00361429. URL http://www.jstor.org/stable/2949580. Publisher: Society for Industrial and Applied Mathematics.

Giulia Denevi, Carlo Ciliberto, Dimitris Stamos, and Massimiliano Pontil. Learning to learn around a common mean. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/b9a25e422ba96f7572089a00b838c3f8-Paper.pdf.

Giulia Denevi, Massimiliano Pontil, and Carlo Ciliberto. The advantage of conditional meta-learning for biased regularization and fine tuning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 964–974. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/0a716fe8c7745e51a3185fc8be6ca23a-Paper.pdf.

Shaofeng Deng, Shuyang Ling, and Thomas Strohmer. Strong consistency, graph Laplacians, and the stochastic block model. *Journal of Machine Learning Research*, 22(117):1–44, 2021. URL http://jmlr.org/papers/v22/20-391.html.

Simon Shaolei Du, Wei Hu, Sham M. Kakade, Jason D. Lee, and Qi Lei. Few-shot learning via learning the representation, provably. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=pW2Q2xLwIMD.

Chen Fang, Ye Xu, and Daniel N Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1657–1664, 2013.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2020.

Jeff Z HaoChen, Colin Wei, Ananya Kumar, and Tengyu Ma. Beyond separability: Analyzing the linear transferability of contrastive representations to related subpopulations. *Advances in Neural Information Processing Systems*, 35: 26889–26902, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

James J Heckman. Sample selection bias as a specification error. *Econometrica: Journal of the econometric society*, pages 153–161, 1979.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021.

Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In *International Conference on Machine Learning*, pages 5468–5479. PMLR, 2020.

Ananya Kumar, Tengyu Ma, Percy Liang, and Aditi Raghunathan. Calibrated ensembles can mitigate accuracy tradeoffs under distribution shift. In *Uncertainty in Artificial Intelligence*, pages 1041–1051. PMLR, 2022a.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2022b.

Qi Lei, Wei Hu, and Jason Lee. Near-optimal linear regression under distribution shift. In *International Conference on Machine Learning*, pages 6164–6174. PMLR, 2021.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017.

Evan Z Liu, Behzad Haghgoo, Annie S Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning*, pages 6781–6792. PMLR, 2021a.

Hong Liu, Jeff Z HaoChen, Adrien Gaidon, and Tengyu Ma. Self-supervised learning is more robust to dataset imbalance. In *International Conference on Learning Representations*, 2021b.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *Advances in neural information processing systems*, 31, 2018.

Chaochao Lu, Yuhuai Wu, Jośe Miguel Hernández-Lobato, and Bernhard Schölkopf. Nonlinear invariant risk

minimization: A causal approach. *arXiv preprint arXiv:2102.12353*, 2021.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684, 2020.

Geon Yeong Park and Sang Wan Lee. Information-theoretic regularization for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9214–9223, October 2021.

Reese Pathak, Cong Ma, and Martin Wainwright. A new similarity measure for covariate shift with applications to nonparametric regression. In *International Conference on Machine Learning*, pages 17517–17530. PMLR, 2022.

Joaquin Quinonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence. *Dataset shift in machine learning*. Mit Press, 2008.

Alexandre Rame, Matthieu Kirchmeyer, Thibaud Rahier, Alain Rakotomamonjy, Patrick Gallinari, and Matthieu Cord. Diverse weight averaging for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 35:10821–10836, 2022.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, pages 213–226. Springer, 2010.

Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning*, pages 8346–8356. PMLR, 2020.

Ziliang Samuel Zhong and Shuyang Ling. Improved theoretical guarantee for rank aggregation via spectral method. *Information and Inference: A Journal of the IMA*, 13(3):iaae020, 08 2024. ISSN 2049-8772. doi: 10.1093/imaiai/iaae020. URL https://doi.org/10.1093/imaiai/iaae020.

Kendrick Shen, Robbie M Jones, Ananya Kumar, Sang Michael Xie, Jeff Z HaoChen, Tengyu Ma, and Percy Liang. Connect, not collapse: Explaining contrastive learning for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 19847–19878. PMLR, 2022a.

Ruoqi Shen, Sébastien Bubeck, and Suriya Gunasekar. Data augmentation as feature manipulation. In *International conference on machine learning*, pages 19773–19808. PMLR, 2022b.

Zhenmei Shi, Yifei Ming, Ying Fan, Frederic Sala, and Yingyu Liang. Domain generalization via nuclear norm regularization. *arXiv preprint arXiv:2303.07527*, 2023.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

Megha Srivastava, Tatsunori Hashimoto, and Percy Liang. Robustness to spurious correlations via human annotations. In *International Conference on Machine Learning*, pages 9109–9119. PMLR, 2020.

Qian Sun, Rita Chattopadhyay, Sethuraman Panchanathan, and Jieping Ye. A two-stage weighting framework for multi-source domain adaptation. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper_files/paper/2011/file/d709f38ef758b5066ef31b18039b8ce5-Paper.pdf.

Nilesh Tripuraneni, Michael Jordan, and Chi Jin. On the theory of transfer learning: The importance of task diversity. *Advances in neural information processing systems*, 33:7852–7862, 2020.

Nilesh Tripuraneni, Chi Jin, and Michael Jordan. Provable meta-learning of linear representations. In *International Conference on Machine Learning*, pages 10434–10443. PMLR, 2021.

Joel A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, August 2012. ISSN 1615-3375, 1615-3383. doi: 10.1007/s10208-011-9099-z. URL http://link.springer.com/10.1007/s10208-011-9099-z.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017.

Martin J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019. doi: 10.1017/9781108627771.

Jindong Wang and Wang Lu. Deepdg: Deep domain generalization toolkit. https://github.com/jindongwang/transferlearning/tree/master/code/DeepDG.

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

Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. Theoretical analysis of self-training with deep networks on unlabeled data. In *International Conference on Learning Representations*, 2020.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning*, pages 23965–23998. PMLR, 2022.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 33:6256–6268, 2020a.

Sang Michael Xie, Ananya Kumar, Robbie Jones, Fereshte Khani, Tengyu Ma, and Percy Liang. In-n-out: Pre-training and self-training using auxiliary information for out-of-distribution robustness. *arXiv preprint arXiv:2012.04550*, 2020b.

Shuo Yang, Yijun Dong, Rachel Ward, Inderjit S Dhillon, Sujay Sanghavi, and Qi Lei. Sample efficiency of data augmentation consistency regularization. In *International Conference on Artificial Intelligence and Statistics*, pages 3825–3853. PMLR, 2023.

Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the twenty-first international conference on Machine learning*, page 114, 2004.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]
2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]
3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Yes]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A General nonlinear representations

In this section, we will present the omitted details in Section 3 including Assumption A.1 and A.2, the complete version of Theorem 3.4 and 3.5, and the proof of them.

We make the following assumptions on the representation function class $\mathbf{\Phi}_D$ and the input distribution p, which ensures the concentration properties of the representation covariances.

Assumption A.1 (Point-wise concentration) *For some failure probability $\delta \in (0,1)$, there exists a positive integer $N_{\text{point}}(\mathbf{\Phi}_D, p, \delta)$ such that if $n \geq N_{\text{point}}(\mathbf{\Phi}_D, p, \delta)$, then for any two given representation functions $\phi \in \mathbf{\Phi}_{d_1}$, $\phi' \in \mathbf{\Phi}_{d_2}$, $d_1, d_2 \in \{D, k, D-k\}$, n i.i.d. samples of p will with probability $1-\delta$ satisfy*

$$0.9 S_p(\phi, \phi') \preceq S_{\widehat{p}}(\phi, \phi') \preceq 1.1 S_p(\phi, \phi')$$

where $\widehat{p}$ is the empirical distribution over the n samples.

Assumption A.2 (Uniform concentration) *For some failure probability $\delta \in (0,1)$, there exists a positive integer $N_{\text{unif}}(\mathbf{\Phi}_D, q, \delta)$ such that if $n \geq N_{\text{unif}}(\mathbf{\Phi}_D, q, \delta)$, then n i.i.d. samples of p will with probability $1-\delta$ satisfy*

$$0.9 S_p(\phi, \phi') \preceq S_{\widehat{p}}(\phi, \phi') \preceq 1.1 S_p(\phi, \phi'), \quad \forall \phi \in \mathbf{\Phi}_{d_1}, \phi' \in \mathbf{\Phi}_{d_2},$$

where $d_1, d_2 \in \{D, k, D-k\}$ and $\widehat{p}$ is the empirical distribution over the n samples.

Assumption A.1 and A.2 are the conditions of the representation function class and the input distribution that ensure the concentration of empirical covariances to their population counterpart. Since $\mathbf{\Phi}_k / \mathbf{\Phi}_{D-k}$ are function classes induced by some fixed $\mathbf{\Phi}_D$, the concentration of $S_{\widehat{p}}$ in $\mathbf{\Phi}_D$ automatically guarantees its concentration in $\mathbf{\Phi}_k$, $\mathbf{\Phi}_{D-k}$, and their union.

Since uniform concentration is stronger than point-wise concentration, we expect $N_{\text{unif}}(\mathbf{\Phi}_D, q, \delta) \gtrsim N_{\text{point}}(\mathbf{\Phi}_D, q, \delta)$. In particular, in the linear case discussed in Section B.2 where $\mathbf{\Phi}_D = \{f : f(\boldsymbol{x}) = R^\top \boldsymbol{x}, R \in \mathcal{O}(d)\}$ and $\mathbf{\Phi}_k = \{f : f(\boldsymbol{x}) = R^\top \boldsymbol{x}, R^\top R = I_k, R \in \mathbb{R}^{d \times k}\}$, we show that $N_{\text{unif}}(\mathbf{\Phi}_D, q, \delta) = O(d)$ and $N_{\text{point}}(\mathbf{\Phi}_D, q, \delta) = O(k)$ in Claim B.8 and B.9.

We present the complete version of Theorem 3.4 and 3.5 as follows.

Theorem A.3 (Full version of Theorem 3.4) *Under Assumption A.2, 3.2, and 3.3, for some failure probability $\delta = o(1)$, we further assume $n_1 \gtrsim N_{\text{unif}}(\mathbf{\Phi}_D, q, \delta)$, then the average excess risk across the source environments with probability $1 - o(1)$ satisfies*

$$\overline{\text{ER}}(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^1, \ldots, \widehat{\boldsymbol{\theta}}^E) := \frac{1}{E} \sum_{e=1}^{E} \text{ER}_e(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^e) \lesssim \sigma \sqrt{\frac{\mathcal{C}_{cont}}{n_1 E}} + \mathcal{C}_{env}$$

where

$$\mathcal{C}_{cont} := \left\| \Sigma_p(\phi^*_{D[1:k]}) \right\| \left(\|\boldsymbol{\theta}^*\|^2 + \text{Tr}(\Lambda_{11}) \right), \quad \mathcal{C}_{env} := \left\| \Sigma_p(\phi^{*e}_{D[k+1:D]}) \right\| \text{Tr}(\Lambda_{22}).$$

Theorem A.4 (Full version of Theorem 3.5) *Under Assumption A.1, A.2, 3.2, and 3.3, for some failure probability $\delta = o(1)$, we further assume $n_1 \gtrsim N_{\text{unif}}(\mathbf{\Phi}_D, q, \delta)$, $n_2 \gtrsim N_{\text{point}}(\mathbf{\Phi}_D, q, \delta)$. Under the choice of λ_1 and λ_2 in 17, the excess risk of the learned predictor $\boldsymbol{x} \mapsto \langle \widehat{\phi}_D^{E+1}(\boldsymbol{x}), \widehat{\boldsymbol{\theta}}^{E+1} \rangle$ in equation 6 on the target domain with probability $1 - o(1)$ satisfies*

$$\mathbb{E}_{\boldsymbol{\theta}^{*E+1}} \left[\text{ER}_{E+1}(\widehat{\phi}_D^{E+1}, \widehat{\boldsymbol{\theta}}^{E+1}) \right] \lesssim \sigma \sqrt{\frac{\overline{\mathcal{C}'}_{env}}{n_2}} + \sigma \sqrt{\frac{k}{n_2}} + \sigma \sqrt{\frac{RE}{n_2}},$$

where

$$\overline{\mathcal{C}'}_{env} := \left(\|\boldsymbol{\theta}^*\|^2 + \text{Tr}(\Lambda_{11}) \right) \max_{\phi \in \mathbf{\Phi}_{D-k}} \text{Tr}(\Sigma_p(\phi)), \quad \text{RE} := \sigma \sqrt{\frac{\mathcal{C}_{cont}}{n_1 E}} + \mathcal{C}_{env}.$$

A.1 Representation divergence

First, we introduce the definition of representation divergence, which is helpful in the analysis of the excess risk. Note that it is a generalization of the representation divergence defined in (Du et al., 2021) where the authors only consider the divergence between representations in the same function class.

Definition A.5 *Let q be a distribution over $\mathbb{R}^d$ and $\mathbf{\Phi}_D$ be some function class with $\mathbf{\Phi}_k/\mathbf{\Phi}_{D-k}$ defined in equation 1. For two representation functions $\phi \in \mathbf{\Phi}_{d_1}$, $\phi' \in \mathbf{\Phi}_{d_2}$, $d_1, d_2 \in \{D, k, D-k\}$, we define the divergence between ϕ and ϕ' with respect to q as*

$$D_q(\phi, \phi') = \Sigma_q(\phi', \phi') - \Sigma_q(\phi', \phi)(\Sigma_q(\phi, \phi))^\dagger \Sigma_q(\phi, \phi') \in \mathbb{R}^{d_2 \times d_2}.$$

It can be verified that $D_q(\phi, \phi') \succeq 0$, $D_q(\phi, \phi') = 0$ for any ϕ, ϕ' and q. The following lemma states the relation between covariance and divergence between representations. Note that in (Du et al., 2021), it holds for $\phi, \phi' \in \mathbf{\Phi}$ for some function class Φ. It can be generalized to $\phi \in \mathbf{\Phi}_{d_1}$, $\phi' \in \mathbf{\Phi}_{d_2}$, $d_1, d_2 \in \{D, k, D-k\}$ using the same proof in (Du et al., 2021) due to the fact that $\mathbf{\Phi}_k$ and $\mathbf{\Phi}_{D-k}$ are induced by $\mathbf{\Phi}_D$.

Lemma A.6 *Given two representation functions $\phi \in \mathbf{\Phi}_{d_1}$, $\phi' \in \mathbf{\Phi}_{d_2}$, $d_1, d_2 \in \{D, k, D-k\}$, and two distributions q, q' over $\mathbb{R}^d$. If*

$$S_q(\phi, \phi') \succeq \alpha \cdot S_{q'}(\phi, \phi')$$

then

$$D_q(\phi, \phi') \succeq \alpha \cdot D_{q'}(\phi, \phi').$$

Proof. We will only show the case when $\phi \in \mathbf{\Phi}_k$ and $\phi' \in \mathbf{\Phi}_{D-k}$ since the technique is the same. For any $\boldsymbol{v} \in \mathbb{R}^{D-k}$, we will show that $\boldsymbol{v}^\top D_q(\phi, \phi')\boldsymbol{v} \geq \alpha \cdot \boldsymbol{v}^\top D_{q'}(\phi, \phi')\boldsymbol{v}$. We define the quadratic function $f(\boldsymbol{w}) := [\boldsymbol{w}^\top, -\boldsymbol{v}^\top] S_q(\phi, \phi')[\boldsymbol{w}^\top, -\boldsymbol{v}^\top]^\top$. Using the definition of $S_q(\phi, \phi')$, we get

$$\begin{aligned} f(\boldsymbol{w}) &= \boldsymbol{w}^\top \Sigma_q(\phi, \phi)\boldsymbol{w} - \boldsymbol{v}^\top \Sigma_q(\phi', \phi)\boldsymbol{w} - \boldsymbol{w}^\top \Sigma_q(\phi, \phi')\boldsymbol{v} + \boldsymbol{v}^\top \Sigma_q(\phi', \phi')\boldsymbol{v} \\ &= \mathbb{E}_{\boldsymbol{x} \sim q}\left[(\langle \phi(\boldsymbol{x}), \boldsymbol{w}\rangle - \langle \phi'(\boldsymbol{x}), \boldsymbol{v}\rangle)^2\right] \geq 0. \end{aligned}$$

Note that $f(\boldsymbol{w})$ has maximized at $\boldsymbol{w}^* = (\Sigma_q(\phi, \phi))^\dagger \Sigma_q(\phi, \phi')\boldsymbol{v}$

$$\min_{\boldsymbol{w} \in \mathbb{R}^k} f(\boldsymbol{w}) = f(\boldsymbol{w}^*) = \boldsymbol{v}^\top D_q(\phi, \phi')\boldsymbol{v}.$$

Similarly, let $g(\boldsymbol{w}) := [\boldsymbol{w}^\top, -\boldsymbol{v}^\top] S_{q'}(\phi, \phi')[\boldsymbol{w}^\top, -\boldsymbol{v}^\top]^\top$. We have

$$\min_{\boldsymbol{w} \in \mathbb{R}^k} g(\boldsymbol{w}) = \boldsymbol{v}^\top D_{q'}(\phi, \phi')\boldsymbol{v}.$$

Note that

$$S_q(\phi, \phi') \succeq \alpha \cdot S_{q'}(\phi, \phi')$$

implies $f(\boldsymbol{w}) \geq \alpha g(\boldsymbol{w})$ for any $\boldsymbol{w} \in \mathbb{R}^k$. Recall that $f(\boldsymbol{w})$ is minimized at $\boldsymbol{w}^*$, we have

$$\alpha \boldsymbol{v}^\top D_{q'}(\phi, \phi')\boldsymbol{v} = \alpha \min_{\boldsymbol{w} \in \mathbb{R}^k} g(\boldsymbol{w}) \leq \alpha g(\boldsymbol{w}^*) \leq f(\boldsymbol{w}^*) = \boldsymbol{v}^\top D_q(\phi, \phi')\boldsymbol{v},$$

which finishes the proof. □

A.2 Proof of Theorem A.3

By the definition of average excess risk across source environment,

$$\overline{\mathrm{ER}}(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^1, \ldots, \widehat{\boldsymbol{\theta}}^E) = \frac{1}{E}\sum_{e=1}^{E} \mathrm{ER}_e(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^e). \tag{9}$$

For $e \in [E]$, we define the empirical excess risk on n_1 i.i.d. samples $(\boldsymbol{x}_1^e, y_1^e), \ldots, (\boldsymbol{x}_{n_1}^e, y_{n_1}^e) \sim \mu^e$ as

$$\widehat{\mathrm{ER}}(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^e) := \frac{1}{2n_1}\left\|\phi_D^*(X^e)\boldsymbol{\theta}^{*e} - \widehat{\phi}_k^E(X^e)\widehat{\boldsymbol{\theta}}^e\right\|^2. \tag{10}$$

Then each $\mathrm{ER}_e(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^e)$ can be upper bounded by its empirical counterpart via

$$\begin{aligned}
\mathrm{ER}_e(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^e) &= \frac{1}{2}\,\mathbb{E}_{\boldsymbol{x}\sim p}[(\langle \phi_D^*(\boldsymbol{x}), \boldsymbol{\theta}^{*E+1}\rangle - \langle \widehat{\phi}_k^E(\boldsymbol{x}), \widehat{\boldsymbol{\theta}}^e\rangle)^2] \\
&= \frac{1}{2}\begin{bmatrix}\widehat{\boldsymbol{\theta}}^e \\ -\boldsymbol{\theta}^{*E+1}\end{bmatrix}^\top S_p\left(\widehat{\phi}_k^E, \phi_D^*\right)\begin{bmatrix}\widehat{\boldsymbol{\theta}}^e \\ -\boldsymbol{\theta}^{*E+1}\end{bmatrix} \\
&\lesssim \begin{bmatrix}\widehat{\boldsymbol{\theta}}^e \\ -\boldsymbol{\theta}^{*E+1}\end{bmatrix}^\top S_{\widehat{p}^e}\left(\widehat{\phi}_k^E, \phi_D^*\right)\begin{bmatrix}\widehat{\boldsymbol{\theta}}^e \\ -\boldsymbol{\theta}^{*E+1}\end{bmatrix} \\
&= 2\widehat{\mathrm{ER}}_e(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^e).
\end{aligned}$$

Then to estimate equation 9, it suffices to upper bound the average empirical excess risk

$$\frac{1}{E}\sum_e^E \widehat{\mathrm{ER}}_e(\widehat{\phi}_k^E, \widehat{\boldsymbol{\theta}}^e) = \frac{1}{2n_1E}\sum_{e=1}^E \left\|\phi_D^*(X^e)\boldsymbol{\theta}^{*e} - \widehat{\phi}_k^E(X^e)\widehat{\boldsymbol{\theta}}^e\right\|^2. \tag{11}$$

By the optimality of $\widehat{\phi}_k, \widehat{\boldsymbol{\theta}}_1, \ldots, \widehat{\boldsymbol{\theta}}_E$, the empirical risk of the source environments satisfies

$$\frac{1}{2n_1E}\sum_{e=1}^E \left\|\boldsymbol{y}^e - \widehat{\phi}_k^E(X^e)\widehat{\boldsymbol{\theta}}^e\right\|^2 \le \frac{1}{2n_1E}\sum_{e=1}^E \left\|\boldsymbol{y}^e - \phi_{D[1:k]}^*(X^e)\widehat{\boldsymbol{\theta}}^e_{[1:k]}\right\|^2,$$

which implies that equation 11 can be decomposed into two terms

$$\begin{aligned}
\frac{1}{2n_1E}\sum_{e=1}^E \left\|\phi_D^*(X^e)\boldsymbol{\theta}^{*e} - \widehat{\phi}_k^E(X^e)\widehat{\boldsymbol{\theta}}^e\right\|^2 \le{}& \underbrace{\frac{1}{n_1E}\sum_{e=1}^E \langle \widehat{\phi}_k^E(X^e)\widehat{\boldsymbol{\theta}}^e - \phi_{D[1:k]}^*(X^e)\boldsymbol{\theta}^{*e}_{[1:k]}, \boldsymbol{z}^e\rangle}_{T_1} \\
&+ \underbrace{\frac{1}{2n_1E}\sum_{e=1}^E \left\|\phi_{D[k+1:D]}^*(X^e)\boldsymbol{\theta}^{*e}_{D[k+1:D]}\right\|^2}_{T_2}
\end{aligned} \tag{12}$$

where we use

$$\boldsymbol{y}^e = \phi_{D[1:k]}^*(X^e)\boldsymbol{\theta}^{*e}_{[1:k]} + \phi_{D[k+1:D]}^*(X^e)\boldsymbol{\theta}^{*e}_{[k+1:D]} + \boldsymbol{z}^e.$$

Estimation of T_1. Using $\widehat{\phi}_k^E(X^e)\widehat{\boldsymbol{\theta}}^e = \mathcal{P}_{\widehat{\phi}_k^E(X^e)}\boldsymbol{y}^e$, each summand of T_1 can be decomposed into

$$\begin{aligned}
\langle \widehat{\phi}_k^E(X^e)\widehat{\boldsymbol{\theta}}^e - \phi_{D[1:k]}^*(X^e)\boldsymbol{\theta}^{*e}_{[1:k]}, \boldsymbol{z}^e\rangle = {}& \underbrace{\langle -\mathcal{P}^\perp_{\widehat{\phi}_k^E(X^e)}\phi_{D[1:k]}^*(X^e)\boldsymbol{\theta}^{*e}_{[1:k]}, \boldsymbol{z}^e\rangle}_{T_{1,1,e}} \\
&+ \underbrace{\langle \mathcal{P}_{\widehat{\phi}_k^E(X^e)}\phi_{D[k+1:D]}^*(X^e)\boldsymbol{\theta}^{*e}_{[k+1:D]}, \boldsymbol{z}^e\rangle}_{T_{1,2,e}} \\
&+ \underbrace{\langle \mathcal{P}_{\widehat{\phi}_k^E(X^e)}\boldsymbol{z}^e, \boldsymbol{z}^e\rangle}_{T_{1,3,e}}
\end{aligned}$$

Let $\boldsymbol{v}^e = n_1^{-1/2}\phi^*_{D[1:k]}(X^e)\boldsymbol{\theta}^{*e}_{[1:k]}$ which is independent of $\boldsymbol{z}^e$. Then

$$\begin{aligned}
\frac{1}{n_1 E}\sum_{e=1}^{E} T_{1,1,e} &= \frac{1}{n_1 E}\sum_{e=1}^{E}\langle -\mathcal{P}^{\perp}_{\widehat{\phi}_k^E(X^e)}\phi^*_{D[1:k]}(X^e)\boldsymbol{\theta}^{*e}_{[1:k]}, \boldsymbol{z}^e\rangle \\
&\le \frac{1}{\sqrt{n_1 E}}\sum_{e=1}^{E}\langle \frac{1}{\sqrt{n_1}}\phi^*_{D[1:k]}(X^e)\boldsymbol{\theta}^{*e}_{[1:k]}, \frac{1}{\sqrt{E}}\boldsymbol{z}^e\rangle \\
&= \frac{1}{\sqrt{n_1 E}}\sum_{e=1}^{E}\langle \boldsymbol{v}^e, \frac{1}{\sqrt{E}}\boldsymbol{z}^e\rangle \\
&\lesssim \frac{1}{\sqrt{n_1 E}}\sqrt{\frac{\sigma^2\sum_{e=1}^{E}\left\|\boldsymbol{v}^e\right\|^2}{E}}
\end{aligned}$$

where the last inequality is obtained via Bernstein's inequality. Note that we estimate $\left\|\boldsymbol{v}^e\right\|^2$ via

$$\begin{aligned}
\left\|\boldsymbol{v}^e\right\|^2 &= \left\| n_1^{-1/2}\phi^*_{D[1:k]}(X^e)\boldsymbol{\theta}^{*e}_{[1:k]}\right\|^2 \\
&= \left\|\Sigma^{1/2}_{\widehat{p}}(\phi^*_{D[1:k]})\boldsymbol{\theta}^{*e}_{[1:k]}\right\|^2 \\
&\le \left\|\Sigma_{\widehat{p}}(\phi^*_{D[1:k]})\right\|\left(\left\|\boldsymbol{\theta}^*\right\|^2 + \mathrm{Tr}(\Lambda_{11})\right) \\
&\lesssim \left\|\Sigma_{p}(\phi^*_{D[1:k]})\right\|\left(\left\|\boldsymbol{\theta}^*\right\|^2 + \mathrm{Tr}(\Lambda_{11})\right)
\end{aligned}$$

where we use $\boldsymbol{\theta}^{*e}_{[1:k]} \sim \mathcal{N}(\boldsymbol{\theta}^*, \Lambda_{11})$ and Assumption A.2. Thus,

$$\frac{1}{n_1 E}\sum_{e=1}^{E} T_{1,1,e} \lesssim \sigma\sqrt{\frac{\left\|\Sigma_{p}(\phi^*_{D[1:k]})\right\|\left(\left\|\boldsymbol{\theta}^*\right\|^2 + \mathrm{Tr}(\Lambda_{11})\right)}{n_1 E}}.$$

Similarly,

$$\frac{1}{n_1 E}\sum_{e=1}^{E} T_{1,2,e} \lesssim \sigma\sqrt{\frac{\left\|\Sigma_{p}(\phi^*_{D[k+1:D]})\right\|\sum_{j=D-k+1}^{D}\left(\Lambda_{22}\right)_j}{n_1 E}}$$

where $\sum_{j=D-k+1}^{D}\left(\Lambda_{22}\right)_j$ is the sum of top k entries in Λ_{22}, and

$$\frac{1}{n_1 E}\sum_{e=1}^{E} T_{1,3,e} \lesssim \frac{\sigma^2 k}{n_1}.$$

Then we obtain the estimation of T_1

$$\begin{aligned}
T_1 &= \frac{1}{n_1 E}\sum_{e=1}^{E}\left(T_{1,1,e} + T_{1,2,e} + T_{1,3,e}\right) \\
&\lesssim \sigma\sqrt{\frac{\left\|\Sigma_{p}(\phi^*_{D[1:k]})\right\|\left(\left\|\boldsymbol{\theta}^*\right\|^2 + \mathrm{Tr}(\Lambda_{11})\right)}{n_1 E}} + \sigma\sqrt{\frac{\left\|\Sigma_{p}(\phi^*_{D[k+1:D]})\right\|\sum_{j=D-k+1}^{D}\left(\Lambda_{22}\right)_j}{n_1 E}} + \frac{\sigma^2 k}{n_1} \\
&\lesssim \sigma\sqrt{\frac{\left\|\Sigma_{p}(\phi^*_{D[1:k]})\right\|\left(\left\|\boldsymbol{\theta}^*\right\|^2 + \mathrm{Tr}(\Lambda_{11})\right)}{n_1 E}} \\
&= \sigma\sqrt{\frac{\mathcal{C}_{\text{cont}}}{n_1 E}}
\end{aligned} \tag{13}$$

where we use Assumption 3.2 and omit the third term which is of small order in the last inequality.

Estimation of T_2. Note that T_2 is the non-vanishing term as $n_1, E \to \infty$ because we only learn the content features from source environments. This term is the approximation error due to the environmental features, thus it can be bounded by the complexity of the environmental features.

$$\begin{aligned} T_2 &= \frac{1}{2n_1 E}\sum_{e=1}^{E}\left\|\phi^*_{D[k+1:D]}(X^e)^{*e}\boldsymbol{\theta}^{*e}_{[k+1:D]}\right\|^2 \\ &= \frac{1}{E}\sum_{e=1}^{E}\left\|\Sigma^{1/2}_{\widehat{p}}(\phi^{*e}_{D[k+1:D]})\boldsymbol{\theta}^{*e}_{[k+1:D]}\right\|^2 \\ &\lesssim \left\|\Sigma_p(\phi^{*e}_{D[k+1:D]})\right\|\mathrm{Tr}(\Lambda_{22}) \\ &= \mathcal{C}_{\text{env}} \end{aligned} \tag{14}$$

Thus, with probability $1-o(1)$, the average excess risk across source environments satisfies

$$\overline{\mathrm{ER}}(\widehat{\phi}^E_k, \widehat{\boldsymbol{\theta}}^1, \ldots, \widehat{\boldsymbol{\theta}}^E) = \frac{1}{E}\sum_{e=1}^{E}\mathrm{ER}_e(\widehat{\phi}^E_k, \widehat{\boldsymbol{\theta}}^e) \lesssim T_1 + T_2 \lesssim \sigma\sqrt{\frac{\mathcal{C}_{\text{cont}}}{n_1 E}} + \mathcal{C}_{\text{env}}.$$

A.3 Proof of Theorem A.4

In this section, we abbreviate $\mathrm{ER}_{E+1}(\widehat{\phi}^{E+1}_D, \widehat{\boldsymbol{\theta}}^{E+1})$ as $\mathrm{ER}(\widehat{\phi}^{E+1}_D, \widehat{\boldsymbol{\theta}}^{E+1})$ and $\widehat{\mathrm{ER}}_{E+1}(\widehat{\phi}^{E+1}_D, \widehat{\boldsymbol{\theta}}^{E+1})$ as $\widehat{\mathrm{ER}}(\widehat{\phi}^{E+1}_D, \widehat{\boldsymbol{\theta}}^{E+1})$. We will first bound the empirical excess risk of $\boldsymbol{x} \mapsto \langle\widehat{\phi}^{E+1}_D(\boldsymbol{x}), \widehat{\boldsymbol{\theta}}^{E+1}\rangle$ and then prove that it is close to its population counterpart. By the optimality of $(\widehat{\phi}^{E+1}_D, \widehat{\boldsymbol{\theta}}^{E+1})$ for equation 6, the empirical risk satisfies

$$\begin{aligned} &\frac{1}{2n_2}\left\|\boldsymbol{y}^{E+1} - \widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right\|^2 \\ &\le \frac{1}{2n_2}\left\|\boldsymbol{y}^{E+1} - \widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right\|^2 + \frac{\lambda_1}{n_2}\left\|\mathcal{P}^{\perp}_{\widehat{\phi}^E_k(X^{E+1})}\widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right\|^2 + \frac{\lambda_2}{2}\left\|\widehat{\boldsymbol{\theta}}^{E+1}\right\|^2 \\ &\le \frac{1}{2n_2}\left\|\boldsymbol{y}^{E+1} - \phi^*_D(X^{E+1})\theta^{*E+1}\right\|^2 + \frac{\lambda_1}{n_2}\left\|\mathcal{P}^{\perp}_{\widehat{\phi}^E_k(X^{E+1})}\phi^*_D(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2 + \frac{\lambda_2}{2}\left\|\boldsymbol{\theta}^{*E+1}\right\|^2. \end{aligned} \tag{15}$$

Let $A = \left[\phi^*_{D[1:k]}(X^{E+1}), \widehat{\phi}^E_k(X^{E+1})\right]$. By plugging $\boldsymbol{y}^{E+1} = \phi^*_D(X^{E+1})\boldsymbol{\theta}^{*E+1} + \boldsymbol{z}^{E+1}$ in equation 15, the empirical excess risk on the target task satisfies

$$
\begin{aligned}
\widehat{\mathrm{ER}}(\widehat{\phi}^{E+1}_D, \widehat{\boldsymbol{\theta}}^{E+1}) &= \frac{1}{2n_2}\left\|\phi^*_D(X^{E+1})\theta^{*E+1} - \widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right\|^2 \\
&\le -\frac{1}{n_2}\left\langle \boldsymbol{z}^{E+1}, \phi^*_D(X^{E+1})\theta^{*E+1} - \widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right\rangle \\
&+ \frac{\lambda_1}{n_2}\left\|\mathcal{P}^{\perp}_{\widehat{\phi}^E_k(X^{E+1})}\phi^*_D(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2 + \frac{\lambda_2}{2}\left\|\boldsymbol{\theta}^{*E+1}\right\|^2 \\
&= -\frac{1}{n_2}\left\langle \boldsymbol{z}^{E+1}, \mathcal{P}_A\left(\phi^*_D(X^{E+1})\theta^{*E+1} - \widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right)\right\rangle \\
&- \frac{1}{n_2}\left\langle \boldsymbol{z}^{E+1}, \mathcal{P}^{\perp}_A\left(\phi^*_D(X^{E+1})\theta^{*E+1} - \widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right)\right\rangle \\
&+ \frac{\lambda_1}{n_2}\left\|\mathcal{P}^{\perp}_{\widehat{\phi}^E_k(X^{E+1})}\phi^*_D(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2 + \frac{\lambda_2}{2}\left\|\boldsymbol{\theta}^{*E+1}\right\|^2 \\
&\le \underbrace{\frac{1}{n_2}\left\|\mathcal{P}_A\boldsymbol{z}^{E+1}\right\|\left\|\phi^*_D(X^{E+1})\theta^{*E+1} - \widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right\|}_{T_1} \\
&+ \underbrace{\frac{1}{n_2}\left\langle \boldsymbol{z}^{E+1}, \mathcal{P}^{\perp}_A\phi^*_D(X^{E+1})\theta^{*E+1}\right\rangle}_{T_2} \\
&+ \underbrace{\frac{1}{n_2}\left\langle \boldsymbol{z}^{E+1}, \mathcal{P}^{\perp}_A\widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right\rangle}_{T_3} \\
&+ \underbrace{\frac{\lambda_1}{n_2}\left\|\mathcal{P}^{\perp}_{\widehat{\phi}^E_k(X^{E+1})}\phi^*_D(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2 + \frac{\lambda_2}{2}\left\|\boldsymbol{\theta}^{*E+1}\right\|^2}_{T_4}
\end{aligned}
\tag{16}
$$

We will bound the 4 terms one by one.

Estimation of T_1. Note that

$$\left\|\mathcal{P}_A\boldsymbol{z}^{E+1}\right\| \lesssim \sigma\sqrt{k},$$

then

$$T_1 \lesssim \frac{\sigma\sqrt{k}}{n_2}\left\|\phi^*_D(X^{E+1})\theta^{*E+1} - \widehat{\phi}^{E+1}_D(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right\| = 2\sigma\sqrt{\frac{k\widehat{\mathrm{ER}}(\widehat{\phi}^{E+1}_D, \widehat{\boldsymbol{\theta}}^{E+1})}{n_2}}.$$

Estimation of T_2.

$$
\begin{aligned}
T_2 = \frac{1}{n_2}\left\langle \boldsymbol{z}^{E+1}, \mathcal{P}^{\perp\top}_A\phi^*_D(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\rangle &= \frac{1}{n_2}\left\|\phi^*_D(X^{E+1})^\top\mathcal{P}^{\perp\top}_A\boldsymbol{z}^{E+1}\right\|\left\|\boldsymbol{\theta}^{*E+1}\right\| \\
&= \frac{1}{n_2}\left\|\phi^*_{D[k+1:D]}(X^{E+1})^\top\mathcal{P}^{\perp\top}_A\boldsymbol{z}^{E+1}\right\|\left\|\boldsymbol{\theta}^{*E+1}\right\| \\
&\lesssim \sigma\sqrt{\frac{\mathrm{Tr}\Big(\Sigma_p(\phi^*_{D[k+1:D]})\Big)\left(\left\|\boldsymbol{\theta}^*\right\|^2 + \mathrm{Tr}(\Lambda_{11}) + \mathrm{Tr}(\Lambda_{22})\right)}{n_2}} \\
&= \lesssim \sigma\sqrt{\frac{\mathcal{C}'_{\text{env}}}{n_2}}
\end{aligned}
$$

where we use Assumption 3.2 and the fact that $A \perp \phi^*_{D[1:k]}(X^{E+1})$ and define $\mathcal{C}'_{\text{env}} := \mathrm{Tr}\Big(\Sigma_p(\phi^*_{D[k+1:D]})\Big)\left(\left\|\boldsymbol{\theta}^*\right\|^2 + \mathrm{Tr}(\Lambda_{11}) + \mathrm{Tr}(\Lambda_{22})\right)$.

Estimation of T_3. Similarly to T_2, T_3 is bounded via

$$\begin{aligned} T_3 &= \frac{1}{n_2}\left\langle \boldsymbol{z}^{E+1}, \mathcal{P}_A^{\perp}\widehat{\phi}_D^{E+1}(X^{E+1})\widehat{\boldsymbol{\theta}}^{E+1}\right\rangle \\ &\le \frac{1}{n_2}\left\|\widehat{\phi}_D^{E+1}(X^{E+1})^{\top}\mathcal{P}_A^{\perp\top}\boldsymbol{z}^{E+1}\right\|\left\|\widehat{\boldsymbol{\theta}}^{E+1}\right\| \\ &\lesssim \sigma\sqrt{\frac{\mathrm{Tr}\Big(\Sigma_p(\widehat{\phi}_D^{E+1})\Big)\left(\|\boldsymbol{\theta}^*\|^2+\mathrm{Tr}(\Lambda_{11})+\mathrm{Tr}(\Lambda_{22})\right)}{n_2}} \\ &\lesssim \sigma\sqrt{\frac{\overline{\mathcal{C}'}_{\text{env}}}{n_2}} \end{aligned}$$

where $\overline{\mathcal{C}}_{\text{env}}$ is the maximum complexity of the environment features defined as

$$\overline{\mathcal{C}'}_{\text{env}} := \left(\|\boldsymbol{\theta}^*\|^2+\mathrm{Tr}(\Lambda_{11})\right)\max_{\phi\in\boldsymbol{\Phi}_{D-k}}\mathrm{Tr}(\Sigma_p(\phi)),$$

where we apply the assymption $\|\boldsymbol{\theta}^*\|^2+\mathrm{Tr}(\Lambda_{11})+\mathrm{Tr}(\Lambda_{22})\le(1+\gamma)(\|\boldsymbol{\theta}^*\|^2+\mathrm{Tr}(\Lambda_{11}))$. Thus $T3\gtrsim T_2$ by definition.

Estimation of T_4. The following lemma estimates $n_2^{-1}\left\|\mathcal{P}^{\perp}_{\widehat{\phi}_k^E(X^{E+1})}\phi_D^*(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2$.

Lemma A.7 *Under the conditions in Theorem A.4, it holds with probability $1-o(1)$ that*

$$\frac{1}{n_2}\left\|\mathcal{P}^{\perp}_{\widehat{\phi}_k^E(X^{E+1})}\phi_D^*(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2 \lesssim \underbrace{\sigma\sqrt{\frac{\mathcal{C}_{cont}}{n_1E}}+\mathcal{C}_{env}}_{:=RE}.$$

Then T_4 is bounded via

$$\begin{aligned} T_4 &= \frac{\lambda_1}{n_2}\left\|\mathcal{P}^{\perp}_{\widehat{\phi}_k^E(X^{E+1})}\phi_D^*(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2+\frac{\lambda_2}{2}\left\|\boldsymbol{\theta}^{*E+1}\right\|^2 \\ &\lesssim \lambda_1\mathrm{RE}+\lambda_2(\|\boldsymbol{\theta}^*\|^2+\mathrm{Tr}(\Lambda_{11})+\mathrm{Tr}(\Lambda_{22})). \end{aligned}$$

Combining the above 4 terms, and under the choice

$$\lambda_1=\sqrt{\frac{\sigma^2}{n_2\mathrm{RE}}},\quad \lambda_2=\frac{\sigma\sqrt{\mathrm{RE}}}{\sqrt{n_2}(\|\boldsymbol{\theta}^*\|^2+\mathrm{Tr}(\Lambda_{11})+\mathrm{Tr}(\Lambda_{22}))}, \tag{17}$$

we have the following quadratic inequality

$$\begin{aligned} \widehat{\mathrm{ER}}(\widehat{\phi}_D^{E+1},\widehat{\boldsymbol{\theta}}^{E+1}) &\lesssim 2\sigma\sqrt{\frac{k\widehat{\mathrm{ER}}(\widehat{\phi}_D^{E+1},\widehat{\boldsymbol{\theta}}^{E+1})}{n_2}}+\sigma\sqrt{\frac{\overline{\mathcal{C}'}_{\text{env}}}{n_2}} \\ &\quad+\lambda_1\mathrm{RE}+\lambda_2(\|\boldsymbol{\theta}^*\|^2+\mathrm{Tr}(\Lambda_{11})+\mathrm{Tr}(\Lambda_{22})) \\ &\lesssim 2\sigma\sqrt{\frac{k\widehat{\mathrm{ER}}(\widehat{\phi}_D^{E+1},\widehat{\boldsymbol{\theta}}^{E+1})}{n_2}}+\sigma\sqrt{\frac{\overline{\mathcal{C}'}_{\text{env}}}{n_2}}+\sigma\sqrt{\frac{\mathrm{RE}}{n_2}}, \end{aligned} \tag{18}$$

which gives the solution

$$\widehat{\mathrm{ER}}(\widehat{\phi}_D^{E+1},\widehat{\boldsymbol{\theta}}^{E+1})\lesssim\sigma\sqrt{\frac{\overline{\mathcal{C}'}_{\text{env}}}{n_2}}+\sigma\sqrt{\frac{k}{n_2}}+\sigma\sqrt{\frac{\mathrm{RE}}{n_2}}.$$

Finally, we will prove that the empirical excess risk is close to its population counterpart

$$\begin{aligned}
\mathrm{ER}(\widehat{\phi}_D^{E+1}, \widehat{\boldsymbol{\theta}}^{E+1}) &= \frac{1}{2}\,\mathbb{E}_{\boldsymbol{x}\sim p}[(\langle \phi_D^*(\boldsymbol{x}), \boldsymbol{\theta}^{*E+1}\rangle - \langle \widehat{\phi}_D^{E+1}(\boldsymbol{x}), \widehat{\boldsymbol{\theta}}^{E+1}\rangle)^2] \\
&= \frac{1}{2}\begin{bmatrix} \widehat{\boldsymbol{\theta}}^{E+1} \\ -\boldsymbol{\theta}^{*E+1}\end{bmatrix}^\top S_p\left(\widehat{\phi}_D^{E+1}, \phi_D^*\right)\begin{bmatrix} \widehat{\boldsymbol{\theta}}^{E+1} \\ -\boldsymbol{\theta}^{*E+1}\end{bmatrix} \\
&\lesssim \begin{bmatrix} \widehat{\boldsymbol{\theta}}^{E+1} \\ -\boldsymbol{\theta}^{*E+1}\end{bmatrix}^\top S_{\widehat{p}^{E+1}}\left(\widehat{\phi}_D^{E+1}, \phi_D^*\right)\begin{bmatrix} \widehat{\boldsymbol{\theta}}^{E+1} \\ -\boldsymbol{\theta}^{*E+1}\end{bmatrix} \\
&= 2\widehat{\mathrm{ER}}(\widehat{\phi}_D^{E+1}, \widehat{\boldsymbol{\theta}}^{E+1}) \\
&\lesssim \sigma\sqrt{\frac{\mathcal{C}'_{\text{env}}}{n_2}} + \sigma\sqrt{\frac{k}{n_2}} + \sigma\sqrt{\frac{\mathrm{RE}}{n_2}}.
\end{aligned} \tag{19}$$

A.4 Proof of Lemma A.7

Note that we have

$$\frac{1}{n_2}\left\|\mathcal{P}^\perp_{\widehat{\phi}_k^E(X^{E+1})}\phi_D^*(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2 \le \frac{1}{n_2}\left\|\mathcal{P}^\perp_{\widehat{\phi}_k^E(X^{E+1})}\phi_D^*(X^{E+1})\right\|_F^2\left\|\boldsymbol{\theta}^{*E+1}\right\|^2.$$

Let $\Theta^* = [\boldsymbol{\theta}^{*1}, \ldots, \boldsymbol{\theta}^{*E}]$ and $\sigma_1(\Theta^*)$ be its smallest singular value. We have the following chain of inequalities

$$\begin{aligned}
&\frac{1}{n_2}\left\|\mathcal{P}^\perp_{\widehat{\phi}_k^E(X^{E+1})}\phi_D^*(X^{E+1})\right\|_F^2\frac{\sigma_1^2(\Theta^*)}{E} \\
&\le \frac{1}{E}\operatorname{Tr}\Big(D_{\widehat{p}^{E+1}}(\widehat{\phi}_k^E, \phi_D^*)\Big)\sigma_1^2(\Theta^*) \\
&\lesssim \frac{1}{E}\operatorname{Tr}\Big(D_p(\widehat{\phi}_k^E, \phi_D^*)\Big)\sigma_1^2(\Theta^*) \\
&\le \frac{1}{E}\left\|\Big(D_p(\widehat{\phi}_k^E, \phi_D^*)\Big)^{1/2}\Theta^*\right\|_F^2 \\
&= \frac{1}{E}\sum_{e=1}^E \boldsymbol{\theta}^{*e\top} D_p(\widehat{\phi}_k^E, \phi_D^*)\boldsymbol{\theta}^{*e} \\
&\lesssim \frac{1}{E}\sum_{e=1}^E \boldsymbol{\theta}^{*e\top} D_{\widehat{p}^e}(\widehat{\phi}_k^E, \phi_D^*)\boldsymbol{\theta}^{*e} \\
&= \frac{1}{n_1 E}\sum_{e=1}^E \boldsymbol{\theta}^{*e\top}(\phi_D^*(X^e))^\top\left(I_{n_1} - \widehat{\phi}_k^E(X^e)\left[(\widehat{\phi}_k^E(X^e))^\top\widehat{\phi}_k^E(X^e)\right]^\dagger\widehat{\phi}_k^E(X^e)\right)\phi_D^*(X^e))\boldsymbol{\theta}^{*e} \\
&= \frac{1}{n_1 E}\sum_{e=1}^E \left\|\mathcal{P}^\perp_{\widehat{\phi}(X^e)}\phi_D^*(X^e)\boldsymbol{\theta}^{*e}\right\|^2 \\
&\lesssim \frac{1}{2n_1 E}\sum_{e=1}^E \left\|\phi_D^*(X^e)\boldsymbol{\theta}^{*e} - \widehat{\phi}_k^E(X^e)\widehat{\boldsymbol{\theta}}^e\right\|^2 \\
&\lesssim \underbrace{\sigma\sqrt{\frac{\mathcal{C}_{\text{cont}}}{n_1 E}} + \mathcal{C}_{\text{env}}}_{\text{RE}}.
\end{aligned}$$

where we apply Assumption A.1 and A.2 and equation 10. Then using the gaussianality of $\boldsymbol{\theta}^{*E+1}$, we get

$$\begin{aligned}
\frac{1}{n_2}\left\|\mathcal{P}^\perp_{\widehat{\phi}_k^E(X^{E+1})}\phi_D^*(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2 &\le \frac{1}{n_2}\left\|\mathcal{P}^\perp_{\widehat{\phi}_k^E(X^{E+1})}\phi_D^*(X^{E+1})\right\|_F^2\left\|\boldsymbol{\theta}^{*E+1}\right\|^2 \\
&\lesssim \frac{E(\|\boldsymbol{\theta}^*\|^2\operatorname{Tr}(\Lambda_{11}) + \operatorname{Tr}(\Lambda_{22}))}{\sigma_1^2(\Theta^*)}\mathrm{RE}.
\end{aligned}$$

Finally, under Assumption 3.2 and 3.3,

$$\frac{1}{n_2}\left\|\mathcal{P}^{\perp}_{\widehat{\phi}^E_k(X^{E+1})}\phi^*_D(X^{E+1})\boldsymbol{\theta}^{*E+1}\right\|^2 \lesssim \frac{E(\|\boldsymbol{\theta}^*\|^2\operatorname{Tr}(\Lambda_{11})+\operatorname{Tr}(\Lambda_{22}))}{\sigma_1^2(\Theta^*)}\text{RE} \lesssim \text{RE}.$$

B Linear representations

In the linear case, the ground-truth representation function class is

$$\boldsymbol{\Phi}_d = \{f : f(\boldsymbol{x}) = R^\top \boldsymbol{x}, R \in \mathcal{O}(d)\}.$$

In particular, we let the ground-truth representation function be $\phi^*_d(\boldsymbol{x}) = R^{*\top}\boldsymbol{x}$ where $R^* \in \mathcal{O}(d)$. Then the data generation process $(\boldsymbol{x}, y) \sim \mu_e$ for $e \in [E+1]$ can be described as

$$y = \langle \boldsymbol{x}, R^*\boldsymbol{\theta}^{e*}\rangle + z.$$

Given the meta distribution in equation 3, $R^*\boldsymbol{\theta}^{e*}$'s are i.i.d. rotated multivariate Gaussian random variables

$$\begin{aligned} R^*\boldsymbol{\theta}^{e*} &\sim \mathcal{N}\left(R_1^*\boldsymbol{\theta}^*, \Sigma_{\text{RM}}\right), \\ \Sigma_{\text{RM}} &= R_1^*\Lambda_{11}R_1^{*\top} + R_2^*\Lambda_{22}R_2^{*\top}, \end{aligned}$$

where $R^* = [R_1^*, R_2^*]$, R_1^* is the first k columns of R^* and R_2^* is the rest $(d-k)$ columns. The "RM" here stands for "rotated meta distribution".

B.1 The Meta-Representation Learning Algorithm

Our goal is to learn a low-dimensional representation

$$\widehat{\phi}^E_k \in \boldsymbol{\Phi}_k = \{f : f(\boldsymbol{x}) = R^\top\boldsymbol{x}, R^\top R = I_k, R \in \mathbb{R}^{d\times k}\}$$

from the source environments that capture the content features. In the linear case, it is equivalent to finding a $d \times k$ matrix with orthogonal columns to approximate R_1^*. If E is large enough, *i.e.,* we have enough source environments, a natural way of estimating R_1^* is to use the least k eigenvectors of the sample covariance matrix of $\widehat{\boldsymbol{\theta}}^e$'s. This motivates the following learning process.

For each source environment $e \in [E]$, we obtain $\widehat{\boldsymbol{\theta}}^e$ via empirical risk minimization

$$\widehat{\boldsymbol{\theta}}^e = \underset{\boldsymbol{\theta}^e\in\mathbb{R}^d}{\operatorname{argmin}} \frac{1}{2n_1}\|\boldsymbol{y}^e - X^e\boldsymbol{\theta}^e\|^2. \tag{20}$$

Let $\bar{\boldsymbol{\theta}}^E = E^{-1}\sum_{e=1}^E \widehat{\boldsymbol{\theta}}^e$ be the "sample mean" of these learned parameters and consider the "sample covariance" with its eigendecomposition

$$\begin{aligned} \Sigma_{\widehat{\boldsymbol{\theta}}} &= \frac{1}{E}\sum_{e=1}^E \left(\widehat{\boldsymbol{\theta}}^e - \bar{\boldsymbol{\theta}}^E\right)\left(\widehat{\boldsymbol{\theta}}^e - \bar{\boldsymbol{\theta}}^E\right)^\top \\ &= \widehat{R}_1\widehat{\Lambda}_{11}\widehat{R}_1^\top + \widehat{R}_2\widehat{\Lambda}_{22}\widehat{R}_2^\top \end{aligned} \tag{21}$$

where $\widehat{\Lambda}_{11} \in \mathbb{R}^{k\times k}$ and $\widehat{\Lambda}_{22} \in \mathbb{R}^{(d-k)\times(d-k)}$ are diagonal matrices with ascending entries, $\widehat{\Lambda}_{11}$ consists of the least k eigenvalues of $\Sigma_{\widehat{\boldsymbol{\theta}}}$ with eigenvectors $\widehat{R}_1 \in \mathbb{R}^{d\times k}$ and $\widehat{\Lambda}_{22}$ consists of the remaining $d-k$ eigenvalues with eigenvectors $\widehat{R}_2 \in \mathbb{R}^{d\times(d-k)}$.

The learned representation $\widehat{R}_1$ and the average parameter $\bar{\boldsymbol{\theta}}^E$ will be applied to the fine-tunning phase on the target environment via

$$\begin{aligned} \widehat{\boldsymbol{\theta}}^{E+1} = \underset{\boldsymbol{\theta}^{E+1}\in\mathbb{R}^d}{\operatorname{argmin}} &\frac{1}{2n_2}\left\|\boldsymbol{y}^{E+1} - X^{E+1}\boldsymbol{\theta}^{E+1}\right\|^2 \\ &+ \underbrace{\frac{\lambda_1}{2}\left\|\mathcal{P}_{\widehat{R}_1}(\boldsymbol{\theta}^{E+1} - \bar{\boldsymbol{\theta}}^E)\right\|^2}_{\text{Reg}_1} + \underbrace{\frac{\lambda_2}{2}\left\|\mathcal{P}^{\perp}_{\widehat{R}_1}(\boldsymbol{\theta}^{E+1})\right\|^2}_{\text{Reg}_2}. \end{aligned} \tag{22}$$

This optimization is the empirical risk minimization on the target domain with two regularizing terms. Reg_1 penalizes the difference between $\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}^E$ on the subspace spanned by the column space of the learned representation $\widehat{R}_1$. This regularization encourages the learning dynamics to capture more content features approximately shared through all the environments. On the other hand, Reg_2 penalizes the weight in the directions that are perpendicular to the column space of $\widehat{R}_1$, which discourages the learning dynamics from capturing the environmental features. This regularization-based fine-tuning process is motivated by the biased regularization (Denevi et al., 2018, 2020) which has been widely applied to the theoretical analysis of transfer learning.

We are interested in the excess risk of the learned predictor $\boldsymbol{x} \mapsto \langle \boldsymbol{x}, \widehat{\boldsymbol{\theta}}^{E+1} \rangle$ on the target environment, *i.e.,* how much our learned model performs worse than the optimal model on the target task:

$$\mathrm{ER}(\widehat{\boldsymbol{\theta}}^{E+1}) = \frac{1}{2}\,\mathbb{E}_{(\boldsymbol{x},y)\sim\mu_{E+1}}[\langle \boldsymbol{x}, R^*\boldsymbol{\theta}^{*E+1} - \widehat{\boldsymbol{\theta}}^{E+1}\rangle^2]. \tag{23}$$

We often calculate the expected excess risk with respect to the meta distribution, *i.e.,* $\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}[\mathrm{ER}(\widehat{\boldsymbol{\theta}}^{E+1})]$.

B.2 Theoretical Analysis

Before stating the main theorem, we first make some statistical assumptions on the input data. For $e \in [E+1]$, we assume $\mathbb{E}_{\boldsymbol{x}\sim p_e}[\boldsymbol{x}] = \mathbf{0}$ and let $\Sigma_{X^e} = \mathbb{E}_{\boldsymbol{x}\sim p_e}[\boldsymbol{x}\boldsymbol{x}^\top]$. Note that a sample $\boldsymbol{x} \sim p_e$ can be generated from $\boldsymbol{x} = \Sigma_{X^e}^{1/2}\bar{\boldsymbol{x}}$ where $\bar{\boldsymbol{x}} \sim \bar{p}_e$ where $\mathbb{E}_{\boldsymbol{x}\sim\bar{p}_e}[\bar{\boldsymbol{x}}] = \mathbf{0}$ and $\mathbb{E}_{\boldsymbol{x}\sim\bar{p}_e}[\bar{\boldsymbol{x}}\bar{\boldsymbol{x}}^\top] = I_d$ (p_e is called the whitening of p_e). We make the following assumptions on the input distribution $p_1, \ldots, p_{E+1}$.

Assumption B.1 (Subgaussian input) *There exists $\rho > 0$ such that for $e \in [E+1]$, $\bar{\boldsymbol{x}} \sim \bar{p}_e$ is ρ^2-subgaussian.*

Assumption B.2 (Covariance dominance) *There exists $c > 0, c_2 > c_1 > 0$ and $\Sigma_X \succeq 0$ such that for $e \in [E]$, $c_1 \cdot \Sigma_X \preceq c \cdot \Sigma_{X^{E+1}} \preceq \Sigma_{X^e} \preceq c_2 \cdot \Sigma_X$.*

Assumption B.1 is a standard assumption in statistical learning to obtain probabilistic tail bounds used in the proof. It might be replaced with other moment or boundedness conditions if we use different tail bounds in the analysis.

Assumption B.2 says that every direction spanned by Σ_{E+1} should be spanned by Σ_e, $e \in [E]$ and the parameter c quantifies how "easy" it is for Σ_e to cover Σ_{E+1}. We remark that instead of having $c \cdot \Sigma_{X^{E+1}} \preceq \Sigma_{X^e}$ for all $e \in [E]$, as long as this holds for a constant fraction of $[E]$, our result is valid. We also assume that Σ_{X^e}'s are uniformly bounded by some PSD matrix Σ_X up to some constant, which facilitates our theoretical analysis.

Recall that we distinguish the content and environmental features based on the variation of their correlation with y. The following assumption guarantees the well-separation between these two features. Let $\Delta_\Lambda := \min(\Lambda_{22}) - \|\Lambda_{11}\|$ be the eigengap between environmental and content features.

Assumption B.3 (Well-speration)

$$\left\|\widehat{\Sigma}_{\widehat{\boldsymbol{\theta}}} - \Sigma_{\mathrm{RM}}\right\| \lesssim \Delta_\Lambda.$$

This assumption indicates the eigengap between the content feature space and environmental feature space in the covariance matrix Σ_{RM} is large enough so that the Davis-Kahan bound $\min_{O\in\mathcal{O}(k)} \left\|\widehat{R}_1 - R_1^* O\right\|$ is meaningful. One sufficient condition for the separation assumption is E and n_1 being sufficiently large: $\sqrt{\frac{d}{E}}\|\Lambda_{22}\| + \frac{\sigma^2}{n_1\lambda_1(\Sigma_X)} \lesssim \Delta_\Lambda$.

Assumption B.4 (Content features dominance)

$$\frac{\mathrm{Tr}\left(R_1^{*\top}\Sigma_X R_1^*\right)}{\mathrm{Tr}\left(R_2^{*\top}\Sigma_X R_2^*\right)} \gtrsim \frac{\|\Lambda_{22}\|}{\|\Lambda_{11}\|}.$$

Remark B.5 *In the liner case, equation 4 can be written as*

$$y^e = \boldsymbol{x}^\top R_1^* \boldsymbol{\theta}^{*e}_{[1:k]} + \boldsymbol{x}^\top R_2^* \boldsymbol{\theta}^{*e}_{[k+1:D]} + z.$$

Assumption B.4 gaurentees

$$\mathbb{E}_{\theta^{*e},(\boldsymbol{x},y)}\left|\boldsymbol{x}^\top R_1^* \boldsymbol{\theta}^{*e}_{[1:k]}\right| \gtrsim \mathbb{E}_{\theta^{*e},(\boldsymbol{x},y)}\left|\boldsymbol{x}^\top R_2^* \boldsymbol{\theta}^{*e}_{[k+1:D]}\right|.$$

The following theorem gives a high probability bound of the excess risk for linear representations on the target domain.

Theorem B.6 *Under Assumption B.1, B.2, B.3 and B.4, we further assume that $k \le d \le E$ and the sample size in source and target environments satisfies $n_1 \gtrsim \rho^4 d, n_2 \gtrsim \rho^4 k$ and $n_1 \gtrsim n_2$. Under the choice of λ_1 and λ_2 in equation 27, with probability $1 - o(1)$, the excess risk of the learned predictor $\boldsymbol{x} \mapsto \langle \boldsymbol{x}, \widehat{\boldsymbol{\theta}}^{E+1} \rangle$ in equation 22 is*

$$\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}[\mathrm{ER}(\widehat{\boldsymbol{\theta}}^{E+1})] \lesssim \frac{\sigma^2 \|\Lambda_{11}\| \operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^*\right)}{n_2} + \frac{\sigma^2 \|\Lambda_{22}\| \operatorname{Tr}\left(R_2^{*\top} \Sigma_X R_2^*\right)}{n_2} + \frac{\sigma \|\Lambda_{22}\| \|\boldsymbol{\theta}^*\|}{n_2(\min(\Lambda_{22}) - \|\Lambda_{11}\|))} \sqrt{\frac{d \operatorname{Tr}\left(R_2^{*\top} \Sigma_X R_2^*\right)}{n_2 E}} + \xi(\sigma, n_1, n_2, d, k, E)$$

where

$$\xi(\sigma, n_1, n_2, d, k, E) = \frac{\sigma^2 \|\Sigma_X\| \operatorname{Tr}(\Lambda_{11})}{n_2 E} + \frac{\sigma^4 \|\Sigma_X\| \operatorname{Tr}\left(R_1^{*\top} \Sigma_X^{-1} R_1^*\right)}{n_1 n_2 E} + \frac{\sigma}{n_2} \sqrt{\frac{\operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^*\right) \operatorname{Tr}(\Lambda_{11})}{n_2 E}} + \frac{\sigma^2}{n_2} \sqrt{\frac{\operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^*\right) \operatorname{Tr}\left(R_1^{*\top} \Sigma_X^{-1} R_1^*\right)}{n_1 n_2 E}}$$

is the lower order terms.

Remark B.7 *In our setting, traditional ridge regression will yield a $\sqrt{\frac{\sigma^2 \|\boldsymbol{\theta}^{*E+1}\|^2 \operatorname{Tr}(\Sigma_X)}{n_2}}$ (and $\left\|\boldsymbol{\theta}^{*E+1}\right\|^2$ is bounded by $\|\boldsymbol{\theta}^*\|^2 + \|\Lambda_{11}\| + \|\Lambda_{22}\|$) rate. In comparison, our rate not only achieves a fast rate, but also manages to 1) quickly eliminate the error caused by the variation in utilizing content and environmental features (reflected by the Λ_{11} and Λ_{22} in the first two terms), and 2) fully utilize the shared part learned jointly from all E environments (reflected by the third term that involves $\|\boldsymbol{\theta}^*\|$ and is lower-order).*

B.3 Proof of Theorem B.6

We first prove two claims on the covariance concentration for both source and target tasks.

Claim B.8 (covariance concentration of source tasks) *Suppose $n_1 \gtrsim \rho^4 d$. Then it holds with probability $1 - o(1)$ that for $e \in [E]$,*

$$0.9 \Sigma_{X^e} \preceq \frac{1}{n_1} X^{e\top} X^e \preceq 1.1 \Sigma_{X^e}.$$

Proof. For $e \in [E]$, we write $X^e = \bar{X}^e \Sigma_{X^e}^{1/2}$. Lemma C.2 gives

$$0.9 I_d \preceq \frac{1}{n_1} \bar{X}^{e\top} \bar{X}^e \preceq 1.1 I_d,$$

which implies that

$$0.9 \Sigma_{X^e} \preceq \frac{1}{n_2} \Sigma_{X^e}^{1/2} \bar{X}^{e\top} \bar{X}^e \Sigma_{X^e}^{1/2} \preceq 1.1 \Sigma_{X^e}.$$

Taking a union bound over all $e \in [E]$ finishes the proof. □

Claim B.9 (covariance concentration of target tasks) *Suppose $n_2 \gtrsim \rho^4 k$. Then for any $B \in \mathbb{R}^{d \times k}$, $k \le d$, independent of X^{E+1}, it holds with probability $1 - o(1)$ that*

$$0.9 B^\top \Sigma_{X^{E+1}} B \preceq \frac{1}{n_1} B^\top X^{E+1\top} X^{E+1} B \preceq 1.1 B^\top \Sigma_{X^{E+1}} B.$$

Proof. Let $X^{E+1} = \bar{X}^{E+1} \Sigma_{X^{E+1}}^{1/2}$. Let the SVD of $\Sigma_{X^{E+1}}^{1/2} B$ be $UDV^\top$ where $U \in \mathbb{R}^{d \times k}$, $D, V \in \mathbb{R}^{k \times k}$. Then it can be verified that the rows of $X^{E+1} U$ are k-dimensional i.i.d. ρ^2-subgaussian random vectors with zero mean and identity covariance. Then similarly to the proof of the source covariance concentration, Lemma C.2 together with some algebra operations finish the proof. □

Then we will analyze the expected excess risk with respect to the meta distribution

$$\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}[\mathrm{ER}(\widehat{\boldsymbol{\theta}}^{E+1})] = \frac{1}{2}\,\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\,\mathbb{E}_{(\boldsymbol{x},y)\sim\mu_{E+1}}[\langle \boldsymbol{x}, R^*\boldsymbol{\theta}^{*E+1} - \widehat{\boldsymbol{\theta}}^{E+1}\rangle^2]. \tag{24}$$

The learned parameter of the target task from equation 22 has the closed form

$$\begin{aligned}\widehat{\boldsymbol{\theta}}^{E+1} &= \left(\frac{1}{n_2}X^{E+1\top}X^{E+1} + \lambda_1\mathcal{P}_{\widehat{R}_1} + \lambda_2\mathcal{P}^{\perp}_{\widehat{R}_1}\right)^{-1}\left(\frac{1}{n_2}X^{E+1\top}\boldsymbol{y}^{E+1} + \lambda_1\mathcal{P}_{\widehat{R}_1}\bar{\boldsymbol{\theta}}^{E}\right)\\ &= C^{-1}_{n_2,\lambda_1,\lambda_2}\left(\frac{1}{n_2}X^{E+1\top}\boldsymbol{y}^{E+1} + \lambda_1\mathcal{P}_{\widehat{R}_1}\bar{\boldsymbol{\theta}}^{E}\right).\end{aligned}$$

where

$$C_{n_2,\lambda_1,\lambda_2} = \left(\frac{1}{n_2}X^{E+1\top}X^{E+1} + \lambda_1\mathcal{P}_{\widehat{R}_1} + \lambda_2\mathcal{P}^{\perp}_{\widehat{R}_1}\right).$$

Recall that for $e \in [E+1]$, the rotated meta distribution is

$$R^*\boldsymbol{\theta}^{e*} \overset{\text{i.i.d.}}{\sim} \mathcal{N}\left(R_1^*\boldsymbol{\theta}^*, \Sigma_{\text{RM}}\right), \quad \Sigma_{\text{RM}} = R_1^*\Lambda_{11}R_1^{*\top} + R_2^*\Lambda_{22}R_2^{*\top}.$$

Then it can be verified that

$$\begin{aligned}\widehat{\boldsymbol{\theta}}^{E+1} - R^*\boldsymbol{\theta}^{*E+1} &= C^{-1}_{n_2,\lambda_1,\lambda_2}\left(\frac{1}{n_2}X^{E+1\top}\boldsymbol{y}^{E+1} + \lambda_1\mathcal{P}_{\widehat{R}_1}\bar{\boldsymbol{\theta}}^{E}\right) - R^*\boldsymbol{\theta}^{*E+1}\\ &= C^{-1}_{n_2,\lambda_1,\lambda_2}\left(\frac{1}{n_2}X^{E+1\top}\left(X^{E+1}R^*\boldsymbol{\theta}^{*E+1} + \boldsymbol{z}_{n_2}\right) + \lambda_1\mathcal{P}_{\widehat{R}_1}\bar{\boldsymbol{\theta}}^{E}\right) - R^*\boldsymbol{\theta}^{*E+1}\\ &= \lambda_1 C^{-1}_{n_2,\lambda_1,\lambda_2}\mathcal{P}_{\widehat{R}_1}\left(\bar{\boldsymbol{\theta}}^{E} - R^*\boldsymbol{\theta}^{*E+1}\right) - \lambda_2 C^{-1}_{n_2,\lambda_1,\lambda_2}\mathcal{P}^{\perp}_{\widehat{R}_1}R^*\boldsymbol{\theta}^{*E+1}\\ &\quad + \frac{1}{n_2}C^{-1}_{n_2,\lambda_1,\lambda_2}X^{E+1\top}\boldsymbol{z}_{n_2}\end{aligned} \tag{25}$$

where in the second equality we use $\boldsymbol{y}^{E+1} = X^{E+1}R^*\boldsymbol{\theta}^{*E+1} + \boldsymbol{z}_{n_2}$. Plugging equation 25 into equation 24, the excess risk can be decomposed into 5 terms and we will bound each of them:

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}[\mathrm{ER}(\widehat{\boldsymbol{\theta}}^{E+1})] &= \frac{1}{2}\,\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\left(\widehat{\boldsymbol{\theta}}^{E+1} - R^*\boldsymbol{\theta}^{*E+1}\right)^{\top}\Sigma_{X^{E+1}}\left(\widehat{\boldsymbol{\theta}}^{E+1} - R^*\boldsymbol{\theta}^{*E+1}\right)\right]\\ &= \frac{1}{2}(T_1 + T_2 + T_3 + T_4 + T_5),\end{aligned} \tag{26}$$

where

$$\begin{aligned}T_1 &= \mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\lambda_1^2\left(\bar{\boldsymbol{\theta}}^{E} - \boldsymbol{\theta}^{*E+1}\right)^{\top}\mathcal{P}_{\widehat{R}_1}C^{-1}_{n_2,\lambda_1,\lambda_2}\Sigma_{X^{E+1}}C^{-1}_{n_2,\lambda_1,\lambda_2}\mathcal{P}_{\widehat{R}_1}\left(\bar{\boldsymbol{\theta}}^{E} - \boldsymbol{\theta}^{*E+1}\right)\right],\\ T_2 &= \mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\lambda_2^2\boldsymbol{\theta}^{*E+1\top}\mathcal{P}^{\perp}_{\widehat{R}_1}C^{-1}_{n_2,\lambda_1,\lambda_2}\Sigma_{X^{E+1}}C^{-1}_{n_2,\lambda_1,\lambda_2}\mathcal{P}^{\perp}_{\widehat{R}_1}\boldsymbol{\theta}^{*E+1}\right],\\ T_3 &= \frac{1}{n_2^2}\boldsymbol{z}_{n_2}^{\top}X^{E+1}C^{-1}_{n_2,\lambda_1,\lambda_2}\Sigma_{X^{E+1}}C^{-1}_{n_2,\lambda_1,\lambda_2}X^{E+1\top}\boldsymbol{z}_{n_2},\\ T_4 &= \mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\langle\lambda_1 C^{-1}_{n_2,\lambda_1,\lambda_2}\mathcal{P}_{\widehat{R}_1}\left(\bar{\boldsymbol{\theta}}^{E} - R^*\boldsymbol{\theta}^{*E+1}\right), \frac{1}{n_2}C^{-1}_{n_2,\lambda_1,\lambda_2}X^{E+1\top}\boldsymbol{z}_{n_2}\rangle\right],\\ T_5 &= \mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\langle-\lambda_2 C^{-1}_{n_2,\lambda_1,\lambda_2}\mathcal{P}^{\perp}_{\widehat{R}_1}R^*\boldsymbol{\theta}^{*E+1}, \frac{1}{n_2}C^{-1}_{n_2\lambda_1,\lambda_2}X^{E+1\top}\boldsymbol{z}_{n_2}\rangle\right].\end{aligned}$$

Estimation of T_1. Let $\boldsymbol{v} = R^* \boldsymbol{\theta}^{*E+1} - R_1^* \boldsymbol{\theta}^*$. We have $\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}[\boldsymbol{v}] = 0$ and $\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\boldsymbol{v}\boldsymbol{v}^\top\right] = \Sigma_{\mathrm{RM}}$. Then the first term T_1 can be written as

$$
\begin{aligned}
T_1 &= \mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\lambda_1^2 \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^* + \boldsymbol{v}\right)^\top \mathcal{P}_{\widehat{R}_1} C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_{X^{E+1}} C_{n_2,\lambda_1,\lambda_2}^{-1} \mathcal{P}_{\widehat{R}_1} \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^* + \boldsymbol{v}\right)\right] \\
&= \lambda_1^2 \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^*\right)^\top \mathcal{P}_{\widehat{R}_1} C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_{X^{E+1}} C_{n_2,\lambda_1,\lambda_2}^{-1} \mathcal{P}_{\widehat{R}_1} \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^*\right) \\
&+ \lambda_1^2 \mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\boldsymbol{v}^\top \mathcal{P}_{\widehat{R}_1} C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_{X^{E+1}} C_{n_2,\lambda_1,\lambda_2}^{-1} \mathcal{P}_{\widehat{R}_1} \boldsymbol{v}\right] \\
&= \lambda_1^2 \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^*\right)^\top \mathcal{P}_{\widehat{R}_1} C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_{X^{E+1}} C_{n_2,\lambda_1,\lambda_2}^{-1} \mathcal{P}_{\widehat{R}_1} \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^*\right) \\
&+ \operatorname{Tr}\left(\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\widehat{R}_1^\top \boldsymbol{v}\boldsymbol{v}^\top \widehat{R}_1\right] \lambda_1^2 \widehat{R}_1^\top C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_{X^{E+1}} C_{n_2,\lambda_1,\lambda_2}^{-1} \widehat{R}_1\right) \\
&\leq \underbrace{\left\|\lambda_1 \Sigma_X^{1/2} C_{n_2,\lambda_1,\lambda_2}^{-1} \widehat{R}_1 \widehat{R}_1^\top \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^*\right)\right\|^2}_{T_{1,1}} + \underbrace{\lambda_1^2 \operatorname{Tr}\left(\widehat{R}_1^\top C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_X C_{n_2,\lambda_1,\lambda_2}^{-1} \widehat{R}_1\right) \left\|\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\widehat{R}_1^\top \boldsymbol{v}\boldsymbol{v}^\top \widehat{R}_1\right]\right\|}_{T_{1,2}}.
\end{aligned}
$$

To bound $T_{1,1}$ we need the performance guarantee of $\bar{\boldsymbol{\theta}}^E$ on the direction of the content features, which is given by the following lemma.

Lemma B.10 (source task guarantee on the content feature space) *Under the conditions of Theorem B.6, with probability $1 - o(1)$,*

$$
\left\|\mathcal{P}_{\widehat{R}_1}\left(\bar{\boldsymbol{\theta}}^E - R_1 \boldsymbol{\theta}^*\right)\right\| \lesssim \sqrt{\frac{1}{E}\operatorname{Tr}(\Lambda_{11})} + \sqrt{\frac{\sigma^2 \operatorname{Tr}\left(R_1^{*\top} \Sigma_X^{-1} R_1^*\right)}{n_1 E}} + \zeta(n_1, E, d)
$$

and $\zeta(n_1, E, d)$ is the lower terms

$$
\zeta(n_1, E, d) = (\min(\Lambda_{22}) - \|\Lambda_{11}\|))^{-1} \left(\sqrt{\frac{d}{E}}\|\Lambda_{22}\| + \frac{\sigma^2}{n_1 \lambda_1(\Sigma_X)}\right)\left(\sqrt{\frac{\operatorname{Tr}(\Lambda_{11} + \Lambda_{22})}{E}} + \sqrt{\frac{\operatorname{Tr}(\Sigma^{-1})}{n_1 E}}\right)
$$

where $\lambda_i(\Sigma_X)$ denotes the i-th smallest eigenvalue of Σ_X.

Under the choice

$$
\lambda_1 = \lambda_2 = \frac{\lambda_1(\Sigma_X)\sigma}{\sqrt{n_2} - \sigma}, \tag{27}
$$

we have

$$
\begin{aligned}
T_{1,1} &\leq \lambda_1^2 \left\|\Sigma_X^{1/2} C_{n_2,\lambda_1,\lambda_2}^{-1} \widehat{R}_1\right\|^2 \left\|\mathcal{P}_{\widehat{R}_1}\left(\bar{\boldsymbol{\theta}}^E - R_1 \boldsymbol{\theta}^*\right)\right\|^2 \\
&\lesssim \lambda_1^2 \left\|\Sigma_X^{1/2} C_{n_2,\lambda_1,\lambda_2}^{-1} \widehat{R}_1\right\|^2 \left(\frac{\operatorname{Tr}(\Lambda_{11})}{E} + \frac{\sigma^2}{n_1 E}\operatorname{Tr}\left(R_1^{*\top} \Sigma_X^{-1} R_1^*\right)\right) \\
&\lesssim \frac{\sigma^2 \|\Sigma_X\|}{n_2}\left(\frac{\operatorname{Tr}(\Lambda_{11})}{E} + \frac{\sigma^2}{n_1 E}\operatorname{Tr}\left(R_1^{*\top} \Sigma_X^{-1} R_1^*\right)\right) \\
&= \frac{\sigma^2 \|\Sigma_X\| \operatorname{Tr}(\Lambda_{11})}{n_2 E} + \frac{\sigma^4 \|\Sigma_X\| \operatorname{Tr}\left(R_1^{*\top} \Sigma_X^{-1} R_1^*\right)}{n_1 n_2 E}.
\end{aligned}
$$

Note that $T_{1,1}$ is of order $O(\sigma^2 k / n_2 E + \sigma^4 k / n_1 n_2 E)$ which will be omitted as a lower order term.

$$
\begin{aligned}
T_{1,2} =& \lambda_1^2 \operatorname{Tr}\left(\widehat{R}_1^\top C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_X C_{n_2,\lambda_1,\lambda_2}^{-1} \widehat{R}_1\right) \left\|\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\widehat{R}_1^\top \boldsymbol{v}\boldsymbol{v}^\top \widehat{R}_1\right]\right\| \\
&\leq \frac{\sigma^2 \operatorname{Tr}\left(\widehat{R}_1^\top \Sigma_X \widehat{R}_1\right)}{n_2}\left\|\widehat{R}_1^\top R_1^* \Lambda_{11} R_1^{*\top} \widehat{R}_1 + \widehat{R}_1^\top R_2^* \Lambda_{22} R_2^{*\top} \widehat{R}_1\right\| \\
&\lesssim \frac{\sigma^2 \operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^*\right)}{n_2}\left[\left(1 - \left\|R_2^{*\top} \widehat{R}_1\right\|^2\right)\|\Lambda_{11}\|^2 + \left\|R_2^{*\top} \widehat{R}_1\right\|^2 \|\Lambda_{22}\|\right] \\
&\lesssim \frac{\sigma^2 \|\Lambda_{11}\| \operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^*\right)}{n_2}
\end{aligned}
$$

where in the second equality we use $\Sigma_{\mathrm{RM}} = R_1^* \Lambda_{11} R_1^{*\top} + R_2^* \Lambda_{22} R_2^{*\top}$, and in the third inequality we use Lemma B.11 that $\min_{O \in \mathcal{O}(k)} \left\| \widehat{R}_1 - R_1^* O \right\|^2 \asymp \left\| R_2^{*\top} \widehat{R}_1 \right\|^2 = O(d/E + \sigma^2/n_1^2)$. $T_{1,2}$ is of order $O(\sigma^2 k/n_2)$ which is one of the dominant terms in the main theorem.

Estimation of T_2. Similarly to the first term, the second term can be written as

$$\begin{aligned}
T_2 &= \mathbb{E}_{\boldsymbol{\theta}^{*E+1}} \left[\lambda_2^2 \boldsymbol{\theta}^{*E+1\top} \mathcal{P}_{\widehat{R}_1}^{\perp} C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_{X^{E+1}} C_{n_2,\lambda_1,\lambda_2}^{-1} \mathcal{P}_{\widehat{R}_1}^{\perp} \boldsymbol{\theta}^{*E+1} \right] \\
&\lesssim \operatorname{Tr}\left(\mathbb{E}_{\boldsymbol{\theta}^{*E+1}} \left[\widehat{R}_2^\top \boldsymbol{\theta}^{*E+1} \boldsymbol{\theta}^{*E+1\top} \widehat{R}_2 \right] \lambda_2^2 \widehat{R}_2^\top C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_X C_{n_2,\lambda_1,\lambda_2}^{-1} \widehat{R}_2 \right) \\
&\lesssim \lambda_2^2 \left(\left(1 - \left\| R_2^{*\top} \widehat{R}_1 \right\|^2 \right) \|\Lambda_{22}\| + \left\| R_2^{*\top} \widehat{R}_1 \right\|^2 \|\Lambda_{11}\| \right) \operatorname{Tr}\left(\widehat{R}_2^\top C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma C_{n_2,\lambda_1,\lambda_2}^{-1} \widehat{R}_2 \right)
\end{aligned}$$

Similarly to the second term of T_1,

$$T_2 \lesssim \frac{\sigma^2 \|\Lambda_{22}\| \operatorname{Tr}\left(R_2^{*\top} \Sigma_X R_2^* \right)}{n_2}.$$

Under Assumption B.4, one can verify that $T_2 \lesssim T_{1,2}$.

Estimation of T_3.

$$\begin{aligned}
T_3 &= \frac{1}{n_2^2} \boldsymbol{z}_{n_2}^\top X^{E+1} C_{n_2,\lambda_1,\lambda_2}^{-1} \Sigma_{X^{E+1}} C_{n_2,\lambda_1,\lambda_2}^{-1} X^{E+1\top} \boldsymbol{z}_{n_2} \\
&= \frac{1}{n_2^2} \left\| \Sigma_{X^{E+1}}^{1/2} C_{n_2,\lambda_1,\lambda_2}^{-1} X^{E+1\top} \boldsymbol{z}_{n_2} \right\|^2 \\
&\lesssim \frac{1}{n_2^2} \left\| \mathcal{P}_{R_1^*} \Sigma_X^{1/2} C_{n_2,\lambda_1,\lambda_2}^{-1} X^{E+1\top} \boldsymbol{z}_{n_2} \right\|^2 + \frac{1}{n_2^2} \left\| \mathcal{P}_{R_2^*} \Sigma_X^{1/2} C_{n_2,\lambda_1,\lambda_2}^{-1} X^{E+1\top} \boldsymbol{z}_{n_2} \right\|^2 \\
&\lesssim \frac{\sigma^2 \operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^* \right)}{n_2} + \frac{\sigma^2 \operatorname{Tr}\left(R_2^{*\top} \Sigma_X R_2^* \right)}{n_2} \\
&\lesssim \left(1 + \frac{\|\Lambda_{11}\|}{\|\Lambda_{22}\|} \right) \frac{\sigma^2 \operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^* \right)}{n_2}
\end{aligned}$$

where we use Assumption B.4 in the last step.

Estimation of T_4.

$$\begin{aligned}
T_4 &= \mathbb{E}_{\boldsymbol{\theta}^{*E+1}} \left[\langle \lambda_1 C_{n_2,\lambda_1,\lambda_2}^{-1} \mathcal{P}_{\widehat{R}_1} \left(\bar{\boldsymbol{\theta}}^E - R^* \boldsymbol{\theta}^{*E+1} \right), \frac{1}{n_2} C_{n_2,\lambda_1,\lambda_2}^{-1} X^{E+1\top} \boldsymbol{z}_{n_2} \rangle \right] \\
&= \langle \lambda_1 C_{n_2,\lambda_1,\lambda_2}^{-1} \mathcal{P}_{\widehat{R}_1} \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^* \right), \frac{1}{n_2} C_{n_2,\lambda_1,\lambda_2}^{-1} X^{E+1\top} \boldsymbol{z}_{n_2} \rangle \\
&= \frac{\lambda_1}{n_2} \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^* \right)^\top \mathcal{P}_{\widehat{R}_1} C_{n_2,\lambda_1,\lambda_2}^{-2} X^{E+1\top} \boldsymbol{z}_{n_2} \\
&\le \left\| \mathcal{P}_{\widehat{R}_1} \left(\bar{\boldsymbol{\theta}}^E - R_1^* \boldsymbol{\theta}^* \right) \right\| \left\| \frac{\lambda_1}{n_2} \mathcal{P}_{\widehat{R}_1} C_{n_2,\lambda_1,\lambda_2}^{-2} X^{E+1\top} \boldsymbol{z}_{n_2} \right\| \\
&\lesssim \left(\sqrt{\frac{1}{E} \operatorname{Tr}(\Lambda_{11})} + \sqrt{\frac{\sigma^2 \operatorname{Tr}\left(R_1^{*\top} \Sigma_X^{-1} R_1^* \right)}{n_1 E}} \right) \left(\frac{\lambda_1 \sigma}{\sqrt{n_2}} \sqrt{\operatorname{Tr}\left(\widehat{R}_1^\top C_{n_2,\lambda_1,\lambda_2}^{-2} \Sigma_X C_{n_2,\lambda_1,\lambda_2}^{-2} \widehat{R}_1 \right)} \right) \\
&\lesssim \frac{\sigma}{n_2} \sqrt{\frac{\operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^* \right)}{n_2}} \left(\sqrt{\frac{1}{E} \operatorname{Tr}(\Lambda_{11})} + \sqrt{\frac{\sigma^2 \operatorname{Tr}\left(R_1^{*\top} \Sigma_X^{-1} R_1^* \right)}{n_1 E}} \right) \\
&\lesssim \frac{\sigma}{n_2} \sqrt{\frac{\operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^* \right) \operatorname{Tr}(\Lambda_{11})}{n_2 E}} + \frac{\sigma^2}{n_2} \sqrt{\frac{\operatorname{Tr}\left(R_1^{*\top} \Sigma_X R_1^* \right) \operatorname{Tr}\left(R_1^{*\top} \Sigma_X^{-1} R_1^* \right)}{n_1 n_2 E}}
\end{aligned}$$

where in the second inequality we plug in our choice of λ_1 in equation 27.

Estimation of T_5. The estimation of T_5 is almost the same as that of T_4.

$$\begin{aligned}
T_5 &= \mathbb{E}_{\boldsymbol{\theta}^{*E+1}}\left[\langle -\lambda_2 C^{-1}_{n_2,\lambda_1,\lambda_2}\mathcal{P}^{\perp}_{\widehat{R}_1} R^*\boldsymbol{\theta}^{*E+1}, \frac{1}{n_2} C^{-1}_{n_2,\lambda_1,\lambda_2} X^{E+1\top}\boldsymbol{z}_{n_2}\rangle\right] \\
&= \langle -\lambda_2 C^{-1}_{n_2,\lambda_1,\lambda_2}\mathcal{P}^{\perp}_{\widehat{R}_1} R_1^*\boldsymbol{\theta}^{*}, \frac{1}{n_2} C^{-1}_{n_2,\lambda_1,\lambda_2} X^{E+1\top}\boldsymbol{z}_{n_2}\rangle \\
&\lesssim \left\|\mathcal{P}^{\perp}_{\widehat{R}_1} R_1^*\boldsymbol{\theta}^{*}\right\|\left\|\frac{\lambda_2}{n_2}\mathcal{P}_{\widehat{R}_1} C^{-2}_{n_2,\lambda_1,\lambda_2} X^{E+1\top}\boldsymbol{z}_{n_2}\right\| \\
&\lesssim \frac{\|\boldsymbol{\theta}^*\|}{(\min(\Lambda_{22}) - \|\Lambda_{11}\|))}\left(\sqrt{\frac{d}{E}}\|\Lambda_{22}\| + \frac{\sigma^2}{n_1\lambda_1(\Sigma_X)}\right)\frac{\sigma}{n_2}\sqrt{\frac{\operatorname{Tr}\big(R_2^{*\top}\Sigma_X R_2^*\big)}{n_2}} \\
&\lesssim \frac{\|\boldsymbol{\theta}^*\|}{(\min(\Lambda_{22}) - \|\Lambda_{11}\|))}\left(\frac{\sigma\|\Lambda_{22}\|}{n_2}\sqrt{\frac{d\operatorname{Tr}\big(R_2^{*\top}\Sigma_X R_2^*\big)}{n_2 E}} + \frac{\sigma^3\lambda_1^{-1}(\Sigma_X)}{n_1 n_2}\sqrt{\frac{\operatorname{Tr}\big(R_2^{*\top}\Sigma_X R_2^*\big)}{n_2}}\right) \\
&\lesssim \frac{\sigma\|\Lambda_{22}\|\|\boldsymbol{\theta}^*\|}{n_2(\min(\Lambda_{22}) - \|\Lambda_{11}\|))}\sqrt{\frac{d\operatorname{Tr}\big(R_2^{*\top}\Sigma_X R_2^*\big)}{n_2 E}}.
\end{aligned}$$

With $T_{1,2}, T_2, T_3, T_5$ being dominant terms and $T_{1,1}, T_4$ being the lower order terms, we obtain the final bound

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\theta}^{*E+1}}[\mathrm{ER}(\widehat{\boldsymbol{\theta}}^{E+1})] &= \frac{1}{2}(T_1 + T_2 + T_3 + T_4 + T_5) \\
&\lesssim \frac{\sigma^2\|\Lambda_{11}\|\operatorname{Tr}\left(R_1^{*\top}\Sigma_X R_1^*\right)}{n_2} + \frac{\sigma^2\|\Lambda_{22}\|\operatorname{Tr}\left(R_2^{*\top}\Sigma_X R_2^*\right)}{n_2} \\
&\quad + \frac{\sigma\|\Lambda_{22}\|\|\boldsymbol{\theta}^*\|}{n_2(\min(\Lambda_{22}) - \|\Lambda_{11}\|))}\sqrt{\frac{d\operatorname{Tr}\big(R_2^{*\top}\Sigma_X R_2^*\big)}{n_2 E}} + \xi(\sigma, n_1, n_2, d, k, E)
\end{aligned}$$

where

$$\begin{aligned}
\xi(\sigma, n_1, n_2, d, k, E) &= \frac{\sigma^2\|\Sigma_X\|\operatorname{Tr}(\Lambda_{11})}{n_2 E} + \frac{\sigma^4\|\Sigma_X\|\operatorname{Tr}\big(R_1^{*\top}\Sigma_X^{-1}R_1^*\big)}{n_1 n_2 E} + \frac{\sigma}{n_2}\sqrt{\frac{\operatorname{Tr}\big(R_1^{*\top}\Sigma_X R_1^*\big)\operatorname{Tr}(\Lambda_{11})}{n_2 E}} \\
&\quad + \frac{\sigma^2}{n_2}\sqrt{\frac{\operatorname{Tr}\big(R_1^{*\top}\Sigma_X R_1^*\big)\operatorname{Tr}\big(R_1^{*\top}\Sigma_X^{-1}R_1^*\big)}{n_1 n_2 E}}.
\end{aligned}$$

B.4 Proof of Lemma B.10

For $1 \le e \le E$, $\widehat{\boldsymbol{\theta}}^e$ is the OLS estimator:

$$\begin{aligned}
\widehat{\boldsymbol{\theta}}^e &= \left(X^{e\top}X^e\right)^{-1}X^{e\top}\boldsymbol{y}^e \\
&= \left(X^{e\top}X^e\right)^{-1}X^{e\top}\left(X^e\boldsymbol{\theta}^{*e} + \boldsymbol{z}^e\right) \\
&= \boldsymbol{\theta}^{*e} + \left(X^{e\top}X^e\right)^{-1}X^{e\top}\boldsymbol{z}^e.
\end{aligned}$$

Let $\widehat{O} \in \mathcal{O}(k)$ be such that

$$\widehat{O} = \operatorname*{argmin}_{O\in\mathcal{O}(k)}\left\|\widehat{R}_1 - R_1 O\right\|.$$

Then the ℓ_2 error can be decomposed as:

$$\begin{aligned}\left\|\widehat{R}_1^\top\left(\frac{1}{E}\sum_{e=1}^{E}\widehat{\boldsymbol{\theta}}^e - R_1^*\boldsymbol{\theta}^*\right)\right\| \le& \left\|R_1^{*\top}\left(\frac{1}{E}\sum_{e=1}^{E}R^*\boldsymbol{\theta}^{*e} - R_1\boldsymbol{\theta}^*\right)\right\| \\ &+ \left\|R_1^{*\top}\left(\frac{1}{E}\sum_{e=1}^{E}\left(X^{e\top}X^e\right)^{-1}X^{e\top}\boldsymbol{z}^e\right)\right\| \\ &+ \left\|\left(\widehat{R}_1^\top - \widehat{O}^\top R_1^{*\top}\right)\left(\frac{1}{E}\sum_{e=1}^{E}R^*\boldsymbol{\theta}^{*e} - R_1\boldsymbol{\theta}^*\right)\right\| \\ &+ \left\|\left(\widehat{R}_1^\top - \widehat{O}^\top R_1^{*\top}\right)\left(\frac{1}{E}\sum_{e=1}^{E}\left(X^{e\top}X^e\right)^{-1}X^{e\top}\boldsymbol{z}^e\right)\right\|.\end{aligned}$$

where we use the fact that ℓ_2 norm is orthogonal invariant. Note that

$$R_1^{*\top}\left(\frac{1}{E}\sum_{e=1}^{E}R^*\boldsymbol{\theta}^{*e} - R_1\boldsymbol{\theta}^*\right) \sim \mathcal{N}\left(\mathbf{0}, \frac{1}{E}\Lambda_{11}\right).$$

The Chernoff bound gives that with probability $1-o(1)$,

$$\left\|R_1^{*\top}\left(\frac{1}{E}\sum_{e=1}^{E}R^*\boldsymbol{\theta}^{*e} - R_1\boldsymbol{\theta}^*\right)\right\| \lesssim \sqrt{\frac{1}{E}\operatorname{Tr}(\Lambda_{11})}.$$

Note that

$$R_1^{*\top}\left(\frac{1}{E}\sum_{e=1}^{E}\left(X^{e\top}X^e\right)^{-1}X^{e\top}\boldsymbol{z}^e\right) \sim \mathcal{N}\left(\mathbf{0}, \frac{\sigma^2}{n_1E^2}\sum_{e=1}^{E}R_1^{*\top}\left(\frac{X^{e\top}X^e}{n_1}\right)^{-1}R_1^*\right).$$

The Chernoff bound gives that with probability $1-o(1)$,

$$\left\|R_1^{*\top}\left(\frac{1}{E}\sum_{e=1}^{E}\left(X^{e\top}X^e\right)^{-1}X^{e\top}\boldsymbol{z}^e\right)\right\| \lesssim \frac{\sigma}{\sqrt{n_1E}}\sqrt{\operatorname{Tr}\left(R_1^{*\top}\Sigma_X^{-1}R_1^*\right)}$$

The following lemma provides the estimation of $\left\|\widehat{R}_1 - R_1^*\widehat{O}\right\|$.

Lemma B.11 *Under the conditions in Theorem B.6, with probability* $1-o(1)$,

$$\left\|\widehat{R}_1 - R_1^*\widehat{O}\right\| \lesssim (\min(\Lambda_{22}) - \|\Lambda_{11}\|))^{-1}\left(\sqrt{\frac{d}{E}}\|\Lambda_{22}\| + \frac{\sigma^2}{n_1\lambda_1(\Sigma_X)}\right).$$

Then the remaining two terms are of lower order.

$$\begin{aligned}&\left\|\left(\widehat{R}_1^\top - \widehat{O}^\top R_1^\top\right)\left(\frac{1}{E}\sum_{e=1}^{E}\boldsymbol{\theta}^{*e} - R_1\boldsymbol{\theta}^*\right)\right\| \lesssim \left\|\widehat{R}_1 - R_1\widehat{O}\right\|\left\|\frac{1}{E}\sum_{e=1}^{E}\boldsymbol{\theta}^{*e} - R_1\boldsymbol{\theta}^*\right\| \\ &\lesssim (\min(\Lambda_{22}) - \|\Lambda_{11}\|))^{-1}\left(\sqrt{\frac{d}{E}}\|\Lambda_{22}\| + \frac{\sigma^2}{n_1\lambda_1(\Sigma_X)}\right)\sqrt{\frac{\operatorname{Tr}(\Lambda_{11}+\Lambda_{22})}{E}}.\end{aligned}$$

Putting the terms together, we get

$$\begin{aligned}&\left\|\left(\widehat{R}_1^\top - \widehat{O}^\top R_1^{*\top}\right)\left(\frac{1}{E}\sum_{e=1}^{E}\left(X^{e\top}X^e\right)^{-1}X^{e\top}\boldsymbol{z}^e\right)\right\| \lesssim \left\|\widehat{R}_1 - R_1\widehat{O}\right\|\left\|\frac{1}{E}\sum_{e=1}^{E}\left(X^{e\top}X^e\right)^{-1}X^{e\top}\boldsymbol{z}^e\right\| \\ &\lesssim (\min(\Lambda_{22}) - \|\Lambda_{11}\|))^{-1}\left(\sqrt{\frac{d}{E}}\|\Lambda_{22}\| + \frac{\sigma^2}{n_1\lambda_1(\Sigma_X)}\right)\sqrt{\frac{\operatorname{Tr}(\Sigma^{-1})}{n_1E}}.\end{aligned}$$

B.5 Proof of Lemma B.11

Let $\widehat{O} \in \mathcal{O}(k)$ be such that

$$\widehat{O} = \operatorname{argmin}_{O \in \mathcal{O}(k)} \left\| \widehat{R}_1 - R_1 O \right\|.$$

It can be verified that $\widehat{O} = \bar{U}\bar{V}^\top$ given the SVD of $R_1^{*\top}\widehat{R}_1$ being $\bar{U}\bar{D}\bar{V}^\top$. Then Davis-Kahan Theorem (Theorem C.1) gives the bound

$$\begin{aligned}
\left\| \widehat{R}_1 - R_1\widehat{O} \right\| &= \left\| \left(I_d - R_1^* R_1^{*\top}\right) \widehat{R}_1 \right\| + \left\| R_1^{*\top}\widehat{R}_1 - \widehat{O} \right\| \\
&\le 2 \left\| \sin\left(R_1, \widehat{R}_1\right) \right\| \\
&\le \frac{2\left\| \widehat{\Sigma}_{\widehat{\theta}} - \Sigma_{\mathrm{RM}} \right\|}{\min(\Lambda_{22}) - \|\Lambda_{11}\|) - \left\| \widehat{\Sigma}_{\widehat{\theta}} - \Sigma_{\mathrm{RM}} \right\|}.
\end{aligned}$$

Under Assumption B.3,

$$\left\| \widehat{R}_1 - R_1\widehat{O} \right\| \lesssim \frac{\left\| \widehat{\Sigma}_{\widehat{\theta}} - \Sigma_{\mathrm{RM}} \right\|}{\min(\Lambda_{22}) - \|\Lambda_{11}\|)}.$$

For readability of the proof, we define the auxiliary quantities as follows. Let

$$\begin{aligned}
\Delta^e &= \left(X^{e\top}X^e\right)^{-1} X^{e\top}\boldsymbol{z}^e \\
\bar{\boldsymbol{\theta}}^{*E} &= \frac{1}{E}\sum_{e=1}^{E} \boldsymbol{\theta}^{*e}, \\
\bar{\Delta}^E &= \frac{1}{E}\sum_{e=1}^{E} \Delta^e, \\
\widehat{\Sigma}_{\boldsymbol{\theta}^*} &= \frac{1}{E}\sum_{e=1}^{E} \left(\boldsymbol{\theta}^{*e} - \bar{\boldsymbol{\theta}}^{*E}\right)\left(\boldsymbol{\theta}^{*e} - \bar{\boldsymbol{\theta}}^{*E}\right)^\top, \\
\widehat{\Sigma}^0_{\boldsymbol{\theta}^*} &= \frac{1}{E}\sum_{e=1}^{E} \left(\boldsymbol{\theta}^{*e} - R_1^*\boldsymbol{\theta}^*\right)\left(\boldsymbol{\theta}^{*e} - R_1^*\boldsymbol{\theta}^*\right)^\top, \\
\widehat{\Sigma}_{\Delta} &= \frac{1}{E}\sum_{e=1}^{E} \left(\Delta^e - \bar{\Delta}^E\right)\left(\Delta^e - \bar{\Delta}^E\right)^\top, \\
\widehat{\Sigma}^0_{\Delta} &= \frac{1}{E}\sum_{e=1}^{E} \Delta^e\Delta^{e\top}.
\end{aligned}$$

Then $\left\| \widehat{\Sigma}_{\widehat{\theta}} - \Sigma_{\mathrm{RM}} \right\|$ can be decomposed as:

$$\begin{aligned}
\left\| \widehat{\Sigma}_{\widehat{\theta}} - \Sigma_{\mathrm{RM}} \right\| &= \left\| \frac{1}{E}\sum_{e=1}^{E} \left(\widehat{\boldsymbol{\theta}}^e - \bar{\boldsymbol{\theta}}^E\right)\left(\widehat{\boldsymbol{\theta}}^e - \bar{\boldsymbol{\theta}}^E\right)^\top - \Sigma_{\mathrm{RM}} \right\| \\
&= \left\| \frac{1}{E}\sum_{e=1}^{E} \left(\boldsymbol{\theta}^{*e} - \bar{\boldsymbol{\theta}}^{*E} + \Delta^e - \bar{\Delta}^E\right)\left(\boldsymbol{\theta}^{*e} - \bar{\boldsymbol{\theta}}^{*E} + \Delta^e - \bar{\Delta}^E\right)^\top - \Sigma_{\mathrm{RM}} \right\| \\
&= \left\| \widehat{\Sigma}_{\boldsymbol{\theta}^*} - \Sigma_{\mathrm{RM}} \right\| + \left\| \widehat{\Sigma}_{\Delta} \right\| \\
&\quad + \left\| \frac{1}{E}\sum_{e=1}^{E} \boldsymbol{\theta}^{*e}\Delta^{e\top} - \bar{\boldsymbol{\theta}}^{*E}\bar{\Delta}^{E\top} + \frac{1}{E}\sum_{e=1}^{E} \Delta^e\boldsymbol{\theta}^{*e\top} - \bar{\Delta}^E\bar{\boldsymbol{\theta}}^{*E\top} \right\|.
\end{aligned}$$

Estimation of $\left\|\widehat{\Sigma}_{\boldsymbol{\theta}^*} - \Sigma_{\mathrm{RM}}\right\|$. We have that

$$\left\|\widehat{\Sigma}_{\boldsymbol{\theta}^*} - \Sigma_{\mathrm{RM}}\right\| \leq \left\|\widehat{\Sigma}^0_{\boldsymbol{\theta}^*} - \Sigma_{\mathrm{RM}}\right\| + \left\|\left(\bar{\boldsymbol{\theta}}^{*E} - R_1^*\boldsymbol{\theta}^*\right)\left(\bar{\boldsymbol{\theta}}^{*E} - R_1^*\boldsymbol{\theta}^*\right)^\top\right\|.$$

Since $\boldsymbol{\theta}^e - R_1^*\boldsymbol{\theta}^*$ is centered Gaussian with covariance Σ_{RM}, by the standard covariance estimation in (Wainwright, 2019), with probability $1 - o(1)$,

$$\left\|\widehat{\Sigma}^0_{\boldsymbol{\theta}^*} - \Sigma_{\mathrm{RM}}\right\| \lesssim \sqrt{\frac{d}{E}}\|\Lambda_{22}\|.$$

The second term upper bounded by the squared norm of the Gaussian vector $\left\|\bar{\boldsymbol{\theta}}^{*E} - R_1^*\boldsymbol{\theta}^*\right\|$

$$\left\|\left(\bar{\boldsymbol{\theta}}^{*E} - R_1^*\boldsymbol{\theta}^*\right)\left(\bar{\boldsymbol{\theta}}^{*E} - R_1^*\boldsymbol{\theta}^*\right)^\top\right\| = \left\|\bar{\boldsymbol{\theta}}^{*E} - R_1^*\boldsymbol{\theta}^*\right\|^2 \lesssim \frac{\operatorname{Tr}\left(\Lambda_{11} + \Lambda_{22}\right)}{E}.$$

Estimation of $\left\|\widehat{\Sigma}_\Delta\right\|$. Similarly, Δ^e is a centered Gaussian with covariance $\sigma^2\Sigma_{X^e}^{-1}/n_1$. Then

$$\begin{aligned}\left\|\widehat{\Sigma}_\Delta\right\| &\leq \|\Sigma_\Delta\| + \left\|\Sigma_\Delta - \widehat{\Sigma^0_\Delta}\right\| + \left\|\bar{\Delta}^E\bar{\Delta}^{E\top}\right\| \\ &\lesssim \frac{\sigma^2}{n_1\lambda_1(\Sigma)}\left(1 + \sqrt{\frac{d}{E}}\right) + \frac{\sigma^2\operatorname{Tr}\left(\Sigma_X^{-1}\right)}{n_1 E}\end{aligned}$$

Estimation of $\left\|\frac{1}{E}\sum_{e=1}^E \boldsymbol{\theta}^{*e}\Delta^{e\top} - \bar{\boldsymbol{\theta}}^{*E}\bar{\Delta}^{E\top} + \frac{1}{E}\sum_{e=1}^E \Delta^e\boldsymbol{\theta}^{*e\top} - \bar{\Delta}^E\bar{\boldsymbol{\theta}}^{*E\top}\right\|$. Since the two parts of the sum are transpose of each other, it suffices to find the upper bound one of them only. Then we have that

$$\left\|\frac{1}{E}\sum_{e=1}^E \boldsymbol{\theta}^{*e}\Delta^{e\top} - \bar{\boldsymbol{\theta}}^{*E}\bar{\Delta}^{E\top}\right\| \leq \left\|\frac{1}{E}\sum_{e=1}^E(\boldsymbol{\theta}^{*e} - R_1^*\boldsymbol{\theta}^*)\Delta^{e\top}\right\| + \left\|(R_1^*\boldsymbol{\theta}^* - \bar{\boldsymbol{\theta}}^{*E})\bar{\Delta}^{E\top}\right\|$$

Chernoff bound gives the upper bound of the second term. With probability $1 - o(1)$,

$$\begin{aligned}\left\|(R_1^*\boldsymbol{\theta}^* - \bar{\boldsymbol{\theta}}^{*E})\bar{\Delta}^{E\top}\right\| &\leq \left\|R_1^*\boldsymbol{\theta}^* - \bar{\boldsymbol{\theta}}^{*E}\right\|\left\|\bar{\Delta}^E\right\| \\ &\lesssim \sqrt{\frac{\operatorname{Tr}(\Lambda_{11} + \Lambda_{22})}{E}\frac{\sigma^2\operatorname{Tr}\left(\Sigma_X^{-1}\right)}{n_1 E}}\end{aligned}$$

Then we will apply Matrix Bernstein (Theorem C.3) to find the upper bound of the first term. Note that each term is bounded by

$$\left\|\frac{1}{E}(\boldsymbol{\theta}^{*e} - R_1^*\boldsymbol{\theta}^*)\Delta^{e\top}\right\| \lesssim \frac{\sigma\sqrt{\operatorname{Tr}(\Lambda_{11} + \Lambda_{22})\operatorname{Tr}\left(\Sigma_X^{-1}\right)}}{\sqrt{n_1}E}.$$

Ther variance proxy is

$$\begin{aligned}\left\|\mathbb{E}\left[\sum_{e=1}^E \frac{1}{E^2}(\boldsymbol{\theta}^{*e} - R_1^*\boldsymbol{\theta}^*)\Delta^{e\top}\Delta^e(\boldsymbol{\theta}^{*e} - R_1^*\boldsymbol{\theta}^*)^\top\right]\right\| &\lesssim \frac{\mathbb{E}_{\boldsymbol{\theta}^*}\|\boldsymbol{\theta}^{*e} - R_1^*\boldsymbol{\theta}^*\|^2\|\Delta^e\|^2}{E} \\ &\lesssim \frac{\sigma^2\operatorname{Tr}(\Lambda_{11} + \Lambda_{22})\operatorname{Tr}\left(\Sigma_X^{-1}\right)}{n_1 E}.\end{aligned}$$

Then, by Bernstein's inequality,

$$\begin{aligned}\left\|\frac{1}{E}\sum_{e=1}^E(\boldsymbol{\theta}^{*e} - R_1^*\boldsymbol{\theta}^*)\Delta^{e\top}\right\| &\lesssim \sqrt{\frac{\sigma^2\operatorname{Tr}(\Lambda_{11} + \Lambda_{22})\operatorname{Tr}\left(\Sigma_X^{-1}\right)\log(E)}{n_1 E}} + \frac{\sigma\sqrt{\operatorname{Tr}(\Lambda_{11} + \Lambda_{22})\operatorname{Tr}\left(\Sigma_X^{-1}\right)}\log(E)}{\sqrt{n_1}E} \\ &\lesssim \sqrt{\frac{\sigma^2\operatorname{Tr}(\Lambda_{11} + \Lambda_{22})\operatorname{Tr}\left(\Sigma_X^{-1}\right)\log(E)}{n_1 E}}.\end{aligned}$$

Thus, we combine the terms together and omit the lower order terms:

$$\left\|\widehat{\Sigma}_{\widehat{\theta}} - \Sigma_{\mathrm{RM}}\right\| \lesssim \sqrt{\frac{d}{E}}\|\Lambda_{22}\| + \frac{\sigma^2}{n_1\lambda_1(\Sigma_X)}$$

and

$$\left\|\widehat{R}_1 - R_1^*\widehat{O}\right\| \lesssim \frac{\left\|\widehat{\Sigma}_{\widehat{\theta}} - \Sigma_{\mathrm{RM}}\right\|}{\min(\Lambda_{22}) - \|\Lambda_{11}\|)} \lesssim (\min(\Lambda_{22}) - \|\Lambda_{11}\|))^{-1}\left(\sqrt{\frac{d}{E}}\|\Lambda_{22}\| + \frac{\sigma^2}{n_1\lambda_1(\Sigma_X)}\right).$$

C Technical ingredients

Theorem C.1 (Generalized Davis-Kahan theorem (Deng et al., 2021; Samuel Zhong and Ling, 2024)) *Consider the eigenvalue problem $N^{-1}M\boldsymbol{u} = \lambda\boldsymbol{u}$ where M and N are both Hermitian, and N is positive definite. Let X be the matrix that has the eigenvectors of $N^{-1}M$ as columns. Then $N^{-1}M$ is diagonalizable and can be written as*

$$N^{-1}M = X\Lambda X^H = X_1\Lambda_1 X_1^H + X_2\Lambda_2 X_2^H$$

where

$$X^{-1} = \begin{bmatrix} X_1 & X_2 \end{bmatrix}^{-1} = \begin{bmatrix} Y_1^H \\ Y_2^H \end{bmatrix}, \quad \Lambda = \begin{bmatrix} \Lambda_1 & \\ & \Lambda_2 \end{bmatrix}.$$

Suppose $\delta = \min_i |(\Lambda_2)_{ii} - \widehat{\lambda}|$ is the absolute separation of $\widehat{\lambda}$ from $(\Lambda_2)_{ii}$, then for any vector $\widehat{\boldsymbol{u}}$ we have

$$\|P\widehat{\boldsymbol{u}}\| \le \frac{\sqrt{\kappa(N)}\left\|(N^{-1}M - \widehat{\lambda}I_n)\widehat{\boldsymbol{u}}\right\|}{\delta}.$$

where $P = (Y_2^\dagger)^H(Y_2)^H = I - (X_1^\dagger)^H(X_1)^H$ is the orthogonal projection matrix onto the orthogonal complement of the column space of X_1, $\kappa(N) = \|N\|\left\|N^{-1}\right\|$ is the condition number of N and $Y_2^\dagger$ is the Moore-Penrose inverse of Y_2.

When $N = I$ and $(\hat{\lambda}, \widehat{\boldsymbol{u}})$ be an eigen-pair of a matrix $\widehat{M}$, we have

$$\sin\theta \le \frac{\left\|(M - \widehat{M})\widehat{\boldsymbol{u}}\right\|}{\delta}$$

where θ is the canonical angle between $\widehat{\boldsymbol{u}}$ and the column space of X_1. In this case the theorem reduces to the classical Davis-Kahan theorem (Davis and Kahan, 1970).

Lemma C.2 ((Du et al., 2021), Lemma A.6) *Let $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ be i.i.d. $d-$dimensional random vectors such that $\mathbb{E}[\boldsymbol{a}_i] = 0$, $\mathbb{E}[\boldsymbol{a}_i\boldsymbol{a}_i^\top] = I_d$, and $\boldsymbol{a}_i$ is ρ^2-subgaussian. If $n \gtrsim \rho^4 d$, then it holds with probability $1 - o(1)$ that*

$$0.9I_d \preceq \frac{1}{n}\sum_{i=1}^n \boldsymbol{a}_i\boldsymbol{a}_i^\top \preceq 1.1I_d.$$

Theorem C.3 (Matrix Bernstein (Tropp, 2012)) *Consider a finite sequence of independent random matrices $\{Z_k\}$. Assume that each random matrix satisfies*

$$\mathbb{E}Z_k = 0, \quad \|Z_k\| \le R.$$

Then for all $t \ge 0$,

$$\mathbb{P}\left(\left\|\sum_k Z_k\right\| \ge t\right) \le (d_1 + d_2)\cdot\exp\left(-\frac{t^2/2}{\sigma^2 + Rt/3}\right).$$

where

$$\sigma^2 = \max\left\{\left\|\sum_k \mathbb{E}Z_k^\top Z_k\right\|, \left\|\sum_k \mathbb{E}Z_k Z_k^\top\right\|\right\}.$$

Then with probability at least $1 - n^{-\gamma+1}$,

$$\left\| \sum_k Z_k \right\| \leq \sqrt{2\gamma\sigma^2 \log(d_1 + d_2)} + \frac{2\gamma R \log(d_1 + d_2)}{3}.$$

Table 2: Dataset Statistics: the number of classes and samples in each dataset with different ratios

dataset	target_envs	source_train	source_val	target_val	source_test	target_test	target_finetune					
							0.1	0.2	0.4	0.6	0.8	1
pacs	0	6,356	793	204	794	205	163	327	655	983	1,311	1,639
	1	6,119	763	234	765	234	187	375	750	1,125	1,500	1,876
	2	6,659	830	167	832	167	133	267	534	801	1,068	1,336
	3	4,851	605	392	606	393	314	628	1,257	1,886	2,515	3,144
vlcs	0	6,386	797	141	798	142	113	226	452	679	905	1,132
	1	6,371	795	143	797	143	114	229	458	688	917	1,147
	2	4,980	621	317	623	317	253	507	1,015	1,522	2,030	2,538
	3	4,817	601	337	602	338	270	540	1,080	1,620	2,160	2,701
officehome	0	10,530	1,314	242	1,317	243	194	388	776	1,165	1,553	1,942
	1	8,980	1,120	436	1,123	437	349	698	1,396	2,095	2,793	3,492
	2	8,920	1,113	443	1,116	444	355	710	1,420	2,131	2,841	3,552
	3	8,986	1,121	435	1,124	436	348	697	1,394	2,091	2,788	3,486
terraincognita	0	15,672	1,958	474	1,959	474	379	758	1,517	2,275	3,034	3,793
	1	11,676	1,459	973	1,459	974	778	1,557	3,115	4,673	6,231	7,789
	2	16,289	2,035	397	2,036	397	317	635	1,270	1,905	2,540	3,176
	3	14,758	1,844	588	1,845	588	470	941	1,882	2,824	3,765	4,707

D Experimental Setup

D.1 Target Finetuning

In target finetuning, we initialize the classifier from source pretrained $\boldsymbol{\theta}$ over different feature parts: For example, PN-Y is the backbone with the classifier only using the target-specific feature. Hyperparameter tuning and model selection are based on performance on the target data validation set.

E Experimental Results

We show the full results in the following tables. B stands for the base feature, Y stands for the target-specific feature, S stands for the shared feature, and E stands for the environment-specific feature. We use the Adam optimizer with lr $\in$ [1e-5, 5e-5, 1e-4], batch size $\in$ [32, 64, 128]. We choose the best model based on the source validation accuracy. For the λ_1 and λ_2 hyperparameters, we just need to make sure it is not too small so the target is well-regularized and make the value comparable to the loss value, we use λ_1 = 1.0 and λ_2 = 1.0. We have different training hyperparameter spaces with DomainBed. The absolute performance may differ from the DomainBed benchmark, but the relative rank is consistent.

E.1 Linear Probing Results

We fixed the source validation dataset based on random seed and use that validation dataset for hyperparameter tuning. We vary the hyperparameters of the logistic regression models: C $\in$ [1e-5, 1e-4, 1e-3, 1e-2, 1e-1], lbfgs solver, max iter = 1000. We report the results in the following Table 3, Table 4, Table 5, Table 6.

E_t	Method	0.10	0.20	0.40	0.60	0.80	1.00	mean
0	DANN	0.4527 ± 0.0107	0.5984 ± 0.0109	0.6527 ± 0.0090	0.7111 ± 0.0054	0.7193 ± 0.0092	0.7177 ± 0.0087	0.6420 ± 0.0090
	DIWA	0.5835 ± 0.0132	0.6675 ± 0.0097	0.7276 ± 0.0094	0.7358 ± 0.0082	0.7720 ± 0.0118	0.7909 ± 0.0048	0.7129 ± 0.0095
	ERM	0.5745 ± 0.0105	0.6848 ± 0.0046	0.7243 ± 0.0107	0.7539 ± 0.0108	0.7827 ± 0.0082	0.7926 ± 0.0085	0.7188 ± 0.0089
	NUC-0.01	0.4840 ± 0.0129	0.6543 ± 0.0043	0.7407 ± 0.0097	0.7934 ± 0.0136	0.8049 ± 0.0040	0.8165 ± 0.0062	0.7156 ± 0.0084
	NUC-0.1	0.4733 ± 0.0072	0.6782 ± 0.0111	**0.7687 ± 0.0117**	**0.8049 ± 0.0038**	0.8132 ± 0.0062	0.8099 ± 0.0077	0.7247 ± 0.0080
	PN-B	0.5852 ± 0.0051	0.6733 ± 0.0115	0.7449 ± 0.0068	0.7613 ± 0.0065	**0.8165 ± 0.0025**	0.8074 ± 0.0053	0.7314 ± 0.0063
	PN-B-Y	0.5770 ± 0.0076	0.6955 ± 0.0050	0.7490 ± 0.0068	0.7761 ± 0.0053	0.8074 ± 0.0094	0.8329 ± 0.0074	0.7396 ± 0.0069
	PN-B-Y-S-E	0.5844 ± 0.0057	0.7119 ± 0.0114	0.7366 ± 0.0077	0.7580 ± 0.0054	0.8041 ± 0.0110	**0.8337 ± 0.0055**	0.7381 ± 0.0078
	PN-Y	0.5877 ± 0.0080	**0.7169 ± 0.0081**	0.7481 ± 0.0102	0.7588 ± 0.0083	0.8066 ± 0.0075	0.8280 ± 0.0126	0.7410 ± 0.0091
	PN-Y-S	0.5745 ± 0.0070	0.7062 ± 0.0095	0.7432 ± 0.0135	0.7860 ± 0.0135	0.8041 ± 0.0127	0.8222 ± 0.0080	0.7394 ± 0.0107
	PN-Y-S-E	**0.5901 ± 0.0098**	0.7029 ± 0.0105	0.7498 ± 0.0068	0.7778 ± 0.0066	0.8082 ± 0.0028	0.8296 ± 0.0094	**0.7431 ± 0.0077**
1	DANN	0.4842 ± 0.0128	0.5986 ± 0.0077	0.6856 ± 0.0078	0.7011 ± 0.0090	0.7263 ± 0.0047	0.7318 ± 0.0072	0.6546 ± 0.0082
	DIWA	0.5716 ± 0.0082	0.6577 ± 0.0101	0.7121 ± 0.0077	0.7487 ± 0.0067	0.7808 ± 0.0050	0.7725 ± 0.0082	0.7072 ± 0.0077
	ERM	0.5538 ± 0.0071	0.6490 ± 0.0065	0.6989 ± 0.0054	0.7204 ± 0.0034	0.7432 ± 0.0038	0.7803 ± 0.0089	0.6909 ± 0.0058
	NUC-0.01	0.4838 ± 0.0070	0.6293 ± 0.0049	0.6879 ± 0.0055	0.7414 ± 0.0075	0.7730 ± 0.0063	0.7867 ± 0.0053	0.6837 ± 0.0061
	NUC-0.1	0.4824 ± 0.0101	0.6449 ± 0.0066	0.6728 ± 0.0095	0.7382 ± 0.0087	0.7744 ± 0.0061	0.7794 ± 0.0067	0.6820 ± 0.0080
	PN-B	0.5542 ± 0.0082	0.6568 ± 0.0028	0.7281 ± 0.0062	0.7684 ± 0.0059	0.7812 ± 0.0063	0.7973 ± 0.0026	0.7143 ± 0.0053
	PN-B-Y	0.5593 ± 0.0092	0.6545 ± 0.0035	**0.7465 ± 0.0045**	0.7808 ± 0.0092	0.7844 ± 0.0053	0.8009 ± 0.0064	**0.7211 ± 0.0064**
	PN-B-Y-S-E	0.5712 ± 0.0066	0.6526 ± 0.0076	0.7217 ± 0.0060	0.7698 ± 0.0060	**0.7863 ± 0.0009**	**0.8064 ± 0.0044**	0.7180 ± 0.0052
	PN-Y	0.5584 ± 0.0052	0.6563 ± 0.0068	0.7245 ± 0.0070	**0.7817 ± 0.0058**	0.7826 ± 0.0128	0.8018 ± 0.0056	0.7175 ± 0.0072
	PN-Y-S	**0.5744 ± 0.0113**	0.6517 ± 0.0025	0.7263 ± 0.0046	0.7744 ± 0.0070	0.7826 ± 0.0071	0.7899 ± 0.0092	0.7166 ± 0.0070
	PN-Y-S-E	0.5707 ± 0.0078	**0.6613 ± 0.0035**	0.7359 ± 0.0047	0.7661 ± 0.0034	0.7808 ± 0.0085	0.7995 ± 0.0030	0.7191 ± 0.0052
2	DANN	0.6649 ± 0.0056	0.7423 ± 0.0018	0.8230 ± 0.0062	0.8509 ± 0.0058	0.8667 ± 0.0036	0.8784 ± 0.0049	0.8044 ± 0.0046
	DIWA	0.7734 ± 0.0082	0.8189 ± 0.0059	0.8649 ± 0.0049	0.8959 ± 0.0036	0.9063 ± 0.0052	0.9045 ± 0.0055	0.8607 ± 0.0055
	ERM	0.7743 ± 0.0069	0.8401 ± 0.0019	0.8716 ± 0.0048	0.9113 ± 0.0038	0.8995 ± 0.0039	0.9122 ± 0.0033	0.8682 ± 0.0041
	NUC-0.01	0.7176 ± 0.0030	0.8320 ± 0.0049	**0.9077 ± 0.0031**	0.9126 ± 0.0046	**0.9261 ± 0.0024**	**0.9302 ± 0.0028**	**0.8710 ± 0.0035**
	NUC-0.1	0.7257 ± 0.0085	0.8279 ± 0.0052	0.8977 ± 0.0052	**0.9198 ± 0.0032**	0.9212 ± 0.0053	0.9248 ± 0.0023	0.8695 ± 0.0050
	PN-B	0.7689 ± 0.0024	0.8365 ± 0.0037	0.8689 ± 0.0053	0.8919 ± 0.0026	0.9140 ± 0.0034	0.9203 ± 0.0038	0.8667 ± 0.0035
	PN-B-Y	0.7721 ± 0.0083	0.8351 ± 0.0044	0.8613 ± 0.0039	0.8955 ± 0.0044	0.9144 ± 0.0034	0.9117 ± 0.0039	0.8650 ± 0.0047
	PN-B-Y-S-E	**0.7797 ± 0.0038**	0.8459 ± 0.0051	0.8730 ± 0.0039	0.8914 ± 0.0049	0.9162 ± 0.0008	0.9185 ± 0.0041	0.8708 ± 0.0038
	PN-Y	0.7527 ± 0.0071	0.8446 ± 0.0054	0.8770 ± 0.0042	0.8923 ± 0.0046	0.8991 ± 0.0060	0.9234 ± 0.0052	0.8649 ± 0.0054
	PN-Y-S	0.7703 ± 0.0074	**0.8468 ± 0.0065**	0.8671 ± 0.0033	0.8955 ± 0.0034	0.9027 ± 0.0039	0.9176 ± 0.0021	0.8667 ± 0.0044
	PN-Y-S-E	0.7770 ± 0.0069	0.8455 ± 0.0090	0.8829 ± 0.0074	0.8959 ± 0.0053	0.9023 ± 0.0054	0.9225 ± 0.0027	0.8710 ± 0.0061
3	DANN	0.6385 ± 0.0043	0.7183 ± 0.0051	0.7518 ± 0.0067	0.7638 ± 0.0044	0.7853 ± 0.0067	0.7862 ± 0.0058	0.7407 ± 0.0055
	DIWA	0.7867 ± 0.0085	0.8000 ± 0.0034	0.8271 ± 0.0040	0.8408 ± 0.0044	0.8344 ± 0.0037	0.8560 ± 0.0073	0.8242 ± 0.0052
	ERM	**0.8248 ± 0.0050**	0.8367 ± 0.0053	0.8619 ± 0.0044	0.8555 ± 0.0079	0.8546 ± 0.0084	0.8555 ± 0.0054	0.8482 ± 0.0061
	NUC-0.01	0.7156 ± 0.0052	0.8147 ± 0.0085	0.8596 ± 0.0033	0.8670 ± 0.0036	0.8656 ± 0.0049	0.8812 ± 0.0112	0.8339 ± 0.0061
	NUC-0.1	0.7165 ± 0.0111	0.8165 ± 0.0058	0.8587 ± 0.0017	0.8679 ± 0.0049	0.8624 ± 0.0030	0.8771 ± 0.0078	0.8332 ± 0.0057
	PN-B	0.7894 ± 0.0077	0.8312 ± 0.0036	0.8628 ± 0.0029	0.8651 ± 0.0031	0.8693 ± 0.0062	0.8633 ± 0.0066	0.8469 ± 0.0050
	PN-B-Y	0.7940 ± 0.0058	0.8252 ± 0.0023	0.8615 ± 0.0006	0.8656 ± 0.0034	**0.8817 ± 0.0071**	0.8647 ± 0.0039	0.8488 ± 0.0038
	PN-B-Y-S-E	0.7872 ± 0.0066	0.8353 ± 0.0099	0.8587 ± 0.0034	0.8633 ± 0.0039	0.8683 ± 0.0064	0.8665 ± 0.0060	0.8466 ± 0.0060
	PN-Y	0.7986 ± 0.0037	0.8422 ± 0.0074	0.8495 ± 0.0051	0.8615 ± 0.0046	0.8688 ± 0.0022	0.8720 ± 0.0028	0.8488 ± 0.0043
	PN-Y-S	0.7812 ± 0.0038	0.8381 ± 0.0030	**0.8679 ± 0.0060**	0.8628 ± 0.0039	0.8693 ± 0.0061	**0.8835 ± 0.0028**	0.8505 ± 0.0043
	PN-Y-S-E	0.7904 ± 0.0051	**0.8454 ± 0.0051**	0.8661 ± 0.0039	**0.8706 ± 0.0048**	0.8679 ± 0.0039	0.8697 ± 0.0059	**0.8517 ± 0.0048**

Table 3: Linear Probing results on OfficeHome dataset.

E_t	Method	0.10	0.20	0.40	0.60	0.80	1.00	mean
0	DANN	0.2956 ± 0.0118	0.3337 ± 0.0150	0.3171 ± 0.0079	0.3698 ± 0.0179	0.4059 ± 0.0192	0.4215 ± 0.0124	0.3572 ± 0.0140
	DIWA	**0.9405 ± 0.0081**	**0.9346 ± 0.0033**	**0.9512 ± 0.0077**	**0.9551 ± 0.0036**	**0.9473 ± 0.0076**	**0.9444 ± 0.0065**	**0.9455 ± 0.0061**
	ERM	0.8702 ± 0.0043	0.8673 ± 0.0042	0.8849 ± 0.0068	0.9044 ± 0.0059	0.9141 ± 0.0072	0.9190 ± 0.0037	0.8933 ± 0.0053
	NUC-0.01	0.8429 ± 0.0052	0.8780 ± 0.0056	0.8937 ± 0.0036	0.9015 ± 0.0066	0.8966 ± 0.0070	0.8966 ± 0.0054	0.8849 ± 0.0056
	NUC-0.1	0.8273 ± 0.0070	0.8702 ± 0.0099	0.8839 ± 0.0081	0.9024 ± 0.0044	0.8976 ± 0.0083	0.8937 ± 0.0099	0.8792 ± 0.0079
	PN-B	0.8810 ± 0.0077	0.8878 ± 0.0074	0.9239 ± 0.0065	0.9190 ± 0.0059	0.9307 ± 0.0071	0.9288 ± 0.0065	0.9119 ± 0.0068
	PN-B-Y	0.8849 ± 0.0072	0.8868 ± 0.0076	0.9190 ± 0.0037	0.9180 ± 0.0039	0.9337 ± 0.0045	0.9346 ± 0.0075	0.9128 ± 0.0057
	PN-B-Y-S-E	0.9024 ± 0.0044	0.9054 ± 0.0043	0.9220 ± 0.0046	0.9337 ± 0.0077	0.9434 ± 0.0061	0.9405 ± 0.0047	0.9246 ± 0.0053
	PN-Y	0.9054 ± 0.0048	0.9005 ± 0.0092	0.9268 ± 0.0034	0.9317 ± 0.0053	0.9337 ± 0.0048	0.9268 ± 0.0031	0.9208 ± 0.0051
	PN-Y-S	0.9122 ± 0.0084	0.8917 ± 0.0068	0.9220 ± 0.0074	0.9327 ± 0.0039	0.9337 ± 0.0037	0.9220 ± 0.0051	0.9190 ± 0.0059
	PN-Y-S-E	0.9073 ± 0.0034	0.9024 ± 0.0067	0.9171 ± 0.0094	0.9239 ± 0.0067	0.9415 ± 0.0041	0.9210 ± 0.0086	0.9189 ± 0.0065
1	DANN	0.3419 ± 0.0137	0.3983 ± 0.0048	0.4256 ± 0.0048	0.4359 ± 0.0102	0.4538 ± 0.0148	0.4906 ± 0.0068	0.4244 ± 0.0092
	DIWA	0.8906 ± 0.0098	0.9154 ± 0.0067	0.9231 ± 0.0072	0.9402 ± 0.0045	0.9402 ± 0.0038	0.9342 ± 0.0089	0.9239 ± 0.0068
	ERM	0.8752 ± 0.0114	0.9111 ± 0.0065	0.9231 ± 0.0062	0.9316 ± 0.0076	0.9470 ± 0.0037	**0.9581 ± 0.0086**	0.9244 ± 0.0074
	NUC-0.01	0.8436 ± 0.0083	0.8581 ± 0.0075	0.8897 ± 0.0044	0.8974 ± 0.0045	0.8906 ± 0.0042	0.9179 ± 0.0048	0.8829 ± 0.0056
	NUC-0.1	0.8308 ± 0.0046	0.8624 ± 0.0044	0.8923 ± 0.0076	0.8949 ± 0.0086	0.9034 ± 0.0068	0.9145 ± 0.0030	0.8830 ± 0.0058
	PN-B	0.9034 ± 0.0064	0.9205 ± 0.0064	0.9308 ± 0.0067	0.9256 ± 0.0026	0.9265 ± 0.0028	0.9436 ± 0.0046	0.9251 ± 0.0049
	PN-B-Y	**0.9222 ± 0.0056**	0.9162 ± 0.0064	0.9179 ± 0.0095	0.9171 ± 0.0032	0.9350 ± 0.0031	0.9359 ± 0.0076	0.9241 ± 0.0059
	PN-B-Y-S-E	0.9128 ± 0.0032	0.9248 ± 0.0072	**0.9342 ± 0.0052**	0.9359 ± 0.0041	**0.9487 ± 0.0027**	0.9487 ± 0.0043	0.9342 ± 0.0044
	PN-Y	0.9068 ± 0.0104	0.9282 ± 0.0048	0.9231 ± 0.0052	0.9359 ± 0.0019	0.9308 ± 0.0078	0.9487 ± 0.0014	0.9289 ± 0.0052
	PN-Y-S	0.9162 ± 0.0053	**0.9342 ± 0.0017**	0.9265 ± 0.0046	**0.9427 ± 0.0048**	0.9368 ± 0.0046	0.9556 ± 0.0052	**0.9353 ± 0.0044**
	PN-Y-S-E	0.9077 ± 0.0074	0.9342 ± 0.0032	0.9291 ± 0.0040	0.9342 ± 0.0066	0.9479 ± 0.0039	0.9513 ± 0.0055	0.9340 ± 0.0051
2	DANN	0.6431 ± 0.0062	0.6311 ± 0.0140	0.6862 ± 0.0122	0.6862 ± 0.0082	0.7186 ± 0.0143	0.7138 ± 0.0088	0.6798 ± 0.0106
	DIWA	0.9737 ± 0.0031	0.9737 ± 0.0015	0.9820 ± 0.0033	0.9880 ± 0.0000	0.9796 ± 0.0031	0.9844 ± 0.0031	0.9802 ± 0.0023
	ERM	0.9533 ± 0.0035	0.9569 ± 0.0035	0.9653 ± 0.0040	0.9677 ± 0.0056	0.9677 ± 0.0052	0.9593 ± 0.0035	0.9617 ± 0.0042
	NUC-0.01	0.9569 ± 0.0064	0.9737 ± 0.0045	0.9749 ± 0.0055	**0.9916 ± 0.0031**	0.9808 ± 0.0022	0.9880 ± 0.0019	0.9776 ± 0.0039
	NUC-0.1	0.9545 ± 0.0024	0.9760 ± 0.0019	0.9784 ± 0.0049	0.9868 ± 0.0022	0.9808 ± 0.0035	**0.9904 ± 0.0024**	0.9778 ± 0.0029
	PN-B	0.9701 ± 0.0060	0.9796 ± 0.0036	**0.9796 ± 0.0041**	0.9808 ± 0.0048	0.9737 ± 0.0031	0.9784 ± 0.0045	0.9770 ± 0.0043
	PN-B-Y	0.9784 ± 0.0059	0.9820 ± 0.0054	**0.9856 ± 0.0031**	0.9832 ± 0.0029	0.9772 ± 0.0048	0.9820 ± 0.0050	0.9814 ± 0.0045
	PN-B-Y-S-E	**0.9796 ± 0.0041**	0.9772 ± 0.0048	0.9808 ± 0.0029	0.9796 ± 0.0031	0.9784 ± 0.0052	0.9880 ± 0.0019	0.9806 ± 0.0037
	PN-Y	0.9749 ± 0.0029	**0.9856 ± 0.0031**	0.9832 ± 0.0040	0.9760 ± 0.0033	**0.9892 ± 0.0012**	0.9808 ± 0.0022	**0.9816 ± 0.0028**
	PN-Y-S	0.9725 ± 0.0024	0.9820 ± 0.0050	0.9784 ± 0.0062	0.9737 ± 0.0045	0.9784 ± 0.0024	0.9880 ± 0.0019	0.9788 ± 0.0037
	PN-Y-S-E	0.9641 ± 0.0066	0.9760 ± 0.0063	0.9820 ± 0.0038	**0.9904 ± 0.0024**	0.9784 ± 0.0015	0.9832 ± 0.0040	0.9790 ± 0.0041
3	DANN	0.6132 ± 0.0038	0.6626 ± 0.0069	0.7008 ± 0.0071	0.6962 ± 0.0120	0.7298 ± 0.0049	0.7394 ± 0.0091	0.6903 ± 0.0073
	DIWA	0.8718 ± 0.0055	**0.8896 ± 0.0042**	**0.9170 ± 0.0032**	0.9074 ± 0.0034	0.9115 ± 0.0053	0.9160 ± 0.0018	0.9022 ± 0.0039
	ERM	**0.8936 ± 0.0039**	0.8891 ± 0.0030	0.9033 ± 0.0052	**0.9206 ± 0.0057**	**0.9277 ± 0.0046**	**0.9232 ± 0.0041**	**0.9096 ± 0.0044**
	NUC-0.01	0.6972 ± 0.0055	0.7659 ± 0.0091	0.8081 ± 0.0034	0.8036 ± 0.0061	0.8193 ± 0.0023	0.8249 ± 0.0060	0.7865 ± 0.0054
	NUC-0.1	0.6987 ± 0.0042	0.7603 ± 0.0056	0.7969 ± 0.0103	0.8107 ± 0.0059	0.8249 ± 0.0064	0.8275 ± 0.0051	0.7865 ± 0.0063
	PN-B	0.8427 ± 0.0066	0.8718 ± 0.0097	0.8840 ± 0.0040	0.8967 ± 0.0039	0.9033 ± 0.0039	0.9013 ± 0.0061	0.8833 ± 0.0057
	PN-B-Y	0.8601 ± 0.0040	0.8636 ± 0.0026	0.8947 ± 0.0049	0.9018 ± 0.0013	0.8901 ± 0.0041	0.9069 ± 0.0061	0.8862 ± 0.0038
	PN-B-Y-S-E	0.8570 ± 0.0033	0.8616 ± 0.0057	0.8941 ± 0.0030	0.8987 ± 0.0035	0.9074 ± 0.0061	0.9018 ± 0.0055	0.8868 ± 0.0045
	PN-Y	0.8595 ± 0.0033	0.8585 ± 0.0035	0.8926 ± 0.0029	0.8997 ± 0.0030	0.9125 ± 0.0024	0.9120 ± 0.0039	0.8891 ± 0.0032
	PN-Y-S	0.8534 ± 0.0068	0.8555 ± 0.0026	0.8906 ± 0.0053	0.8911 ± 0.0027	0.9038 ± 0.0066	0.8947 ± 0.0057	0.8815 ± 0.0049
	PN-Y-S-E	0.8631 ± 0.0047	0.8784 ± 0.0057	0.8835 ± 0.0049	0.8962 ± 0.0045	0.9013 ± 0.0025	0.8997 ± 0.0051	0.8870 ± 0.0046

Table 4: Linear Probing results on PACS dataset.

E_t	Method	0.10	0.20	0.40	0.60	0.80	1.00	mean
0	DANN	0.9620 ± 0.0042	0.9761 ± 0.0065	0.9831 ± 0.0053	0.9859 ± 0.0039	0.9930 ± 0.0022	0.9930 ± 0.0031	0.9822 ± 0.0042
	DIWA	0.9873 ± 0.0052	0.9930 ± 0.0039	0.9972 ± 0.0028	**1.0000 ± 0.0000**	0.9887 ± 0.0053	**1.0000 ± 0.0000**	0.9944 ± 0.0029
	ERM	0.9901 ± 0.0036	0.9873 ± 0.0034	0.9873 ± 0.0056	0.9775 ± 0.0068	0.9831 ± 0.0036	0.9972 ± 0.0017	0.9871 ± 0.0041
	NUC-0.01	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	0.9944 ± 0.0056	0.9991 ± 0.0009
	NUC-0.1	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	PN-B	0.9887 ± 0.0036	0.9930 ± 0.0022	0.9944 ± 0.0026	0.9972 ± 0.0017	0.9944 ± 0.0026	0.9986 ± 0.0014	0.9944 ± 0.0024
	PN-B-Y	0.9901 ± 0.0017	0.9944 ± 0.0014	0.9930 ± 0.0022	0.9972 ± 0.0017	0.9972 ± 0.0017	0.9958 ± 0.0017	0.9946 ± 0.0018
	PN-B-Y-S-E	0.9958 ± 0.0017	0.9944 ± 0.0026	0.9958 ± 0.0017	0.9944 ± 0.0014	0.9901 ± 0.0028	0.9972 ± 0.0017	0.9946 ± 0.0020
	PN-Y	0.9958 ± 0.0028	0.9915 ± 0.0041	0.9915 ± 0.0026	0.9901 ± 0.0017	0.9986 ± 0.0014	0.9972 ± 0.0017	0.9941 ± 0.0024
	PN-Y-S	0.9901 ± 0.0036	0.9845 ± 0.0041	0.9944 ± 0.0026	0.9930 ± 0.0000	0.9986 ± 0.0014	0.9915 ± 0.0026	0.9920 ± 0.0024
	PN-Y-S-E	0.9944 ± 0.0026	0.9972 ± 0.0017	0.9972 ± 0.0017	0.9873 ± 0.0014	0.9915 ± 0.0026	0.9972 ± 0.0017	0.9941 ± 0.0020
1	DANN	0.5429 ± 0.0039	0.5526 ± 0.0034	0.6293 ± 0.0076	0.6391 ± 0.0160	0.6429 ± 0.0081	0.6368 ± 0.0089	0.6073 ± 0.0080
	DIWA	0.7414 ± 0.0034	0.7504 ± 0.0102	0.7722 ± 0.0057	**0.7729 ± 0.0070**	**0.7805 ± 0.0069**	**0.7992 ± 0.0096**	0.7694 ± 0.0071
	ERM	**0.7519 ± 0.0083**	0.7263 ± 0.0088	0.7549 ± 0.0082	0.7586 ± 0.0101	0.7707 ± 0.0090	0.7609 ± 0.0096	0.7539 ± 0.0090
	NUC-0.01	0.7331 ± 0.0117	0.7677 ± 0.0088	0.7692 ± 0.0094	0.7722 ± 0.0070	0.7602 ± 0.0038	0.7654 ± 0.0104	0.7613 ± 0.0085
	NUC-0.1	0.7331 ± 0.0092	0.7451 ± 0.0150	0.7654 ± 0.0128	0.7609 ± 0.0054	0.7549 ± 0.0060	0.7737 ± 0.0102	0.7555 ± 0.0098
	PN-B	0.7113 ± 0.0066	0.7331 ± 0.0087	0.7669 ± 0.0110	0.7662 ± 0.0038	0.7421 ± 0.0044	0.7519 ± 0.0067	0.7452 ± 0.0069
	PN-B-Y	0.7083 ± 0.0066	0.7617 ± 0.0059	0.7699 ± 0.0084	0.7729 ± 0.0076	0.7571 ± 0.0104	0.7707 ± 0.0072	0.7568 ± 0.0077
	PN-B-Y-S-E	0.7248 ± 0.0069	0.7511 ± 0.0048	0.7767 ± 0.0035	0.7564 ± 0.0099	0.7481 ± 0.0128	0.7609 ± 0.0081	0.7530 ± 0.0077
	PN-Y	0.7346 ± 0.0050	**0.7797 ± 0.0130**	**0.7992 ± 0.0116**	0.7677 ± 0.0129	0.7519 ± 0.0059	0.7842 ± 0.0102	**0.7695 ± 0.0098**
	PN-Y-S	0.7203 ± 0.0059	0.7782 ± 0.0098	0.7714 ± 0.0077	0.7669 ± 0.0102	0.7466 ± 0.0112	0.7534 ± 0.0061	0.7561 ± 0.0085
	PN-Y-S-E	0.7256 ± 0.0086	0.7774 ± 0.0129	0.7774 ± 0.0076	0.7571 ± 0.0137	0.7654 ± 0.0165	0.7511 ± 0.0069	0.7590 ± 0.0110
2	DANN	0.7927 ± 0.0080	0.8012 ± 0.0114	0.7817 ± 0.0060	0.7976 ± 0.0056	0.7854 ± 0.0053	0.8091 ± 0.0126	0.7946 ± 0.0081
	DIWA	**0.8201 ± 0.0075**	0.8073 ± 0.0068	0.8189 ± 0.0039	0.8146 ± 0.0077	0.8049 ± 0.0066	0.8104 ± 0.0080	0.8127 ± 0.0067
	ERM	0.8104 ± 0.0077	**0.8268 ± 0.0046**	0.8293 ± 0.0048	0.8152 ± 0.0105	**0.8299 ± 0.0078**	0.8311 ± 0.0124	0.8238 ± 0.0080
	NUC-0.01	0.8067 ± 0.0100	0.8213 ± 0.0050	0.8262 ± 0.0043	**0.8329 ± 0.0079**	0.8140 ± 0.0027	0.8189 ± 0.0079	0.8200 ± 0.0063
	NUC-0.1	0.8006 ± 0.0072	0.8262 ± 0.0061	**0.8354 ± 0.0042**	0.8274 ± 0.0091	0.8220 ± 0.0058	**0.8341 ± 0.0059**	**0.8243 ± 0.0064**
	PN-B	0.7945 ± 0.0067	0.8030 ± 0.0055	0.8049 ± 0.0093	0.8152 ± 0.0033	0.8128 ± 0.0059	0.8073 ± 0.0047	0.8063 ± 0.0059
	PN-B-Y	0.7823 ± 0.0100	0.7848 ± 0.0101	0.7915 ± 0.0030	0.8171 ± 0.0057	0.8165 ± 0.0054	0.8104 ± 0.0095	0.8004 ± 0.0073
	PN-B-Y-S-E	0.8073 ± 0.0095	0.7970 ± 0.0051	0.7957 ± 0.0041	0.8067 ± 0.0058	0.7963 ± 0.0050	0.8165 ± 0.0113	0.8033 ± 0.0068
	PN-Y	0.7982 ± 0.0118	0.7933 ± 0.0049	0.8073 ± 0.0053	0.8183 ± 0.0053	0.8079 ± 0.0042	0.8055 ± 0.0089	0.8051 ± 0.0067
	PN-Y-S	0.7963 ± 0.0064	0.7866 ± 0.0059	0.8067 ± 0.0059	0.8030 ± 0.0062	0.8110 ± 0.0058	0.8293 ± 0.0042	0.8055 ± 0.0057
	PN-Y-S-E	0.7957 ± 0.0062	0.7939 ± 0.0104	0.8024 ± 0.0090	0.8073 ± 0.0095	0.8213 ± 0.0074	0.8244 ± 0.0101	0.8075 ± 0.0088
3	DANN	0.5018 ± 0.0121	0.5302 ± 0.0022	0.5846 ± 0.0061	0.5840 ± 0.0012	0.5976 ± 0.0075	0.5716 ± 0.0037	0.5616 ± 0.0055
	DIWA	0.8491 ± 0.0095	0.8728 ± 0.0047	0.8550 ± 0.0028	0.8651 ± 0.0039	0.8550 ± 0.0028	0.8580 ± 0.0048	0.8592 ± 0.0047
	ERM	0.8604 ± 0.0069	0.8728 ± 0.0129	0.8722 ± 0.0036	0.8663 ± 0.0077	0.8787 ± 0.0066	0.8722 ± 0.0049	0.8704 ± 0.0071
	NUC-0.01	0.8456 ± 0.0045	0.8645 ± 0.0042	0.8817 ± 0.0021	0.8828 ± 0.0093	0.8846 ± 0.0099	0.8716 ± 0.0074	0.8718 ± 0.0062
	NUC-0.1	0.8467 ± 0.0071	0.8686 ± 0.0050	0.8757 ± 0.0039	**0.8864 ± 0.0064**	**0.8947 ± 0.0047**	0.8751 ± 0.0068	**0.8746 ± 0.0056**
	PN-B	0.8609 ± 0.0047	0.8556 ± 0.0067	0.8740 ± 0.0050	0.8763 ± 0.0071	0.8657 ± 0.0061	0.8609 ± 0.0078	0.8656 ± 0.0062
	PN-B-Y	0.8621 ± 0.0052	0.8556 ± 0.0044	0.8722 ± 0.0065	0.8675 ± 0.0076	0.8651 ± 0.0123	0.8734 ± 0.0029	0.8660 ± 0.0065
	PN-B-Y-S-E	**0.8728 ± 0.0039**	0.8550 ± 0.0049	0.8574 ± 0.0055	0.8627 ± 0.0049	0.8680 ± 0.0058	0.8651 ± 0.0061	0.8635 ± 0.0052
	PN-Y	0.8521 ± 0.0064	**0.8781 ± 0.0054**	**0.8911 ± 0.0025**	0.8598 ± 0.0050	0.8722 ± 0.0044	**0.8769 ± 0.0056**	0.8717 ± 0.0049
	PN-Y-S	0.8669 ± 0.0085	0.8686 ± 0.0050	0.8710 ± 0.0091	0.8621 ± 0.0060	0.8805 ± 0.0070	0.8669 ± 0.0068	0.8693 ± 0.0071
	PN-Y-S-E	0.8633 ± 0.0037	0.8692 ± 0.0041	0.8692 ± 0.0090	0.8615 ± 0.0059	0.8633 ± 0.0081	0.8746 ± 0.0030	0.8669 ± 0.0057

Table 5: Linear Probing results on VLCS dataset.

E_t	Method	0.10	0.20	0.40	0.60	0.80	1.00	mean
0	DANN	0.6823 ± 0.0044	0.7198 ± 0.0051	0.7684 ± 0.0049	0.7827 ± 0.0032	0.7852 ± 0.0052	0.7886 ± 0.0087	0.7545 ± 0.0053
	DIWA	0.8249 ± 0.0058	0.8734 ± 0.0037	0.9055 ± 0.0062	0.9291 ± 0.0025	0.9422 ± 0.0076	0.9418 ± 0.0020	0.9028 ± 0.0046
	ERM	0.8346 ± 0.0055	0.8616 ± 0.0064	0.9084 ± 0.0028	0.9329 ± 0.0043	0.9367 ± 0.0032	0.9397 ± 0.0024	0.9023 ± 0.0041
	NUC-0.01	0.7928 ± 0.0052	0.8329 ± 0.0056	0.8730 ± 0.0059	0.9105 ± 0.0021	0.9219 ± 0.0020	0.9291 ± 0.0037	0.8767 ± 0.0041
	NUC-0.1	0.7975 ± 0.0110	0.8409 ± 0.0054	0.8624 ± 0.0039	0.9110 ± 0.0051	0.9139 ± 0.0036	0.9295 ± 0.0044	0.8759 ± 0.0055
	PN-B	0.8278 ± 0.0058	**0.8844 ± 0.0045**	**0.9211 ± 0.0011**	0.9405 ± 0.0018	**0.9578 ± 0.0009**	0.9603 ± 0.0044	0.9153 ± 0.0031
	PN-B-Y	0.8253 ± 0.0049	0.8700 ± 0.0049	0.9135 ± 0.0049	0.9473 ± 0.0033	0.9502 ± 0.0023	0.9603 ± 0.0035	0.9111 ± 0.0040
	PN-B-Y-S-E	0.8295 ± 0.0042	0.8819 ± 0.0024	0.9181 ± 0.0037	0.9405 ± 0.0020	0.9532 ± 0.0018	0.9553 ± 0.0017	0.9131 ± 0.0026
	PN-Y	**0.8392 ± 0.0041**	0.8819 ± 0.0035	0.9131 ± 0.0012	0.9540 ± 0.0026	0.9519 ± 0.0026	**0.9608 ± 0.0032**	0.9168 ± 0.0029
	PN-Y-S	0.8287 ± 0.0044	0.8747 ± 0.0051	0.9165 ± 0.0011	0.9502 ± 0.0029	0.9506 ± 0.0041	0.9578 ± 0.0019	0.9131 ± 0.0032
	PN-Y-S-E	0.8300 ± 0.0032	0.8844 ± 0.0041	0.9190 ± 0.0061	**0.9586 ± 0.0047**	0.9515 ± 0.0012	0.9586 ± 0.0029	**0.9170 ± 0.0037**
1	DANN	0.6875 ± 0.0026	0.7294 ± 0.0037	0.7760 ± 0.0028	0.7690 ± 0.0067	0.7727 ± 0.0052	0.7729 ± 0.0038	0.7512 ± 0.0041
	DIWA	**0.8405 ± 0.0027**	0.8692 ± 0.0027	0.8893 ± 0.0016	0.8955 ± 0.0047	0.9181 ± 0.0018	0.9281 ± 0.0026	0.8901 ± 0.0027
	ERM	0.8351 ± 0.0041	0.8643 ± 0.0016	0.8916 ± 0.0016	0.9064 ± 0.0022	**0.9341 ± 0.0026**	0.9300 ± 0.0022	0.8936 ± 0.0024
	NUC-0.01	0.7947 ± 0.0033	0.8246 ± 0.0038	0.8561 ± 0.0030	0.8836 ± 0.0021	0.8949 ± 0.0030	0.8998 ± 0.0016	0.8589 ± 0.0028
	NUC-0.1	0.7951 ± 0.0040	0.8366 ± 0.0059	0.8690 ± 0.0039	0.8852 ± 0.0045	0.8869 ± 0.0033	0.9008 ± 0.0024	0.8623 ± 0.0040
	PN-B	0.8310 ± 0.0061	**0.8713 ± 0.0044**	**0.9074 ± 0.0013**	0.9117 ± 0.0019	0.9220 ± 0.0021	0.9347 ± 0.0030	**0.8963 ± 0.0031**
	PN-B-Y	0.8302 ± 0.0066	0.8622 ± 0.0026	0.8996 ± 0.0037	0.9140 ± 0.0043	0.9300 ± 0.0016	**0.9398 ± 0.0036**	0.8960 ± 0.0037
	PN-B-Y-S-E	0.8267 ± 0.0050	0.8637 ± 0.0025	0.8922 ± 0.0029	0.9156 ± 0.0031	0.9296 ± 0.0047	0.9368 ± 0.0015	0.8941 ± 0.0033
	PN-Y	0.8248 ± 0.0036	0.8550 ± 0.0047	0.8895 ± 0.0021	**0.9179 ± 0.0040**	0.9290 ± 0.0011	0.9326 ± 0.0037	0.8915 ± 0.0032
	PN-Y-S	0.8197 ± 0.0053	0.8544 ± 0.0031	0.8926 ± 0.0015	0.9117 ± 0.0037	0.9257 ± 0.0023	0.9304 ± 0.0026	0.8891 ± 0.0031
	PN-Y-S-E	0.8230 ± 0.0028	0.8706 ± 0.0021	0.8834 ± 0.0033	0.9166 ± 0.0055	0.9324 ± 0.0020	0.9347 ± 0.0011	0.8935 ± 0.0028
2	DANN	0.5275 ± 0.0066	0.6045 ± 0.0061	0.6453 ± 0.0065	0.6403 ± 0.0081	0.6665 ± 0.0056	0.6690 ± 0.0043	0.6255 ± 0.0062
	DIWA	0.7264 ± 0.0102	0.7909 ± 0.0046	0.8040 ± 0.0039	0.8458 ± 0.0062	0.8499 ± 0.0091	0.8872 ± 0.0045	0.8174 ± 0.0064
	ERM	0.7330 ± 0.0112	0.7864 ± 0.0084	0.8317 ± 0.0043	0.8640 ± 0.0044	0.8831 ± 0.0055	0.8912 ± 0.0029	0.8316 ± 0.0061
	NUC-0.01	0.6448 ± 0.0048	0.7380 ± 0.0064	0.7889 ± 0.0045	0.8388 ± 0.0066	0.8589 ± 0.0071	0.8615 ± 0.0048	0.7885 ± 0.0057
	NUC-0.1	0.6534 ± 0.0067	0.7345 ± 0.0101	0.7950 ± 0.0103	0.8368 ± 0.0049	0.8615 ± 0.0063	0.8680 ± 0.0030	0.7915 ± 0.0069
	PN-B	**0.7521 ± 0.0081**	**0.7945 ± 0.0082**	0.8247 ± 0.0059	0.8443 ± 0.0048	0.8675 ± 0.0017	0.8811 ± 0.0036	0.8274 ± 0.0054
	PN-B-Y	0.7436 ± 0.0040	0.7884 ± 0.0069	0.8327 ± 0.0068	0.8630 ± 0.0083	0.8670 ± 0.0066	0.8922 ± 0.0054	0.8312 ± 0.0063
	PN-B-Y-S-E	0.7345 ± 0.0063	0.7904 ± 0.0058	0.8277 ± 0.0027	**0.8670 ± 0.0063**	0.8856 ± 0.0047	0.8992 ± 0.0047	0.8341 ± 0.0051
	PN-Y	0.7295 ± 0.0044	0.7748 ± 0.0089	0.8322 ± 0.0045	0.8620 ± 0.0059	0.8831 ± 0.0033	**0.9103 ± 0.0029**	0.8320 ± 0.0050
	PN-Y-S	0.7325 ± 0.0073	0.7904 ± 0.0079	**0.8368 ± 0.0106**	0.8660 ± 0.0037	0.8836 ± 0.0075	0.9008 ± 0.0023	**0.8350 ± 0.0065**
	PN-Y-S-E	0.7451 ± 0.0070	0.7788 ± 0.0039	0.8287 ± 0.0046	0.8610 ± 0.0084	**0.8932 ± 0.0060**	0.8957 ± 0.0044	0.8338 ± 0.0057
3	DANN	0.4942 ± 0.0087	0.5143 ± 0.0048	0.5014 ± 0.0034	0.5099 ± 0.0098	0.5218 ± 0.0020	0.5296 ± 0.0069	0.5118 ± 0.0059
	DIWA	0.7156 ± 0.0030	0.7340 ± 0.0052	0.8116 ± 0.0023	0.8255 ± 0.0030	0.8616 ± 0.0036	0.8810 ± 0.0030	0.8049 ± 0.0034
	ERM	0.7367 ± 0.0067	0.7554 ± 0.0093	0.8119 ± 0.0053	0.8265 ± 0.0045	0.8527 ± 0.0055	0.8779 ± 0.0010	0.8102 ± 0.0054
	NUC-0.01	0.6442 ± 0.0070	0.7061 ± 0.0072	0.7687 ± 0.0062	0.8122 ± 0.0052	0.8337 ± 0.0036	0.8456 ± 0.0033	0.7684 ± 0.0054
	NUC-0.1	0.6633 ± 0.0061	0.7139 ± 0.0055	0.7759 ± 0.0069	0.8027 ± 0.0046	0.8340 ± 0.0025	0.8442 ± 0.0053	0.7723 ± 0.0051
	PN-B	0.7310 ± 0.0068	0.7735 ± 0.0060	0.8221 ± 0.0049	0.8398 ± 0.0062	0.8660 ± 0.0029	0.8810 ± 0.0074	0.8189 ± 0.0057
	PN-B-Y	0.7241 ± 0.0069	0.7701 ± 0.0024	**0.8265 ± 0.0038**	0.8507 ± 0.0054	0.8724 ± 0.0058	0.8827 ± 0.0041	0.8211 ± 0.0047
	PN-B-Y-S-E	**0.7446 ± 0.0054**	0.7721 ± 0.0037	0.8173 ± 0.0032	**0.8520 ± 0.0031**	**0.8861 ± 0.0061**	0.8857 ± 0.0056	**0.8263 ± 0.0045**
	PN-Y	0.7357 ± 0.0037	0.7748 ± 0.0042	0.8126 ± 0.0065	0.8500 ± 0.0037	0.8680 ± 0.0046	0.8847 ± 0.0041	0.8210 ± 0.0045
	PN-Y-S	0.7391 ± 0.0058	0.7568 ± 0.0030	0.8184 ± 0.0070	0.8463 ± 0.0067	0.8694 ± 0.0082	0.8816 ± 0.0013	0.8186 ± 0.0053
	PN-Y-S-E	0.7432 ± 0.0031	**0.7823 ± 0.0065**	0.7980 ± 0.0028	0.8432 ± 0.0010	0.8782 ± 0.0049	**0.8898 ± 0.0032**	0.8224 ± 0.0036

Table 6: Linear Probing results on Terrain-Cognita dataset.

E.2 Target Finetuning Results

We use the source pretrained feature encoder and classifier as the initialization for the target finetuning. We use the same hyperparameter range as the source pretraining. The best model is choosen based on the target validation accuracy. We report the results in the following Table 7, Table 8, Table 9, Table 10.

E_t	Method	0.10	0.20	0.40	0.60	0.80	1.00	mean
0	DANN	0.6263 ± 0.0091	0.6650 ± 0.0098	0.7251 ± 0.0062	0.7588 ± 0.0051	0.7786 ± 0.0035	0.7984 ± 0.0045	0.7254 ± 0.0064
	DIWA	0.6329 ± 0.0423	0.6872 ± 0.0194	0.7366 ± 0.0069	0.7514 ± 0.0075	0.7630 ± 0.0119	0.8148 ± 0.0054	0.7310 ± 0.0156
	ERM	0.7407 ± 0.0079	0.7786 ± 0.0054	0.8000 ± 0.0042	0.8272 ± 0.0034	0.8362 ± 0.0040	0.8593 ± 0.0033	0.8070 ± 0.0047
	NUC-0.01	0.4535 ± 0.0069	0.6049 ± 0.0125	0.7103 ± 0.0065	0.7646 ± 0.0066	0.7770 ± 0.0067	0.8148 ± 0.0072	0.6875 ± 0.0077
	NUC-0.1	0.4461 ± 0.0051	0.6058 ± 0.0129	0.7029 ± 0.0073	0.7556 ± 0.0067	0.7852 ± 0.0067	0.8173 ± 0.0108	0.6855 ± 0.0083
	PN-Y	0.7588 ± 0.0100	0.7893 ± 0.0086	0.8082 ± 0.0059	**0.8305 ± 0.0030**	**0.8519 ± 0.0037**	**0.8675 ± 0.0038**	0.8177 ± 0.0058
	PN-Y-S	**0.7621 ± 0.0035**	**0.7926 ± 0.0059**	**0.8239 ± 0.0049**	0.8247 ± 0.0050	0.8502 ± 0.0046	0.8568 ± 0.0015	**0.8184 ± 0.0043**
	PN-Y-S-E	0.7605 ± 0.0082	0.7819 ± 0.0062	0.8140 ± 0.0072	0.8296 ± 0.0053	0.8510 ± 0.0044	0.8601 ± 0.0041	0.8162 ± 0.0059
1	DANN	0.4641 ± 0.0030	0.6105 ± 0.0124	0.7382 ± 0.0036	0.7922 ± 0.0048	0.8110 ± 0.0079	0.8352 ± 0.0049	0.7085 ± 0.0061
	DIWA	0.6229 ± 0.0045	**0.7121 ± 0.0170**	0.7510 ± 0.0160	0.8018 ± 0.0122	0.8110 ± 0.0114	0.8384 ± 0.0063	0.7562 ± 0.0113
	ERM	**0.6526 ± 0.0100**	0.7062 ± 0.0036	**0.7629 ± 0.0072**	0.7931 ± 0.0095	0.8220 ± 0.0073	0.8279 ± 0.0039	0.7608 ± 0.0069
	NUC-0.01	0.4288 ± 0.0057	0.5922 ± 0.0107	0.7231 ± 0.0064	0.7844 ± 0.0085	0.8165 ± 0.0018	**0.8426 ± 0.0063**	0.6979 ± 0.0066
	NUC-0.1	0.4279 ± 0.0051	0.5931 ± 0.0103	0.7199 ± 0.0067	0.7945 ± 0.0039	0.8151 ± 0.0081	0.8375 ± 0.0069	0.6980 ± 0.0068
	PN-Y	0.6526 ± 0.0028	0.7053 ± 0.0100	0.7616 ± 0.0041	**0.8087 ± 0.0043**	**0.8247 ± 0.0006**	0.8412 ± 0.0020	**0.7657 ± 0.0040**
	PN-Y-S	0.6293 ± 0.0051	0.6783 ± 0.0107	0.7442 ± 0.0033	0.7858 ± 0.0047	0.8014 ± 0.0051	0.8183 ± 0.0020	0.7429 ± 0.0052
	PN-Y-S-E	0.6302 ± 0.0062	0.6870 ± 0.0089	0.7368 ± 0.0067	0.7785 ± 0.0043	0.8046 ± 0.0028	0.8183 ± 0.0025	0.7426 ± 0.0052
2	DANN	0.7770 ± 0.0091	0.8450 ± 0.0053	0.8986 ± 0.0050	0.9099 ± 0.0043	0.9252 ± 0.0033	0.9365 ± 0.0024	0.8821 ± 0.0049
	DIWA	0.7856 ± 0.0131	0.8568 ± 0.0081	0.8878 ± 0.0066	0.9072 ± 0.0058	0.9207 ± 0.0086	0.9275 ± 0.0061	0.8809 ± 0.0080
	ERM	**0.8572 ± 0.0201**	**0.8851 ± 0.0150**	**0.9068 ± 0.0082**	**0.9221 ± 0.0074**	**0.9342 ± 0.0011**	0.9360 ± 0.0039	**0.9069 ± 0.0093**
	NUC-0.01	0.6923 ± 0.0082	0.8059 ± 0.0063	0.8820 ± 0.0031	0.9113 ± 0.0057	0.9324 ± 0.0042	0.9315 ± 0.0009	0.8592 ± 0.0047
	NUC-0.1	0.6995 ± 0.0058	0.8054 ± 0.0062	0.8838 ± 0.0036	0.9122 ± 0.0038	0.9342 ± 0.0037	0.9356 ± 0.0023	0.8618 ± 0.0042
	PN-Y	0.8266 ± 0.0124	0.8563 ± 0.0108	0.9023 ± 0.0049	0.9149 ± 0.0029	0.9324 ± 0.0031	**0.9419 ± 0.0030**	0.8957 ± 0.0062
	PN-Y-S	0.8176 ± 0.0117	0.8437 ± 0.0080	0.8919 ± 0.0064	0.8991 ± 0.0034	0.9207 ± 0.0028	0.9279 ± 0.0020	0.8835 ± 0.0057
	PN-Y-S-E	0.8288 ± 0.0205	0.8455 ± 0.0054	0.8842 ± 0.0031	0.9063 ± 0.0034	0.9185 ± 0.0039	0.9270 ± 0.0034	0.8851 ± 0.0066
3	DANN	0.7440 ± 0.0097	0.8128 ± 0.0080	0.8339 ± 0.0034	0.8477 ± 0.0039	0.8463 ± 0.0036	0.8537 ± 0.0017	0.8231 ± 0.0050
	DIWA	0.7972 ± 0.0120	0.8179 ± 0.0106	0.8399 ± 0.0090	0.8151 ± 0.0147	0.8349 ± 0.0143	0.8339 ± 0.0095	0.8232 ± 0.0117
	ERM	0.8335 ± 0.0067	0.8541 ± 0.0033	0.8596 ± 0.0020	0.8578 ± 0.0025	0.8624 ± 0.0065	0.8638 ± 0.0030	0.8552 ± 0.0040
	NUC-0.01	0.6445 ± 0.0087	0.7583 ± 0.0058	0.8266 ± 0.0037	0.8381 ± 0.0080	0.8495 ± 0.0064	0.8596 ± 0.0051	0.7961 ± 0.0063
	NUC-0.1	0.6450 ± 0.0098	0.7592 ± 0.0032	0.8174 ± 0.0041	0.8408 ± 0.0059	0.8431 ± 0.0084	0.8472 ± 0.0083	0.7921 ± 0.0066
	PN-Y	0.8564 ± 0.0065	0.8651 ± 0.0048	0.8780 ± 0.0045	0.8789 ± 0.0011	0.8798 ± 0.0021	0.8743 ± 0.0031	0.8721 ± 0.0037
	PN-Y-S	**0.8596 ± 0.0038**	**0.8697 ± 0.0052**	**0.8803 ± 0.0032**	0.8771 ± 0.0039	**0.8849 ± 0.0028**	0.8775 ± 0.0041	**0.8748 ± 0.0038**
	PN-Y-S-E	0.8537 ± 0.0071	0.8647 ± 0.0031	0.8771 ± 0.0037	**0.8803 ± 0.0034**	0.8826 ± 0.0029	**0.8817 ± 0.0041**	0.8733 ± 0.0041

Table 7: Target Finetuning results on OfficeHome dataset.

E_t	Method	0.10	0.20	0.40	0.60	0.80	1.00	mean
0	DANN	0.7132 ± 0.0216	0.8166 ± 0.0138	0.8644 ± 0.0079	0.8946 ± 0.0113	0.9063 ± 0.0036	0.9112 ± 0.0064	0.8511 ± 0.0108
	DIWA	**0.9454 ± 0.0061**	**0.9463 ± 0.0067**	**0.9541 ± 0.0040**	0.9580 ± 0.0057	0.9571 ± 0.0066	0.9649 ± 0.0056	0.9543 ± 0.0058
	ERM	0.9190 ± 0.0025	0.9356 ± 0.0036	0.9483 ± 0.0025	0.9620 ± 0.0028	0.9590 ± 0.0043	0.9512 ± 0.0034	0.9459 ± 0.0032
	NUC-0.01	0.8878 ± 0.0072	0.9102 ± 0.0045	0.9307 ± 0.0066	0.9483 ± 0.0025	0.9571 ± 0.0028	0.9541 ± 0.0037	0.9314 ± 0.0046
	NUC-0.1	0.8820 ± 0.0062	0.9141 ± 0.0045	0.9395 ± 0.0080	0.9502 ± 0.0024	0.9473 ± 0.0056	0.9551 ± 0.0036	0.9314 ± 0.0051
	PN-Y	0.9395 ± 0.0025	0.9463 ± 0.0038	0.9541 ± 0.0020	0.9600 ± 0.0032	0.9639 ± 0.0043	0.9698 ± 0.0028	**0.9556 ± 0.0031**
	PN-Y-S	0.9298 ± 0.0045	0.9415 ± 0.0034	0.9532 ± 0.0033	**0.9639 ± 0.0029**	**0.9659 ± 0.0031**	**0.9707 ± 0.0027**	0.9541 ± 0.0033
	PN-Y-S-E	0.9366 ± 0.0034	0.9346 ± 0.0072	0.9493 ± 0.0020	0.9590 ± 0.0040	0.9600 ± 0.0036	0.9620 ± 0.0024	0.9502 ± 0.0038
1	DANN	0.7812 ± 0.0120	0.8573 ± 0.0156	0.9043 ± 0.0118	0.9299 ± 0.0072	0.9419 ± 0.0074	0.9513 ± 0.0035	0.8943 ± 0.0096
	DIWA	**0.9342 ± 0.0058**	**0.9453 ± 0.0053**	**0.9667 ± 0.0025**	**0.9701 ± 0.0030**	**0.9735 ± 0.0016**	**0.9726 ± 0.0022**	**0.9604 ± 0.0034**
	ERM	0.8940 ± 0.0041	0.9051 ± 0.0031	0.9350 ± 0.0041	0.9453 ± 0.0034	0.9538 ± 0.0028	0.9538 ± 0.0021	0.9312 ± 0.0033
	NUC-0.01	0.8299 ± 0.0133	0.8846 ± 0.0081	0.9179 ± 0.0028	0.9274 ± 0.0056	0.9453 ± 0.0031	0.9538 ± 0.0055	0.9098 ± 0.0064
	NUC-0.1	0.8291 ± 0.0158	0.8829 ± 0.0068	0.9179 ± 0.0048	0.9376 ± 0.0064	0.9402 ± 0.0049	0.9547 ± 0.0052	0.9104 ± 0.0073
	PN-Y	0.9197 ± 0.0051	0.9325 ± 0.0096	0.9530 ± 0.0014	0.9590 ± 0.0010	0.9632 ± 0.0017	0.9658 ± 0.0019	0.9489 ± 0.0035
	PN-Y-S	0.9051 ± 0.0102	0.9214 ± 0.0100	0.9479 ± 0.0021	0.9496 ± 0.0016	0.9564 ± 0.0031	0.9607 ± 0.0034	0.9402 ± 0.0051
	PN-Y-S-E	0.9051 ± 0.0083	0.9256 ± 0.0067	0.9479 ± 0.0034	0.9521 ± 0.0021	0.9556 ± 0.0029	0.9581 ± 0.0025	0.9407 ± 0.0043
2	DANN	0.8970 ± 0.0179	0.9497 ± 0.0094	0.9760 ± 0.0042	0.9808 ± 0.0022	0.9796 ± 0.0031	0.9808 ± 0.0069	0.9607 ± 0.0073
	DIWA	**0.9928 ± 0.0022**	**0.9928 ± 0.0035**	**0.9976 ± 0.0015**	**0.9988 ± 0.0012**	**0.9988 ± 0.0012**	**0.9976 ± 0.0015**	**0.9964 ± 0.0018**
	ERM	0.9868 ± 0.0035	0.9892 ± 0.0035	0.9952 ± 0.0012	0.9940 ± 0.0033	0.9952 ± 0.0029	0.9964 ± 0.0015	0.9928 ± 0.0026
	NUC-0.01	0.9665 ± 0.0031	0.9760 ± 0.0054	0.9880 ± 0.0027	0.9856 ± 0.0041	0.9892 ± 0.0029	0.9868 ± 0.0029	0.9820 ± 0.0035
	NUC-0.1	0.9665 ± 0.0031	0.9784 ± 0.0041	0.9880 ± 0.0027	0.9856 ± 0.0015	0.9880 ± 0.0019	0.9868 ± 0.0044	0.9822 ± 0.0029
	PN-Y	0.9880 ± 0.0000	0.9880 ± 0.0000	0.9940 ± 0.0019	0.9916 ± 0.0015	0.9904 ± 0.0015	0.9916 ± 0.0015	0.9906 ± 0.0010
	PN-Y-S	0.9844 ± 0.0024	**0.9928 ± 0.0022**	0.9892 ± 0.0022	0.9940 ± 0.0019	0.9928 ± 0.0012	0.9964 ± 0.0024	0.9916 ± 0.0021
	PN-Y-S-E	0.9868 ± 0.0029	0.9916 ± 0.0024	0.9928 ± 0.0012	0.9952 ± 0.0022	0.9940 ± 0.0000	0.9928 ± 0.0012	0.9922 ± 0.0017
3	DANN	0.8336 ± 0.0041	0.8804 ± 0.0049	0.9150 ± 0.0061	0.9272 ± 0.0040	0.9298 ± 0.0029	0.9496 ± 0.0010	0.9059 ± 0.0038
	DIWA	**0.9323 ± 0.0041**	**0.9298 ± 0.0048**	**0.9399 ± 0.0035**	**0.9537 ± 0.0023**	**0.9580 ± 0.0043**	0.9552 ± 0.0033	**0.9448 ± 0.0037**
	ERM	0.8758 ± 0.0099	0.9135 ± 0.0029	0.9328 ± 0.0061	0.9486 ± 0.0055	0.9506 ± 0.0013	**0.9557 ± 0.0013**	0.9295 ± 0.0045
	NUC-0.01	0.7847 ± 0.0119	0.8687 ± 0.0076	0.9053 ± 0.0067	0.9170 ± 0.0037	0.9288 ± 0.0048	0.9338 ± 0.0065	0.8897 ± 0.0069
	NUC-0.1	0.7817 ± 0.0111	0.8687 ± 0.0021	0.8952 ± 0.0038	0.9150 ± 0.0046	0.9277 ± 0.0036	0.9450 ± 0.0034	0.8889 ± 0.0047
	PN-Y	0.9043 ± 0.0013	0.9216 ± 0.0028	0.9257 ± 0.0019	0.9389 ± 0.0055	0.9486 ± 0.0026	0.9476 ± 0.0037	0.9311 ± 0.0030
	PN-Y-S	0.8957 ± 0.0030	0.9099 ± 0.0022	0.9277 ± 0.0025	0.9384 ± 0.0062	0.9471 ± 0.0031	0.9466 ± 0.0031	0.9276 ± 0.0033
	PN-Y-S-E	0.8967 ± 0.0039	0.9104 ± 0.0046	0.9308 ± 0.0035	0.9379 ± 0.0047	0.9461 ± 0.0035	0.9496 ± 0.0052	0.9286 ± 0.0043

Table 8: Target Finetuning results on PACS dataset.

E_t	Method	0.10	0.20	0.40	0.60	0.80	1.00	mean
0	DANN	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	DIWA	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	ERM	0.9915 ± 0.0026	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	0.9986 ± 0.0004
	NUC-0.01	0.9930 ± 0.0022	0.9915 ± 0.0026	0.9972 ± 0.0017	0.9986 ± 0.0014	0.9958 ± 0.0017	**1.0000 ± 0.0000**	0.9960 ± 0.0016
	NUC-0.1	0.9930 ± 0.0022	0.9930 ± 0.0022	0.9972 ± 0.0017	0.9986 ± 0.0014	0.9958 ± 0.0017	**1.0000 ± 0.0000**	0.9962 ± 0.0016
	PN-Y	0.9958 ± 0.0028	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	0.9993 ± 0.0005
	PN-Y-S	0.9958 ± 0.0017	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	0.9993 ± 0.0003
	PN-Y-S-E	0.9972 ± 0.0017	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	0.9995 ± 0.0003
1	DANN	0.7053 ± 0.0080	0.7331 ± 0.0043	0.7586 ± 0.0111	0.7812 ± 0.0048	0.7699 ± 0.0070	0.7782 ± 0.0044	0.7544 ± 0.0066
	DIWA	**0.7692 ± 0.0091**	**0.7617 ± 0.0061**	**0.7842 ± 0.0085**	**0.7872 ± 0.0061**	**0.7857 ± 0.0036**	**0.8000 ± 0.0036**	**0.7813 ± 0.0062**
	ERM	0.7391 ± 0.0044	0.7474 ± 0.0087	0.7647 ± 0.0065	0.7850 ± 0.0018	0.7797 ± 0.0055	0.7699 ± 0.0069	0.7643 ± 0.0056
	NUC-0.01	0.7511 ± 0.0061	0.7504 ± 0.0079	0.7835 ± 0.0073	0.7729 ± 0.0082	0.7797 ± 0.0059	0.7977 ± 0.0048	0.7726 ± 0.0067
	NUC-0.1	0.7496 ± 0.0031	0.7436 ± 0.0036	0.7752 ± 0.0093	0.7850 ± 0.0084	0.7812 ± 0.0077	0.7782 ± 0.0043	0.7688 ± 0.0061
	PN-Y	0.7481 ± 0.0082	0.7541 ± 0.0057	0.7564 ± 0.0047	0.7805 ± 0.0035	0.7699 ± 0.0038	0.7759 ± 0.0081	0.7642 ± 0.0057
	PN-Y-S	0.7226 ± 0.0083	0.7421 ± 0.0149	0.7556 ± 0.0063	0.7489 ± 0.0090	0.7504 ± 0.0042	0.7692 ± 0.0068	0.7481 ± 0.0083
	PN-Y-S-E	0.7331 ± 0.0145	0.7083 ± 0.0051	0.7346 ± 0.0064	0.7534 ± 0.0059	0.7609 ± 0.0076	0.7647 ± 0.0107	0.7425 ± 0.0083
2	DANN	0.7476 ± 0.0078	0.7829 ± 0.0081	0.7994 ± 0.0066	0.8067 ± 0.0077	0.8238 ± 0.0059	0.8274 ± 0.0100	0.7980 ± 0.0077
	DIWA	0.7835 ± 0.0065	0.8110 ± 0.0050	**0.8262 ± 0.0066**	0.8335 ± 0.0048	0.8372 ± 0.0079	0.8378 ± 0.0054	0.8215 ± 0.0060
	ERM	0.7811 ± 0.0045	0.7933 ± 0.0054	0.8128 ± 0.0039	0.8110 ± 0.0062	0.8122 ± 0.0073	0.8323 ± 0.0067	0.8071 ± 0.0057
	NUC-0.01	0.7567 ± 0.0072	0.7884 ± 0.0080	0.8134 ± 0.0092	0.8262 ± 0.0039	**0.8409 ± 0.0060**	0.8256 ± 0.0066	0.8085 ± 0.0068
	NUC-0.1	0.7622 ± 0.0026	0.7841 ± 0.0077	0.8134 ± 0.0083	0.8244 ± 0.0080	0.8256 ± 0.0050	0.8171 ± 0.0064	0.8045 ± 0.0063
	PN-Y	**0.8055 ± 0.0077**	**0.8226 ± 0.0039**	0.8244 ± 0.0027	**0.8396 ± 0.0071**	0.8366 ± 0.0055	0.8329 ± 0.0024	**0.8269 ± 0.0049**
	PN-Y-S	0.8018 ± 0.0055	0.8165 ± 0.0046	0.8091 ± 0.0055	0.8250 ± 0.0028	0.8274 ± 0.0021	**0.8433 ± 0.0052**	0.8205 ± 0.0043
	PN-Y-S-E	0.7890 ± 0.0095	0.8061 ± 0.0061	0.8220 ± 0.0064	0.8293 ± 0.0054	0.8323 ± 0.0033	0.8366 ± 0.0053	0.8192 ± 0.0060
3	DANN	0.7811 ± 0.0070	0.8000 ± 0.0033	0.8178 ± 0.0072	0.8373 ± 0.0032	0.8396 ± 0.0049	0.8432 ± 0.0057	0.8198 ± 0.0052
	DIWA	0.8556 ± 0.0030	0.8598 ± 0.0035	0.8627 ± 0.0040	0.8704 ± 0.0046	0.8734 ± 0.0029	0.8686 ± 0.0062	0.8651 ± 0.0040
	ERM	0.8698 ± 0.0047	0.8586 ± 0.0068	0.8757 ± 0.0055	0.8746 ± 0.0054	0.8882 ± 0.0011	0.8799 ± 0.0057	0.8745 ± 0.0049
	NUC-0.01	0.8432 ± 0.0032	0.8598 ± 0.0042	0.8675 ± 0.0057	0.8704 ± 0.0052	0.8805 ± 0.0050	0.8846 ± 0.0034	0.8677 ± 0.0045
	NUC-0.1	0.8414 ± 0.0035	0.8592 ± 0.0035	0.8627 ± 0.0035	0.8669 ± 0.0026	0.8751 ± 0.0025	0.8746 ± 0.0080	0.8633 ± 0.0039
	PN-Y	0.8734 ± 0.0030	0.8793 ± 0.0060	0.8882 ± 0.0047	**0.8953 ± 0.0038**	0.8941 ± 0.0050	**0.9012 ± 0.0043**	0.8886 ± 0.0045
	PN-Y-S	**0.8781 ± 0.0036**	0.8811 ± 0.0024	**0.8893 ± 0.0033**	0.8905 ± 0.0026	**0.9041 ± 0.0018**	0.8964 ± 0.0036	**0.8899 ± 0.0029**
	PN-Y-S-E	0.8722 ± 0.0059	**0.8899 ± 0.0047**	0.8846 ± 0.0050	0.8893 ± 0.0020	0.8964 ± 0.0030	0.8935 ± 0.0028	0.8877 ± 0.0039

Table 9: Target Finetuning results on VLCS dataset.

E_t	Method	0.10	0.20	0.40	0.60	0.80	1.00	mean
0	DANN	0.8076 ± 0.0082	0.8650 ± 0.0055	0.9169 ± 0.0041	0.9549 ± 0.0023	0.9675 ± 0.0025	0.9764 ± 0.0012	0.9147 ± 0.0040
	DIWA	0.8793 ± 0.0043	0.8979 ± 0.0040	0.9401 ± 0.0045	0.9570 ± 0.0022	**0.9726 ± 0.0013**	0.9726 ± 0.0042	0.9366 ± 0.0034
	ERM	0.8696 ± 0.0055	0.8937 ± 0.0014	0.9409 ± 0.0043	0.9515 ± 0.0038	0.9671 ± 0.0035	**0.9806 ± 0.0022**	0.9339 ± 0.0035
	NUC-0.01	0.7793 ± 0.0040	0.8426 ± 0.0074	0.9143 ± 0.0020	0.9460 ± 0.0045	0.9591 ± 0.0032	0.9700 ± 0.0023	0.9019 ± 0.0039
	NUC-0.1	0.7772 ± 0.0031	0.8494 ± 0.0028	0.9177 ± 0.0030	0.9464 ± 0.0046	0.9553 ± 0.0031	0.9709 ± 0.0016	0.9028 ± 0.0030
	PN-Y	**0.8835 ± 0.0046**	**0.9093 ± 0.0030**	0.9414 ± 0.0049	**0.9608 ± 0.0036**	0.9679 ± 0.0026	0.9751 ± 0.0016	**0.9397 ± 0.0034**
	PN-Y-S	0.8772 ± 0.0045	0.9051 ± 0.0037	0.9376 ± 0.0017	0.9553 ± 0.0020	0.9658 ± 0.0017	0.9730 ± 0.0014	0.9357 ± 0.0025
	PN-Y-S-E	0.8787 ± 0.0020	0.9013 ± 0.0023	**0.9430 ± 0.0024**	0.9527 ± 0.0033	0.9679 ± 0.0039	0.9759 ± 0.0027	0.9366 ± 0.0028
1	DANN	0.8361 ± 0.0061	0.8949 ± 0.0032	0.9329 ± 0.0041	0.9556 ± 0.0020	0.9649 ± 0.0014	0.9747 ± 0.0008	0.9265 ± 0.0029
	DIWA	**0.8797 ± 0.0051**	**0.9203 ± 0.0054**	**0.9446 ± 0.0031**	**0.9575 ± 0.0051**	**0.9731 ± 0.0010**	**0.9764 ± 0.0009**	**0.9419 ± 0.0034**
	ERM	0.8333 ± 0.0071	0.8947 ± 0.0045	0.9331 ± 0.0035	0.9515 ± 0.0027	0.9653 ± 0.0028	0.9749 ± 0.0020	0.9255 ± 0.0038
	NUC-0.01	0.8070 ± 0.0062	0.8793 ± 0.0031	0.9265 ± 0.0042	0.9466 ± 0.0022	0.9620 ± 0.0020	0.9698 ± 0.0014	0.9152 ± 0.0032
	NUC-0.1	0.8154 ± 0.0032	0.8723 ± 0.0039	0.9331 ± 0.0036	0.9458 ± 0.0028	0.9624 ± 0.0015	0.9713 ± 0.0006	0.9167 ± 0.0026
	PN-Y	0.8591 ± 0.0037	0.9021 ± 0.0038	0.9312 ± 0.0023	0.9487 ± 0.0011	0.9620 ± 0.0012	0.9745 ± 0.0018	0.9296 ± 0.0023
	PN-Y-S	0.8472 ± 0.0050	0.8943 ± 0.0028	0.9370 ± 0.0025	0.9520 ± 0.0014	0.9651 ± 0.0020	0.9749 ± 0.0003	0.9284 ± 0.0023
	PN-Y-S-E	0.8509 ± 0.0030	0.9004 ± 0.0035	0.9312 ± 0.0027	0.9472 ± 0.0025	0.9645 ± 0.0012	0.9731 ± 0.0020	0.9279 ± 0.0025
2	DANN	0.6816 ± 0.0133	0.7673 ± 0.0072	0.8529 ± 0.0056	0.8922 ± 0.0072	0.9179 ± 0.0061	0.9441 ± 0.0042	0.8427 ± 0.0073
	DIWA	**0.7778 ± 0.0104**	**0.8292 ± 0.0056**	**0.8826 ± 0.0065**	**0.9128 ± 0.0043**	**0.9370 ± 0.0052**	**0.9471 ± 0.0029**	**0.8811 ± 0.0058**
	ERM	0.7526 ± 0.0056	0.8096 ± 0.0052	0.8579 ± 0.0062	0.8997 ± 0.0062	0.9179 ± 0.0045	0.9194 ± 0.0035	0.8595 ± 0.0052
	NUC-0.01	0.6222 ± 0.0102	0.7375 ± 0.0126	0.8489 ± 0.0054	0.8977 ± 0.0058	0.9224 ± 0.0033	0.9390 ± 0.0043	0.8280 ± 0.0069
	NUC-0.1	0.6121 ± 0.0090	0.7365 ± 0.0083	0.8489 ± 0.0065	0.8987 ± 0.0019	0.9249 ± 0.0037	0.9395 ± 0.0055	0.8268 ± 0.0058
	PN-Y	0.7647 ± 0.0045	0.8227 ± 0.0063	0.8665 ± 0.0052	0.8922 ± 0.0041	0.9123 ± 0.0043	0.9295 ± 0.0035	0.8647 ± 0.0047
	PN-Y-S	0.7446 ± 0.0049	0.8050 ± 0.0031	0.8584 ± 0.0074	0.8856 ± 0.0049	0.9098 ± 0.0041	0.9285 ± 0.0028	0.8553 ± 0.0046
	PN-Y-S-E	0.7446 ± 0.0031	0.7953 ± 0.0049	0.8559 ± 0.0042	0.8872 ± 0.0033	0.9159 ± 0.0048	0.9300 ± 0.0045	0.8548 ± 0.0041
3	DANN	0.6983 ± 0.0047	0.7969 ± 0.0062	0.8714 ± 0.0037	0.9034 ± 0.0022	**0.9296 ± 0.0045**	0.9364 ± 0.0019	0.8560 ± 0.0039
	DIWA	**0.7806 ± 0.0067**	**0.8289 ± 0.0090**	**0.8806 ± 0.0041**	**0.9065 ± 0.0050**	0.9259 ± 0.0039	0.9276 ± 0.0022	**0.8750 ± 0.0052**
	ERM	0.7534 ± 0.0062	0.8207 ± 0.0056	0.8704 ± 0.0047	0.8980 ± 0.0063	0.9231 ± 0.0027	0.9320 ± 0.0028	0.8663 ± 0.0047
	NUC-0.01	0.6476 ± 0.0060	0.7769 ± 0.0117	0.8571 ± 0.0085	0.8980 ± 0.0058	0.9184 ± 0.0042	0.9361 ± 0.0022	0.8390 ± 0.0064
	NUC-0.1	0.6374 ± 0.0029	0.7718 ± 0.0109	0.8585 ± 0.0083	0.8993 ± 0.0053	0.9214 ± 0.0029	0.9374 ± 0.0014	0.8376 ± 0.0053
	PN-Y	0.7673 ± 0.0037	0.8289 ± 0.0077	0.8721 ± 0.0088	0.8952 ± 0.0054	0.9252 ± 0.0032	0.9384 ± 0.0035	0.8712 ± 0.0054
	PN-Y-S	0.7486 ± 0.0041	0.8167 ± 0.0094	0.8714 ± 0.0057	0.8993 ± 0.0047	0.9241 ± 0.0020	0.9374 ± 0.0021	0.8663 ± 0.0047
	PN-Y-S-E	0.7524 ± 0.0066	0.8185 ± 0.0070	0.8673 ± 0.0059	0.8973 ± 0.0049	0.9238 ± 0.0012	**0.9408 ± 0.0022**	0.8667 ± 0.0046

Table 10: Target Finetuning results on TerrainCognita dataset.

E.3 Learning Rate Decay Results

We test the learning rate decay for the feature encoder, which is commonly used in domain adaptation methods. We use the same setting as the target finetuning experiment, but we decay the learning rate by 0.1. We report the results in Figure 6.

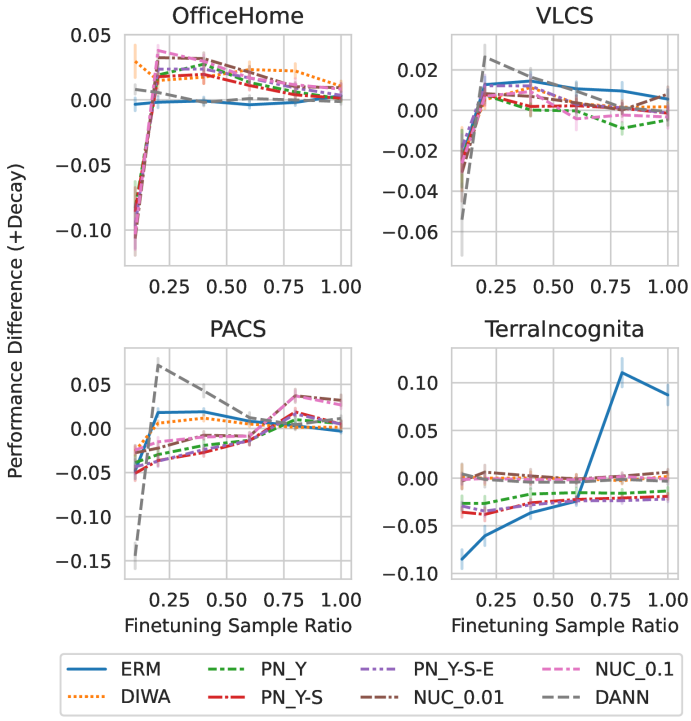


Figure 6: Feature Encoder Learning Rate Decay Results: We show that learning rate decay does not may improve the performance of the model on large FSL regimes but improves the performance on small FSL regimes.

F Experiment Details

Our code is available https://anonymous.4open.science/r/projectionnet.

F.1 Linear synthetic data

We have $E = 999$ source environments and a target environment indexed by 1000. The source and target sample sizes are $n_1 = 50$, $n_2 = 2000$ respectively, with the ambient input dimension $d = 10$, content feature dimension $k = 6$, and variance of noise $\sigma = 0.01$.

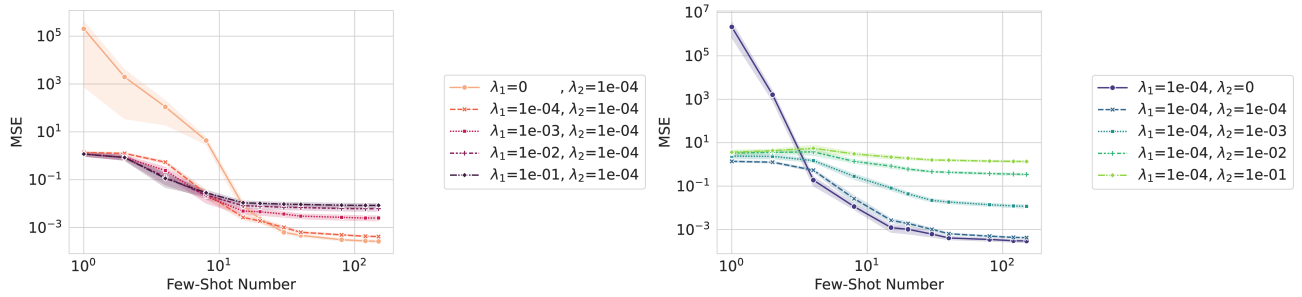
Generation of X^e For $e \in [E]$, $X^e \in \mathbb{R}^{n_1 \times d}$ and for $e = E + 1$, $X^{E+1} \in \mathbb{R}^{n_2 \times d}$. To guarantee Assumption B.4, we generate X^e via its SVD: $X^e = U^e S^e V^{e\top}$ where U^e and V^e are random orthogonal matrices, and the singular values S^e are sampled from the mixture of Gaussian distributions. Specifically, the top k singular values follow $\mathcal{N}(5, 1)$ and the rest $d - k$ follow $\mathcal{N}(0, 1)$.

Generation of $\boldsymbol{\theta}^{*e}$ For $e \in [E + 1]$, $\boldsymbol{\theta}^{*e}$'s are i.i.d. samples of the meta distribution (equation 3) with the parameter choice $\boldsymbol{\theta}^* = 6 \cdot \mathbf{1}_k, \Lambda_{11} = 0.1 \cdot I_k, \Lambda_{22} = 3 \cdot I_{d-k}$.

Generation of $\boldsymbol{y}^e$ For $e \in [E + 1]$, $\boldsymbol{y}^e$'s are generated via

$$\boldsymbol{y}^e = X^e R^* \boldsymbol{\theta}^{*e} + \boldsymbol{z}^e$$

where R^* is some random $d \times d$ orthogonal matrix and $\boldsymbol{z}^e$'s are i.i.d. samples of $\mathcal{N}(0, \sigma^2 I_{n_1/n_2})$ independent of X^e's.



(a) Effect of λ_1

(b) Effect of λ_2

Figure 7: The effect of (λ_1, λ_2) and on the performance of the target task: No regularization leads to an unstable solution and overfitting when FSL is small, but large regularization prevent the full adaptation to the target task when FSL is large.

F.2 Real data

F.2.1 Training Configuration

We use ResNet50 (He et al., 2016) as the backbone network and use the SGD optimizer to train the model, the training setting is the same as (Wang and Lu).

We use $\text{lr}_{\text{decay}} = 0.1, \text{lr}_{\text{Feature}} = \text{lr}_{\text{decay}} * \text{lr}_{\text{Classification}}$. We vary the learning rate decay from 0 (frozen) to 1 (not decayed) to control the (λ_1 and λ_2) regularization strength. In the source training stage, we use the source validation set to select the best model; In the target training stage, we use the target validation set to select the best model.

Computation Resource: All the experiments can be done with A40, RTX 8000 GPU or A100 GPU, 32GB memory, and 16 CPU 2.9GHz cores (Intel Cascade Lake Platinum 8268 chips).

G Additional Experiments

We provide additional experiments results on DomainNet dataset with six domains.

Table 11: Domain Generalization results on DomainNet.

Method	0	1	2	3	4	5
ERM	0.5474 ± 0.0000	0.1889 ± 0.0000	0.4777 ± 0.0001	0.1103 ± 0.0000	0.5575 ± 0.0000	0.4726 ± 0.0002
NUC-0.01	0.5445 ± 0.0000	0.1935 ± 0.0000	0.4782 ± 0.0000	0.1027 ± 0.0000	0.5544 ± 0.0000	0.4720 ± 0.0000
PN-Y	**0.5742 ± 0.0000**	**0.1988 ± 0.0000**	**0.5002 ± 0.0000**	**0.1173 ± 0.0000**	**0.5993 ± 0.0000**	**0.4757 ± 0.0001**

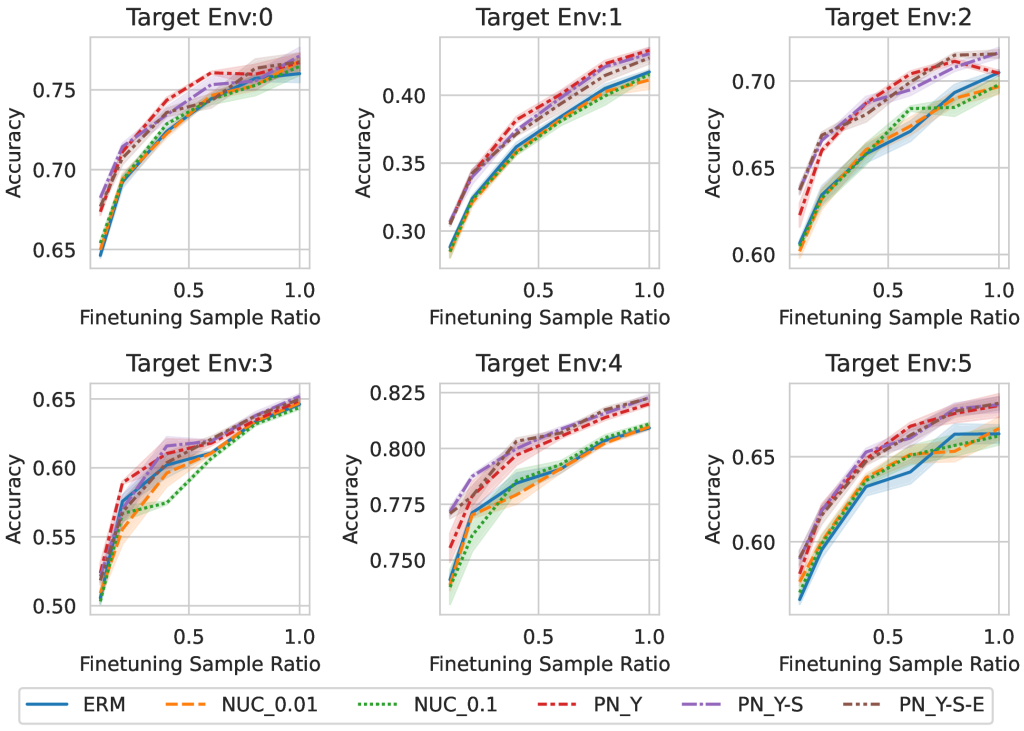


Figure 8: Target Finetuning Results on DomainNet.

G.1 Additional Visualizations

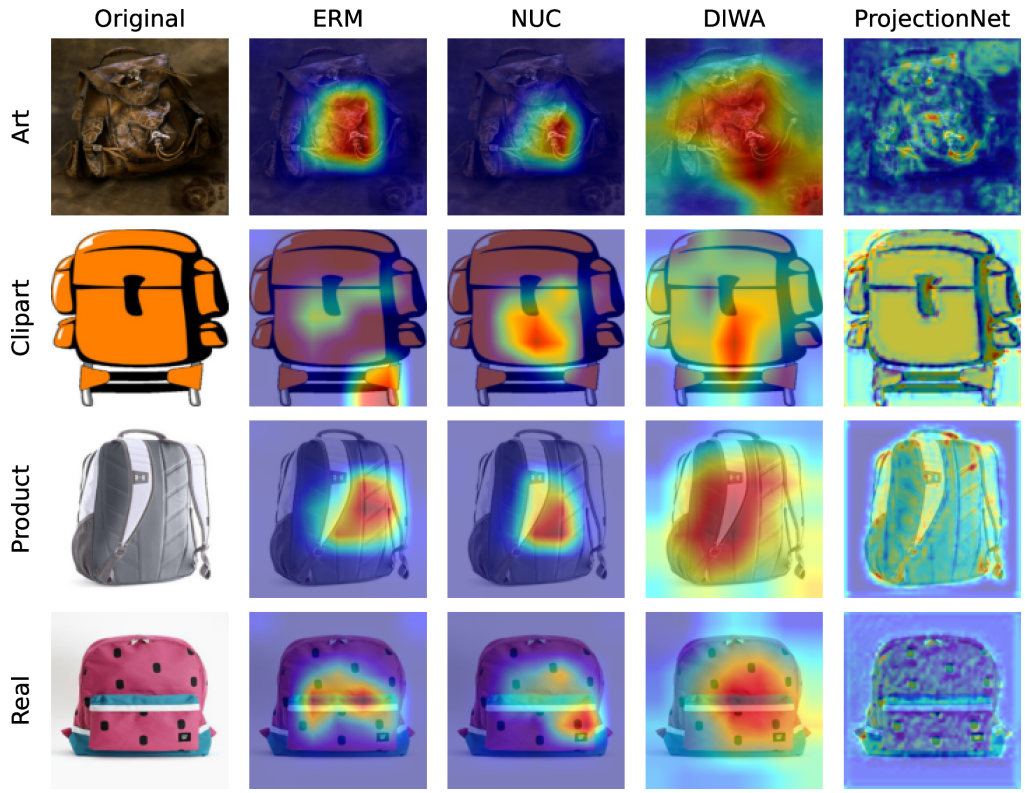


Figure 9: The GradCAM++ visualization of different methods with OfficeHome BackPack class.

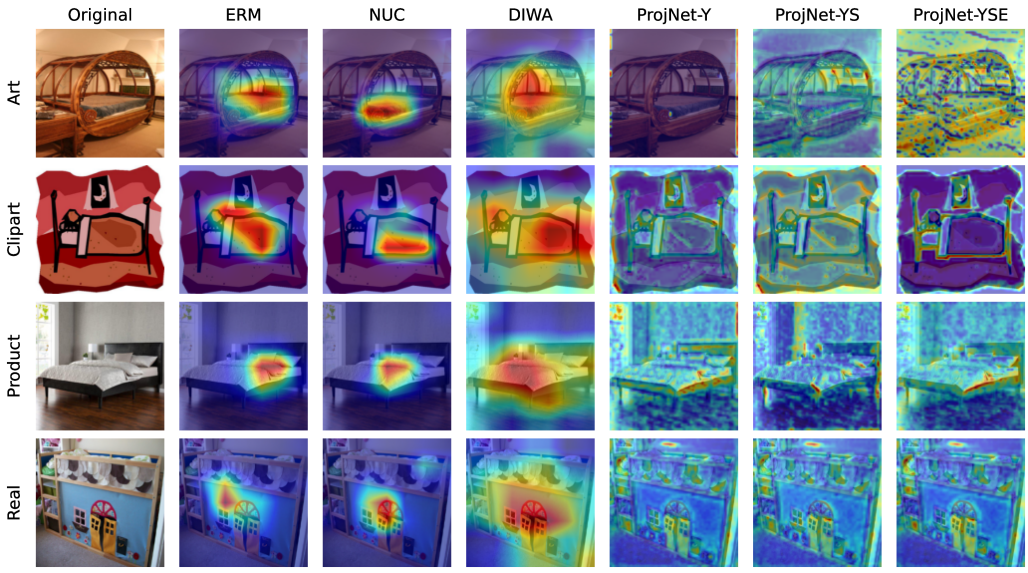


Figure 10: The GradCAM++ visualization of different methods with OfficeHome Bed class.