Transfer Learning of Linear Regression with Multiple Pretrained Models: Benefiting from More Pretrained Models via Overparameterization Debiasing

Daniel Boharon
Faculty of Computer and Information Science
Ben-Gurion University of the Negev
bohadan@post.bgu.ac.il

Yehuda Dar
Faculty of Computer and Information Science
Ben-Gurion University of the Negev
ydar@bgu.ac.il

Abstract

We study transfer learning for a linear regression task using several least-squares pretrained models that can be overparameterized. We formulate the target learning task as optimization that minimizes squared errors on the target dataset with a penalty on the distance of the learned model from the pretrained models. We analytically formulate the test error of the learned target model and provide the corresponding empirical evaluations. Our results elucidate when using more pretrained models can improve transfer learning. Specifically, if the pretrained models are overparameterized, using sufficiently many of them is important for beneficial transfer learning. However, the learning may be compromised by overparameterization bias of pretrained models, i.e., the minimum ℓ_2-norm solution's restriction to a small subspace spanned by the training examples in the high-dimensional parameter space. We propose a simple debiasing via multiplicative correction factor that can reduce the overparameterization bias and leverage more pretrained models to learn a target predictor.

1 Introduction

Transfer learning improves deep neural network training by leveraging auxiliary models pretrained on related tasks. These models serve as parameter initializations or fixed feature extractors, mitigating the high data and computational costs of training. Since many deep networks are overparameterized, containing far more parameters than training examples, transfer learning has become prevalent. This widespread use necessitates a foundational understanding of transfer learning, particularly for overparameterized models.

Transfer learning is useful also for linear models [18, 1, 4, 14, 6, 3]. Specifically, high-dimensional data can require more learnable parameters than the available training examples — making linear models overparameterized and transfer learning beneficial [4, 6]. Therefore, the study of transfer learning of overparameterized linear models is motivated by both practical and theoretical considerations.

In this paper, we study transfer learning for linear regression where a model is learned for a target task using *multiple pretrained models* of related source tasks. While the common transfer learning practice is to use a single pretrained model, we here ask the following central questions: *How beneficial is using more pretrained models? How does overparameterization of pretrained models affect the benefits from using more pretrained models? How can we make multiple overparameterized pretrained models more beneficial for transfer learning?*

We address these questions in a learning setting where a target task of interest has task-specific training data and several pretrained models that were trained for other tasks. The pretrained models were trained using least squares for which we study the parameterization range from underparameterized

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

(the source task has more training examples than parameters) to overparameterized (the source task has less training examples than parameters). Overparameterized pretrained models perfectly fit their training data, i.e., achieve zero training error. Each of our source tasks is related to the target task via noisy linear model where the source task's true parameters (that define the source data distribution) are equal to a linear transformation of the target task's true parameters and additive noise; each of the source tasks can relate to the target task by another task relation. The target model has more learned parameters than training examples and is learned via transfer learning, as described next.

For the target task, we define our transfer learning as a minimization problem of the squared errors for the target dataset with penalties on the distance of the learned target parameters from each of the pretrained source models. By this we extend the transfer learning method by Dar et al. [6] where only a single pretrained model is used. The relations between the source and target tasks are unknown in the transfer learning process. We show that our transfer learning is beneficial even if the linear operators from the task relations are replaced by the identity matrix or its scaled versions. We use random matrix theory tools to analytically formulate the test error in the high-dimensional asymptotic regime, elucidating the effect of the number of pretrained models and their parameterization levels on the transfer learning performance.

Our analytical and empirical results show that transfer learning using multiple pretrained models can outperform the standard use of a single pretrained model. We elucidate that for beneficial transfer learning, the number of pretrained models should be sufficiently high with respect to the learning setting. Specifically, **if the pretrained models are overparameterized, multiple pretrained models can be necessary for beneficial transfer learning**.

We propose a debiasing approach that compensates for the transfer learning bias caused by overparameterization of least-squares pretrained models. Specifically, an overparameterized pretrained model is the minimum ℓ_2-norm solution among the infinitely many solutions to least squares learning of more source parameters than source training examples; such a model has overparameterization bias in the sense of having the expectation of the learned parameters attenuated by a multiplicative factor of $\frac{\widetilde{n}}{d} < 1$, where $\widetilde{n}$ is the number of source training examples and d is the number of learned parameters. If the overparameterization of the pretrained models is too high, the overparameterization bias may significantly degrade transfer learning. Accordingly, when the number $\widetilde{n}$ of source training examples is known, we reduce the bias in our transfer learning by scaling the assumed task relation operators by the inverse of the source overparameterization level $\frac{\widetilde{n}}{d} < 1$. Importantly, when $\widetilde{n}$ is unknown, we choose the scaling factor using a validation dataset. We show, both analytically and empirically, that **our debiasing approach can significantly increase the benefits of using more pretrained models in transfer learning, outperforming the other examined alternatives.**

Our theory and synthetic-data experiments show that overparameterized pretrained models make the transfer learning predictor inconsistent, i.e., its generalization performance does not approach optimality when using more pretrained models. Remarkably, **using our overparameterization debiasing, the transfer learning can provide a consistent predictor.**

We examine the bias-variance tradeoff in our transfer learning with multiple pretrained models. This shows that the **overparameterization debiasing reduces the bias but increases the variance, which in turn can be reduced by using more pretrained models**. Consequently, we get a beneficial bias-variance tradeoff that resolves fundamental difficulties of using many overparameterized pretrained models together in a single transfer learning task.

2 Related Works

Linear Transfer Learning with a Single Pretrained Model: Transfer learning for linear regression using one pretrained model is widely studied [18, 1, 4, 6, 3]. Our work significantly extends the framework of Dar et al. [6] from single to multiple pretrained models. We introduce new research questions unique to the multi-model setting and propose a new overparameterization debiasing approach. See details in Appendix D.1.

Linear Transfer Learning with Multiple Pretrained Models: Prior studies often rely on multiple source *datasets* [14, 15, 22, 16]. Conversely, we address the more challenging scenario where only pretrained models are available. While Singh and Diggavi [20] utilize pretrained representation matrices for underparameterized source tasks, we focus on overparameterized sources. This shift

adds new important aspects that play significant roles in our analysis and proposed algorithm for overparameterization debiasing. Importantly, our central new phenomenon is that overparameterization bias can prevent the benefits from using multiple pretrained models, and this can be mitigated by overparameterization debiasing that yields beneficial transfer as the number of pretrained models increases. This new phenomenon emerges from the important foundational difference: our pretrained models can be overparameterized, in contrast to the underparameterized sources in [20]. See Appendix D.3 for additional discussion on the important differences between the work by Singh and Diggavi [20] and ours.

The Overparameterization Bias: Overparameterized least-squares regression has more learnable parameters than training examples, leading to infinite solutions that perfectly fit the training examples. Among the infinite solutions, the minimum ℓ_2-norm solution has an inductive bias that restricts the learned parameter vector to the linear subspace spanned by the training examples [2, 12]. This dataset-induced subspace resides in the high-dimensional parameter space, therefore, higher overparameterization implies a stronger inductive bias that can increase the bias component of the test error [5]. In the transfer learning, the overparameterization bias played a role in the analysis in [4] of transferred source parameters (from a single pretrained model) that are set fixed in the target task. They showed that a less related source task can be more beneficial if the source task relation compensates for the overparameterization bias. In contrast, **we propose an explicit debiasing approach that unleashes the potential of using multiple pretrained models in optimization penalties. Namely, we show that the overparametrization bias can significantly limit the use of multiple pretrained models, and we resolve this by proposing the overparameterization debiasing.**

3 Problem Formulation and Notations

Source tasks' definitions. We have m source tasks of linear regression. The data distribution of the j^{th} source task, $j \in \{1, \ldots, m\}$, is defined by a d-dimensional random input $\mathbf{z}_j \in \mathbb{R}^d$ with zero mean and covariance matrix $\mathbf{\Sigma}_{\mathbf{z}}$, and an output $v_j \in \mathbb{R}$ such that

$$v_j = \mathbf{z}_j^T \boldsymbol{\theta}_j + \xi_j, \tag{1}$$

$\xi_j \in \mathbb{R}$ is a zero-mean noise variable independent of $\mathbf{z}_j$ with variance $\sigma_{\xi_j}^2$, and $\boldsymbol{\theta}_j \in \mathbb{R}^d$ is an unknown parameter vector of the j^{th} task. The true data distribution of $(\mathbf{z}_j, v_j)$ is unknown for the j^{th} source learning task, which is performed using a dataset $\widetilde{\mathcal{D}}_j \triangleq \left\{ \left(\mathbf{z}_j^{(i)}, v_j^{(i)} \right) \right\}_{j=1}^{\widetilde{n}_j}$ of $\widetilde{n}_j$ independent and identically distributed (i.i.d.) examples of $(\mathbf{z}_j, v_j)$ drawn from the j^{th} source task distribution. The $\widetilde{n}_j$ data examples in $\widetilde{\mathcal{D}}_j$ are reorganized as a $\widetilde{n}_j \times d$ input matrix $\mathbf{Z}_j \triangleq \left[\mathbf{z}_j^{(1)}, \ldots, \mathbf{z}_j^{(\widetilde{n}_j)} \right]^T$ and a $\widetilde{n}_j \times 1$ output vector $\mathbf{v}_j \triangleq \left[v_j^{(1)}, \ldots, v_j^{(\widetilde{n}_j)} \right]^T$. See more details in Appendix E.1. The j^{th} source task is solved via least squares. If multiple solutions exist, the one with the minimum ℓ_2-norm is used,

$$\widehat{\boldsymbol{\theta}}_j = \underset{\mathbf{r} \in \mathbb{R}^d}{\arg\min} \, \|\mathbf{v}_j - \mathbf{Z}_j \mathbf{r}\|_2^2 = \mathbf{Z}_j^+ \mathbf{v}_j, \tag{2}$$

where $\mathbf{Z}_j^+$ is the Moore-Penrose pseudoinverse of $\mathbf{Z}_j$. For an almost-surely full-rank $\mathbf{Z}_j$, the test (squared) error (of the source task without any transfer aspect) peaks around $d = \widetilde{n}_j$ [2, 12], i.e., at the threshold between the under and over parameterized regimes of the j^{th} source model.

Target task definition. Our target task has data $(\mathbf{x}, y) \in \mathbb{R}^d \times \mathbb{R}$ that satisfies $y = \mathbf{x}^T \boldsymbol{\beta} + \epsilon$ where $\mathbf{x}$ is a random input vector with zero mean and covariance matrix $\mathbf{\Sigma}_{\mathbf{x}}$, ϵ is a zero-mean noise variable independent of $\mathbf{x}$ with variance σ_ϵ^2 and $\boldsymbol{\beta} \in \mathbb{R}^d$ is an unknown parameter vector.

The unknown parameter vector $\boldsymbol{\theta}_j$ of the j^{th} source task, $j \in \{1, \ldots, m\}$, is related to the unknown parameter vector of the target task, $\boldsymbol{\beta}$, by the relation

$$\boldsymbol{\theta}_j = \mathbf{H}_j \boldsymbol{\beta} + \boldsymbol{\eta}_j \tag{3}$$

where $\mathbf{H}_j \in \mathbb{R}^{d \times d}$ is a fixed (non-random) matrix and $\boldsymbol{\eta}_j \sim \mathcal{N}\left(\mathbf{0}, (\sigma_{\eta_j}^2/d) \mathbf{I}_d \right)$ is a vector of i.i.d. Gaussian noise components with zero mean and variance $\sigma_{\eta_j}^2/d$. Each source task has its own $\mathbf{H}_j \in \mathbb{R}^{d \times d}$ and $\boldsymbol{\eta}_j$. The random elements $\{\boldsymbol{\eta}_j\}_{j=1}^m$, $\mathbf{x}$, ϵ, $\{\mathbf{z}_j\}_{j=1}^m$, $\{\xi_j\}_{j=1}^m$ are independent.

The task relation in (3) recalls a common data degradation model in inverse problems, which in our case relates to the recovery of the true $\boldsymbol{\beta}$ from the true $\{\boldsymbol{\theta}_j\}_{j=1}^m$. However, in our setting, we do not have the true $\{\boldsymbol{\theta}_j\}_{j=1}^m$ but only their estimates $\left\{\widehat{\boldsymbol{\theta}}_j\right\}_{j=1}^m$ that were learned each for its source task purposes. Moreover, we examine learning settings where $\{\mathbf{H}_j\}_{j=1}^m$ can be known or unknown. In Appendix E.2 we discuss the differences between our task relation (3) and a shared-representation task relation from [10, 21].

The true distribution of $(\mathbf{x}, y)$ is unknown in the target learning task, which is performed based on a dataset $\mathcal{D} \triangleq \left\{\left(\mathbf{x}^{(i)}, y^{(i)}\right)\right\}_{j=1}^n$ that contains n i.i.d. examples drawn from the distribution of $(\mathbf{x}, y)$. The n data examples in $\mathcal{D}$ are organized as a $n \times d$ matrix of input variables $\mathbf{X} \triangleq \left[\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\right]^T$ and an $n \times 1$ vector of outputs $\mathbf{y} \triangleq \left[y^{(1)}, \ldots, y^{(n)}\right]^T$. See more details in Appendix E.1.

A test input-output pair $\left(\mathbf{x}^{(\text{test})}, y^{(\text{test})}\right)$ is independently drawn from the distribution of $(\mathbf{x}, y)$. Given the input $\mathbf{x}^{(\text{test})}$, the target task aims to estimate the output value $y^{(\text{test})}$ by the value $\widehat{y} \triangleq \mathbf{x}^{(\text{test}),T}\widehat{\boldsymbol{\beta}}$, where $\widehat{\boldsymbol{\beta}}$ is learned from $\mathcal{D}$ in a transfer learning process that utilizes the m pretrained source models $\left\{\widehat{\boldsymbol{\theta}}_j\right\}_{j=1}^m$. We evaluate the generalization performance of the target task using the test squared error

$$\mathcal{E} \triangleq \mathbb{E}\left[\left(\widehat{y} - y^{(\text{test})}\right)^2\right] = \sigma_\epsilon^2 + \mathbb{E}\left[\left\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|_{\boldsymbol{\Sigma}_\mathbf{x}}^2\right] \tag{4}$$

where $\|\mathbf{a}\|_{\boldsymbol{\Sigma}_\mathbf{x}}^2 = \mathbf{a}^T\boldsymbol{\Sigma}_\mathbf{x}\mathbf{a}$ for $\mathbf{a} \in \mathbb{R}^d$, and the expectation in the definition of $\mathcal{E}$ is with respect to the test data $\left(\mathbf{x}^{(\text{test})}, y^{(\text{test})}\right)$ of the target task and the training data $\mathcal{D}$, $\left\{\widetilde{\mathcal{D}}_j\right\}_{j=1}^m$ of the target and all source tasks (the randomness of the source datasets indirectly affects $\widehat{\boldsymbol{\beta}}$ via the pretrained models). A lower value of the test error $\mathcal{E}$ reflects better generalization performance of the target task.

3.1 The Proposed Transfer Learning with Multiple Pretrained Models

Our new transfer learning optimization for the target task is

$$\widehat{\boldsymbol{\beta}}_{\text{TL}} = \underset{\mathbf{b}\in\mathbb{R}^d}{\arg\min} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 + n\alpha_{\text{TL}} \sum_{j=1}^m \left\|\widetilde{\mathbf{H}}_j\mathbf{b} - \widehat{\boldsymbol{\theta}}_j\right\|_2^2 \tag{5}$$

where $\alpha_{\text{TL}} > 0$ is a hyperparameter that determines the strength of transferring knowledge from the pretrained models. Our design choice to use a single hyperparameter, and not a separate one for each source task, is for practicality by avoiding tuning multiple (possibly many) hyperparameters; nevertheless, note that our single hyperparameter prevents source weighting and source selection that are potentially useful in practice. Our experiments throughout this paper demonstrate useful performance of a single hyperparameter, even for heterogeneous sources (Appendix T.6), without establishing its optimality. Moreover, $\left\{\widetilde{\mathbf{H}}_j\right\}_{j=1}^m$ are practical substitutes for the unknown task relation operators $\{\mathbf{H}_j\}_{j=1}^m$. When $\mathbf{H}_j$ is unknown, we study the practical substitutes $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$ and $\widetilde{\mathbf{H}}_j = a_j\mathbf{I}_d$ for $a_j > 0$ and show that they can be useful even when the true relation differs substantially from identity (later in this paper, Appendix P explains these choices through the bias-variance tradeoff).

We will analyze how the transfer learning depends on the parametrization levels, number of pretrained models, and the assumed-knowledge on the task relation matrices $\{\mathbf{H}_j\}_{j=1}^m$.

We choose the practical task relation matrices to conform with the following assumption.

Assumption 3.1 (Uniform full-rank condition)**.** There exists a constant $r_H > 0$, independent of d, such that for every d,

$$\sum_{j=1}^m \widetilde{\mathbf{H}}_{j,d}^\top \widetilde{\mathbf{H}}_{j,d} \succeq r_H \mathbf{I}_d. \tag{6}$$

We define $\mathbf{R} \triangleq \left(\sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{1/2}$. Under Assumption 3.1, the closed-form solution for (5) is

$$\widehat{\boldsymbol{\beta}}_{\mathrm{TL}} = \left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\mathbf{R}^2\right)^{-1}\left(\mathbf{X}^T\mathbf{y} + n\alpha_{\mathrm{TL}}\sum_{j=1}^{m}\widetilde{\mathbf{H}}_j^T\widehat{\boldsymbol{\theta}}_j\right) \tag{7}$$

Assumption 3.1 is sufficient for guaranteeing the matrix inverse existence in (7). In Appendix E.3 we provide a source-centered Tikhonov interpretation to our transfer learning (5).

Moreover, our theory will consider an isotropic Gaussian distribution for the true target parameters.

Assumption 3.2. The target task parameter $\boldsymbol{\beta}$ is distributed isotropically scaled by the dimension, i.e., $\boldsymbol{\beta} \sim \mathcal{N}\left(\mathbf{0}, \frac{b}{d}\mathbf{I}_d\right)$ where b is a constant $b > 0$.

Under Assumption 3.2 we will usually analyze the expected test error where the expectation is also over the randomness of $\boldsymbol{\beta}$; this extends the error definition in (4) and will be denoted by a bar over the error symbol, i.e., the expected error for a target model $\widehat{\boldsymbol{\beta}}$ is $\bar{\mathcal{E}} \triangleq \mathbb{E}_{\boldsymbol{\beta}}\left[\mathcal{E}\right] = \sigma_\epsilon^2 + \mathbb{E}\left[\left\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right\|_{\boldsymbol{\Sigma}_{\mathbf{x}}}^2\right]$ where the rightmost expectation is with respect to $\mathbf{x}^{(\mathrm{test})}, y^{(\mathrm{test})}, \mathcal{D}, \left\{\widetilde{\mathcal{D}}_j\right\}_{j=1}^{m}, \boldsymbol{\beta}$.

4 The General Case: Analysis for General Forms of $\mathbf{H}_j$, $\widetilde{\mathbf{H}}_j$ and $\boldsymbol{\Sigma}_{\mathbf{x}}$

Now we analyze a relatively general case (under Assumptions 3.1, 3.2) where the task relation matrices $\{\mathbf{H}_j\}_{j=1}^m$ are unknown. Namely, the transfer learning from (5) uses $\widetilde{\mathbf{H}}_j$ that may differ from the unknown $\mathbf{H}_j$.

While the target input covariance $\boldsymbol{\Sigma}_{\mathbf{x}}$ can be anisotropic, our theory assumes that the source input is isotropic Gaussian (we will empirically-relax this assumption in Section 5.5).

Assumption 4.1. The source input is isotropic Gaussian, i.e., $\mathbf{z}_j \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}_d\right)$, for all source tasks.

Assumption 4.2. The target data is distributed $\mathbf{x} = \boldsymbol{\Sigma}_{\mathbf{x}}^{1/2}\mathbf{t}$, where $\boldsymbol{\Sigma}_{\mathbf{x}}$ has bounded spectral norm and $\mathbf{t}$ has i.i.d. entries with mean 0, variance 1, and finite $(8+\delta)$-th moment for some $\delta > 0$.

Assumption 4.3 (Asymptotic setting)**.** The quantities $d, n, \{\widetilde{n}_j\}_{j=1}^m \to \infty$ such that the target task parameterization level satisfies $\frac{d}{n} \to \gamma_{\mathrm{tgt}} \in (0, \infty)$; for $j \in \{1, \ldots, m\}$, the j^{th} source task parameterization level satisfies $\frac{d}{\widetilde{n}_j} \to \gamma_{\mathrm{src},j} \in (0, \infty)$; the j^{th} task relation model $\boldsymbol{\theta}_j = \mathbf{H}_j\boldsymbol{\beta} + \boldsymbol{\eta}_j$ includes an operator $\mathbf{H}_j$ that satisfies $\frac{1}{d}\left\|\mathbf{H}_j\right\|_{\mathrm{F}}^2 \to \kappa_{\mathbf{H}_j}$; the assumed operator $\widetilde{\mathbf{H}}_j$ satisfies $\frac{1}{d}\left\|\widetilde{\mathbf{H}}_j\right\|_{\mathrm{F}}^2 \to \kappa_{\widetilde{\mathbf{H}}_j}$; unless otherwise specified, the number m of pretrained models is fixed and finite.

We analyze the generalization performance of the target task using the expected test squared error.

Theorem 4.4. *Under Assumptions 3.1, 3.2, 4.1-4.3, the expected test error of the closed-form solution* $\widehat{\boldsymbol{\beta}}_{\mathrm{TL}}$ *from* (7) *is*

$$\bar{\mathcal{E}}_{\mathrm{TL}} \to \sigma_\epsilon^2\left(1 + \gamma_{\mathrm{tgt}} \cdot \mathrm{Tr}\left\{\frac{1}{d}\mathbf{W}\boldsymbol{\Omega}^{-1}\right\} + \gamma_{\mathrm{tgt}}\mathrm{Tr}\left\{\left(\frac{\alpha_{\mathrm{TL}}^2}{\gamma_{\mathrm{tgt}}\sigma_\epsilon^2}\boldsymbol{\Gamma}_{\mathrm{TL},\infty}^m - \frac{\alpha_{\mathrm{TL}}}{d}\mathbf{I}_d\right)\boldsymbol{\Omega}^{-1}s\mathbf{W}\boldsymbol{\Omega}^{-1}\right\}\right) \tag{8}$$

where $\mathbf{W} \triangleq \mathbf{R}^{-1}\boldsymbol{\Sigma}_{\mathbf{x}}\mathbf{R}^{-1}$, $\boldsymbol{\Omega} \triangleq c(\alpha_{\mathrm{TL}})\mathbf{W} + \alpha_{\mathrm{TL}}\mathbf{I}_d$, $c(\alpha_{\mathrm{TL}})$ *is calculated by solving*

$$\frac{1}{c(\alpha_{\mathrm{TL}})} - 1 = \frac{\gamma_{\mathrm{tgt}}}{d}\,\mathrm{Tr}\left\{\mathbf{W}\left(c(\alpha_{\mathrm{TL}})\mathbf{W} + \alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}\right\}, \tag{9}$$

$s \triangleq c'(\alpha_{\mathrm{TL}}) + 1$, $c'(\alpha_{\mathrm{TL}})$ *is computed by solving (10)*

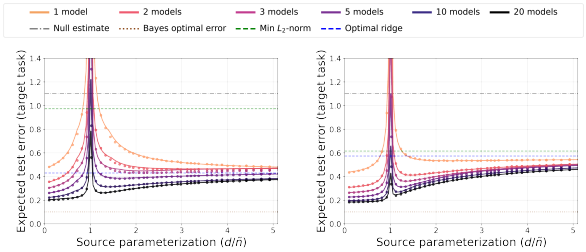
$$c'(\alpha_{\mathrm{TL}}) = \frac{\frac{\gamma_{\mathrm{tgt}}}{d}\left\|\mathbf{W}\left(c(\alpha_{\mathrm{TL}})\mathbf{W} + \alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}\right\|_F^2}{\left(c(\alpha_{\mathrm{TL}})\right)^{-2} - \frac{\gamma_{\mathrm{tgt}}}{d}\left\|\mathbf{W}\left(c(\alpha_{\mathrm{TL}})\mathbf{W} + \alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}\right\|_F^2} \tag{10}$$

where $\|\cdot\|_F$ *is the Frobenius norm.*

In (20), $\mathbf{\Gamma}^{m}_{\mathrm{TL},\infty}$ is formulated using m matrices $\mathbf{\Gamma}^{\mathrm{single},1}_{\mathrm{TL},\infty}, \ldots, \mathbf{\Gamma}^{\mathrm{single},m}_{\mathrm{TL},\infty}$ that reflect transfer learning with a single pretrained model from each of the source tasks; see (21) for the definition of these matrices, and note there the effect of the task relation misspecification via $\mathbf{\Delta}_j \triangleq \mathbf{H}_j - \widetilde{\mathbf{H}}_j$. Moreover, the formulation (20) uses the definition of $\mathbf{A}_j \triangleq \widetilde{\mathbf{H}}_j^T \left(\rho_{j,\infty}\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)$ where

$$\forall j \in \{1,\ldots,m\}, \quad \rho_j \triangleq \begin{cases} 1 & \text{for } \widetilde{n}_j \geq d \\ \frac{\widetilde{n}_j}{d} & \text{for } \widetilde{n}_j < d \end{cases}, \qquad \rho_j \to \rho_{j,\infty} \triangleq \begin{cases} 1 & \text{for } \gamma_{\mathrm{src},j} \leq 1 \\ \frac{1}{\gamma_{\mathrm{src},j}} & \text{for } \gamma_{\mathrm{src},j} > 1. \end{cases} \tag{11}$$

Theorem 4.4 (proved in Appendix F.2) importantly shows how the generalization performance of transfer learning with multiple pretrained models extends transfer learning with a single pretrained model (see detailed discussion in Appendix F.3): **(i) The error depends on a weighted combination of matrices that reflect transfer learning using each of the pretrained models alone. (ii) The error depends on the interactions between pairs of source-target task-relation operators of different source tasks.**



(a) $\gamma_{\mathrm{tgt}} = \frac{8}{7}, \sigma_\xi^2 = \sigma_\eta^2 = 0.5$ (b) $\gamma_{\mathrm{tgt}} = 4, \sigma_\xi^2 = \sigma_\eta^2 = 0.5$

Figure 1: **Test error in the general case of Theorem 4.4.** Here, $\mathbf{H}_j$ is energy preserving subspace of dimension $\frac{d}{2}$ (see Appendix H.1), $d = 128$, and the assumed task relation is $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$. Target covariance is $\mathbf{\Sigma}_{\mathbf{x}} = \mathbf{I}_d$ (left subfig.) and exponential decay $(\mathbf{\Sigma}_{\mathbf{x}})_{il} = 0.5^{|i-l|}$ (right subfig.) See Appendix T.1 for more experiments.

Figures 1 and T.1 compare our analytical formulations from Theorem 4.4 and the empirical evaluation from the corresponding experimental settings. The empirical evaluations (circle markers) of the expected test error match well with the analytical evaluations (solid lines). These figures show expected test error graphs as function of the source task parameterization level $d/\widetilde{n}$ for a fixed target task parameterization level d/n; i.e., in the experiments we set the input dimension d and number of target examples n fixed, and vary the number of source examples $\widetilde{n}$ that each of the pretrained models has. In figures that refer to $\widetilde{n}$, all the source tasks have the same *number* of training examples, i.e., $\widetilde{n}_j = \widetilde{n}$ and the same source parameterization level.

These figures show the expected test error graphs for transfer learning with m pretrained models, including a single pretrained model and up to 20 pretrained models. The comparisons also include the solutions to the target task without transfer learning (nor the pretrained models): minimum ℓ_2-norm solution for least squares regression in green dashed line, optimally tuned ridge regression in blue dashed line, the null estimate of all parameters zero in a black dashed line (its expected error is $\sigma_\epsilon^2 + \frac{b}{d}\,\mathrm{Tr}\,\{\mathbf{\Sigma}_{\mathbf{x}}\}$) and the Bayes-optimal prediction error in brown dotted which is the irreducible target noise σ_ϵ^2. Throughout the paper, we set σ_ϵ^2 to 0.1. Our experiments consider task relation matrices $\mathbf{H}_j$ of the following forms: subspace projection matrices of a r-dimensional subspace ($r < d$), energy preserving projection matrices of a r-dimensional subspace, circulant matrices with high condition number $\kappa(\mathbf{H}_j)$, and the identity $\mathbf{I}_d$; for more details see Appendix H.1. For additional experimental details, see Appendix H.2.2.

Figure 1 and T.1 demonstrate that using multiple pretrained models can be much more beneficial than using a single pretrained model (see discussion in Appendix F.4): **Using multiple pretrained models can resolve negative transfer that occurs for a single pretrained model** (negative transfer is when a non-transfer method such as ridge regression has lower test error than transfer learning). Yet, the generalization gains due to adding more pretrained models can diminish when there are many pretrained models.

5 Overparameterization Prevents the Multi-Model Benefits

5.1 Optimally Tuned Transfer Learning with Multiple Pretrained Models

In Appendix G.1, Theorem G.1, we formulate the optimal transfer hyperparameter $\alpha^{\mathrm{opt}}_{\mathrm{TL}}$ and the optimally-tuned test error in the nonasymptotic case. **Importantly, the transfer learning error**

expression in (40) is the same as for optimally tuned ridge regression (88), except for the different scaling of the identity matrix; here, the scaling $mn\alpha_{\mathrm{TL}}^{\mathrm{opt}}$ depends on the number m of pretrained models and the optimal hyperparameter for transfer learning with the given m pretrained models. This correspondence to the ridge regression will be useful in our analysis.

For further analysis, we will assume all source tasks have the same amount of training data and the same noise statistics (see Assumption G.2). For the asymptotic setting of Assumption 4.3, there is $\gamma_{\text{src}} \in (0, \infty)$ such that $\gamma_{\text{src},j} = \gamma_{\text{src}}$ for all $j \in \{1, \dots, m\}$. Then, the optimal hyperparameter $\alpha_{\mathrm{TL}}^{\mathrm{opt}}$ from Theorem G.1 is simplified as follows.

Corollary 5.1. *Under Assumption G.2, the optimal transfer learning hyperparameter (37):*

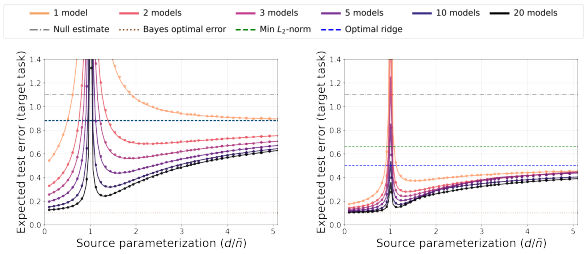
$$\text{for } d \notin \{\widetilde{n}-1, \widetilde{n}, \widetilde{n}+1\}, \quad \alpha_{\mathrm{TL}}^{\mathrm{opt}} = \frac{\sigma_\epsilon^2}{nC + \frac{bn}{d}(m-1)(1-\rho)^2}. \tag{12}$$

Using the results for ridge regression by Dobriban and Wager [9], we provide the following theorem for the asymptotic error of transfer learning with multiple pretrained models. The proof outline is in Appendix J; also, see Figs. 2, T.2.

Theorem 5.2. *Under Assumptions 3.1, 3.2, 4.1-4.3, G.2, and well-specified orthonormal task relation* $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$, $\mathbf{H}_j^T \mathbf{H}_j = \mathbf{I}_d$, *with* $\mathbf{\Sigma}_{\mathbf{x}} = \mathbf{I}_d$ *and* $d \notin \{\widetilde{n}-1, \widetilde{n}, \widetilde{n}+1\}$ *the asymptotic test error of transfer learning with* m *pretrained models is* $\bar{\mathcal{E}}_{\mathrm{TL}} \to \sigma_\epsilon^2 \left(1 + \gamma_{\mathrm{tgt}} \cdot g(-m\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}}; \gamma_{\mathrm{tgt}})\right)$ *where the limiting value of the optimal hyperparameter* $\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}}$ *is formulated in (41), and the Stieltjes transform of the Marchenko-Pastur distribution, i.e., the limiting spectral distribution of the sample covariance of n samples drawn from an isotropic distribution, i.e.* $\mathbf{\Sigma}_x = \mathbf{I}_d$, *is denoted as* $g(-m\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}}; \gamma_{\mathrm{tgt}})$ *and formulated in (G.2).*

5.2 Negative Transfer

Negative transfer occurs when a transfer-learning estimator generalizes worse than a non-transfer baseline. In linear regression, optimally tuned ridge regression serves as an ideal baseline; under isotropic Gaussian data and parameter assumptions, it is the minimum mean square error (MMSE) estimate (i.e., achieves the minimum test error among all non-transfer solutions to the target task). Then, negative transfer is identified if transfer learning yields a higher test error than ridge regression, as illustrated in Figs. 1a, 2a, T.1a, T.1c, T.1d, T.2c. For a detailed analysis, see App. G.3. To avoid negative transfer with multiple pretrained models, we provide the next theorem (proof in App. M).



(a) $\gamma_{\mathrm{tgt}} = 4, \sigma_\xi^2 = \sigma_\eta^2 = 0.5$ (b) $\gamma_{\mathrm{tgt}} = \frac{4}{3}, \sigma_\xi^2 = \sigma_\eta^2 = 0.1$

Figure 2: **Test error in the simple case of Theorem 5.2**. $\mathbf{H}_j = \widetilde{\mathbf{H}}_j = \mathbf{I}_d$. See App. T.2 for more experiments.

Theorem 5.3. *Under the assumptions of Theorem G.1 and Assumption G.2, our transfer learning with* m *pretrained models is beneficial when* $d \notin \{\widetilde{n}-1, \widetilde{n}, \widetilde{n}+1\}$, *if* $\sigma_\eta^2 + \frac{d\sigma_\xi^2}{|d-\widetilde{n}|-1} < b\left(m + (m-1)(1-\rho)\right)$ *where* ρ *is defined based on* $\widetilde{n}$ *and* d *according to (11).*

The condition in Theorem 5.3 requires oracle knowledge of $b, \sigma_\eta^2, \sigma_\xi^2$ and hence it is not directly useful in practice. As we explain in Appendix G.3 for the considered setting, transfer learning using m pretrained models does not perform better than ridge regression performance for any overparameterization level if $\sigma_\eta^2 + \sigma_\xi^2 > b(2m-1)$. Hence, **if the pretrained models are overparameterized, a necessary condition for beneficial transfer is to use sufficiently many pretrained models.**

5.3 Consistency of Optimally Tuned Transfer Learning

Now, we use the error formulation in Theorem 5.2 to elucidate the transfer learning performance as the number of pretrained models increases, i.e., the consistency of the transfer learning with respect to asymptotically increasing the number of pretrained models.

Recall, the Bayes optimal error of the target task is σ_ϵ^2, which is theoretically achievable by setting $\widehat{\boldsymbol{\beta}}$ as the true $\boldsymbol{\beta}$ of the target data model. This is the best prediction performance possible for test data of the target task.

Theorem 5.4. *Under Assumptions 3.1, 3.2, 4.1 and G.2, $d \notin \{\widetilde{n}-1, \widetilde{n}, \widetilde{n}+1\}$, $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{I}_d$, and well-specified orthonormal task relation $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$, $\mathbf{H}_j^T \mathbf{H}_j = \mathbf{I}_d$:*

- *For m **underparameterized** pretrained models ($\gamma_{src} < 1$), the transfer learning is **consistent**:*

$$\lim_{m\to\infty} \bar{\mathcal{E}}_{\mathrm{TL}} = \sigma_\epsilon^2. \tag{13}$$

- *For m **overparameterized** pretrained models ($\gamma_{src} > 1$), the transfer learning is **inconsistent**:*

$$\lim_{m\to\infty} \bar{\mathcal{E}}_{\mathrm{TL}} > \sigma_\epsilon^2. \tag{14}$$

The error $\bar{\mathcal{E}}_{\mathrm{TL}}$ in (13), (14), refers to the error in Thm. 5.2 that already considers asymptotic $d, n, \widetilde{n}$.

The proof of Theorem 5.4 is in Appendix N.2.

In Appendix N.3, we discuss our results that showcase inconsistency when the pretrained models are overparameterized. These emphasize that the potential benefits of using multiple pretrained models can be significantly restricted by their overparameterization. We will address this next.

5.4 Debiasing of Overparameterized Pretrained Models

Theorem 5.4 showed us that under the orthonormal and well specified task relation, the Bayes optimal error is not reachable as the number of pretrained models increases – if these pretrained models are overparameterized. This is mostly because they all suffer from overparameterization bias that attenuates the information of the true parameters in the pretrained model. Specifically, this is reflected in the expectation of overparameterized $\widehat{\boldsymbol{\theta}}_j$ given $\boldsymbol{\beta}$, in which $\boldsymbol{\beta}$ is attenuated by a factor $\frac{\widetilde{n}_j}{d}$ (the inverse overparameterization level of the pretrained model):

$$\text{for } d \le \widetilde{n}_j, \ \ \mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j \mid \boldsymbol{\beta}\right] = \mathbf{H}_j\boldsymbol{\beta}; \qquad \text{for } d > \widetilde{n}_j, \ \ \mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j \mid \boldsymbol{\beta}\right] = \frac{\widetilde{n}_j}{d}\mathbf{H}_j\boldsymbol{\beta}. \tag{15}$$

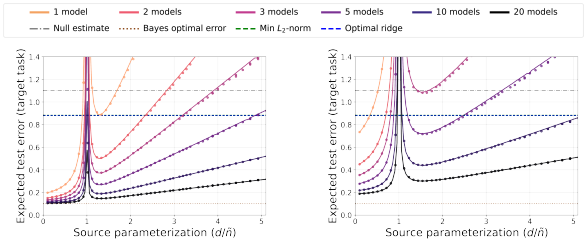
See details in Appendix K. Next, recall that bias of a predictor $\widehat{\boldsymbol{\beta}}$ is defined as $\mathrm{Bias}(\widehat{\boldsymbol{\beta}}) \triangleq \mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right] - \boldsymbol{\beta}$. The predictor is unbiased if $\mathrm{Bias}(\widehat{\boldsymbol{\beta}}) = \mathbf{0}$. Our analysis (see (121) in Appendix Q.1) shows that the bias of the transfer learning predictor $\widehat{\boldsymbol{\beta}}_{\mathrm{TL}}$ depends on the difference between $\widetilde{\mathbf{H}}_j\boldsymbol{\beta}$ and $\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j \mid \boldsymbol{\beta}\right]$ from (15). Hence, using $\widetilde{\mathbf{H}}_j = \frac{\widetilde{n}_j}{d}\mathbf{H}_j$ for overparameterized pretrained models provides unbiased transfer learning, as stated next (proof in Appendix Q.1).

Theorem 5.5. *Under Assumptions 3.1, 3.2, 4.1, assuming $\sum_{j=1}^m \mathbf{H}_j^T\mathbf{H}_j$ is full rank, our transfer learning (7) using m pretrained models provides an unbiased predictor if*

- *$\widetilde{\mathbf{H}}_j = \mathbf{H}_j$ for any underparameterized pretrained model, i.e., $\forall j \in \{1,\dots,m\}$ such that $d \le \widetilde{n}_j$;*
- *$\widetilde{\mathbf{H}}_j = \frac{\widetilde{n}_j}{d}\mathbf{H}_j$ for any overparameterized pretrained model, i.e., $\forall j \in \{1,\dots,m\}$ such that $d > \widetilde{n}_j$.*

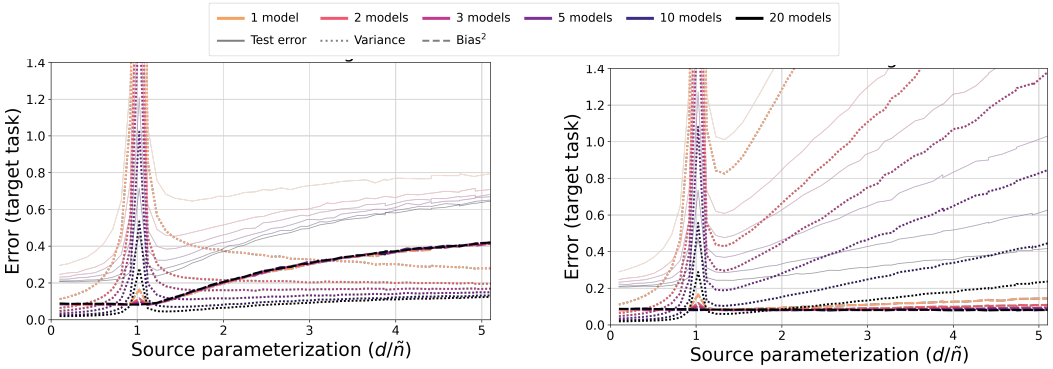
Theorem 5.5 motivates our proposed approach in Algorithm 1 for overparameterization debiasing when the true operators $\{\mathbf{H}_j\}_{j=1}^m$ are **unknown**; this is demonstrated in Fig. 3. Fig. 4 demonstrates the bias-variance decomposition of the test error, for the well-specified and misspecified case, and the usefulness of the overparameterization debiasing; see the detailed discussion in Appendices O.2, P, Q.

Our theory in Appendix O.3 showcases the benefits of the overparameterization debiasing. Our theory shows that debiasing enables consistency (proved in Appendix R.2).



(a) $\sigma_\xi^2 = \sigma_\eta^2 = 0.1$ (b) $\sigma_\xi^2 = \sigma_\eta^2 = 0.5$

Figure 3: In Fig. 3a, $\mathbf{H}_j = \mathbf{I}_d$, and in Fig. 3b, $\mathbf{H}_j$ is a subspace projection of dimension $\frac{3d}{4}$; see Appendix H.1. The assumed task relation is $\widetilde{\mathbf{H}}_j = \rho\,\mathbf{I}_d$ and $\gamma_{\mathrm{tgt}} = 4$. For more figures, see Appendix T.3.



(a) Without debiasing (b) With debiasing

Figure 4: **Bias-variance decomposition.** Dashed and dotted lines denote the bias and variance terms, respectively. Here, $\mathbf{H}_j$ is circulant with condition number $\kappa = 1000$, and $\sigma_\xi^2 = \sigma_\eta^2 = 0.1$. Without debiasing (a) uses $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$; with debiasing (b) uses $\widetilde{\mathbf{H}}_j = \rho_j \mathbf{I}_d$. See App. P for more results.

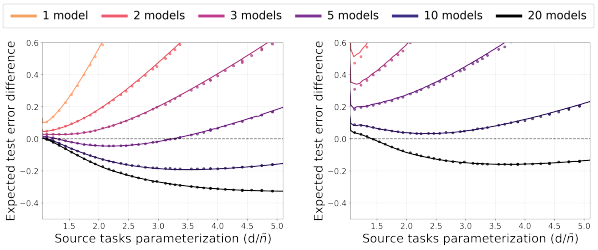
Theorem 5.6. *Under Assumptions 3.1, 3.2, 4.1 and G.2, overparameterized pretrained models* $\gamma_{\text{src}} > 1$, $\mathbf{\Sigma_x} = \mathbf{I}_d$, *task relation* $\mathbf{H}_j = \mathbf{I}_d$, $\forall j \in \{1, \dots, m\}$, *and setting* $\widetilde{\mathbf{H}}$ *using the debiasing Algorithm 1. Then, the transfer learning using* m ***overparameterized*** *pretrained models of a fixed pretrained overparameterization level* γ_{src} *and* ***debiasing*** *is* ***consistent****:*

$$\lim_{m \to \infty} \bar{\mathcal{E}}_{\text{TLdeb}} = \sigma_\epsilon^2. \tag{16}$$

$\bar{\mathcal{E}}_{\text{TLdeb}}$ *in (16) refers to the error in (113) that already considers asymptotic* $d, n, \widetilde{n}$.

Theorem 5.6 implies that the overparameterization debiasing can provide a consistent predictor for the target task, i.e., the debiasing can help to get close to the Bayes optimal error as sufficiently many pretrained models are used. **Remarkably, Theorem 5.4 shows that overparameterized pretrained models prevent transfer learning consistency; Theorem 5.6 shows (in the isotropic identity-relation setting) that the proposed overparameterization debiasing resolves this problem and enables transfer learning consistency.**

As overparameterization debiasing can help to achieve the Bayes optimal error when asymptotically many pretrained models are used ($m \to \infty$), we now ask when debiasing is beneficial for a finite number m of pretrained models. The test error formulations for with/without debiasing ((112)/(40), respectively) are the same except for the scaling of the identity matrix; consequently, debiasing is beneficial when $\alpha_{\text{TL}}^{\text{opt}} < \alpha_{\text{deb}}^{\text{opt}} \frac{\widetilde{n}^2}{d^2}$ and then the following theorem holds (proof in Appendix R.3).



(a) $\sigma_\xi^2 = \sigma_\eta^2 = 0.05$ (b) $\sigma_\xi^2 = \sigma_\eta^2 = 0.5$

Figure 5: **Difference** in target test error between transfer learning **with and without debiasing** for $\gamma_{\text{tgt}} = 4$. Negative values denote beneficial debiasing. Here, $\mathbf{H}_j = \mathbf{I}_d$; see Appendix T.4 for other task relations.

Theorem 5.7. *Under Assumptions 3.1, 3.2, 4.1,G.2, target data with isotropic input covariance, well-specified orthonormal task relation* $\widetilde{\mathbf{H}}_j = \rho_j \mathbf{H}_j$, $\mathbf{H}_j^T \mathbf{H}_j = \mathbf{I}_d$ *and* $d \geq \tilde{n} + 2$, *the overparameterization debiasing of Alg. 1 is beneficial when* $\left(\sigma_\eta^2 + \frac{d\sigma_\xi^2}{d - \widetilde{n} - 1}\right)\left(\frac{d}{\widetilde{n}} + \frac{2d}{d - \widetilde{n}}\right) < \left(m - 1 - \frac{d}{\widetilde{n}}\right) b$.

Corollary 5.8. *In the setting of Theorem 5.7, a necessary but not sufficient condition for beneficial debiasing is to use at least one more model than the overparameterization level of the pretrained models, i.e.,* $m > 1 + \frac{d}{\widetilde{n}}$.

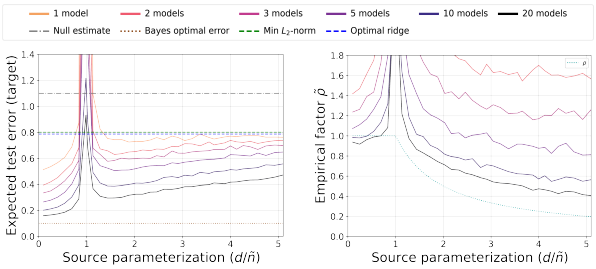
The oracle condition in Theorem 5.7 implies that, for any fixed pretrained overparameterization and noise levels, there is a number m of sufficiently many pretrained models that yields beneficial

debiasing. This behavior is shown in Fig. 5. Moreover, the condition for beneficial debiasing in Theorem 5.7 depends *non-monotonically* on the pretrained overparameterization level $\frac{d}{\tilde{n}}$. This implies that beneficial debiasing requires sufficiently many pretrained models whose overparameterization levels are not too high. See detailed discussion in Appendix R.4. Moreover, in Figs. T.4 we show that the *qualitative* principles of Theorem 5.7 and Corollary 5.8 can be generally observed in our more complex settings. In Appendix U, we outline how much of our proposed theory extends beyond the specialized assumptions.

5.5 Anisotropic Source Data Model

Debiasing in the isotropic *source* model is motivated by the pretrained model expectation in (15) that, for asymptotic overparameterization $\gamma_{\text{src},j} > 1$, implies the same *scalar* shrinkage factor $1/\gamma_{\text{src},j}$ for all the coordinates. Yet, for anisotropic source input, (15) no longer holds: the mean learned predictor can depend on *anisotropic* shrinkage stemming from the unknown anisotropic covariance $\mathbf{\Sigma_z}$.

In practice, $\mathbf{\Sigma_z}$ is unknown and the source data $\mathbf{Z}_j$ is unavailable in our transfer setting; thus estimating the anisotropic shrinkage is impractical without further assumptions. Hence, we **replace the isotropic factor** $1/\gamma_{\text{src}}$ in the debiasing surrogate $\widetilde{\mathbf{H}}_j = \frac{1}{\gamma_{\text{src}}}\mathbf{I}_d$ by a *single scalar* $\tilde{\rho} > 0$, **chosen empirically by validation** and used as $\widetilde{\mathbf{H}}_j = \tilde{\rho}\mathbf{I}_d$ (Algorithm 2).



(a) Test error (b) Empirical factor

Figure 6: **Empirical tuning of the shrinkage factor.** The cyan dotted line is the isotropic baseline $\rho = 1/\gamma_{\text{src}}$. Here, $\mathbf{H}_j$ is energy preserving with dimension $\frac{3d}{4}$, the assumed task relation is $\widetilde{\mathbf{H}}_j = \tilde{\rho}\,\mathbf{I}_d$, and $(\mathbf{\Sigma_x})_{il} = (\mathbf{\Sigma_z})_{il} = 0.5^{|i-l|}$. For experiments under more skewed covariance structures, see Fig. T.5.

Figs. 6, T.5 show the empirically-chosen $\tilde{\rho}$ for anisotropic Gaussian sources $\mathbf{z}_j \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma_z})$ where $\mathbf{\Sigma_z}$ is an exponential-decay covariance (Appendix H.2.1). For comparison, the baseline (cyan dotted line) is the simple isotropic debiasing $\rho = 1/\gamma_{\text{src}}$. For each learning we *jointly* tune $\tilde{\rho}$ and α_{TL} using cross-validation, and report the average (over 100 experiments) test error and chosen $\tilde{\rho}$. Notably, $\tilde{\rho}$ can exceed 1, mostly for underparameterized and low number of overparameterized source models; this can be interpreted as a *hybrid* transfer learning with an implicit ridge regularization (Appendix S.2). For sufficiently many overparameterized pretrained models, we get $\tilde{\rho} < 1$, qualitatively conforming with our overparameterization debiasing theory for the isotropic case.

6 Conclusion

We studied transfer learning using multiple pretrained models for a target linear regression task. We showed that while multiple pretrained models can significantly improve transfer learning performance, excessive overparameterization introduces bias that can degrade results. To mitigate this, we proposed a debiasing approach that reduces this bias and yields beneficial transfer as the number of models increases. These findings provide theoretical and conceptual foundations for leveraging multiple overparameterized models, offering new directions for research beyond linear regression. In Appendix V we discuss the potential implications for neural networks.

References

[1] H. Bastani. Predicting with proxies: Transfer learning in high dimension. *Management Science*, 67(5):2964–2984, 2021.

[2] M. Belkin, D. Hsu, and J. Xu. Two models of double descent for weak features. *SIAM Journal on Mathematics of Data Science*, 2(4):1167–1180, 2020.

[3] E. Craig, M. Pilanci, T. Le Menestrel, B. Narasimhan, M. A Rivas, S.-E. Gullaksen, R. Dehghannasiri, J. Salzman, J. Taylor, and R. Tibshirani. Pretraining and the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, page qkaf050, 2025.

[4] Y. Dar and R. G. Baraniuk. Double double descent: On generalization errors in transfer learning between linear regression tasks. *SIAM Journal on Mathematics of Data Science*, 4(4): 1447–1472, 2022.

[5] Y. Dar, V. Muthukumar, and R. G Baraniuk. A farewell to the bias-variance tradeoff? An overview of the theory of overparameterized machine learning. *arXiv preprint arXiv:2109.02355*, 2021.

[6] Y. Dar, D. LeJeune, and R. G. Baraniuk. The common intuition to transfer learning can win or lose: Case studies for linear regression. *SIAM Journal on Mathematics of Data Science*, 6(2): 454–480, 2024.

[7] O. Dhifallah and Y. M. Lu. Phase transitions in transfer learning for high-dimensional perceptrons. *Entropy*, 23(4):400, 2021.

[8] E. Dobriban and Y. Sheng. WONDER: Weighted one-shot distributed ridge regression in high dimensions. *The Journal of Machine Learning Research*, 21(66):1–52, 2020.

[9] E. Dobriban and S. Wager. High-dimensional asymptotics of prediction: Ridge regression and classification. *The Annals of Statistics*, 46(1):247–279, 2018.

[10] Y. Duan and K. Wang. Adaptive and robust multi-task learning. *The Annals of Statistics*, 51(5): 2015–2039, 2023.

[11] F. Gerace, L. Saglietti, S. S. Mannelli, A. Saxe, and L. Zdeborová. Probing transfer learning with a model of synthetic correlated datasets. *Mach. Learn.: Sci. Technol.*, 3(1):015030, 2022.

[12] T. Hastie, A. Montanari, S. Rosset, and R. J. Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *Annals of statistics*, 50(2):949, 2022.

[13] S. Hendy and Y. Dar. TL-PCA: Transfer learning of principal component analysis. *arXiv preprint arXiv:2410.10805*, 2024.

[14] S. Li, T. Cai, and H. Li. Transfer learning for high-dimensional linear regression: Prediction, estimation and minimax optimality. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(1):149–173, 2022.

[15] S. Li, L. Zhang, T. Cai, and H. Li. Estimation and inference for high-dimensional generalized linear models with knowledge transfer. *Journal of the American Statistical Association*, 119 (546):1274–1285, 2024.

[16] K. Meng, Y. Gai, X. Wang, M. Yao, and X. Sun. Transfer learning for high-dimensional linear regression via the elastic net. *Knowledge-Based Systems*, 304:112525, 2024.

[17] P. Nakkiran, P. Venkat, S. Kakade, and T. Ma. Optimal regularization can mitigate double descent. In *International Conference on Learning Representations (ICLR)*, 2021.

[18] D. Obst, B. Ghattas, J. Cugliari, G. Oppenheim, S. Claudel, and Y. Goude. Transfer learning for linear regression: a statistical test of gain. *arXiv preprint arXiv:2102.09504*, 2021.

[19] F. Rubio and X. Mestre. Spectral convergence for a general class of random matrices. *Statistics & probability letters*, 81(5):592–602, 2011.

[20] N. Singh and S. Diggavi. Representation transfer learning via multiple pre-trained models for linear regression. *IEEE Journal of Selected Topics in Signal Processing*, 19(1):208–220, 2025.

[21] Y. Tian, Y. Gu, and Y. Feng. Learning from similar linear representations: Adaptivity, minimaxity, and robustness. *arXiv preprint arXiv:2303.17765*, 2023.

[22] Ye Tian and Yang Feng. Transfer learning under high-dimensional generalized linear models. *Journal of the American Statistical Association*, 118(544):2684–2697, 2023.

Appendices

A Limitations

Our work is theoretical with empirical support of claims. The theory and extensive experiments focus on linear models and synthetic data. The theory assumes isotropic source data and allows anisotropic target data; the experiments examine both isotropic and anisotropic covariances for the source data and target data.

B Broader Impact

This paper presents a mathematical theory and a foundational debiasing principle for transfer learning using multiple pretrained models. Our foundational insights can potentially contribute to improve the area of overparameterized transfer learning and the use of pretrained model. It should be noted that using more pretrained models can unknowingly introduce more societal and other biases to the target model; therefore, the learning designers should be aware of the pretrained models that they use and implement relevant safeguards to the target model.

C Experiments Compute Resources

All experiments were conducted on an internal CPU cluster.

D Related Works: Additional Discussion

D.1 Transfer Learning of Linear or Simple Models using a Single Pretrained Model

Transfer learning of linear regression using a *single* pretrained model has received considerable attention in recent years. Obst et al. [18] examined gradient descent training, initialized with a pretrained model, for ordinary least squares (i.e., underparameterized models). Bastani [1] studied the single underparameterized source linear case where the difference between the true parameters of the source and target tasks is sparse. Dar and Baraniuk [4] analyzed the minimum ℓ_2-norm interpolating solution of overparameterized least squares with subset of target parameters set fixed on their corresponding pretrained source parameters. Dar et al. [6] studied transfer learning for high-dimensional linear regression with a penalty on the distance between the learned target parameters and the pretrained parameters. Craig et al. [3] studied lasso regression using a pretrained model. Notable examples for analyses of pretraining in simple models beyond linear regression include the regression/classification perceptron models [7], two-layer neural networks [11], and principal component analysis [13].

D.2 Main Differences between the Work by Dar et al. [6] and Ours

We study transfer learning with multiple pretrained models; in [6], there was only a single pretrained model. Our learning setting and error formulation (both without the debiasing) are extensions of [6] from single to multiple pretrained models; importantly, the extended formulas are not the same as in [6]. The purpose of these extensions is to study new research questions completely different than [6].

Our main research questions and findings stem from our multi-model setting and are new and significant:

- We elucidate that for beneficial transfer learning, the number of pretrained models should be sufficiently high with respect to the learning setting.
- If the pretrained models are overparameterized, multiple pretrained models can be necessary for beneficial transfer learning.
- We propose overparameterization debiasing for making multiple pretrained models more beneficial for transfer learning.
- We show that overparameterized pretrained models make the transfer learning predictor inconsistent, i.e., its generalization performance does not approach optimality when using

more pretrained models. Remarkably, using our overparameterization debiasing, the transfer learning can provide a consistent predictor that benefits from more pretrained models.

- We examine the effects of debiasing and multiple pretrained models on the bias-variance tradeoff. We show that the overparameterization debiasing reduces the bias but increases the variance, which in turn can be reduced by using more pretrained models. Hence, we get a beneficial bias-variance tradeoff that resolves inherent difficulties of using many overparameterized pretrained models together in a transfer learning task.

D.3 Main Differences between the Work by Singh and Diggavi [20] and Ours

The main important differences between the work by Singh and Diggavi [20] and ours:

- Their target task is overparameterized but their source tasks are underparameterized.
  In contrast, our source tasks can be overparameterized. This adds new important aspects that play significant roles in the analysis and proposed algorithm, e.g., in our proposed overparameterization debiasing of the pretrained models.
- They use only the pretrained representation matrices.
  In contrast, we use the pretrained predictors – a setting that refers to another practical scenario where the pretrained models are available predictors that were not trained in architectural forms that were mainly intended to facilitate transfer learning of other tasks.
- Their method is composed of distinct phases: Combining the pretrained representation matrices via computing the orthonormal basis that spans their columns; using this orthonormal basis to learn an intermediate predictor for the target task using (least squares on) a subset of the target dataset; then using the intermediate predictor as initialization for learning a predictor via gradient descent on least squares using the remainder of the target dataset that was not used earlier in the process.
  In contrast, our transfer learning is algorithmically and conceptually simpler – we define a penalized least squares optimization that uses all the target data and all the pretrained predictors and solve it via its closed-form solution.

There are additional differences between the two works, including in the relations between the source and target models. All the above imply that in this research we provide many more contributions beyond the interesting work by Singh and Diggavi [20].

E Additional Details for Section 3

E.1 Additional Formulations and Definitions for the Proofs

The training dataset of the j^{th} source task satisfies $\mathbf{v}_j = \mathbf{Z}_j\boldsymbol{\theta}_j + \boldsymbol{\xi}_j$ where $\boldsymbol{\xi}_j \triangleq \left[\xi_j^{(1)}, \ldots, \xi^{(\widetilde{n}_j)}\right]^T$ is an unknown noise vector whose i^{th} component $\xi^{(i)}$ originates in the i^{th} data example's relation $v_j^{(i)} = \mathbf{z}_j^{(i),T}\boldsymbol{\theta}_j + \xi_j^{(i)}$ of the j^{th} source task.

The training dataset of the target task satisfies $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon} \triangleq \left[\epsilon^{(1)}, \ldots, \epsilon^{(n)}\right]^T$ is an unknown noise vector whose i^{th} component $\epsilon^{(i)}$ originates in the i^{th} data example's relation $y^{(i)} = \mathbf{x}^{(i),T}\boldsymbol{\beta} + \epsilon^{(i)}$.

E.2 Our Task Relation Compared to the Shared Representation

The existing literature (e.g., [10, 21]) proposes a shared representation model $\boldsymbol{\beta}_j = \mathbf{B}\mathbf{w}_j + \boldsymbol{\eta}_j$ to formulate multi task relations. This shared-representation formulation is naturally symmetric across tasks: all task parameters are represented in or near the common subspace spanned by $\mathbf{B}$, while each task has its own coefficient vector $\mathbf{w}_j$; it does not inherently designate one task as the target.

Our transfer learning setting and task relation model (3) are deliberately asymmetric. There is one target parameter vector $\boldsymbol{\beta}$ to be estimated, while the m source pretrained models provide auxiliary information about that target. Accordingly, the task relation between the true parameters of the target

and the j^{th} source $\boldsymbol{\theta}_j = \mathbf{H}_j\boldsymbol{\beta} + \boldsymbol{\eta}_j$ is target-centered and models each source parameter vector $\boldsymbol{\theta}_j$ as noisy, source-specific linearly transformed view of $\boldsymbol{\beta}$. Different sources may reveal different transformations or components of the target parameters, as illustrated by the identity, projection, energy-preserving projection, and ill-conditioned operators in our paper. This also motivates the inverse-problem interpretation mentioned after (3). Moreover, as shown throughout our analysis, the target-centered formulation is particularly useful for our theoretical question because the effect of isotropic source overparameterization appears explicitly in the conditional mean of the pretrained model $\widehat{\boldsymbol{\theta}}_j$: $\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j|\boldsymbol{\beta}\right] = \boldsymbol{\rho}_j\mathbf{H}_j\boldsymbol{\beta}$. It therefore lets us distinguish task-relation misspecification, $\mathbf{H}_j - \widetilde{\mathbf{H}}_j$, from minimum ℓ_2-norm attenuation and analyze their combined effect through $\boldsymbol{\rho}_j\mathbf{H}_j - \widetilde{\mathbf{H}}_j$.

In a shared-representation model, the relation between a source and the designated target is mediated by their common representation $\mathbf{B}$ and separate coefficient vectors $\mathbf{w}_j$; translating this formulation into our direct target-centered relation would require additional assumptions linking those coefficient vectors (i.e., each source to be linked to the single designated target).

Hence, our task relation model and the shared representation model are different, complementary and problem-dependent rather than either is universally preferred over the other.

E.3 A Source-Centered Tikhonov Interpretation to Our Transfer Learning

Define $\mathbf{R}^2 = \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j$ and $\widehat{\boldsymbol{\beta}}_{\text{src}} = \mathbf{R}^{-2}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widehat{\boldsymbol{\theta}}_j$. Then, under Assumption 3.1, $\widehat{\boldsymbol{\beta}}_{\text{src}}$ is exactly the source-only least squares estimate obtained via

$$\widehat{\boldsymbol{\beta}}_{\text{src}} = \underset{\mathbf{b}\in\mathbb{R}^d}{\arg\min} \sum_{j=1}^m \left\|\widetilde{\mathbf{H}}_j\mathbf{b} - \widehat{\boldsymbol{\theta}}_j\right\|_2^2. \tag{17}$$

Note that

$$\sum_{j=1}^m \left\|\widetilde{\mathbf{H}}_j\mathbf{b} - \widehat{\boldsymbol{\theta}}_j\right\|_2^2 = \left\|\mathbf{R}\left(\mathbf{b} - \widehat{\boldsymbol{\beta}}_{\text{src}}\right)\right\|_2^2 + \text{const} \tag{18}$$

where the constant does not depend on $\mathbf{b}$ and therefore does not affect an argminimum optimization w.r.t. $\mathbf{b}$. Therefore, our transfer learning (5) can be equivalently written as

$$\widehat{\boldsymbol{\beta}}_{\text{TL}} = \underset{\mathbf{b}\in\mathbb{R}^d}{\arg\min} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 + n\alpha_{\text{TL}}\left\|\mathbf{R}\left(\mathbf{b} - \widehat{\boldsymbol{\beta}}_{\text{src}}\right)\right\|_2^2. \tag{19}$$

The source-only estimator $\widehat{\boldsymbol{\beta}}_{\text{src}}$ does not use the target data and inherits source-estimation noise, task-relation noise, overparameterization attenuation, and any mismatch between $\mathbf{H}_j$ and $\widetilde{\mathbf{H}}_j$. The formulation (19) shows that our transfer learning (5) uses $\widehat{\boldsymbol{\beta}}_{\text{src}}$ as a structured source anchor while allowing the target data to correct it; α_{TL} controls this source-target tradeoff.

When using $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$ for all j, $\widehat{\boldsymbol{\beta}}_{\text{src}} = \frac{1}{m}\sum_{j=1}^m \widehat{\boldsymbol{\theta}}_j$. Hence, regularization toward the average of pretrained models is a special case of our transfer learning estimator.

F Additional Details and Discussion for Section 4

F.1 Theorem 4.4: Additional Details

For all j such that $\gamma_{\text{src},j} \neq 1$,

$$\boldsymbol{\Gamma}_{\text{TL},\infty}^m \triangleq \mathbf{R}^{-1}\left(\frac{b}{d}\sum_{\substack{j,l=1\\ l\neq j}}^m \mathbf{A}_j\,\mathbf{A}_l^T + \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\,\boldsymbol{\Gamma}_{\text{TL},\infty}^{\text{single},j}\,\widetilde{\mathbf{H}}_j\right)\mathbf{R}^{-1} \tag{20}$$

otherwise (i.e., $\gamma_{\text{src},j} = 1$), $\boldsymbol{\Gamma}_{\text{TL},\infty}^m = \infty$. In (20), $\mathbf{A}_j \triangleq \widetilde{\mathbf{H}}_j^T\left(\rho_{j,\infty}\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)$.

The single source tasks affect the overall via the following matrices: $\forall j \in \{1,\ldots,m\}$,

$$\boldsymbol{\Gamma}_{\mathrm{TL},\infty}^{\mathrm{single},j} \triangleq \begin{cases} \frac{1}{d}\left(\sigma_{\eta_j}^2 + \frac{\gamma_{\mathrm{src},j}\sigma_{\xi_j}^2}{1-\gamma_{\mathrm{src},j}}\right)\mathbf{I}_d + \frac{b}{d}\boldsymbol{\Delta}_j\boldsymbol{\Delta}_j^T, & \gamma_{\mathrm{src},j} < 1, \\ \infty, & \gamma_{\mathrm{src},j} = 1, \\ \dfrac{b}{d\gamma_{\mathrm{src},j}}\boldsymbol{\Delta}_j\boldsymbol{\Delta}_j^T + \dfrac{1}{d\gamma_{\mathrm{src},j}}\left(\sigma_{\eta_j}^2 + \dfrac{\gamma_{\mathrm{src},j}\sigma_{\xi_j}^2}{\gamma_{\mathrm{src},j}-1}\right)\mathbf{I}_d & \\ + \dfrac{b(\gamma_{\mathrm{src},j}-1)}{d\gamma_{\mathrm{src},j}^2}\left(\gamma_{\mathrm{src},j}\widetilde{\mathbf{H}}_j\widetilde{\mathbf{H}}_j^T - \mathbf{H}_j\mathbf{H}_j^T + \kappa_{\mathbf{H}_j}\mathbf{I}_d - \dfrac{1}{d}\mathrm{diag}\left(\mathbf{H}_j\mathbf{H}_j^T\right)\right), & \gamma_{\mathrm{src},j} > 1. \end{cases} \tag{21}$$

where

$$\boldsymbol{\Delta}_j \triangleq \mathbf{H}_j - \widetilde{\mathbf{H}}_j.$$

In (21), $\left[\mathbf{H}_j\mathbf{H}_j^T\right]_{kk}$ is the k-th component on the main diagonal of $\mathbf{H}_j\mathbf{H}_j^T$. The notation $\mathrm{diag}(\cdot)$ refers to the $d \times d$ diagonal matrix whose main diagonal values are the d given values.

F.2 Proof of Theorem 4.4

The expected test error of the transfer learning solution to the target task is developed as follows:

$$\begin{aligned} \bar{\mathcal{E}}_{\mathrm{TL}} &\triangleq \sigma_\epsilon^2 + \mathbb{E}\left[\left\|\widehat{\boldsymbol{\beta}}_{TL} - \boldsymbol{\beta}\right\|_{\boldsymbol{\Sigma}_\mathbf{x}}^2\right] \\ &= \sigma_\epsilon^2 + \mathbb{E}\left[\left\|\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\left(\mathbf{X}^T\mathbf{y} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widehat{\boldsymbol{\theta}}_j\right) - \boldsymbol{\beta}\right\|_{\boldsymbol{\Sigma}_\mathbf{x}}^2\right] && (22) \\ &= \sigma_\epsilon^2 + \mathbb{E}\left[\left\|\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1} \times\right.\right. && (23) \\ &\qquad \left.\left.\left(\mathbf{X}^T\mathbf{y} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widehat{\boldsymbol{\theta}}_j - \mathbf{X}^T\mathbf{X}\boldsymbol{\beta} - n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right\|_{\boldsymbol{\Sigma}_\mathbf{x}}^2\right] && (24) \\ &= \sigma_\epsilon^2 + \mathbb{E}\left[\left\|\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\left(\mathbf{X}^T\epsilon + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right)\right\|_{\boldsymbol{\Sigma}_\mathbf{x}}^2\right] && (25) \end{aligned}$$

Let us define

$$\mathbf{R} \triangleq \left(\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{1/2} \tag{26}$$

as the unique positive definite square root of the matrix $\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j$, which is a full-rank (by Assumption 3.1) positive definite matrix (as explained next) and therefore has a unique positive definite square root. Specifically, note that $\widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j$ is a symmetric and positive semi definite matrix for any $j \in \{1,\ldots,m\}$, hence, $\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j$ is symmetric and (using the full rank Assumption 3.1) positive definite matrix. Therefore, $\mathbf{R}$ is a positive definite, symmetric matrix.

Using the cyclic property of trace, the definition of $\mathbf{R}$ from (26), and the definitions

$$\mathbf{W} \triangleq \mathbf{R}^{-1}\boldsymbol{\Sigma}_\mathbf{x}\mathbf{R}^{-1} \tag{27}$$

$$\mathbf{X}_{\mathbf{R}^{-1}} \triangleq \mathbf{X}\mathbf{R}^{-1} \tag{28}$$

the test error can be further developed as follows:

$$\bar{\mathcal{E}}_{\mathrm{TL}} = \sigma_\epsilon^2 + \mathbb{E}\left[\mathrm{Tr}\left\{\left(\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\left(\mathbf{X}^T\epsilon + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right)\right)^T \times \right.\right.$$
$$\left.\left.\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\left(\mathbf{X}^T\epsilon + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right)\right\}\right]$$
$$= \sigma_\epsilon^2 + \mathbb{E}\left[\mathrm{Tr}\left\{\left(\mathbf{X}^T\epsilon + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right)\left(\mathbf{X}^T\epsilon + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right)^T \times \right.\right.$$
$$\left.\left.\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\right\}\right]$$

Using the zero mean of ϵ and its independence from other random elements, we get

$$\begin{aligned}
\bar{\mathcal{E}}_{\mathrm{TL}} =& \\
=&\sigma_\epsilon^2 + \mathbb{E}\left[\mathrm{Tr}\left\{\left(\sigma_\epsilon^2\mathbf{X}^T\mathbf{X} + n^2\alpha_{\mathrm{TL}}^2\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\sum_{j=1}^m \left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)^T\widetilde{\mathbf{H}}_j\right) \times \right.\right. \\
&\left.\left.\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\right\}\right] \\
=&\sigma_\epsilon^2 + \\
&+ \mathbb{E}\left[\mathrm{Tr}\left\{\left(n^2\alpha_{\mathrm{TL}}^2\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\sum_{j=1}^m \left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)^T\widetilde{\mathbf{H}}_j - \sigma_\epsilon^2 n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right) \times \right.\right. \\
&\left.\left.\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\right\}\right] \\
&+ \mathbb{E}\left[\mathrm{Tr}\left\{\sigma_\epsilon^2\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\right\}\right] \\
=&\sigma_\epsilon^2 + \mathbb{E}\left[\mathrm{Tr}\left\{\sigma_\epsilon^2\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\right\}\right] \\
&+ \mathbb{E}\left[\mathrm{Tr}\left\{\left(n^2\alpha_{\mathrm{TL}}^2\mathbf{R}^{-1}\left(\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\sum_{j=1}^m \left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)^T\widetilde{\mathbf{H}}_j\right)\mathbf{R}^{-1} - \sigma_\epsilon^2 n\alpha_{\mathrm{TL}}\mathbf{I}_d\right) \times \right.\right. \\
&\left.\left.\left(\mathbf{X}_{\mathbf{R}^{-1}}^T\mathbf{X}_{\mathbf{R}^{-1}} + n\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}\mathbf{W}\left(\mathbf{X}_{\mathbf{R}^{-1}}^T\mathbf{X}_{\mathbf{R}^{-1}} + n\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}\right\}\right]
\end{aligned}$$

By defining

$$\boldsymbol{\Gamma}_{\mathrm{TL}} = \mathbb{E}\left[\mathbf{R}^{-1}\left(\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}_j} - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right)\left(\sum_{j=1}^m \left(\widehat{\boldsymbol{\theta}_j} - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)^T\widetilde{\mathbf{H}}_j\right)\mathbf{R}^{-1}\right] \tag{29}$$

we can get

$$\bar{\mathcal{E}}_{\mathrm{TL}} = \sigma_\epsilon^2\left(1 + \frac{d}{n}\mathbb{E}\left[\mathrm{Tr}\left\{\frac{1}{d}\mathbf{W}\left(\frac{1}{n}\mathbf{X}_{\mathbf{R}^{-1}}^T\mathbf{X}_{\mathbf{R}^{-1}} + \alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}\right\}\right] \tag{30}$$
$$+\frac{d}{n}\mathbb{E}\left[\mathrm{Tr}\left\{\left(\frac{n\alpha_{\mathrm{TL}}^2}{d\sigma_\epsilon^2}\boldsymbol{\Gamma}_{\mathrm{TL}} - \frac{\alpha_{\mathrm{TL}}}{d}\mathbf{I}_d\right)\left(\frac{1}{n}\mathbf{X}_{\mathbf{R}^{-1}}^T\mathbf{X}_{\mathbf{R}^{-1}} + \alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}\mathbf{W}\left(\frac{1}{n}\mathbf{X}_{\mathbf{R}^{-1}}^T\mathbf{X}_{\mathbf{R}^{-1}} + \alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}\right\}\right]\right). \tag{31}$$

To simplify $\boldsymbol{\Gamma}_{\mathrm{TL}}$ from (29), for $l \neq j$ using Appendix K and the independence of different pretrained models, we get

$$\mathbb{E}\left[\widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\left(\widehat{\boldsymbol{\theta}}_l - \widetilde{\mathbf{H}}_l\boldsymbol{\beta}\right)^T\widetilde{\mathbf{H}}_l\right] = \frac{b}{d}\widetilde{\mathbf{H}}_j^T\left(\rho_j\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)\left(\rho_l\mathbf{H}_l - \widetilde{\mathbf{H}}_l\right)^T\widetilde{\mathbf{H}}_l. \tag{32}$$

For $l = j$, we can use the asymptotic result by Dar et al. [6] (their Eq. (6.5)) for a single source task, which in our notation is

$$\mathbb{E}\left[\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)^T\right] \tag{33}$$

$$\to \begin{cases} \frac{1}{d}\left(\sigma_{\eta_j}^2 + \frac{\gamma_{\mathrm{src},j}\cdot\sigma_{\xi_j}^2}{1-\gamma_{\mathrm{src},j}}\right)\mathbf{I}_d + \frac{b}{d}\left(\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)\left(\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)^T & \text{for } \gamma_{\mathrm{src},j} < 1, \\ \infty & \text{for } \gamma_{\mathrm{src},j} = 1, \\ \frac{b(\gamma_{\mathrm{src},j}-1)}{d\gamma_{\mathrm{src},j}^2}\left(\gamma_{\mathrm{src},j}\widetilde{\mathbf{H}}_j\widetilde{\mathbf{H}}_j^T - \mathbf{H}_j\mathbf{H}_j^T + \kappa_{\mathbf{H}_j}\mathbf{I}_d - \frac{1}{d}\mathrm{diag}\left(\left\{\left[\mathbf{H}_j\mathbf{H}_j^T\right]_{kk}\right\}_{k=1,\ldots,d}\right)\right) \\ \quad + \frac{b}{d\gamma_{\mathrm{src},j}}\left(\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)\left(\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)^T + \frac{1}{d\gamma_{\mathrm{src},j}}\left(\sigma_{\eta_j}^2 + \frac{\gamma_{\mathrm{src},j}\cdot\sigma_{\xi_j}^2}{\gamma_{\mathrm{src},j}-1}\right)\mathbf{I}_d \\ & \text{for } \gamma_{\mathrm{src},j} > 1. \end{cases}$$

Then, we use the following lemma by Dar et al. [6], which can be proved using results by Rubio and Mestre [19], Dobriban and Sheng [8] (for more details and proof see Lemma E.1 by [6]).

Lemma F.1. *Consider a random data matrix $\mathbf{X} = [\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}]^T$ composed of i.i.d. $\mathbf{x}^{(i)}$ distributed with covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{d\times d}$ that satisfies assumption 4.2, and $\boldsymbol{\theta} \in \mathbb{R}^{d\times d}$ such that $\mathrm{Tr}\left\{\left(\boldsymbol{\theta}^T\boldsymbol{\theta}\right)^{1/2}\right\}$ is uniformly bounded in d, and $\boldsymbol{\Xi} \in \mathbb{R}^{d\times d}$ is a positive semi-definite matrix. Then, with probability one, for each $\alpha > 0$, as $n, d \to \infty$ such that $d/n \to \gamma_{\mathrm{tgt}}$,*

$$\mathrm{Tr}\left\{\boldsymbol{\theta}\left(\left(\tfrac{1}{n}\mathbf{X}^T\mathbf{X} + \alpha\mathbf{I}_d\right)^{-1} - \left(c(\alpha)\boldsymbol{\Sigma} + \alpha\mathbf{I}_d\right)^{-1}\right)\right\} \to 0 \tag{34}$$

and

$$\mathrm{Tr}\Big\{\boldsymbol{\theta}\Big(\left(\tfrac{1}{n}\mathbf{X}^T\mathbf{X} + \alpha\mathbf{I}_d\right)^{-1}\boldsymbol{\Xi}\left(\tfrac{1}{n}\mathbf{X}^T\mathbf{X} + \alpha\mathbf{I}_d\right)^{-1} - \left(c(\alpha)\boldsymbol{\Sigma} + \alpha\mathbf{I}_d\right)^{-1}\left(c'(\alpha)\boldsymbol{\Sigma} + \boldsymbol{\Xi}\right)\left(c(\alpha)\boldsymbol{\Sigma} + \alpha\mathbf{I}_d\right)^{-1}\Big)\Big\} \to 0, \tag{35}$$

where $c(\alpha)$ is the unique solution c of $\frac{1}{c} - 1 = \frac{\gamma_{\mathrm{tgt}}}{d}\mathrm{Tr}\left\{\boldsymbol{\Sigma}(c\boldsymbol{\Sigma} + \alpha\mathbf{I}_d)^{-1}\right\}$, and

$$c'(\alpha) = \frac{\frac{\gamma_{\mathrm{tgt}}}{d}\mathrm{Tr}\left\{\boldsymbol{\Sigma}(c(\alpha)\boldsymbol{\Sigma} + \alpha\mathbf{I}_d)^{-1}\boldsymbol{\Xi}(c(\alpha)\boldsymbol{\Sigma} + \alpha\mathbf{I}_d)^{-1}\right\}}{c(\alpha)^{-2} - \frac{\gamma_{\mathrm{tgt}}}{d}\mathrm{Tr}\left\{\boldsymbol{\Sigma}(c(\alpha)\boldsymbol{\Sigma} + \alpha\mathbf{I}_d)^{-1}\boldsymbol{\Sigma}(c(\alpha)\boldsymbol{\Sigma} + \alpha\mathbf{I}_d)^{-1}\right\}}. \tag{36}$$

Let us continue the proof of Theorem 4.4 from (30)– (31), considering $\gamma_{\mathrm{src},j} \neq 1$ for every j.

To bound $\boldsymbol{\Gamma}_{\mathrm{TL}}$, let

$$\mathbf{a}_j \triangleq \mathbf{R}^{-1}\widetilde{\mathbf{H}}_j^T\left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right).$$

Expanding the product of the two sums in (29) gives

$$\begin{aligned}\boldsymbol{\Gamma}_{\mathrm{TL}} &= \mathbb{E}\left[\left(\sum_{j=1}^m \mathbf{a}_j\right)\left(\sum_{l=1}^m \mathbf{a}_l\right)^T\right] \\ &= \sum_{j=1}^m \mathbb{E}[\mathbf{a}_j\mathbf{a}_j^T] + \sum_{j=1}^m\sum_{\substack{l=1\\l\neq j}}^m \mathbb{E}[\mathbf{a}_j\mathbf{a}_l^T].\end{aligned}$$

Therefore, taking the trace,

$$\mathrm{Tr}\{\boldsymbol{\Gamma}_{\mathrm{TL}}\} = \sum_{j=1}^m \mathbb{E}[\|\mathbf{a}_j\|_2^2] + \sum_{j=1}^m\sum_{\substack{l=1\\l\neq j}}^m \mathbb{E}[\mathbf{a}_l^T\mathbf{a}_j].$$

We bound these two types of terms separately:

- For $j = l$, the definition of $\mathbf{R}$ gives

$$\mathbf{R}^2 - \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j = \sum_{\substack{k=1 \\ k \neq j}}^{m} \widetilde{\mathbf{H}}_k^T \widetilde{\mathbf{H}}_k \succeq \mathbf{0}.$$

Since $\mathbf{R}^{-1}$ is symmetric, congruence by $\mathbf{R}^{-1}$ yields

$$\mathbf{R}^{-1} \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j \mathbf{R}^{-1} \preceq \mathbf{R}^{-1} \mathbf{R}^2 \mathbf{R}^{-1} = \mathbf{I}_d.$$

Consequently,

$$\|\mathbf{R}^{-1} \widetilde{\mathbf{H}}_j^T\|_{\mathrm{op}}^2 = \lambda_{\max}\Big(\mathbf{R}^{-1} \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j \mathbf{R}^{-1}\Big) \leq 1,$$

and hence $\|\mathbf{R}^{-1} \widetilde{\mathbf{H}}_j^T\|_{\mathrm{op}} \leq 1$. Using the conditional mean and covariance from Appendix K and Assumption 3.2, we get, for all sufficiently large d,

$$\begin{aligned}
\mathbb{E}[\|\mathbf{a}_j\|_2^2] &\leq \mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j \boldsymbol{\beta}\right\|_2^2\right] \\
&= \frac{b}{d}\left(\rho_j(1-\rho_j)\,\|\mathbf{H}_j\|_{\mathrm{F}}^2 + \left\|\rho_j \mathbf{H}_j - \widetilde{\mathbf{H}}_j\right\|_{\mathrm{F}}^2\right) + \rho_j \sigma_{\eta_j}^2 + \sigma_{\xi_j}^2 \frac{\min\{d, \widetilde{n}_j\}}{|d - \widetilde{n}_j| - 1} \\
&\leq \frac{2b}{d}\left(\|\mathbf{H}_j\|_{\mathrm{F}}^2 + \left\|\widetilde{\mathbf{H}}_j\right\|_{\mathrm{F}}^2\right) + \sigma_{\eta_j}^2 + \sigma_{\xi_j}^2 \frac{\min\{d, \widetilde{n}_j\}}{|d - \widetilde{n}_j| - 1} \leq C_j,
\end{aligned}$$

where $C_j < \infty$ is independent of d. Here we used

$$\left\|\rho_j \mathbf{H}_j - \widetilde{\mathbf{H}}_j\right\|_{\mathrm{F}}^2 \leq 2\rho_j^2 \|\mathbf{H}_j\|_{\mathrm{F}}^2 + 2\left\|\widetilde{\mathbf{H}}_j\right\|_{\mathrm{F}}^2, \qquad \rho_j(1-\rho_j) + 2\rho_j^2 \leq 2.$$

The existence of C_j follows from Assumption 4.3 and

$$\frac{\min\{d, \widetilde{n}_j\}}{|d - \widetilde{n}_j| - 1} \longrightarrow \frac{\min\{1, \gamma_{\mathrm{src},j}\}}{|1 - \gamma_{\mathrm{src},j}|} < \infty.$$

- For $j \neq l$, the Cauchy–Schwarz inequality gives

$$\begin{aligned}
\left|\mathbb{E}[\mathbf{a}_l^T \mathbf{a}_j]\right| &\leq \mathbb{E}[\|\mathbf{a}_l\|_2 \|\mathbf{a}_j\|_2] \\
&\leq \sqrt{\mathbb{E}[\|\mathbf{a}_l\|_2^2]\,\mathbb{E}[\|\mathbf{a}_j\|_2^2]} \leq \sqrt{C_l C_j} \leq \frac{C_l + C_j}{2}.
\end{aligned}$$

This bound allows the cross terms to be nonzero and does not require independence of $\mathbf{a}_j$ and $\mathbf{a}_l$.

Combining the two cases, we obtain

$$0 \leq \mathrm{Tr}\{\boldsymbol{\Gamma}_{\mathrm{TL}}\} \leq \sum_{j=1}^{m} C_j + \frac{1}{2} \sum_{j=1}^{m} \sum_{\substack{l=1 \\ l \neq j}}^{m} (C_j + C_l) = m \sum_{j=1}^{m} C_j.$$

The last equality holds because each C_j occurs $m - 1$ times in each of the two off-diagonal sums. Since m is fixed, this is a constant independent of d.

Consequently, by Assumptions 3.1– 4.3, there exists a constant $C > 0$, independent of d, such that, for all sufficiently large d,

$$\begin{aligned}
\|\mathbf{W}\|_{\mathrm{op}} &\leq r_H^{-1} \|\boldsymbol{\Sigma}_{\mathbf{x}}\|_{\mathrm{op}} \leq C, \\
\left\|\frac{1}{d}\mathbf{W}\right\|_* &= \frac{1}{d}\,\mathrm{Tr}\{\mathbf{W}\} \leq C, \\
\left\|\frac{n\alpha_{\mathrm{TL}}^2}{d\sigma_\epsilon^2}\boldsymbol{\Gamma}_{\mathrm{TL}} - \frac{\alpha_{\mathrm{TL}}}{d}\mathbf{I}_d\right\|_* &\leq \frac{n\alpha_{\mathrm{TL}}^2}{d\sigma_\epsilon^2}\,\mathrm{Tr}\{\boldsymbol{\Gamma}_{\mathrm{TL}}\} + \alpha_{\mathrm{TL}} \leq C,
\end{aligned}$$

where $\|\cdot\|_*$ denotes the nuclear norm.

We can therefore apply (34) and (35) with the respective test matrices above, $\mathbf{X} = \mathbf{X}_{\mathbf{R}^{-1}}$, $\mathbf{\Sigma} = \mathbf{W}$, $\alpha = \alpha_{\mathrm{TL}}$, and $\mathbf{\Xi} = \mathbf{W}$ in the second application. For the transformed design, the lemma applies after the orthogonal change of coordinates given by the polar decomposition of $\mathbf{R}^{-1}{\mathbf{\Sigma}_{\mathbf{x}}}^{1/2}$; simultaneous orthogonal conjugation preserves the traces and the bounds above.

It remains to justify taking expectations in these almost-sure limits. Let

$$\mathbf{Q} \triangleq \left(\frac{1}{n}\mathbf{X}_{\mathbf{R}^{-1}}^{T}\mathbf{X}_{\mathbf{R}^{-1}} + \alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}.$$

Then $\|\mathbf{Q}\|_{\mathrm{op}} \leq \alpha_{\mathrm{TL}}^{-1}$ and $\|\mathbf{\Omega}^{-1}\|_{\mathrm{op}} \leq \alpha_{\mathrm{TL}}^{-1}$. Also, since $0 \preceq c(\alpha_{\mathrm{TL}})\mathbf{W}\mathbf{\Omega}^{-1} \preceq \mathbf{I}_d$,

$$\frac{\gamma_{\mathrm{tgt}}}{d}\operatorname{Tr}\left\{\left(c(\alpha_{\mathrm{TL}})\mathbf{W}\mathbf{\Omega}^{-1}\right)^2\right\} \leq \frac{\gamma_{\mathrm{tgt}}}{d}\operatorname{Tr}\left\{c(\alpha_{\mathrm{TL}})\mathbf{W}\mathbf{\Omega}^{-1}\right\} = 1 - c(\alpha_{\mathrm{TL}}).$$

Using the definitions of $c(\alpha_{\mathrm{TL}})$ and s, this gives

$$1 \leq s \leq \frac{1}{c(\alpha_{\mathrm{TL}})} \leq 1 + \frac{\gamma_{\mathrm{tgt}}C}{\alpha_{\mathrm{TL}}}.$$

Therefore, the two trace differences are bounded as follows:

$$\left|\operatorname{Tr}\left\{\frac{1}{d}\mathbf{W}(\mathbf{Q} - \mathbf{\Omega}^{-1})\right\}\right| \leq \frac{2C}{\alpha_{\mathrm{TL}}},$$

$$\left|\operatorname{Tr}\left\{\left(\frac{n\alpha_{\mathrm{TL}}^2}{d\sigma_\epsilon^2}\mathbf{\Gamma}_{\mathrm{TL}} - \frac{\alpha_{\mathrm{TL}}}{d}\mathbf{I}_d\right)\left(\mathbf{Q}\mathbf{W}\mathbf{Q} - \mathbf{\Omega}^{-1}s\mathbf{W}\mathbf{\Omega}^{-1}\right)\right\}\right| \leq \frac{C^2}{\alpha_{\mathrm{TL}}^2}\left(2 + \frac{\gamma_{\mathrm{tgt}}C}{\alpha_{\mathrm{TL}}}\right).$$

Both right-hand sides are deterministic constants independent of d. For either trace difference, denoted by Δ_d, Lemma F.1 gives $\Delta_d \to 0$ almost surely. The dominated convergence theorem consequently yields

$$\lim_{d\to\infty}\mathbb{E}[|\Delta_d|] = \mathbb{E}\left[\lim_{d\to\infty}|\Delta_d|\right] = 0.$$

In particular, $\mathbb{E}[\Delta_d] \to 0$. Thus each expected random trace can be replaced by its deterministic approximation with a difference tending to zero. Since $d/n \to \gamma_{\mathrm{tgt}}$, this justifies the expected trace approximations used in (8).

F.3 Additional Discussion on the Insights from Theorem 4.4

The error depends on a weighted combination of matrices that reflect transfer learning using each of the pretrained models alone: m matrices $\left\{\mathbf{\Gamma}_{\mathrm{TL},\infty}^{\mathrm{single},j}\right\}_{j=1}^{m}$, where $\mathbf{\Gamma}_{\mathrm{TL},\infty}^{\mathrm{single},j}$ is formulated in (21) and reflects the j^{th} pretrained model effect *alone* on transfer learning using a *single* pretrained model (this matrix formulation appeared in Theorem 6.2 by Dar et al. [6] for transfer learning using a single pretrained model). Here, our extension to multiple pretrained models combines these single-task matrices in (20).

The error depends on the interactions between pairs of source-target task-relation operators of different source tasks: Eq. (20) includes matrix products with pairs of matrices from the set of assumed, possibly misspecified, linear operators $\left\{\widetilde{\mathbf{H}}_j\right\}_{j=1}^{m}$; here, the possible misspecification of a matrix refers to differences between the assumed matrix and its true unknown form in the task relation model. Moreover, (20) includes products of pairs from the set of (asymptotic) misspecification errors of the linear operators $\left\{\rho_{j,\infty}\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right\}_{j=1}^{m}$, these misspecification errors include the asymptotic overparameterization bias factor $\rho_{j,\infty}$ (11); this overparameterization bias factor will play a significant role in our proposed debiasing approach in subsection 5.4.

F.4 Additional Discussion on the General Case Results

Figures 1, 2, 3, T.1, T.2 and T.3 demonstrate that using multiple pretrained models can be much more beneficial than using a single pretrained model:

- **Using multiple pretrained models can resolve negative transfer that occurs for a single pretrained model.** Such resolved negative transfer cases are observed in Figs. T.3a and T.3d at source parameterization levels where the orange error curve of the single pretrained model is *above* the errors of no-transfer methods (i.e., the least squares and/or ridge regression that appear as green and blue dashed lines, respectively) but the error curves of multiple pretrained models are below the errors of no-transfer methods — implying that, at such source parameterization levels, transfer learning is beneficial only when using more than one pretrained model.
- **The generalization gains due to adding more pretrained models can diminish when there are already many pretrained models.** As demonstrated in Figs. 1 and T.1, using $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$ as the assumed task relation shows that two or three pretrained models can significantly outperform the use of a single pretrained model; however, using 20 pretrained models provides marginal gains compared to using 10 pretrained models. These marginal gains are despite that, for overparameterized pretrained models, there may be a significant room for improvement when the error is much greater than the Bayes optimal error σ_ϵ^2 of the target task. We will address this issue using our overparameterization debiasing approach in subsection 5.4.

G Optimally Tuned Transfer Learning with Multiple Pretrained Models: Nonasymptotic Setting with Noisy-Rotation Task Relation and Isotropic Inputs

G.1 Optimally-Tuned Transfer Learning: The Nonasymptotic Case

Here we will formulate the optimal transfer hyperparameter $\alpha_{\mathrm{TL}}^{\mathrm{opt}}$ using the derivative of the error expression $\bar{\mathcal{E}}_{\mathrm{TL}}$.

Theorem G.1. *Under Assumptions 3.1, 3.2 and 4.1, target data with isotropic input covariance and well-specified orthonormal task relation $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$, $\mathbf{H}_j^T\mathbf{H}_j = \mathbf{I}_d$, the optimal hyperparameter α_{TL} for transfer learning with m pretrained models when $d \notin \{\widetilde{n}_j - 1, \widetilde{n}_j, \widetilde{n}_j + 1\}$ is*

$$\alpha_{\mathrm{TL}}^{\mathrm{opt}} = \frac{m\sigma_\epsilon^2}{n\left(\sum_{j=1}^m C_j + \frac{2b}{d}\sum_{l=1}^{m-1}\sum_{j=l+1}^m (1-\rho_j)(1-\rho_l)\right)} \tag{37}$$

where

$$C_j \triangleq \tag{38}$$

$$\begin{cases} \frac{\sigma_{\eta_j}^2}{d} + \frac{\sigma_{\xi_j}^2}{\widetilde{n}_j - d - 1} & \text{for } d \le \widetilde{n}_j - 2, \\ \left(1 - \frac{\widetilde{n}_j}{d}\right)\frac{b}{d} + \frac{\widetilde{n}_j}{d}\left(\frac{\sigma_{\eta_j}^2}{d} + \frac{\sigma_{\xi_j}^2}{d - \widetilde{n}_j - 1}\right) & \text{for } d \ge \widetilde{n}_j + 2. \end{cases} \tag{39}$$

Then, the optimally tuned transfer learning has the following expected test error:

$$\bar{\mathcal{E}}_{\mathrm{TL}} = \sigma_\epsilon^2\left(1 + \mathbb{E}_{\mathbf{X}}\left[\mathrm{Tr}\left\{\left(\mathbf{X}^T\mathbf{X} + mn\alpha_{\mathrm{TL}}^{\mathrm{opt}}\mathbf{I}_d\right)^{-1}\right\}\right]\right). \tag{40}$$

The proof is provided in Appendix I.

For additional analysis, the following assumption will be useful.

Assumption G.2 (Sources have the same parameterization and noise levels)**.** All pretrained models share the same parameterization level and identical distributions for feature and task relation noise, i.e., there are $\widetilde{n}$, σ_η^2, σ_ξ^2 such that

$$\widetilde{n}_j = \widetilde{n}, \qquad \sigma_{\eta_j}^2 = \sigma_\eta^2, \qquad \sigma_{\xi_j}^2 = \sigma_\xi^2, \quad \forall j \in \{1, \ldots, m\}.$$

Assumption G.2 implies that $\rho_j = \rho$ and $C_j = C$, for all $j \in \{1, \ldots, m\}$, where ρ and C are computed by plugging $\widetilde{n}, \sigma_\eta^2, \sigma_\xi^2$ in (11) and (38), respectively.

G.2 Optimally-Tuned Transfer Learning: The Asymptotic Case

In Theorem 5.2, the limiting value of the optimal hyperparameter $\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}}$ is

$$\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}} = \sigma_\epsilon^2 \gamma_{\mathrm{tgt}} \times \begin{cases} \left(\sigma_\eta^2 + \frac{\gamma_{\mathrm{src}} \cdot \sigma_\xi^2}{1-\gamma_{\mathrm{src}}}\right)^{-1} & \text{for } \gamma_{\mathrm{src}} < 1, \\ \left(\frac{\gamma_{\mathrm{src}}-1}{\gamma_{\mathrm{src}}} b + (m-1) b (\frac{1-\gamma_{\mathrm{src}}}{\gamma_{\mathrm{src}}})^2 + \frac{1}{\gamma_{\mathrm{src}}}\left(\sigma_\eta^2 + \frac{\gamma_{\mathrm{src}} \cdot \sigma_\xi^2}{\gamma_{\mathrm{src}}-1}\right)\right)^{-1} & \text{for } \gamma_{\mathrm{src}} > 1. \end{cases} \tag{41}$$

The Stieltjes transform of the Marchenko-Pastur distribution is formulated as

$$g(-m\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}}; \gamma_{\mathrm{tgt}}) = \frac{-(1-\gamma_{\mathrm{tgt}} + m\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}}) + \sqrt{(1-\gamma_{\mathrm{tgt}} + m\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}})^2 + 4\gamma_{\mathrm{tgt}} m \alpha_{\mathrm{TL},\infty}^{\mathrm{opt}}}}{2\gamma_{\mathrm{tgt}} m \alpha_{\mathrm{TL},\infty}^{\mathrm{opt}}}.$$

G.3 Negative Transfer

For a start, we consider the case of poor pretrained models (due to remote relation to target task or due to inadequate generalization in the source task) such that the transfer learning predictor $\widehat{\boldsymbol{\beta}}_{\mathrm{TL}}$ from (7) negligibly uses them by having $\alpha_{\mathrm{TL}} \to 0^+$.

Proposition G.3. *If the target task is underparameterized with $d \leq n-2$, $\mathbf{X}^T\mathbf{X}$ is almost surely invertible and therefore*

$$\lim_{\alpha_{\mathrm{TL}} \to 0^+} \widehat{\boldsymbol{\beta}}_{\mathrm{TL}} \;=\; \widehat{\boldsymbol{\beta}}_{\mathrm{ML2N}}. \tag{42}$$

If the target task is overparameterized with $d \geq n+2$, $\mathbf{X}^T\mathbf{X}$ is not invertible and

$$\lim_{\alpha_{\mathrm{TL}} \to 0^+} \widehat{\boldsymbol{\beta}}_{\mathrm{TL}} \;=\; \widehat{\boldsymbol{\beta}}_{\mathrm{ML2N}} \;+\; \mathbf{P}_{\mathcal{N}(\mathbf{X})} \left(\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{-1} \left(\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widehat{\boldsymbol{\theta}}_j\right) \tag{43}$$

where $\mathbf{P}_{\mathcal{N}(\mathbf{X})} \triangleq \mathbf{I}_d - \mathbf{X}^+\mathbf{X}$ denotes the orthogonal projector onto the null space of $\mathbf{X}$ and $ML2N$ refers to the minimum ℓ_2-norm solution

$$\widehat{\boldsymbol{\beta}}_{\mathrm{ML2N}} = \operatorname*{arg\,min}_{\mathbf{b} \in \mathbb{R}^d} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 = \mathbf{X}^+\mathbf{y}. \tag{44}$$

Eq. (43) shows that for $\alpha_{\mathrm{TL}} \to 0^+$, the transfer learning predictor is composed of the minimum ℓ_2–norm least squares predictor in the column space of $\mathbf{X}^T$ and depends on the pretrained models in the null–space of $\mathbf{X}^T$. This implies that for an overparameterized target task, poor pretrained models can inevitably degrade transfer learning performance compared to the minimum ℓ_2–norm least squares predictor, which by itself cannot perform better than optimally tuned ridge regression. This explains a potential scenario of negative transfer such as we observe as the target task error peaks when the source parameterization level is close to 1 (see, e.g., Figs. 1, 2, T.1 and T.2).

The statistics of the pretrained learned models (Appendix K) imply that

$$\lim_{d/\widetilde{n}_j \to \infty} \mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j \mid \boldsymbol{\beta}\right] = \mathbf{0}. \tag{45}$$

Under the isotropic Gaussian source model, for $d > \widetilde{n}_j + 1$, we have

$$\begin{aligned} \mathbb{E}\left[\|\widehat{\boldsymbol{\theta}}_j\|_2^2 \mid \boldsymbol{\beta}\right] = \frac{\widetilde{n}_j}{d}\left(\|\mathbf{H}_j\boldsymbol{\beta}\|_2^2 + \sigma_{\eta_j}^2\right) \\ + \frac{\widetilde{n}_j \sigma_{\xi_j}^2}{d - \widetilde{n}_j - 1}. \end{aligned} \tag{46}$$

Thus, for bounded source signal energies and noise variances, $\widehat{\boldsymbol{\theta}}_j$ converges to zero in L^2 as $d/\widetilde{n}_j \to \infty$. For fixed m and uniformly bounded assumed task relation operators, Assumption 3.1 and the closed-form solution in 7 then imply that the transfer learning predictor converges in L^2 to the Tikhonov predictor.

$\widetilde{\mathbf{H}}_j$ independent of the source parametrization for any j, we get that

$$\lim_{\forall j,\ d/\widetilde{n}_j \to \infty} \widehat{\boldsymbol{\beta}}_{\mathrm{TL}} = \widehat{\boldsymbol{\beta}}_{\text{Tikhonov},\mathbf{R}} \tag{47}$$

where $\mathbf{R}$ is the unique positive definite square root of $\left(\sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{1/2}$, as in Theorem 4.4, and $\widehat{\boldsymbol{\beta}}_{\text{Tikhonov},\mathbf{R}}$ is learned using Tikhonov regularization with $\mathbf{R}$ as the Tikhonov matrix and a regularization hyperparameter α_{TL} as for our transfer learning, i.e.,

$$\widehat{\boldsymbol{\beta}}_{\text{Tikhonov},\mathbf{R}} = \underset{\mathbf{b}\in\mathbb{R}^d}{\arg\min} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 + n\alpha_{\mathrm{TL}} \|\mathbf{R}\mathbf{b}\|_2^2 \tag{48}$$

whose closed-form solution for a full rank $\mathbf{R}$ is

$$\widehat{\boldsymbol{\beta}}_{\text{Tikhonov},\mathbf{R}} = \left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\mathbf{R}^T\mathbf{R}\right)^{-1}\mathbf{X}^T\mathbf{y}. \tag{49}$$

Importantly, for $\mathbf{R} = a\mathbf{I}_d$, the Tikhonov predictor is equivalent to ridge regression

$$\widehat{\boldsymbol{\beta}}_{\mathrm{ridge}} = \underset{\mathbf{b}\in\mathbb{R}^d}{\arg\min} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 + n\alpha_{\mathrm{ridge}} \|\mathbf{b}\|_2^2 \tag{50}$$

$$= \left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{ridge}}\mathbf{I}_d\right)^{-1}\mathbf{X}^T\mathbf{y} \tag{51}$$

with $\alpha_{\mathrm{ridge}} = \alpha_{\mathrm{TL}}a^2$. Therefore, when all the assumed task relation operators $\left\{\widetilde{\mathbf{H}}_j\right\}_{j=1}^{m}$ are orthonormal matrices, $\mathbf{R} = \sqrt{m}\mathbf{I}_d$ and the predictor $\widehat{\boldsymbol{\beta}}_{\text{Tikhonov},\mathbf{R}}$ becomes ridge regression with hyperparamter $m\alpha_{\mathrm{TL}}$. Hence, for optimal α_{TL} and α_{ridge} we get that

$$\lim_{\forall j,\ d/\widetilde{n}_j \to \infty} \widehat{\boldsymbol{\beta}}_{\mathrm{TL}} = \widehat{\boldsymbol{\beta}}_{\mathrm{ridge}}. \tag{52}$$

For further discussion on the optimally tuned ridge see Appendix L. As can be seen in Fig. 2 for the well specified case, and in Fig. 1 for the misspecified case, both for $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$, the expected test error of the target model approaches the optimal ridge error, either from below nor from above. When assuming orthonormal task relation matrices, except for the asymptotic equivalence of our transfer learning with $\forall j,\ d/\widetilde{n}_j \to \infty$ to ridge regression, for a fixed $d/\widetilde{n}_j$ our transfer learning can generalize better, on par, or worse than ridge regression. This further justifies the use of ridge regression as our baseline for defining negative transfer in our transfer learning.

Under the assumption of a well-specified task relation $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$, the decline in transfer learning performance can stem from poor quality pretrained models $\widehat{\boldsymbol{\theta}}_j$ and the task relation noise $\boldsymbol{\eta}_j$. Dar et al. [6] found that transfer learning with a *single* pretrained model in the orthonormal well-specified case outperforms ridge regression when $\sigma_\eta^2 + \frac{d\sigma_\xi^2}{|d-\widetilde{n}|-1} < b$. For an overparameterized pretrained model $d > \widetilde{n}$, we get $\frac{d}{|d-\widetilde{n}|-1} > 1$, therefore negative transfer occurs for any overparameterization level of the pretrained model if $\sigma_\eta^2 + \sigma_\xi^2 > b(2m-1)$. When σ_η^2 is the task noise, σ_ξ^2 is the source task noise and b is the variance of the true target parameters.

As for avoiding negative transfer with multiple pretrained models, we provide here the following new theorem (the proof is in Appendix M).

Theorem G.4. *Under the assumptions of Theorem G.1 and Assumption G.2, our transfer learning with m pretrained models is beneficial when $d \notin \{\widetilde{n}-1, \widetilde{n}, \widetilde{n}+1\}$, if*

$$\sigma_\eta^2 + \frac{d\sigma_\xi^2}{|d-\widetilde{n}|-1} < b\left(m + (m-1)(1-\rho)\right) \tag{53}$$

where ρ is defined based on $\widetilde{n}$ and d according to (11).

Importantly, recall from (11) that $1-\rho$ reflects the overparameterization bias that does not exist for underparameterized pretrained models (for which $\rho = 1$). Under the same logic of single pretrained model case, transfer learning using m pretrained models does not perform better than ridge regression performance for any overparameterization level if $\sigma_\eta^2 + \sigma_\xi^2 > b(2m-1)$; this shows that signal

strength b is multiplied by the number of pretrained models. As can be seen in Figs. 2a and T.2c where the transfer noises $\sigma_\xi^2 = \sigma_\eta^2 = 0.5$ and the well-specified case, for $m = 1$ the necessary condition is not met and we can observe negative transfer for all overparameterization levels. On the other hand, we can observe in Figs.2b, T.2a, T.2b and T.2d that, for low transfer noises, positive transfer starts from a pretrained parameterization level somewhat to the right to the interpolation threshold, and then the transfer learning test error approaches the optimal ridge error from below as the pretrained overparameterization level increases.

H Additional Experiments Details

H.1 Task Relations

In addition to the task relation of the $d \times d$ identity matrix $\mathbf{I}_d$, we consider additional task relations based on the following non-orthonormal matrices.

H.1.1 Subspace Projection

For task relation based on a r-dimensional linear subspace in $\mathbb{R}^d$, $r < d$, we define a linear operator $\mathbf{H}_j$ for a single source task as follows:

- Draw a random matrix $\mathbf{A} \in \mathbb{R}^{d \times r}$ with i.i.d. entries from $\mathcal{N}(0, 1)$.
- Perform a QR decomposition, $\mathbf{A} = \mathbf{QR}$, to obtain a $\mathbf{Q} \in \mathbb{R}^{d \times r}$ with orthonormal columns that span a r-dimensional linear subspace in $\mathbb{R}^d$. Note that $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}_r$, but $\mathbf{QQ}^T \neq \mathbf{I}_d$.
- Define the task relation operator as
$$\mathbf{H}_j = \mathbf{QQ}^T, \tag{54}$$
which is the orthogonal projection matrix onto the randomly generated r-dimensional subspace. Note that, although we do not denote the matrix $\mathbf{Q}$ with subscript j, the matrix $\mathbf{Q}$ is formed for the j^{th} source task; i.e., different source tasks have different $\mathbf{Q}$ matrices (thus different subspaces) that are formed from different draws of the $\mathbf{A}$ matrix.

H.1.2 Energy-Preserving Subspace Projection

Due to being an orthogonal projection matrix onto r-dimensional linear subspace, $\mathbf{H}_j$ in (54) has r eigenvalues 1 and $d-r$ eigenvalues 0. Setting such $\mathbf{H}_j$ into our general task relation in (3) exemplifies that the true parameter vector $\boldsymbol{\theta}_j$ of the source task may have lower energy (smaller expected ℓ_2-norm), weaker information about the true parameter vector $\boldsymbol{\beta}$ of the target task. Specifically, for isotropic Gaussian $\boldsymbol{\beta}$ that follows Assumption 3.2, we get

$$\mathbb{E}_{\boldsymbol{\beta}}\left[\|\mathbf{H}_j\boldsymbol{\beta}\|_2^2\right] = \mathbb{E}_{\boldsymbol{\beta}}\left[\left\|\mathbf{QQ}^T\boldsymbol{\beta}\right\|_2^2\right] = \frac{r}{d}b \tag{55}$$

and, therefore,

$$\mathbb{E}\left[|\boldsymbol{\theta}_j|_2^2\right] = \mathbb{E}_{\boldsymbol{\beta},\boldsymbol{\eta}_j}\left[\left\|\mathbf{H}_j\boldsymbol{\beta} + \boldsymbol{\eta}_j\right\|_2^2\right] = \mathbb{E}_{\boldsymbol{\beta}}\left[\|\mathbf{H}_j\boldsymbol{\beta}\|_2^2\right] + \mathbb{E}_{\boldsymbol{\eta}_j}\left[\left\|\boldsymbol{\eta}_j\right\|_2^2\right] = \frac{r}{d}b + \sigma_{\eta_j}^2 \tag{56}$$

where $\boldsymbol{\eta}_j$ and $\boldsymbol{\beta}$ are independent. Note that although $\mathbf{Q}$ is constructed randomly, it is treated as fixed (non-random) in this expectation. More importantly, as the subspace dimension r gets smaller compared to the data dimension d, the information of the true parameters $\boldsymbol{\beta}$ of the target task diminishes compared to the constant task relation noise level $\sigma_{\eta_j}^2$. This motivates us to define an additional projection-based task relation operator that preserves the energy of the target parameters.

For the *energy-preserving subspace-projection task relation*, we form the matrix $\mathbf{Q}$ as explained above, but use it to define the the task relation operator $\mathbf{H}_j$ with a dimension-dependent scaling:

$$\mathbf{H}_j = \sqrt{\frac{d}{r}}\,\mathbf{QQ}^T. \tag{57}$$

This scaling ensures that the projection preserves the expected energy of the true target parameters, i.e.,

$$\mathbb{E}\left[\boldsymbol{\theta}_j\right] = \mathbb{E}_{\boldsymbol{\beta},\boldsymbol{\eta}_j}\left[\left\|\mathbf{H}_j\boldsymbol{\beta} + \boldsymbol{\eta}_j\right\|_2^2\right] = \mathbb{E}_{\boldsymbol{\beta},\boldsymbol{\eta}_j}\left[\left\|\sqrt{\frac{d}{r}}\mathbf{QQ}^T\boldsymbol{\beta} + \boldsymbol{\eta}_j\right\|_2^2\right] = b + \sigma_{\eta_j}^2. \tag{58}$$

Throughout this paper we will use both types of task relation operators (54), (57), in separate experiments.

H.1.3 Circulant Matrix with Condition Number κ

In the circulant case we set the same $d \times d$ matrix $\mathbf{H}$ for all j, $\mathbf{H}_j = \mathbf{H}$, such that $\mathbf{H}$ is circulant with condition number κ and it is formed as follows. For a condition number κ and even dimension d, we set the eigenvalues $\lambda_1, .., \lambda_d$, such that they are symmetric around the index $\frac{d}{2} + 1$ with $\frac{\lambda_1}{\lambda_{\frac{d}{2}+1}} = \kappa$ and $\lambda_1^2 + \lambda_{\frac{d}{2}+1}^2 = 2$. The construction gives us unique $\lambda_{\frac{d}{2}+1}$ and λ_1. Then we set the eigenvalues $\lambda_2, .., \lambda_{\frac{d}{2}}$ to be the square root of equally-spaced numbers in the interval $[\lambda_1^2, \lambda_{\frac{d}{2}+1}^2]$, this way achieving $\sum_{i=1}^{d} |\lambda_i|^2 = d$. Taking this set of eigenvalues $\lambda_1, .., \lambda_d$ using the Discrete Fourier Transform gives us the next circulant matrix, $\mathbf{H} = \mathbf{F}^* diag(\lambda_1, .., \lambda_d)\mathbf{F}$, when the symmetry around the index $\frac{d}{2}+1$ guarantees a real valued matrix, and $\sum_{i=1}^{d} |\lambda_i|^2 = d$ yields that the squared Frobenius norm of $\mathbf{H}$ is d, which is important for Assumption 4.3. This way, for every κ there is a unique construction of this matrix.

H.2 Additional experiment details

H.2.1 Covariance matrix

When referring to an exponential-decay covariance structure for the target data, we construct the covariance matrix as $\Sigma_{ij} = 0.5^{|i-j|}$.

H.2.2 Experiment setup

For the **empirical expected test error** experiments, except for Figures 6 and T.5, we set the dimension to $d = 128$. We evaluated 25 different parametrization levels non-uniformly spaced in the range $\gamma_{\text{src}} \in [0.1, 5.1]$ by setting the training set size to $\tilde{n} = \left\lfloor \frac{128}{\gamma_{\text{src}}} \right\rfloor$. For each parametrization level, the expected error was computed by averaging the test errors of 2000 runs with different random seeds and independently drawn training datasets. In each run, the optimal α_{TL} was empirically-selected using a validation set of size 1000, searching over a grid of 30 logarithmically spaced values in the interval $[10^{-4}, 10^2]$. The test errors were computed using a test set of size 1000, independent of the training and validation sets.

For the **empirical factor validation** as in Figures 6 and T.5, we used 5-fold cross-validation to jointly select the regularization parameter α and the debiasing factor ρ. This was done to ensure a fair comparison with the ridge and transfer baselines. For both ridge regression and the standard transfer estimator, the validation procedure selects only a single hyperparameter, namely the regularization parameter α. In contrast, the empirical debiasing procedure searches over a two-dimensional hyper-parameter space, (α, ρ). Therefore, using the validation set as the other experiments would give the empirical debiasing method an additional model-selection advantage.

For the **bias-variance decomposition**, we employed the same training size selection strategy across 50 different parametrization levels in the range $\gamma_{\text{src}} \in [0.1, 5.1]$, using the same optimal α_{TL} tuning procedure. For each value of γ_{src}, the expectation approximation was done by averaging over 150 main-runs with independently drawn true parameter vector $\boldsymbol{\beta}$. In each main-run, for its $\boldsymbol{\beta}$, we performed 50 independent sub-runs where the training datasets are randomly drawn and the learning of $\widehat{\boldsymbol{\beta}}$ is done independently. By averaging over the sub-runs we get the mean estimator $\mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right]$ for a specific $\boldsymbol{\beta}$. The expectation over $\boldsymbol{\beta}$ of the squared bias term was calculated by averaging the value of $\left(\mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right] - \boldsymbol{\beta}\right)^T \boldsymbol{\Sigma}_{\mathbf{x}} \left(\mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right] - \boldsymbol{\beta}\right)$ over the 150 main-runs. Our bias-variance decomposition evaluations are for $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{I}_d$.

For the variance term, we utilized the computed $\mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right]$ to calculate the covariance matrix by averaging $\widehat{\boldsymbol{\Sigma}}_{\widehat{\boldsymbol{\beta}}} = \left(\widehat{\boldsymbol{\beta}} - \mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right]\right)\left(\widehat{\boldsymbol{\beta}} - \mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right]\right)^T$ over the 50 sub-runs. Finally, the scalar variance error term was obtained by taking the average of the trace, $\text{Tr}\left\{\widehat{\boldsymbol{\Sigma}}_{\widehat{\boldsymbol{\beta}}}\right\}$, over the 50 sub-runs of independently

drawn datasets and over the 150 main-runs of independently-drawn $\boldsymbol{\beta}$. This empirically computes the variance error term for our experiments with $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{I}_d$.

I Proofs for the Simpler Case of Noisy Rotation Task Relation

To prove Theorem G.1, we will first state a lemma and prove it in Appendix I.1. In Appendix I.2 we will use the lemma to prove Theorem G.1.

I.1 Nonasymptotic Test Error Formula of Transfer Learning for Noisy Orthonormal Task Relation: Eigendecomposition Form of the Empirical Covariance

We start by proving the following lemma.

Lemma I.1. *Under Assumptions 3.1, 3.2, 4.1, target data with isotropic input covariance and well-specified orthonormal task relation $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$, $\mathbf{H}_j^T\mathbf{H}_j = \mathbf{I}_d$, the expected error of the transfer learning with m pretrained models and a (not necessarily optimal) hyperparameter $\alpha_{\mathrm{TL}} > 0$ is*

$$\bar{\mathcal{E}}_{\mathrm{TL}} = \sigma_\epsilon^2 + \mathbb{E}\left\{\sum_{k=1}^{d} \frac{\sigma_\epsilon^2\lambda_k + n^2\alpha_{\mathrm{TL}}^2\sum_{j=1}^m C_j + \frac{2bn^2\alpha_{\mathrm{TL}}^2}{d}\sum_{l=1}^{m-1}\sum_{j=l+1}^{m}(1-\rho_j)(1-\rho_l)}{(\lambda_k + nm\alpha_{\mathrm{TL}})^2}\right\} \tag{59}$$

where λ_k is the k^{th} eigenvalue of the $d \times d$ empirical covariance matrix $\mathbf{X}^T\mathbf{X}$ and

$$C_j \triangleq \begin{cases} \frac{\sigma_{\eta_j}^2}{d} + \frac{\sigma_{\xi_j}^2}{\widetilde{n}_j - d - 1}, & \text{for } d \le \widetilde{n}_j - 2, \\ \infty, & \text{for } \widetilde{n}_j - 1 \le d \le \widetilde{n}_j + 1, \\ \left(1 - \frac{\widetilde{n}_j}{d}\right)\frac{b}{d} + \frac{\widetilde{n}_j}{d}\left(\frac{\sigma_{\eta_j}^2}{d} + \frac{\sigma_{\xi_j}^2}{d - \widetilde{n}_j - 1}\right), & \text{for } d \ge \widetilde{n}_j + 2. \end{cases} \tag{60}$$

The lemma proof is as follows. From (25) and that here we have $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{I}_d$ and $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$ for any j, we get

$$\bar{\mathcal{E}}_{\mathrm{TL}} = \sigma_\epsilon^2 + \mathbb{E}\left[\left\|\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \mathbf{H}_j^T\mathbf{H}_j\right)^{-1}\left(\mathbf{X}^T\boldsymbol{\epsilon} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \mathbf{H}_j^T\left(\boldsymbol{\eta}_j + \left(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j\right)\right)\right)\right\|_2^2\right]. \tag{61}$$

Here $\mathbf{H}_j$ is an orthonormal matrix, i.e., $\mathbf{H}_j^T\mathbf{H}_j = \mathbf{I}_d$, for any j. Then, using the cyclic property of trace, the error expression can be written as

$$\begin{aligned}
&\bar{\mathcal{E}}_{\mathrm{TL}} = \\
&\sigma_\epsilon^2 + \mathrm{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}m\mathbf{I}_d\right)^{-2}\left(\mathbf{X}^T\boldsymbol{\epsilon} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \mathbf{H}_j^T\left(\boldsymbol{\eta}_j + \left(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j\right)\right)\right)\right.\right. \\
&\qquad\left.\left.\left(\mathbf{X}^T\boldsymbol{\epsilon} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \mathbf{H}_j^T\left(\boldsymbol{\eta}_j + \left(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j\right)\right)\right)^T\right]\right\} = \\
&\sigma_\epsilon^2 + \sigma_\epsilon^2\,\mathrm{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + nm\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-2}\mathbf{X}^T\mathbf{X}\right]\right\} + \\
&\mathrm{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}m\mathbf{I}_d\right)^{-2}n^2\alpha_{\mathrm{TL}}^2\sum_{j=1}^m\sum_{l=1}^m \mathbf{H}_j^T(\widehat{\boldsymbol{\theta}}_l - \boldsymbol{\theta}_l + \boldsymbol{\eta}_l)(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j)^T\mathbf{H}_j\right]\right\} = \\
&\sigma_\epsilon^2 + \sigma_\epsilon^2\,\mathrm{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + nm\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-2}\mathbf{X}^T\mathbf{X}\right]\right\} + \\
&n^2\alpha_{\mathrm{TL}}^2\,\mathrm{Tr}\left\{\sum_{j=1}^m\sum_{l=1}^m \mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}m\mathbf{I}_d\right)^{-1}\mathbf{H}_j^T\mathbf{H}_j\mathbf{H}_l^T\mathbf{H}_l\left(\mathbf{X}^T\mathbf{X} + nm\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1}\right.\right. \\
&\qquad\left.\left.\mathbf{H}_l^T(\widehat{\boldsymbol{\theta}}_l - \boldsymbol{\theta}_l + \boldsymbol{\eta}_l)(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j)^T\mathbf{H}_j\right]\right\}
\end{aligned}$$

where the last equality uses the orthonormality property $\mathbf{H}_j^T\mathbf{H}_j = \mathbf{I}_d$, $\mathbf{H}_l^T\mathbf{H}_l = \mathbf{I}_d$.

Now, we define $\mathbf{X}_{\mathbf{H}_j} \triangleq \mathbf{X}\mathbf{H}_j^T$ and $\mathbf{X}_{\mathbf{H}_l} \triangleq \mathbf{X}\mathbf{H}_l^T$, by which we get

$$\begin{aligned}\bar{\mathcal{E}}_{\mathrm{TL}} &= \sigma_\epsilon^2 + \sigma_\epsilon^2 \,\mathrm{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + nm\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-2}\mathbf{X}^T\mathbf{X}\right]\right\} \\ &+ n^2\alpha_{\mathrm{TL}}^2 \,\mathrm{Tr}\Bigg\{\sum_{j=1}^{m}\sum_{l=1}^{m}\mathbb{E}\Big[\mathbf{H}_j\mathbf{H}_l^T(\mathbf{X}_{\mathbf{H}_l}^T\mathbf{X}_{\mathbf{H}_l} + nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}(\widehat{\boldsymbol{\theta}}_l - \boldsymbol{\theta}_l + \boldsymbol{\eta}_l)(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j)^T \\ &\qquad (\mathbf{X}_{\mathbf{H}_j}^T\mathbf{X}_{\mathbf{H}_j} + nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}\Big]\Bigg\} \end{aligned} \tag{62}$$

For a single pretrained model (here indexed by j), Dar et al. [6] have already provided this calculation

$$\mathbb{E}\left[(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j)(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j)^T\right] = C_j\mathbf{I}_d \tag{63}$$

in their Eq. (C.3)-(C.7) of their single pretrained model analysis.

Here, we will develop the case where $j \neq l$, which is new due to our multiple pretrained model setting. First we will simplify the expression and note some important statistics:

$$\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j = \mathbf{Z}_j^+\mathbf{Z}_j\boldsymbol{\theta}_j + \mathbf{Z}_j^+\boldsymbol{\xi}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j = \mathbf{Z}_j^+\mathbf{Z}_j(\mathbf{H}_j\boldsymbol{\beta} + \boldsymbol{\eta}_j) + \mathbf{Z}_j^+\boldsymbol{\xi}_j - \mathbf{H}_j\boldsymbol{\beta} \tag{64}$$

and we will get

$$\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j = (\mathbf{Z}_j^+\mathbf{Z}_j - \mathbf{I}_d)\mathbf{H}_j\boldsymbol{\beta} + \mathbf{Z}_j^+(\mathbf{Z}_j\boldsymbol{\eta}_j + \boldsymbol{\xi}_j) \tag{65}$$

By multiplying terms of different j and l we get

$$\begin{aligned}&(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j)(\widehat{\boldsymbol{\theta}}_l - \boldsymbol{\theta}_l + \boldsymbol{\eta}_l)^T = \\ &= ((\mathbf{Z}_j^+\mathbf{Z}_j - \mathbf{I}_d)\mathbf{H}_j\boldsymbol{\beta} + \mathbf{Z}_j^+(\mathbf{Z}_j\boldsymbol{\eta}_j + \xi_j))((\mathbf{Z}_l^+\mathbf{Z}_l - \mathbf{I}_d)\mathbf{H}_l\boldsymbol{\beta} + \mathbf{Z}_l^+(\mathbf{Z}_l\boldsymbol{\eta}_l + \xi_l))^T \\ &= (\mathbf{Z}_j^+\mathbf{Z}_j - \mathbf{I}_d)\mathbf{H}_j\boldsymbol{\beta}\boldsymbol{\beta}^T\mathbf{H}_l^T(\mathbf{Z}_l^+\mathbf{Z}_l - \mathbf{I}_d)^T + \mathbf{Z}_j^+(\mathbf{Z}_j\boldsymbol{\eta}_j + \xi_j)\boldsymbol{\beta}^T\mathbf{H}_l^T(\mathbf{Z}_l^+\mathbf{Z}_l - \mathbf{I}_d)^T \\ &\quad + (\mathbf{Z}_j^+\mathbf{Z}_j - \mathbf{I}_d)\mathbf{H}_j\boldsymbol{\beta}(\mathbf{Z}_l\boldsymbol{\eta}_l + \xi_l)^T(\mathbf{Z}_l^+)^T + \mathbf{Z}_j^+(\mathbf{Z}_j\boldsymbol{\eta}_j + \xi_j)(\mathbf{Z}_l\boldsymbol{\eta}_l + \xi_l)^T(\mathbf{Z}_l^+)^T. \end{aligned} \tag{66}$$

$\mathbf{Z}_j$ has i.i.d. standard Gaussian components, therefore, using the expectation of the $d \times d$ projection matrix $\mathbf{Z}_j^+\mathbf{Z}_j$ we get almost surely that

$$\mathbb{E}\left[\mathbf{Z}_j^+\mathbf{Z}_j - \mathbf{I}_d\right] = (\rho_j - 1)\mathbf{I}_d \tag{67}$$

where ρ_j is defined in (11). Moreover, under the assumption that $\boldsymbol{\beta}$ is random and has isotropic Gaussian distribution with zero mean and covariance matrix $\mathbf{B}_d = \frac{b}{d}\mathbf{I}_d$,

$$\mathbb{E}\left[\boldsymbol{\beta}\boldsymbol{\beta}^T\right] = \frac{b}{d}\mathbf{I}_d \tag{68}$$

By using (67), (68),

$$\mathbb{E}\left[(\mathbf{Z}_j^+\mathbf{Z}_j - \mathbf{I}_d)\mathbf{H}_j\boldsymbol{\beta}\boldsymbol{\beta}^T\mathbf{H}_l^T(\mathbf{Z}_l^+\mathbf{Z}_l - \mathbf{I}_d)^T\right] = (1-\rho_j)(1-\rho_l)\frac{b}{d}\mathbf{H}_j\mathbf{H}_l^T \tag{69}$$

where we used the independence of $\mathbf{Z}_j$ and $\mathbf{Z}_l$ for $j \neq l$.

Then, due to the independence of $\boldsymbol{\beta}$, $\mathbf{Z}_j$, $\mathbf{Z}_l$, $\boldsymbol{\eta}_j$, $\boldsymbol{\eta}_l$, $\boldsymbol{\xi}_j$, $\boldsymbol{\xi}_l$ for any $j \neq l$, and the zero mean of $\boldsymbol{\eta}_j$, $\boldsymbol{\eta}_l$, $\boldsymbol{\xi}_j$, $\boldsymbol{\xi}_l$:

$$\mathbb{E}\left[\mathbf{Z}_j^+(\mathbf{Z}_j\boldsymbol{\eta}_j + \boldsymbol{\xi}_j)\boldsymbol{\beta}^T\mathbf{H}_l^T(\mathbf{Z}_l^+\mathbf{Z}_l - \mathbf{I}_d)^T\right] = \mathbf{0} \tag{70}$$

$$\mathbb{E}\left[(\mathbf{Z}_j^+\mathbf{Z}_j - \mathbf{I}_d)\mathbf{H}_j\boldsymbol{\beta}(\mathbf{Z}_l\boldsymbol{\eta}_l + \boldsymbol{\xi}_l)^T(\mathbf{Z}_l^+)^T\right] = \mathbf{0} \tag{71}$$

$$\mathbb{E}\left[\mathbf{Z}_j^+(\mathbf{Z}_j\boldsymbol{\eta}_j + \xi_j)(\mathbf{Z}_l\boldsymbol{\eta}_l + \xi_l)^T(\mathbf{Z}_l^+)^T\right] = \mathbf{0} \tag{72}$$

Setting (69)-(72) in (66) gives

$$\mathbb{E}\left[(\widehat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j + \boldsymbol{\eta}_j)(\widehat{\boldsymbol{\theta}}_l - \boldsymbol{\theta}_l + \boldsymbol{\eta}_l)^T\right] = (1-\rho_j)(1-\rho_l)\frac{b}{d}\mathbf{H}_j\mathbf{H}_l^T \tag{73}$$

We use (73) to further develop an expression in the double sum of (62) for $j \neq l$:

$$\begin{aligned}
&\operatorname{Tr}\left\{\mathbb{E}\left[\mathbf{H}_l\mathbf{H}_j^T(\mathbf{X}_{\mathbf{H}_j}^T\mathbf{X}_{\mathbf{H}_j}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}(\widehat{\boldsymbol{\theta}}_j-\boldsymbol{\theta}_j+\boldsymbol{\eta}_j)(\widehat{\boldsymbol{\theta}}_l-\boldsymbol{\theta}_l+\boldsymbol{\eta}_l)^T(\mathbf{X}_{\mathbf{H}_l}^T\mathbf{X}_{\mathbf{H}_l}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}\right]\right\}= \\
&=\operatorname{Tr}\left\{(1-\rho_j)(1-\rho_l)\frac{b}{d}\mathbb{E}\left[\mathbf{H}_l\mathbf{H}_j^T(\mathbf{X}_{\mathbf{H}_j}^T\mathbf{X}_{\mathbf{H}_j}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}\mathbf{H}_j\mathbf{H}_l^T(\mathbf{X}_{\mathbf{H}_l}^T\mathbf{X}_{\mathbf{H}_l}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}\right]\right\} \\
&=\frac{b}{d}(1-\rho_j)(1-\rho_l)\operatorname{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-2}\right]\right\} \qquad (74)
\end{aligned}$$

where we used the orthonormality of $\mathbf{H}_j$ for any j.

So we get

$$\begin{aligned}
&\operatorname{Tr}\Bigg\{\sum_{\substack{j,l\in\{1,\ldots,m\}\\ l\neq j}}\mathbb{E}\left[\mathbf{H}_l\mathbf{H}_j^T(\mathbf{X}_{\mathbf{H}_j}^T\mathbf{X}_{\mathbf{H}_j}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}(\widehat{\boldsymbol{\theta}}_j-\boldsymbol{\theta}_j+\boldsymbol{\eta}_j)\quad\times\right. \\
&\qquad\left.(\widehat{\boldsymbol{\theta}}_l-\boldsymbol{\theta}_l+\boldsymbol{\eta}_l)^T(\mathbf{X}_{\mathbf{H}_l}^T\mathbf{X}_{\mathbf{H}_l}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}\right]\Bigg\}= \\
&=\mathbb{E}\left\{\sum_{k=1}^{d}\frac{\frac{2b}{d}\sum_{l=1}^{m-1}\sum_{j=l+1}^{m}(1-\rho_j)(1-\rho_l)}{(\lambda_k+nm\alpha_{\mathrm{TL}})^2}\right\} \qquad (75)
\end{aligned}$$

For the m times that $j=l$ in the double sum of (62), we get

$$\begin{aligned}
&\operatorname{Tr}\Bigg\{\sum_{j=1}^{m}\mathbb{E}\left[\mathbf{H}_j\mathbf{H}_j^T(\mathbf{X}_{\mathbf{H}_j}^T\mathbf{X}_{\mathbf{H}_j}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}(\widehat{\boldsymbol{\theta}}_j-\boldsymbol{\theta}_j+\boldsymbol{\eta}_j)\quad\times\right. \\
&\qquad\left.(\widehat{\boldsymbol{\theta}}_j-\boldsymbol{\theta}_j+\boldsymbol{\eta}_j)^T(\mathbf{X}_{\mathbf{H}_j}^T\mathbf{X}_{\mathbf{H}_j}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d)^{-1}\right]\Bigg\} \\
&=\mathbb{E}\left\{\sum_{k=1}^{d}\frac{\sum_{j=1}^{m}C_j}{(\lambda_k+nm\alpha_{\mathrm{TL}})^2}\right\} \qquad (76)
\end{aligned}$$

Moreover, we have

$$\operatorname{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X}+nm\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-2}\mathbf{X}^T\mathbf{X}\right]\right\}=\mathbb{E}\left\{\sum_{k=1}^{d}\frac{\lambda_k}{(\lambda_k+nm\alpha_{\mathrm{TL}})^2}\right\} \qquad (77)$$

Setting (75), (76), (77) in (62) gives (59) and proves Lemma I.1.

I.2 Nonasymptotic Test Error Formula of Optimally Tuned Transfer Learning for Noisy Orthonormal Task Relation: Matrix Form of the Empirical Covariance

Now we will formulate the optimal hyperparameter $\alpha_{\mathrm{TL}}^{\mathrm{opt}}$ using the derivative of the error expression $\bar{\mathcal{E}}_{\mathrm{TL}}$. For simplicity, we will denote

$$A=\sum_{j=1}^{m}C_j+\frac{2b}{d}\sum_{l=1}^{m-1}\sum_{j=l+1}^{m}(1-\rho_j)(1-\rho_l) \qquad (78)$$

Now the error expression can be written as

$$\bar{\mathcal{E}}_{\mathrm{TL}}=\sigma_\epsilon^2+\mathbb{E}\left[\sum_{k=1}^{d}\frac{\sigma_\epsilon^2\lambda_k+n^2\alpha_{\mathrm{TL}}^2A}{(\lambda_k+nm\alpha_{\mathrm{TL}})^2}\right] \qquad (79)$$

Taking derivative with respect to α_{TL} gives

$$\begin{aligned}
\frac{\partial\bar{\mathcal{E}}_{\mathrm{TL}}}{\partial\alpha_{\mathrm{TL}}}&=\mathbb{E}\left[\sum_{k=1}^{d}\frac{2n^2\alpha_{\mathrm{TL}}A(\lambda_k+nm\alpha_{\mathrm{TL}})-2nm(\sigma_\epsilon^2\lambda_k+n^2\alpha_{\mathrm{TL}}^2A)}{(\lambda_k+nm\alpha_{\mathrm{TL}})^3}\right] \\
&=\mathbb{E}\left[\sum_{k=1}^{d}\frac{2n^2\alpha_{\mathrm{TL}}A\lambda_k-2nm\sigma_\epsilon^2\lambda_k}{(\lambda_k+nm\alpha_{\mathrm{TL}})^3}\right]=2n(n\alpha_{\mathrm{TL}}A-m\sigma_\epsilon^2)\cdot\mathbb{E}\left[\sum_{k=1}^{d}\frac{\lambda_k}{(\lambda_k+nm\alpha_{\mathrm{TL}})^3}\right]
\end{aligned}$$

For optimality, we will solve $\frac{\partial \bar{\mathcal{E}}_{\mathrm{TL}}}{\partial \alpha_{\mathrm{TL}}} = 0$ and yield

$$\alpha_{\mathrm{TL}}^{\mathrm{opt}} = \frac{m\sigma_\epsilon^2}{nA} = \frac{m\sigma_\epsilon^2}{n\left(\sum_{j=1}^{m} C_j + \frac{2b}{d}\sum_{l=1}^{m-1}\sum_{j=l+1}^{m}(1-\rho_j)(1-\rho_l)\right)} \tag{80}$$

Next, we set the expression for $\alpha_{TL}^{\mathrm{opt}}$ in the error expression from (59) and get

$$\bar{\mathcal{E}}_{\mathrm{TL}} = \sigma_\epsilon^2\left(1 + \mathbb{E}\left[\sum_{k=1}^{d}\frac{1}{\lambda_k + mn\alpha_{\mathrm{TL}}^{\mathrm{opt}}}\right]\right) = \sigma_\epsilon^2\left(1 + \mathbb{E}_{\mathbf{X}}\left[\mathrm{Tr}\left(\left(\mathbf{X}^T\mathbf{X} + mn\alpha_{\mathrm{TL}}^{\mathrm{opt}}\mathbf{I}_d\right)^{-1}\right)\right]\right) \tag{81}$$

which proves Theorem G.1.

J Proof Outline of Theorem 5.2

First, the optimal transfer learning hyperparameter in (12) that uses (38) should be formulated, under Assumptions 3.1, 3.2, 4.1, 4.3 and G.2, in its asymptotic form

$$\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}} = \sigma_\epsilon^2\gamma_{\mathrm{tgt}} \times \begin{cases} \left(\sigma_\eta^2 + \frac{\gamma_{\mathrm{src}}\cdot\sigma_\xi^2}{1-\gamma_{\mathrm{src}}}\right)^{-1} & \text{for } \gamma_{\mathrm{src}} < 1, \\ \left(\frac{\gamma_{\mathrm{src}}-1}{\gamma_{\mathrm{src}}}b + (m-1)b(\frac{1-\gamma_{\mathrm{src}}}{\gamma_{\mathrm{src}}})^2 + \frac{1}{\gamma_{\mathrm{src}}}\left(\sigma_\eta^2 + \frac{\gamma_{\mathrm{src}}\cdot\sigma_\xi^2}{\gamma_{\mathrm{src}}-1}\right)\right)^{-1} & \text{for } \gamma_{\mathrm{src}} > 1. \end{cases} \tag{82}$$

Then, note that the nonasymptotic error form for our optimally tuned transfer learning in (40) is the same as for optimally tuned ridge regression in (88) except for the scaling of the identity matrix.

Therefore, we can use the asymptotic error expression given by Dobriban and Wager [9] for ridge regression with a $n \times d$ random matrix $\mathbf{X}$ with i.i.d rows, means $\mathbf{0}$ and covariance $\mathbf{\Sigma_x} = \mathbf{I}_d$. Accordingly, we can use our different scaling parameter $m\alpha_{\mathrm{TL},\infty}^{\mathrm{opt}}$ and plug it in the result by [9] to get the formulations in (41)-(G.2) of Theorem 5.2. The same proof process was used by Dar et al. [6] in the case of transfer learning with a single pretrained model and its optimal hyperparameter.

K The Second-Order Statistics of Pretrained Source Models

Recall that all the sources tasks regressors $\boldsymbol{\theta}_j$ is unknown and estimated by $\widehat{\boldsymbol{\theta}}_j$, which is the ML2N solution to the suitable source task. The second-order statistics of $\widehat{\boldsymbol{\theta}}_j$ given $\boldsymbol{\beta}$, is as formulated by Dar et al. [6] for a single pretrained model:

- The expected value of $\widehat{\boldsymbol{\theta}}_j$ given $\boldsymbol{\beta}$ is

$$\begin{aligned}\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j \mid \boldsymbol{\beta}\right] &= \begin{cases}\mathbf{H}_j\boldsymbol{\beta} & \text{for } d \le \widetilde{n}_j, \\ \frac{\widetilde{n}_j}{d}\mathbf{H}_j\boldsymbol{\beta} & \text{for } d > \widetilde{n}_j\end{cases} \\ &= \rho_j\mathbf{H}_j\boldsymbol{\beta}\end{aligned} \tag{83}$$

  where ρ_j is defined in (11).

- The covariance matrix $C_{\widehat{\boldsymbol{\theta}}_j|\boldsymbol{\beta}} \triangleq \mathbb{E}\left[\left(\widehat{\boldsymbol{\theta}}_j - \mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j \mid \boldsymbol{\beta}\right]\right)\left(\widehat{\boldsymbol{\theta}}_j - \mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j \mid \boldsymbol{\beta}\right]\right)^T \mid \boldsymbol{\beta}\right]$ is

$$C_{\widehat{\boldsymbol{\theta}}_j|\boldsymbol{\beta}} = \left(\frac{\sigma_{\eta_j}^2}{d} + \frac{\sigma_{\xi_j}^2}{\widetilde{n}_j - d - 1}\right)\mathbf{I}_d \tag{84}$$

  for $d \le \widetilde{n}_j - 2$; and it is possibly anisotropic with

$$\mathrm{Tr}\left\{C_{\widehat{\boldsymbol{\theta}}_j|\boldsymbol{\beta}}\right\} = \frac{\widetilde{n}_j}{d}\left(1 - \frac{\widetilde{n}_j}{d}\right)\|\mathbf{H}_j\boldsymbol{\beta}\|_2^2 + \frac{\widetilde{n}_j}{d}\sigma_{\eta_j}^2 + \frac{\widetilde{n}_j\sigma_{\xi_j}^2}{d - \widetilde{n}_j - 1} \tag{85}$$

  for $d \ge \widetilde{n}_j + 2$. For $d \in \{\widetilde{n}_j - 1, \widetilde{n}_j, \widetilde{n}_j + 1\}$ the covariance matrix is infinite valued.

L Ridge Regression Formulations

Our analysis of transfer learning with multiple pretrained models sometimes uses the mathematical resemblance of its test error to the test error of ridge regression. In this appendix we provide auxiliary details on ridge regression.

The ridge regression for our target task (without the source tasks nor pretrained models) is formulated as

$$\widehat{\boldsymbol{\beta}}_{\mathrm{ridge}} = \underset{\mathbf{b}\in\mathbb{R}^d}{\arg\min} \left\|\mathbf{y}-\mathbf{X}\mathbf{b}\right\|_2^2 + n\alpha_{\mathrm{ridge}} \left\|\mathbf{b}\right\|_2^2 \tag{86}$$

$$= \left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{ridge}}\mathbf{I}_d\right)^{-1}\mathbf{X}^T\mathbf{y} \tag{87}$$

where $\alpha_{\mathrm{ridge}} > 0$ is a hyperparameter that determines the ridge regularization strength. The optimal hyperparameter value $\alpha_{\mathrm{ridge}}^{\mathrm{opt}} = \frac{d\sigma_\epsilon^2}{nb}$ achieves the minimum expected test error of the target task, for isotropic Gaussian $\boldsymbol{\beta}$ that satisfies Assumption 3.2,

$$\bar{\mathcal{E}}_{\mathrm{ridge}}^{\mathrm{opt}} = \sigma_\epsilon^2\left(1 + \mathbb{E}_{\mathbf{X}}\left[\mathrm{Tr}\left\{\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{ridge}}^{\mathrm{opt}}\mathbf{I}_d\right)^{-1}\right\}\right]\right). \tag{88}$$

Similar results for optimally tuned ridge regression were given by, e.g., Nakkiran et al. [17], Dobriban and Wager [9]. Specifically, Dar et al. [6] used the ridge regression error form to study transfer learning with a single pretrained model, hence, the proof outline of (88) is available in their Appendix D.1.

M Proof of Theorem 5.3

Consider Assumptions 3.1 and 3.2, target data with isotropic input covariance and well-specified orthonormal task relation $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$, $\mathbf{H}_j^T\mathbf{H}_j = \mathbf{I}_d$. Then, the test error of transfer learning with multiple pretrained models (40) and ridge regression (88) have the same form, except for the scaling of the identity matrix. From this we get that transfer learning is beneficial, i.e., $\bar{\mathcal{E}}_{\mathrm{TL}} < \bar{\mathcal{E}}_{\mathrm{ridge}}$, if

$$\mathbb{E}_{\mathbf{X}}\left[\mathrm{Tr}\left\{\left(\mathbf{X}^T\mathbf{X} + mn\alpha_{TL}^{\mathrm{opt}}\mathbf{I}_d\right)^{-1}\right\}\right] < \mathbb{E}_{\mathbf{X}}\left[\mathrm{Tr}\left\{\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{ridge}}^{\mathrm{opt}}\mathbf{I}_d\right)^{-1}\right\}\right]. \tag{89}$$

Note that the random matrix $\mathbf{X}^T\mathbf{X}$ has the same distribution on the two sides of this inequality. Moreover, the random eigenvalues of $\mathbf{X}^T\mathbf{X}$ are non-negative. By eigendecomposition, we can write (89) as

$$\sum_{k=1}^{d}\mathbb{E}\left[\frac{1}{\lambda_k\left\{\mathbf{X}^T\mathbf{X}\right\} + mn\alpha_{TL}^{\mathrm{opt}}}\right] < \sum_{k=1}^{d}\mathbb{E}\left[\frac{1}{\lambda_k\left\{\mathbf{X}^T\mathbf{X}\right\} + n\alpha_{\mathrm{ridge}}^{\mathrm{opt}}}\right] \tag{90}$$

where the expectations are over the random eigenvalues $\lambda_1\left\{\mathbf{X}^T\mathbf{X}\right\}, \ldots, \lambda_d\left\{\mathbf{X}^T\mathbf{X}\right\}$.

The inequality (90) together with the non-negativity of its denominators imply a condition on the optimal hyperparameters of the two methods:

$$m\alpha_{\mathrm{TL}}^{\mathrm{opt}} > \alpha_{\mathrm{ridge}}^{\mathrm{opt}}, \tag{91}$$

namely, the larger scaling of the identity matrix yields a lower test error.

We continue to develop (91) as follows:

$$\frac{m\sigma_\epsilon^2}{nC + \frac{bn}{d}(m-1)(1-\rho)^2} > \frac{d\sigma_\epsilon^2}{nb} \tag{92}$$

$$C + \frac{b}{d}(m-1)(1-\rho)^2 < \frac{b}{d}m \tag{93}$$

For overparameterized pretrained models where $d \geq \widetilde{n} + 2$, we set the corresponding formula for C from (38) under Assumption G.2:

$$(1-\rho)\,\frac{b}{d} + \rho\left(\frac{\sigma_\eta^2}{d} + \frac{\sigma_\xi^2}{d-\widetilde{n}-1}\right) + \frac{b}{d}(m-1)(1-\rho)^2 < \frac{b}{d}m$$

$$\sigma_\eta^2 + \frac{d\sigma_\xi^2}{d-\widetilde{n}-1} < b\left(1 - \frac{1}{\rho} + \frac{1}{\rho}m - \frac{1}{\rho}(m-1)\,(1-\rho)^2\right)$$

$$\sigma_\eta^2 + \frac{d\sigma_\xi^2}{d-\widetilde{n}-1} < b\left(1 - \frac{1}{\rho}(m-1)\left(1 - 2\rho + \rho^2 - 1\right)\right)$$

and this gives the condition

$$\sigma_\eta^2 + \frac{d\sigma_\xi^2}{d-\widetilde{n}-1} < b\,(1 + (m-1)\,(2-\rho)) \tag{94}$$

For the underparameterized pretrained models where $d \leq \widetilde{n} - 2$, note that $\rho = 1$, so the condition in (93) is simplified to

$$C < \frac{b}{d}m. \tag{95}$$

By setting the corresponding underparametized formula for C from (38) with Assumption G.2, we get

$$\frac{\sigma_\eta^2}{d} + \frac{\sigma_\xi^2}{\widetilde{n}-d-1} < \frac{b}{d}m \tag{96}$$

i.e.,

$$\sigma_\eta^2 + \frac{d\sigma_\xi^2}{\widetilde{n}-d-1} < bm. \tag{97}$$

From (94) and (97) we get the condition in Theorem 5.3.

N Consistency: Proofs and Discussion

N.1 Auxiliary Lemma

Lemma N.1. *The Stieltjes transform of the Marchenko–Pastur distribution $g(-\phi, \gamma)$ in (G.2) approaches 0 for a fixed γ if and only if $\phi \to \infty$.*

The lemma proof is as follows.

Let $\zeta_{(\phi,\gamma)} \triangleq \phi + 1 - \gamma$.

Then the function can be written as

$$g(-\phi;\gamma) \;=\; \frac{-\zeta_{(\phi,\gamma)} + \sqrt{\zeta_{(\phi,\gamma)}^2 + 4\gamma\phi}}{2\gamma\phi}. \tag{98}$$

We rationalize the numerator:

$$g(-\phi;\gamma) = \frac{\sqrt{\zeta_{(\phi,\gamma)}^2 + 4\gamma\phi} - \zeta_{(\phi,\gamma)}}{2\gamma\phi} = \frac{\left(\sqrt{\zeta_{(\phi,\gamma)}^2 + 4\gamma\phi} - \zeta_{(\phi,\gamma)}\right)\left(\sqrt{\zeta_{(\phi,\gamma)}^2 + 4\gamma\phi} + \zeta_{(\phi,\gamma)}\right)}{2\gamma\phi\left(\sqrt{\zeta_{(\phi,\gamma)}^2 + 4\gamma\phi} + \zeta_{(\phi,\gamma)}\right)} \tag{99}$$

Simplifying the numerator gives

$$g(-\phi;\gamma) = \frac{4\gamma\phi}{2\gamma\phi\left(\sqrt{\zeta_{(\phi,\gamma)}^2 + 4\gamma\phi} + \zeta_{(\phi,\gamma)}\right)} = \frac{2}{\sqrt{\zeta_{(\phi,\gamma)}^2 + 4\gamma\phi} + \zeta_{(\phi,\gamma)}} \tag{100}$$

For fixed $\gamma > 0$, $\zeta_{(\phi,\gamma)} > 0$ for sufficiently large ϕ. Thus, from the previous expression,

$$0 < g(-\phi;\gamma) \leq \frac{1}{\phi + 1 - \gamma} \xrightarrow[\phi\to\infty]{} 0. \tag{101}$$

Conversely, the Marchenko–Pastur distribution with parameter γ is supported on $[0, (1+\sqrt{\gamma})^2]$. Therefore,

$$g(-\phi;\gamma) = \int \frac{1}{\lambda+\phi}\, d\mu_\gamma(\lambda) \geq \frac{1}{\phi + (1+\sqrt{\gamma})^2}, \tag{102}$$

where μ_γ denotes the Marchenko–Pastur distribution. Hence, $g(-\phi;\gamma) \to 0$ implies $\phi \to \infty$, completing the proof.

N.2 Proof of Theorem 5.4

For underparameterized pretrained models, i.e., $\gamma_{\text{src}} < 1$, the optimal hyperparameter $\alpha^{\text{opt}}_{\text{TL},\infty}$ in (41) is a constant independent of m, hence, $\lim_{m\to\infty} m\alpha^{\text{opt}}_{\text{TL},\infty} \to \infty$. By Lemma N.1, this implies that as the number m of pretrained models increases, the transfer learning error in (5.2) approaches to the Bayes optimal error σ_ϵ^2:

$$\lim_{m\to\infty} \bar{\mathcal{E}}_{\text{TL}} = \lim_{m\to\infty} \sigma_\epsilon^2 \left(1 + \gamma_{\text{tgt}} \cdot g(-m\alpha^{\text{opt}}_{\text{TL},\infty}; \gamma_{\text{tgt}})\right) = \sigma_\epsilon^2 \tag{103}$$

This proves the consistency in (13).

For overparameterized pretrained models, i.e., $\gamma_{\text{src}} > 1$, we will use the corresponding optimal hyperparameter value from (41). In this case, $\alpha^{\text{opt}}_{\text{TL},\infty}$ depends on m and therefore we need to analyze where $m\alpha^{\text{opt}}_{\text{TL},\infty}$ tends to when $m \to \infty$:

$$\lim_{m\to\infty} m\alpha^{\text{opt}}_{\text{TL},\infty} = \lim_{m\to\infty} m \frac{\sigma_\epsilon^2 \gamma_{\text{tgt}} m}{\frac{\gamma_{\text{src}}-1}{\gamma_{\text{src}}} b + (m-1) b (\frac{1-\gamma_{\text{src}}}{\gamma_{\text{src}}})^2 + \frac{1}{\gamma_{\text{src}}}\left(\sigma_\eta^2 + \frac{\gamma_{\text{src}}\cdot\sigma_\xi^2}{\gamma_{\text{src}}-1}\right)} = \frac{\sigma_\epsilon^2 \gamma_{\text{tgt}}}{b(\frac{1-\gamma_{\text{src}}}{\gamma_{\text{src}}})^2} < \infty \tag{104}$$

Therefore, by Lemma N.1,

$$\lim_{m\to\infty} \bar{\mathcal{E}}_{\text{TL}} = \lim_{m\to\infty} \sigma_\epsilon^2 \left(1 + \gamma_{\text{tgt}} \cdot g(-m\alpha^{\text{opt}}_{\text{TL},\infty}; \gamma_{\text{tgt}})\right) > \sigma_\epsilon^2, \tag{105}$$

which proves the inconsistency in (14).

N.3 Discussion on Consistency in Our Results

Figure 2 shows results for settings that correspond to the data distributions and task relations of Theorem 5.4. Indeed, for underparameterized pretrained models, the transfer learning errors approach the Bayes optimal error (shown as the dotted horizontal line) as the number of pretrained models increases – this demonstrates consistency. For overparameterized pretrained models, the transfer learning errors can be far from the Bayes optimal error, especially for high source overparameterization levels, despite the increase in the number of pretrained models – this demonstrates inconsistency. Hence, the empirical results support Theorem 5.4.

Figure O.3b demonstrates that the consistency and inconsistency trends of Theorem 5.4 can occur also for task relation operators other than orthonormal matrices. However, for settings that deviate from the setting of Theorem 5.4 (e.g., Figs. 1), usually consistency is not achieved for underparameterized models; specifically, the transfer learning test error converges to some minimal error value, somewhat higher than the Bayes optimal error. Importantly, even in these cases, the minimal transfer learning test error using underparameterized models can be much lower than using overparameterized pretrained models — demonstrating that the potential benefits of using multiple pretrained models can be significantly restricted by their overparameterization. We will address this problem using the proposed debiasing.

O Overparameterization Debiasing under Isotropic Source Input Assumption: Additional Results, Details, and Discussion

O.1 Formulation of the Debiasing Algorithm under Isotropic Source Input Assumption

The overparameterization debiasing procedure is formulated in Algorithm 1.

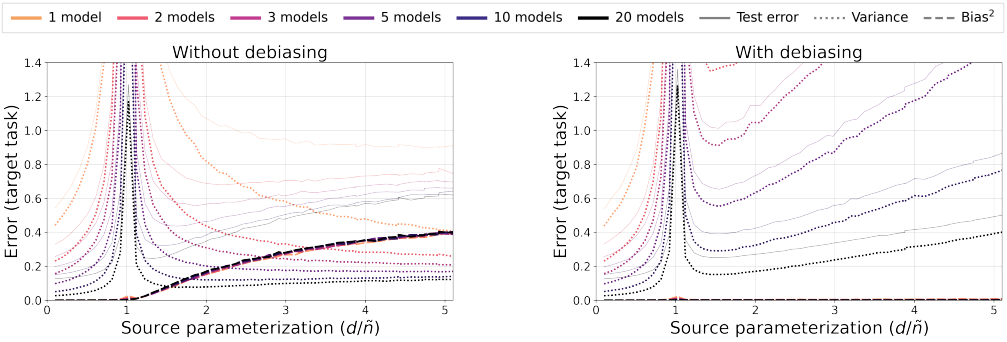
Algorithm 1 Transfer learning with overparameterization debiasing for unknown task relations under isotropic source assumptions

Require: Target training data $(\mathbf{X}, \mathbf{y})$; pretrained models $\{\widehat{\boldsymbol{\theta}}_j\}_{j=1}^m$; source sample sizes $\{\widetilde{n}_j\}_{j=1}^m$; transfer strength $\alpha_{\mathrm{TL}} > 0$.

1: **for** $j = 1, \ldots, m$ **do**
2: **if** $d \leq \widetilde{n}_j$ **then**
3: Set $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$.
4: **else**
5: Set $\widetilde{\mathbf{H}}_j = \frac{\widetilde{n}_j}{d}\mathbf{I}_d$.
6: **end if**
7: **end for**
8: Solve:

$$\widehat{\boldsymbol{\beta}}_{\mathrm{TL}} = \underset{\mathbf{b}\in\mathbb{R}^d}{\arg\min} \left\{ \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \left\| \widetilde{\mathbf{H}}_j \mathbf{b} - \widehat{\boldsymbol{\theta}}_j \right\|_2^2 \right\}. \tag{106}$$

9: **Return** $\widehat{\boldsymbol{\beta}}_{\mathrm{TL}}$.



(a) Without debiasing (b) With debiasing

Figure O.1: **Bias-variance decomposition in well-specified case.** Dashed and dotted lines denote the bias and variance terms, respectively. Here, $\mathbf{H}_j$ is $\mathbf{I}_d$, $\sigma_\xi^2 = \sigma_\eta^2 = 0.5$. Without debiasing (a) uses $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$; with debiasing (b) uses $\widetilde{\mathbf{H}}_j = \rho_j \mathbf{I}_d$.

O.2 Results and Discussion

Figures 4, O.1 and P.1 show the transfer learning test error graphs and their bias-variance decompositions. For the wellspecified case in Fig. O.1 where the task relation matrices are identity matrices ($\widetilde{\mathbf{H}}_j = \mathbf{H}_j = \mathbf{I}_d$) i.e. without debiasing, shows that there is no bias for underparameterized pretrained models (i.e., for source parameterization level is less than 1). Conversely, for overparameterized pretrained models, overparameterization bias is observed, unaffected by the number of pretrained models. In contrast, for with debiasing, Fig. 4 demonstrates that transfer learning with overparameterization debiasing successfully mitigates the overparameterization bias. This empirically supports Theorem 5.5.

When the assumed task relation matrices are set to $\mathbf{I}_d$ and differ from the true task relation matrices (i.e., $\mathbf{H}_j \neq \widetilde{\mathbf{H}}_j = \mathbf{I}_d$), an additional misspecification bias occurs. For example, in Figures 4 we consider a setting where the same circulant matrix is used for all the task relation matrices. For transfer learning without debiasing, the misspecification bias (due to $\mathbf{H}_j \neq \widetilde{\mathbf{H}}_j$) remains a non-zero constant when using underparameterized pretrained models, regardless of the number of pretrained models. This constant error level is then increased by the overparameterization bias when the pretrained models become more overparameterized. Conversely, in transfer learning with debiasing, the overparameterization bias is significantly mitigated. The slightly higher bias for a small number of pretrained models quickly reduces when more pretrained models are used; i.e., the overparameterization bias effectively vanishes as the number of pretrained models increases, leaving only the inherent misspecification bias.

O.3 Theory for the Debiasing in Case of Isotropic Input and Known Task Relations $\mathbf{H}_j = \mathbf{I}_d$

For a start, in this subsection, we analytically examine the proposed debiasing in a relatively simple setting where the true task relation operators are known. Next, in subsection P we will empirically analyze the proposed debiasing in more general settings including unknown true task relation operators.

Theorem O.1. *Under Assumptions 3.1, 3.2, 4.1 and target data with isotropic input covariance, task relation $\mathbf{H}_j = \mathbf{I}_d,\ \forall j \in \{1, \ldots, m\}$, and setting $\widehat{\mathbf{H}}$ according to the debiasing approach in Algorithm 1, the optimal hyperparameter α_{TLdeb} for transfer learning with m overparameterized pretrained models and debiasing when $d \geq \widetilde{n}_j + 2$ is*

$$\alpha_{\mathrm{TLdeb}}^{\mathrm{opt}} = \frac{\sigma_\epsilon^2 \sum_{j=1}^m \tilde{n}_j^2}{n \sum_{j=1}^m \tilde{n}_j^2 C_{\mathrm{deb},j}} \tag{107}$$

where

$$C_{\mathrm{deb},j} \triangleq \frac{\widetilde{n}_j}{d}\left(\left(1 - \frac{\widetilde{n}_j}{d}\right)\frac{b}{d} + \frac{\sigma_{\eta_j}^2}{d} + \frac{\sigma_{\xi_j}^2}{d - \widetilde{n}_j - 1}\right). \tag{108}$$

Then, the optimally tuned transfer learning with m overparameterized pretrained models and debiasing has the following expected test error:

$$\bar{\mathcal{E}}_{\mathrm{TLdeb}} = \sigma_\epsilon^2 \left(1 + \mathbb{E}_{\mathbf{X}}\left[\mathrm{Tr}\left\{\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{deb}}^{\mathrm{opt}} \sum_{j=1}^m \frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d\right)^{-1}\right\}\right]\right). \tag{109}$$

The proof is provided in Appendix R.1.

Corollary O.2. *Under Assumption G.2, the formulations in Theorem O.1 are as follows.*

$$\alpha_{\mathrm{TLdeb}}^{\mathrm{opt}} = \frac{\sigma_\epsilon^2}{n C_{\mathrm{deb}}} \tag{110}$$

where

$$C_{\mathrm{deb}} \triangleq \frac{\widetilde{n}}{d}\left(\left(1 - \frac{\widetilde{n}}{d}\right)\frac{b}{d} + \frac{\sigma_\eta^2}{d} + \frac{\sigma_\xi^2}{d - \widetilde{n} - 1}\right). \tag{111}$$

Then,

$$\bar{\mathcal{E}}_{\mathrm{TLdeb}} = \sigma_\epsilon^2 \left(1 + \mathbb{E}_{\mathbf{X}}\left[\mathrm{Tr}\left\{\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{deb}}^{\mathrm{opt}} m \frac{\widetilde{n}^2}{d^2}\mathbf{I}_d\right)^{-1}\right\}\right]\right). \tag{112}$$

Corollary O.3. *Under Assumption 4.3 and G.2, the test error formulation in Theorem O.1 for an asymptotic setting with a fixed m becomes*

$$\bar{\mathcal{E}}_{\mathrm{TLdeb}} \to \sigma_\epsilon^2 \left(1 + \gamma_{\mathrm{tgt}} \cdot g\left(-\frac{m\alpha_{\mathrm{deb},\infty}^{\mathrm{opt}}}{\gamma_{\mathrm{src}}^2}; \gamma_{\mathrm{tgt}}\right)\right) \tag{113}$$

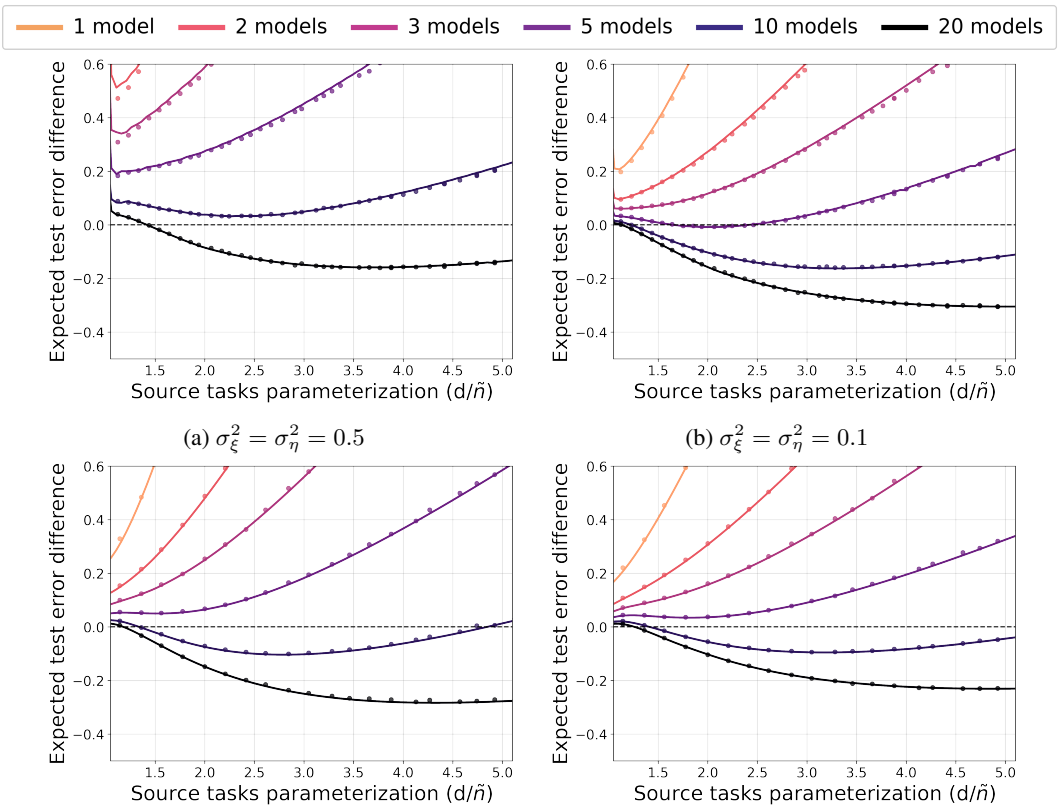
where g is the Stieltjes transform of the Marchenko-Pastur distribution as defined in (G.2) but here it gets a different first argument, and the asymptotically optimal hyperparameter is

$$\alpha_{\mathrm{TLdeb},\infty}^{\mathrm{opt}} = \gamma_{\mathrm{tgt}} \frac{\sigma_\epsilon^2 \gamma_{\mathrm{src}}}{\frac{\gamma_{\mathrm{src}}-1}{\gamma_{\mathrm{src}}} b + \sigma_\eta^2 + \frac{\gamma_{\mathrm{src}}}{\gamma_{\mathrm{src}}-1}\sigma_\xi^2}. \tag{114}$$

P A Bias-Variance Tradeoff Perspective on Overparameterization Debiasing

Now we turn to analyze the bias and variance components of the test error that transfer learning with and without debiasing achieves. Recall that the test error can be formulated as follows [5]:

$$\mathcal{E}\left(\widehat{\boldsymbol{\beta}}\right) = \sigma_\epsilon^2 + \mathrm{ErrBias}^2\left(\widehat{\boldsymbol{\beta}}\right) + \mathrm{ErrVar}\left(\widehat{\boldsymbol{\beta}}\right) \tag{115}$$



(a) $\sigma_\xi^2 = \sigma_\eta^2 = 0.5$ (b) $\sigma_\xi^2 = \sigma_\eta^2 = 0.1$

(c) $\sigma_\xi^2 = \sigma_\eta^2 = 0.5$ (d) $\sigma_\xi^2 = \sigma_\eta^2 = 0.1$

Figure O.2: Difference in expected test error between transfer learning with and without debiasing in the source overparametrized regime. The error difference is examined for a varying source task overparameterization level and for several noise levels $\sigma_\xi^2, \sigma_\eta^2$. A negative error difference implies that debiasing is beneficial. In Figs. O.2a and O.2b, the true relation is $\mathbf{H}_j = \mathbf{I}_d$. The assumed relations are $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$ (without debiasing) and $\widetilde{\mathbf{H}}_j = \frac{\tilde{n}_j}{d}\mathbf{I}_d$ (with debiasing). In Figs. O.2c and O.2d, the true relation matrices are circulant with $\kappa = 1,000$. For Fig. O.2c, we use the matched assumptions: $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$ (without debiasing) and $\widetilde{\mathbf{H}}_j = \frac{\tilde{n}_j}{d}\mathbf{H}_j$ (with debiasing). Conversely, for Fig. O.2d, the assumed relation matrices are identical to those in Figs. O.2a and O.2b. (More figures with different noises and task relations can be seen in Appendix T.4).

where $\mathcal{E}\left(\widehat{\boldsymbol{\beta}}\right)$ is the test error of $\widehat{\boldsymbol{\beta}}$ in the target task, as was defined in (4) as $\mathcal{E}$;

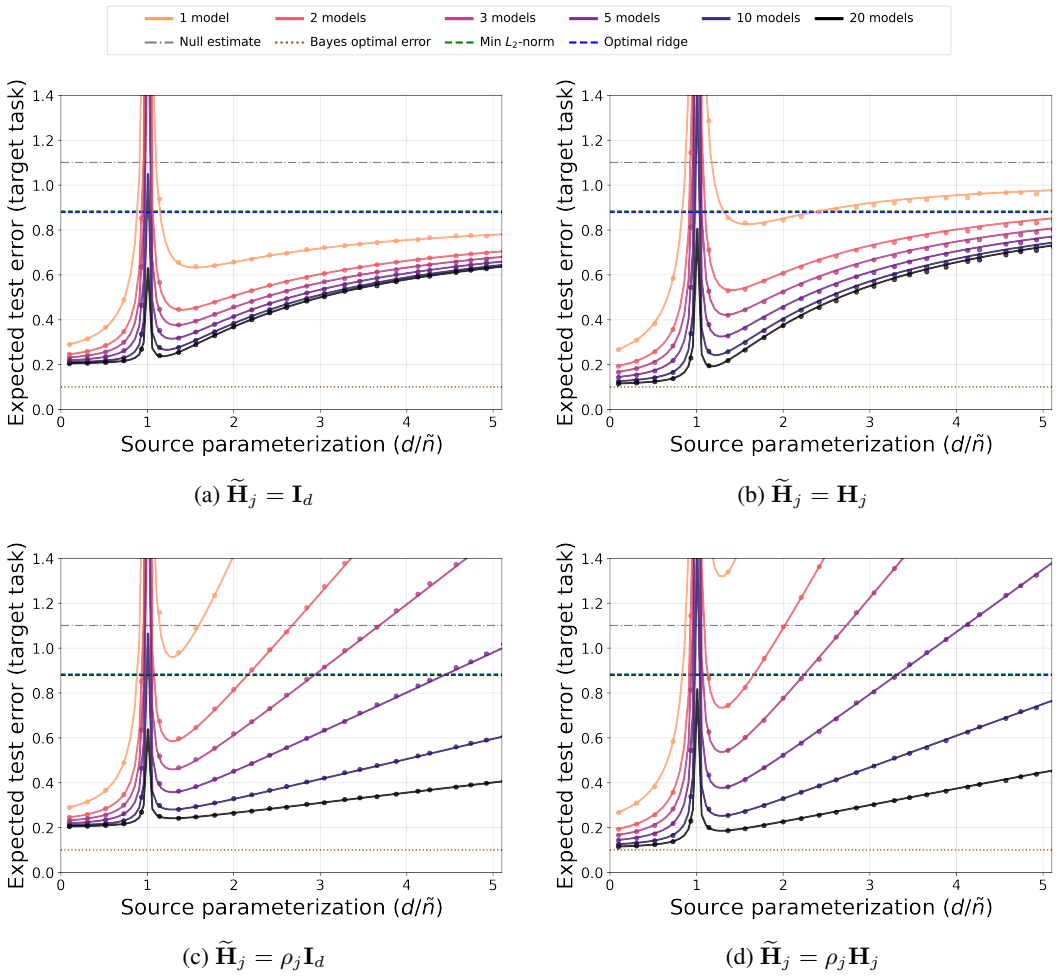
$$\text{ErrBias}^2\left(\widehat{\boldsymbol{\beta}}\right) = \mathbb{E}_{\mathbf{x}}\left[\left(\mathbb{E}_{\mathcal{D}_{\text{all}}}\left[\widehat{\boldsymbol{\beta}}\right]^T\mathbf{x} - \boldsymbol{\beta}^T\mathbf{x}\right)^2\right] \tag{116}$$

is the squared bias error component; and

$$\text{ErrVar}\left(\widehat{\boldsymbol{\beta}}\right) = \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathcal{D}_{\text{all}}}\left[\left(\mathbb{E}_{\mathcal{D}_{\text{all}}}\left[\widehat{\boldsymbol{\beta}}\right]^T\mathbf{x} - \widehat{\boldsymbol{\beta}}^T\mathbf{x}\right)^2\right] \tag{117}$$

is the variance error component. In (116)-(117), $\mathbf{x}$ is a test input drawn the target data model independently of the training data; $\mathcal{D}_{\text{all}}$ is defined as the union of all the training datasets of the target and source tasks. In Appendix Q.2, we further develop the formulations of (116) and (117).

In Fig. 4 we show the empirically-computed decompositions of the test error (in solid lines) to its squared bias component (in dashed lines) and variance component (in dotted lines). These experiments follow Assumption 3.2 and consider the formulations of the test error and its components (116)-(117) also with expectation over isotropic true target parameters $\boldsymbol{\beta}$. These results elucidate how



(a) $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$ (b) $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$

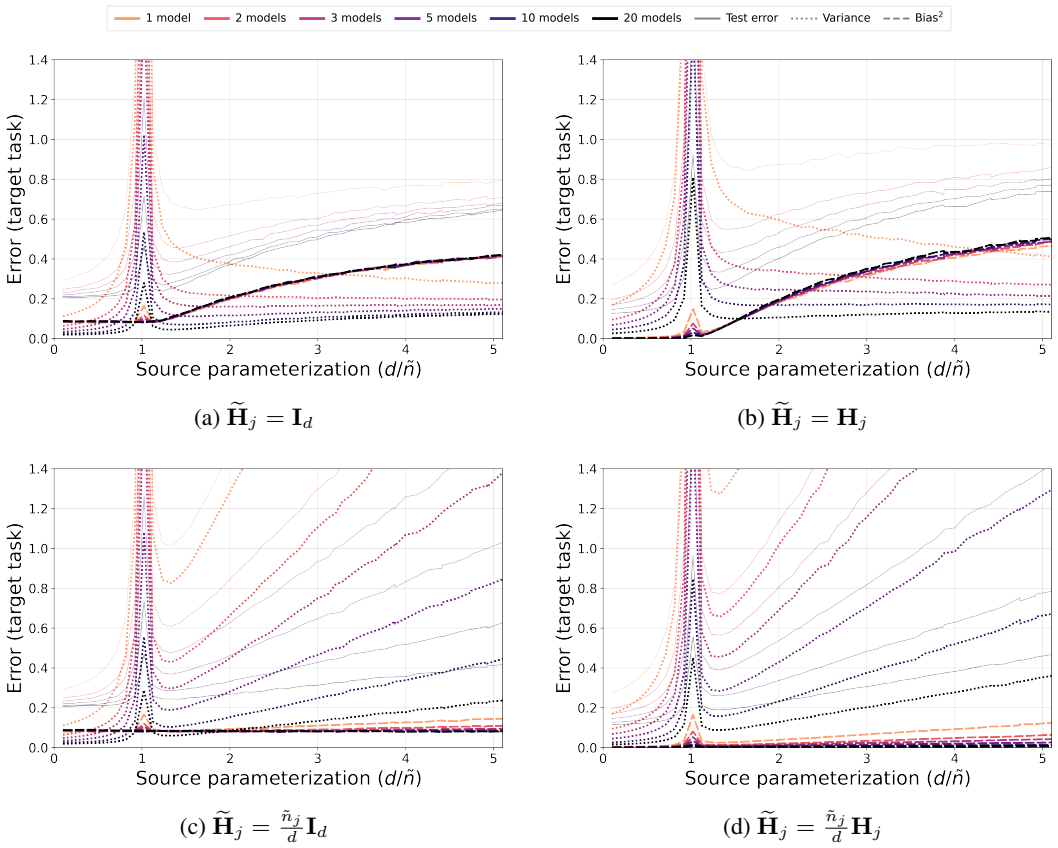
(c) $\widetilde{\mathbf{H}}_j = \rho_j \mathbf{I}_d$ (d) $\widetilde{\mathbf{H}}_j = \rho_j \mathbf{H}_j$

Figure O.3: Comparison between different assumed relation matrices under ill-condition of true relation matrix. In all figures $\mathbf{H}_j$ is a circulant matrix with condition number $\kappa = 1000$ and $\sigma_\xi^2 = \sigma_\eta^2 = 0.1$, $\gamma_{\text{tgt}} = 4$. Left Column displays using the identity matrix or the scaled identity matrix for without and with debiasing, respectively, and the right column displays using the true relation matrix or the scaled true relation matrix for without and with debiasing, respectively. First row displays without debiasing and the second is with. The test error difference between O.3a-O.3b and O.3c-O.3d is provided in Fig. P.2.

the bias and variance components of the test error are affected by the number of pretrained models, their parameterization levels, whether the proposed debiasing approach is applied or not.

Now, we will use the bias-variance decomposition for addressing three principal questions that the proposed debiasing approach raises.

Question #1: Can the transfer learning perform well despite replacing the (unknown) task relation operators $\{\mathbf{H}_j\}_{j=1}^m$ with scaled identity matrices? Yes, it can. Dar et al. [6] showed for transfer learning with a single pretrained model that using $\widetilde{\mathbf{H}} = \mathbf{I}_d$ can significantly outperform the usage of the true task relation operator $\widetilde{\mathbf{H}} = \mathbf{H}$. They showed that this can happen when the target task is overparameterized (i.e., $\mathbf{X}^T\mathbf{X}$ is rank deficient) and the true $\mathbf{H}$ has small singular values that cause numerical instability in the matrix inversion $\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\text{TL}}\widetilde{\mathbf{H}}^T\widetilde{\mathbf{H}}\right)^{-1}$ needed for the predictor in (7) with $m = 1$. Clearly, there is a tradeoff between not using the true $\mathbf{H}$ and reducing the numerical instability that it may incur; indeed, Dar et al. [6] showed that using $\widetilde{\mathbf{H}} = \mathbf{I}_d$ may perform worse when the true operator $\mathbf{H}$ is far from $\mathbf{I}_d$.



(a) $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$ (b) $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$

(c) $\widetilde{\mathbf{H}}_j = \frac{\tilde{n}_j}{d}\mathbf{I}_d$ (d) $\widetilde{\mathbf{H}}_j = \frac{\tilde{n}_j}{d}\mathbf{H}_j$

Figure P.1: Bias-variance decomposition of the expected test error of Fig. O.3. The test error difference between O.3a-O.3b and O.3c-O.3d is provided in Fig. P.2.

The results by Dar et al. [6] underscored that the true task relation operator may not be necessary for overparameterized transfer learning — this lesson motivates us also in our extension to using multiple pretrained models: Our predictor in (7) includes the matrix inversion $\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}$ that can be numerically instable when $\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j$ has small eigenvalues, which may happen for $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$ and resolved by $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$.

Accordingly, our results here show that using $\mathbf{I}_d$ instead of the true $\mathbf{H}$ can be beneficial for transfer learning with and without our debiasing approach; this can be observed by comparing Fig. O.3a to Fig. O.3b, comparing Fig. O.3c to Fig. O.3d. Moreover, this can be observed in our bias-variance decomposition graphs Fig. P.1a compared to Fig. P.1b, and Fig. P.1c compared to Fig. P.1d; these graphs are for a case where all the task relation matrices were the same circulant matrix with a high condition number. It is evident that using $\mathbf{I}_d$ instead of the true $\mathbf{H}$ can be highly beneficial when the source tasks are overparameterized.

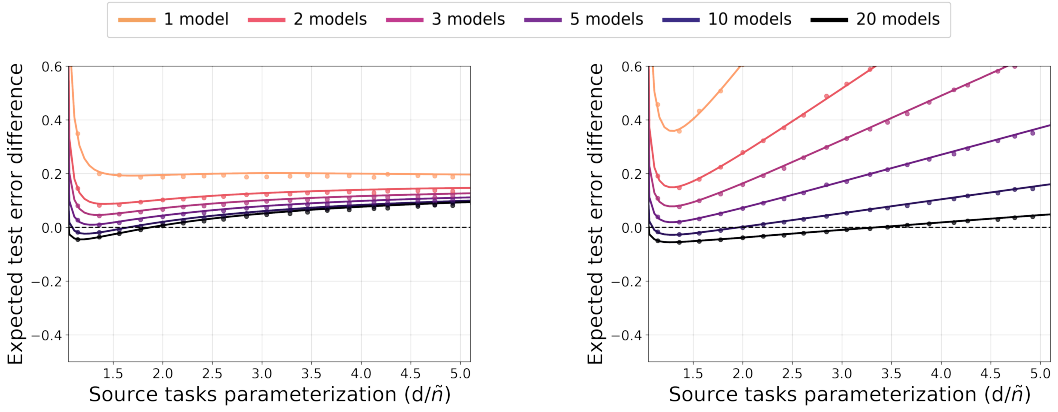
Importantly, while Dar et al. [6] considered replacing the true operator only with the (unscaled) identity matrix, here we provide the new idea of overparameterization debiasing – i.e., replacing the true operator with a scaled identity matrix $\widetilde{\mathbf{H}}_j = \frac{\tilde{n}_j}{d}\mathbf{I}_d$ that compensates for the overparameterization bias of the pretrained model. This debiasing can reduce the bias (see Fig.P.1c). Nevertheless, the reduced bias is at the expense of increased variance (Fig. P.1), which *sometimes* may be very high and cause overall performance degradation due to the debiasing. This leads to our next question.

Question #2: When is the proposed overparameterization debiasing beneficial compared to $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$, $\forall j \in \{1,\dots,m\}$? Debiasing using $\widetilde{\mathbf{H}}_j = \frac{\tilde{n}_j}{d}\mathbf{I}_d$ has the debiasing factor $\frac{\tilde{n}_j}{d}$ that compensates for the overparameterization bias of the corresponding pretrained model. However, for pretrained models with high overparameterization levels, $\frac{\tilde{n}_j}{d}$ can be small such

that $\widetilde{\mathbf{H}}_j = \frac{\widetilde{n}_j}{d}\mathbf{I}_d$ can potentially introduce high numerical instability in the matrix inversion $\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^{m}\widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}$ if $\mathbf{X}^T\mathbf{X}$ is rank deficient (e.g., due to overparameterized target task). For example, if all the pretrained models are overparameterized, then the debiasing requires the inversion $\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^{m}\left(\frac{\widetilde{n}_j}{d}\right)^2\mathbf{I}_d\right)^{-1}$, which can be numerically instable if $\sum_{j=1}^{m}\left(\frac{\widetilde{n}_j}{d}\right)^2$ is relatively small and $\mathbf{X}^T\mathbf{X}$ has a rank lower than d. Such numerical instability is related to the variance error component, and if it is too high it may render the debiasing unbeneficial. This can be observed in the relatively right side of the pretrained overparameterization axis in Figs. P.1c, P.1d showing that the variance error component can become very high due to debiasing highly overparameterized pretrained models (if the number of pretrained models is insufficient; recall that Theorem 5.7 also shows that debiasing benefits depend on the number of pretrained models and their parameterization level). Note that significantly increasing the value of the hyperparameter α_{TL} increases the bias error component (because, e.g., it overly reduces the effective use of the target data of $\mathbf{X}^T\mathbf{X}$) and therefore it cannot sufficiently compensate for the high error that the numerical instability introduces. This implies that the debiasing approach by itself can be beneficial for overparameterized pretrained models whose overparameterization level is not too high.

This leads us to the following question that intends to unleash the benefits of both the debiasing and multiple pretrained models.

Question #3: Can sufficiently many pretrained models and debiasing mutually unleash each other's benefits? At this point we understand that the proposed debiasing is beneficial if the pretrained overparameterization is not too high, because for highly overparameterized pretrained models the debiasing may induce an overly high variance. Remarkably, the variance increase can be compensated by using more pretrained models. First, this can be observed analytically in Theorem 5.7 where a sufficiently large m helps to satisfy the condition for beneficial debiasing. Moreover, this can be observed empirically in the bias-variance graphs for the more general settings (Figs. P.1c, P.1d): Although the variance increases due to the strong debiasing for high overparameterization levels, the overall variance curve can be significantly lower if more pretrained models are used – this yields a lower test error when using debiasing compared to without debiasing. Accordingly, it can be observed in the results that the proposed overparameterization debiasing can significantly improve the utility of overparameterized pretrained models if sufficiently many of them are used.



(a) Without debiasing

(b) With debiasing

Figure P.2: Expected test error difference between using the true relation matrix $\mathbf{H}_j$ to using the identity matrix $\mathbf{I}_d$ as $\widetilde{\mathbf{H}}_j$ in transfer learning without debiasing in P.2a, and using $\rho_j\mathbf{H}_j$ and $\rho_j\mathbf{I}_d$ when debiasing in P.2b corresponding to the figures in O.3. Positive difference implies that using $\mathbf{I}_d$ is better than using the true $\mathbf{H}$.

Q Bias-Variance Decomposition for Various Task Relations

Q.1 Proof of Theorem 5.5

For the solution $\widehat{\boldsymbol{\beta}}_{\mathrm{TL}} = \left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{-1} \left(\mathbf{X}^T\mathbf{y} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widehat{\boldsymbol{\theta}}_j\right)$, we will decompose the bias and variance under two different $\tilde{\mathbf{H}}_j$, in the case where $\mathbf{H}_j$ is known

- $\tilde{\mathbf{H}}_j = \mathbf{H}_j$
- $\tilde{\mathbf{H}}_j = \frac{\widetilde{n}_j}{d}\mathbf{H}_j$

First, we would look at the expected predictor $\mathbb{E}[\widehat{\boldsymbol{\beta}}]$:

$$\begin{aligned}
\mathbb{E}\left[\widehat{\boldsymbol{\beta}}_{\mathrm{TL}}\right] &= \mathbb{E}[(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j)^{-1}(\mathbf{X}^T y + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \hat{\boldsymbol{\theta}}_j)] \\
&= \mathbb{E}[(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j)^{-1} \quad \times \\
&\quad ((\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j)\boldsymbol{\beta} - n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\boldsymbol{\beta} + \mathbf{X}^T\boldsymbol{\epsilon} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \hat{\boldsymbol{\theta}}_j)] \\
&= \mathbb{E}[(\boldsymbol{\beta} + n\alpha_{\mathrm{TL}}(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j)^{-1} \quad \times \\
&\quad (\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \hat{\boldsymbol{\theta}}_j - \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\boldsymbol{\beta}) + (\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j)^{-1}(\mathbf{X}^T\boldsymbol{\epsilon})] \\
&= \boldsymbol{\beta} + n\alpha_{\mathrm{TL}}\mathbb{E}[(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j)^{-1}(\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T (\hat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}))]
\end{aligned}$$

Then, the bias of the transfer learning predictor $\widehat{\boldsymbol{\beta}}_{\mathrm{TL}}$ is

$$\begin{aligned}
\mathrm{Bias}\left(\widehat{\boldsymbol{\beta}}_{\mathrm{TL}}\right) &= \mathbb{E}\left[\widehat{\boldsymbol{\beta}}_{\mathrm{TL}}\right] - \boldsymbol{\beta} && (118)\\
&= n\alpha_{\mathrm{TL}}\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{-1} \left(\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right)\right] && (119)\\
&= n\alpha_{\mathrm{TL}}\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{-1}\right] \mathbb{E}\left[\left(\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right)\right] && (120)\\
&= n\alpha_{\mathrm{TL}}\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{-1}\right] \left(\sum_{j=1}^m \widetilde{\mathbf{H}}_j^T \left(\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j\right] - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right) && (121)
\end{aligned}$$

Importantly, if $\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j\right] - \widetilde{\mathbf{H}}_j\boldsymbol{\beta} = \mathbf{0}$ for all j, then $\mathrm{Bias}\left(\widehat{\boldsymbol{\beta}}\right) = \mathbf{0}$ and the transfer learning predictor is unbiased.

From the expectation of a pretrained model given $\boldsymbol{\beta}$ in (83), we get that

$$\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j\right] - \widetilde{\mathbf{H}}_j\boldsymbol{\beta} = \begin{cases} \left(\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)\boldsymbol{\beta} & \text{for } d \le \widetilde{n}_j, \\ \left(\frac{\widetilde{n}}{d}\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)\boldsymbol{\beta} & \text{for } d > \widetilde{n}_j. \end{cases} \tag{122}$$

Therefore, the predictor is unbiased if

- for any underparameterized pretrained model we use $\widetilde{\mathbf{H}}_j = \mathbf{H}_j$,
- for any overparameterized pretrained model we use $\widetilde{\mathbf{H}}_j = \frac{\widetilde{n}}{d}\mathbf{H}_j$,

This proves Theorem 5.5 and motivates the overparameterization debiasing that we propose in this paper.

Q.2 Additional Formulations of the Bias-Variance Decomposition

The squared bias error term can be formulated [5] as

$$\mathrm{ErrBias}^2\left(\widehat{\boldsymbol{\beta}}\right) = \mathbb{E}_{\mathbf{x}}\left[\left(\mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right]^T\mathbf{x} - \boldsymbol{\beta}^T\mathbf{x}\right)^2\right] \tag{123}$$

where $\mathbf{x}$ is a test input drawn from the target data model independently of the training data. This can be further developed as follows:

$$\mathrm{ErrBias}^2\left(\widehat{\boldsymbol{\beta}}\right) = \left(\mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right] - \boldsymbol{\beta}\right)^T\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right] - \boldsymbol{\beta}\right) \tag{124}$$

$$= \mathrm{Tr}\left\{\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right] - \boldsymbol{\beta}\right)\left(\mathbb{E}\left[\widehat{\boldsymbol{\beta}}\right] - \boldsymbol{\beta}\right)^T\right\} \tag{125}$$

To formulate in more detail the role of the number m of pretrained models and the debiasing by a multiplicative factor, consider the simpler setting where $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{I}_d$, $\mathbf{H}_j = \mathbf{I}_d$ and $\widetilde{\mathbf{H}}_j = a\mathbf{I}_d$ where a is a constant to be set for learning, for all $j \in \{1, \ldots, m\}$ and assumption G.2. Then, the bias in (121) becomes, for overparameterized pretrained models,

$$\mathrm{Bias}\left(\widehat{\boldsymbol{\beta}}\right) = n\alpha_{\mathrm{TL}}\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}ma^2\mathbf{I}_d\right)^{-1}\right]ma\left(\frac{\widetilde{n}}{d} - a\right)\boldsymbol{\beta} \tag{126}$$

this expression can be further developed to show that the bias (and the squared bias error term) increases as the number m of pretrained models is larger. By Assumption 3.2, we get that the squared bias term of the test error, expected w.r.t. $\boldsymbol{\beta}$, is

$$\mathbb{E}_{\boldsymbol{\beta}}\left[\mathrm{ErrBias}^2\right]\left(\widehat{\boldsymbol{\beta}}\right) = n^2\alpha_{\mathrm{TL}}^2m^2a^2\left(\frac{\widetilde{n}}{d} - a\right)^2\frac{b}{d} \tag{127}$$

$$\cdot\,\mathrm{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}ma^2\mathbf{I}_d\right)^{-1}\right]\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}ma^2\mathbf{I}_d\right)^{-1}\right]^T\right\} \tag{128}$$

$$= n^2\alpha_{\mathrm{TL}}^2m^2a^2\left(\frac{\widetilde{n}}{d} - a\right)^2\frac{b}{d}\,\mathrm{Tr}\left\{\left(\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}ma^2\mathbf{I}_d\right)^{-1}\right]\right)^2\right\}$$

$$= n^2\alpha_{\mathrm{TL}}^2a^2\left(\frac{\widetilde{n}}{d} - a\right)^2\frac{b}{d}\,\mathrm{Tr}\left\{\left(\mathbb{E}\left[\left(\frac{1}{m}\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}a^2\mathbf{I}_d\right)^{-1}\right]\right)^2\right\}. \tag{129}$$

The variance error term can be formulated [5] as

$$\mathrm{ErrVar}\left(\widehat{\boldsymbol{\beta}}\right) = \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\left(\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right]^T\mathbf{x} - \widehat{\boldsymbol{\beta}}^T\mathbf{x}\right)^2\right] \tag{130}$$

$$= \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\left(\left(\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right] - \widehat{\boldsymbol{\beta}}\right)^T\mathbf{x}\right)^2\right] \tag{131}$$

$$= \mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\left(\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right] - \widehat{\boldsymbol{\beta}}\right)^T\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right] - \widehat{\boldsymbol{\beta}}\right)\right] \tag{132}$$

$$= \mathrm{Tr}\left\{\boldsymbol{\Sigma}_{\mathbf{x}}\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\left(\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right] - \widehat{\boldsymbol{\beta}}\right)\left(\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right] - \widehat{\boldsymbol{\beta}}\right)^T\right]\right\} \tag{133}$$

$$= \mathrm{Tr}\left\{\boldsymbol{\Sigma}_{\mathbf{x}}\left(\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\widehat{\boldsymbol{\beta}}^T\right] - \mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right]\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right]^T\right)\right\} \tag{134}$$

where, for brevity of expectation notation, $\mathcal{D}_{\mathrm{all}}$ is defined as the union of all the training datasets of the target and source tasks.

For the solution $\widehat{\boldsymbol{\beta}}_{\mathrm{TL}} = \left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m\widetilde{\mathbf{H}}_j^T\widetilde{\mathbf{H}}_j\right)^{-1}\left(\mathbf{X}^T\mathbf{y} + n\alpha_{\mathrm{TL}}\sum_{j=1}^m\widetilde{\mathbf{H}}_j^T\widehat{\boldsymbol{\theta}}_j\right)$,

Using $\mathbf{y} = \mathbf{X}\beta + \boldsymbol{\epsilon}$

$$\widehat{\boldsymbol{\beta}}_{\mathrm{TL}} = \boldsymbol{\beta} + (\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j)^{-1} (\mathbf{X}^T\boldsymbol{\epsilon} + n\alpha_{\mathrm{TL}} \sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T (\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}) \tag{135}$$

And then,

$$\mathbb{E}\left[\widehat{\boldsymbol{\beta}}_{\mathrm{TL}}\right] = \boldsymbol{\beta} + n\alpha_{\mathrm{TL}}\mathbb{E}[(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j)^{-1} (\sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T (\hat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}))]$$

to conclude,

$$\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right] - \widehat{\boldsymbol{\beta}} = n\alpha_{\mathrm{TL}}\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{-1}\right] \left(\sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T \left(\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j\right] - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right) \tag{136}$$

$$- n\alpha_{\mathrm{TL}} \left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{-1} \left(\sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T \left(\widehat{\boldsymbol{\theta}}_j - \widetilde{\mathbf{H}}_j\boldsymbol{\beta}\right)\right) + \boldsymbol{\beta} - \boldsymbol{\beta} \tag{137}$$

$$- \left(\mathbf{X}^T\mathbf{X} + n\alpha_{\mathrm{TL}} \sum_{j=1}^{m} \widetilde{\mathbf{H}}_j^T \widetilde{\mathbf{H}}_j\right)^{-1} \mathbf{X}^T\boldsymbol{\epsilon} \tag{138}$$

Consider the simpler setting where $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{I}_d$, $\mathbf{H}_j = \mathbf{I}_d$ and $\widetilde{\mathbf{H}}_j = \rho_j \mathbf{I}_d$, we get,

$$\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right] - \widehat{\boldsymbol{\beta}} = -n\alpha_{\mathrm{TL}} \left(\mathbf{X}^T\mathbf{X} + mn\rho^2\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1} \left(\sum_{j=1}^{m} \rho\left(\widehat{\boldsymbol{\theta}}_j - \rho\boldsymbol{\beta}\right)\right) \tag{139}$$

$$- \left(\mathbf{X}^T\mathbf{X} + mn\rho^2\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-1} \mathbf{X}^T\boldsymbol{\epsilon} \tag{140}$$

Since the source predictors are independent conditional on $\boldsymbol{\beta}$ and satisfy $\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j \mid \boldsymbol{\beta}\right] = \rho\boldsymbol{\beta}$, the law of iterated expectation gives, for $j \neq \ell$,

$$\mathbb{E}\left[\left(\widehat{\boldsymbol{\theta}}_j - \rho\boldsymbol{\beta}\right)\left(\widehat{\boldsymbol{\theta}}_\ell - \rho\boldsymbol{\beta}\right)^T\right] = \mathbf{0}.$$

Substituting all the above gives,

$$\mathrm{ErrVar}\left(\widehat{\boldsymbol{\beta}}\right) = \mathrm{Tr}\left\{\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\left(\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right] - \widehat{\boldsymbol{\beta}}\right)\left(\mathbb{E}_{\mathcal{D}_{\mathrm{all}}}\left[\widehat{\boldsymbol{\beta}}\right] - \widehat{\boldsymbol{\beta}}\right)^T\right]\right\} \tag{141}$$

$$= \mathrm{Tr}\left\{\mathbb{E}\left[\rho^2 n^2 \alpha_{\mathrm{TL}}^2 \left(\mathbf{X}^T\mathbf{X} + mn\rho^2\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-2} \left(\sum_{j=1}^{m}\left(\widehat{\boldsymbol{\theta}}_j - \rho\boldsymbol{\beta}\right)\right)\left(\sum_{j=1}^{m}\left(\widehat{\boldsymbol{\theta}}_j - \rho\boldsymbol{\beta}\right)\right)^T\right]\right\} \tag{142}$$

$$+ \sigma_\epsilon^2 \,\mathrm{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + mn\rho^2\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-2}\mathbf{X}^T\mathbf{X}\right]\right\} \tag{143}$$

$$= \rho^2 n^2 \alpha_{\mathrm{TL}}^2 \,\mathrm{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + mn\rho^2\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-2}\right] mC_{\widehat{\boldsymbol{\theta}}_j|\boldsymbol{\beta}}\right\} \tag{144}$$

$$+ \sigma_\epsilon^2 \,\mathrm{Tr}\left\{\mathbb{E}\left[\left(\mathbf{X}^T\mathbf{X} + mn\rho^2\alpha_{\mathrm{TL}}\mathbf{I}_d\right)^{-2}\mathbf{X}^T\mathbf{X}\right]\right\} \tag{145}$$

Given that $C_{\widehat{\boldsymbol{\theta}}_j|\boldsymbol{\beta}}$ symmetric positive semidefinite,

$$\mathrm{ErrVar}\left(\widehat{\boldsymbol{\beta}}\right) \leq \rho^2 n^2 \alpha_{TL}^2 \, m\mathbb{E}\left[\sum_{i=1}^{d} \frac{\lambda_{\max}(C_{\widehat{\boldsymbol{\theta}}_j|\boldsymbol{\beta}})}{(\lambda_i + mn\rho^2\alpha_{TL})^2}\right] + \sigma_\epsilon^2 \,\mathbb{E}\left[\sum_{i=1}^{d} \frac{\lambda_i}{(\lambda_i + mn\rho^2\alpha_{TL})^2}\right] \tag{146}$$

Using that α_{TL} is independent of m from 107, we can get,

$$\lim_{m\to\infty} \text{ErrVar}\left(\widehat{\boldsymbol{\beta}}\right) = 0 \tag{147}$$

R Proofs for Transfer Learning with Debiasing

R.1 Proof of Theorem O.1

Consider a setting where all the m pretrained models are overparameterized and, therefore, our debiasing set for them $\widetilde{\mathbf{H}}_j = \frac{\widetilde{n}_j}{d}$ for all j. In this theorem we also have $\mathbf{H}_j = \mathbf{I}_d$ for all j and isotropic target input $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{I}_d$. Then, our predictor (7) becomes

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{X} + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d\right)^{-1}\left(\mathbf{X}^T y + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j}{d}\widehat{\boldsymbol{\theta}}_j\right). \tag{148}$$

The corresponding expected test error is

$$\begin{aligned}
&\mathbb{E}\left[\left|\left|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right|\right|^2\right] = \mathbb{E}\left[\left|\left|(\mathbf{X}^T\mathbf{X} + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d)^{-1}(\mathbf{X}^T\epsilon + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j}{d}(\widehat{\boldsymbol{\theta}}_j - \frac{\widetilde{n}_j}{d}\boldsymbol{\beta}))\right|\right|^2\right] \\
&= \mathbb{E}\Bigg[Tr\Bigg\{(\mathbf{X}^T\mathbf{X} + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d)^{-2}(\mathbf{X}^T\epsilon + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j}{d}(\widehat{\boldsymbol{\theta}}_j - \frac{\widetilde{n}_j}{d}\boldsymbol{\beta})) \quad \times \\
&(\mathbf{X}^T\epsilon + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j}{d}(\widehat{\boldsymbol{\theta}}_j - \frac{\widetilde{n}_j}{d}\boldsymbol{\beta}))^T\Bigg\}\Bigg] \\
&= \mathbb{E}\left[Tr\left\{(\mathbf{X}^T\mathbf{X} + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d)^{-2}\sigma_\epsilon^2\mathbf{X}^T\mathbf{X}\right\}\right] \\
&\quad + \mathbb{E}\left[Tr\left\{(\mathbf{X}^T\mathbf{X} + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d)^{-2}\frac{n^2\alpha_{TLde}^2}{d^2}\sum_{j=1}^{m}\sum_{l=1}^{m}\widetilde{n}_j\widetilde{n}_l(\widehat{\boldsymbol{\theta}}_j - \frac{\widetilde{n}_j}{d}\boldsymbol{\beta})(\widehat{\boldsymbol{\theta}}_l - \frac{\widetilde{n}_l}{d}\boldsymbol{\beta}))^T)\right\}\right]
\end{aligned} \tag{149}$$

From Appendix K, we know that $\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_j \mid \boldsymbol{\beta}\right] = \frac{\widetilde{n}_j}{d}\mathbf{H}_j\boldsymbol{\beta}$, so under our assumption of $\mathbf{H}_j = \mathbf{I}_d$ we get $\mathbb{E}\left[\left(\widehat{\boldsymbol{\theta}}_j - \frac{\widetilde{n}_j}{d}\boldsymbol{\beta}\right)\left(\widehat{\boldsymbol{\theta}}_l - \frac{\widetilde{n}_l}{d}\boldsymbol{\beta}\right)^T\right] = \mathbf{0}$ for any $j \neq l$.

Hence, we get

$$\begin{aligned}
&\mathbb{E}\left[\left|\left|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right|\right|_{\Sigma_X}^2\right] = \mathbb{E}\left[Tr\left\{(\mathbf{X}^T\mathbf{X} + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d)^{-2}\sigma_\epsilon^2\mathbf{X}^T\mathbf{X}\right\}\right] \\
&+ \mathbb{E}\left[Tr\left\{(\mathbf{X}^T\mathbf{X} + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d)^{-2}\frac{n^2\alpha_{TLde}^2}{d^2}\sum_{j=1}^{m}\sum_{l=1}^{m}\widetilde{n}_j\widetilde{n}_l(\widehat{\boldsymbol{\theta}}_j - \frac{\widetilde{n}_j}{d}\boldsymbol{\beta})(\widehat{\boldsymbol{\theta}}_l - \frac{\widetilde{n}_l}{d}\boldsymbol{\beta}))^T)\right\}\right] = \\
&\mathbb{E}\left[Tr\left\{(\mathbf{X}^T\mathbf{X} + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d)^{-2}\sigma_\epsilon^2\mathbf{X}^T\mathbf{X}\right\}\right] \\
&+ \mathbb{E}\left[Tr\left\{(\mathbf{X}^T\mathbf{X} + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d)^{-2}n^2\alpha_{TLde}^2\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}C_{de,j}\right\}\right] = \\
&\sigma_\epsilon^2 + \mathbb{E}\left\{\sum_{k=1}^{d}\frac{\sigma_\epsilon^2\lambda_k + n^2\alpha_{TLde}^2\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2}C_{de,j}}{(\lambda_k + n\alpha_{TLde}\sum_{j=1}^{m}\frac{\widetilde{n}_j^2}{d^2})^2}\right\}
\end{aligned} \tag{150}$$

where $C_{de,j}\mathbf{I}_d = \mathbb{E}\left[(\widehat{\boldsymbol{\theta}}_j - \frac{\widetilde{n}_j}{d}\boldsymbol{\beta})(\widehat{\boldsymbol{\theta}}_j - \frac{\widetilde{n}_j}{d}\boldsymbol{\beta})^T\right]$ when

$$\mathbb{E}\left[(\widehat{\boldsymbol{\theta}}_j - \frac{\tilde{n}_j}{d}\boldsymbol{\beta})(\widehat{\boldsymbol{\theta}}_j - \frac{\tilde{n}_j}{d}\boldsymbol{\beta})^T\right] = \mathbb{E}[\widehat{\boldsymbol{\theta}}_j\widehat{\boldsymbol{\theta}}_j^T] - \frac{\tilde{n}_j}{d}\mathbb{E}[\widehat{\boldsymbol{\theta}}_j\boldsymbol{\beta}^T] - \frac{\tilde{n}_j}{d}\mathbb{E}[\boldsymbol{\beta}\widehat{\boldsymbol{\theta}}_j^T] + \frac{\tilde{n}_j^2}{d^2}\mathbb{E}[\boldsymbol{\beta}\boldsymbol{\beta}^T] \tag{151}$$

Calculating each component:

$$\mathbb{E}[\widehat{\boldsymbol{\theta}}_j\widehat{\boldsymbol{\theta}}_j^T] = \mathbb{E}[(\mathbf{Z}^+\mathbf{Z}\boldsymbol{\theta}_j + \mathbf{Z}^+\xi)(\mathbf{Z}^+\mathbf{Z}\boldsymbol{\theta}_j + \mathbf{Z}^+\xi)^T] = \frac{\tilde{n}_j}{d}(\frac{b+\sigma_\eta^2}{d} + \frac{\sigma_\xi^2}{d-\widetilde{n}_j-1})\mathbf{I}_d$$

$$\mathbb{E}[\widehat{\boldsymbol{\theta}}_j\boldsymbol{\beta}^T] = \mathbb{E}[\boldsymbol{\beta}\widehat{\boldsymbol{\theta}}_j^T] = \frac{\tilde{n}_j}{d}\frac{b}{d}\mathbf{I}_d$$

And,

$$\mathbb{E}[\boldsymbol{\beta}\boldsymbol{\beta}^T] = \frac{b}{d}\mathbf{I}d \tag{152}$$

So we conclude:

$$C_{de,j} = \frac{\tilde{n}_j}{d}(\frac{b+\sigma_{\eta_j}^2}{d} + \frac{\sigma_{\xi_j}^2}{d-\widetilde{n}_j-1}) - \frac{\tilde{n}_j^2}{d^2}\frac{b}{d} = (\frac{\tilde{n}_j}{d} - \frac{\tilde{n}_j^2}{d^2})\frac{b}{d} + \frac{\tilde{n}_j}{d}\left(\frac{\sigma_{\eta_j}^2}{d} + \frac{\sigma_{\xi_j}^2}{d-\widetilde{n}_j-1}\right) \tag{153}$$

Now to find the optimal α_{TLde} we will take derivative:

$$\frac{\partial\bar{\mathcal{E}}}{\partial\alpha_{TLde}} =$$

$$\mathbb{E}\Bigg\{\sum_{k=1}^{d}\frac{2n^2\alpha_{TLde}\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2}C_{de,j}(\lambda_k+n\alpha_{TLde}\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2})^2 - 2n\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2}(\lambda_k+n\alpha_{TLde}\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2})(\sigma_\epsilon^2\lambda_k+n^2\alpha_{TLde}^2\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2}C_{de,j})}{(\lambda_k+n\alpha_{TLde}\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2})^4}\Bigg\}$$

$$= \mathbb{E}\Bigg\{\sum_{k=1}^{d}\frac{2n^2\alpha_{TLde}\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2}C_{de,j}(\lambda_k+n\alpha_{TLde}\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2}) - 2n\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2}(\sigma_\epsilon^2\lambda_k+n^2\alpha_{TLde}^2\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2}C_{de,j})}{(\lambda_k+n\alpha_{TLde}\sum_{j=1}^m\frac{\tilde{n}_j^2}{d^2})^3}\Bigg\}$$

By equating the derivative of the expression to zero, we will get:

$$\left(2n^2\alpha_{TLde}\sum_{j=1}^m\frac{\widetilde{n}_j^2}{d^2}C_{de,j} - 2n\sum_{j=1}^m\frac{\widetilde{n}_j^2}{d^2}\sigma_\epsilon^2\right)\mathbb{E}\Bigg\{\sum_{k=1}^{d}\frac{\lambda_k}{(\lambda_k+n\alpha_{TLde}\sum_{j=1}^m\frac{\widetilde{n}_j^2}{d^2})^3}\Bigg\} = 0 \tag{154}$$

Concluding that the optimal alpha in the debiased case is:

$$\alpha_{TLde}^{opt} = \frac{\sigma_\epsilon^2\sum_{j=1}^m\frac{\widetilde{n}_j^2}{d^2}}{n\sum_{j=1}^m\frac{\widetilde{n}_j^2}{d^2}C_{de,j}} \tag{155}$$

Setting the optimal α in 150 will give us:

$$\bar{\mathcal{E}} = \sigma_\epsilon^2\left(1+\mathbb{E}\Bigg\{\sum_{k=1}^{d}\frac{\lambda_k+n\alpha_{TLde}^{opt}\sum_{j=1}^m\frac{\widetilde{n}_j^2}{d^2}}{(\lambda_k+n\alpha_{TLde}^{opt}\sum_{j=1}^m\frac{\widetilde{n}_j^2}{d^2})^2}\Bigg\}\right) =$$

$$\sigma_\epsilon^2\left(1+\mathbb{E}\Bigg\{\sum_{k=1}^{d}\frac{1}{\lambda_k+n\alpha_{TLde}^{opt}\sum_{j=1}^m\frac{\widetilde{n}_j^2}{d^2}}\Bigg\}\right) =$$

$$\sigma_\epsilon^2\left(1+\mathbb{E}\left[\mathrm{Tr}\left\{(\mathbf{X}^T\mathbf{X}+n\alpha_{TLde}^{opt}\sum_{j=1}^m\frac{\widetilde{n}_j^2}{d^2}\mathbf{I}_d)^{-1}\right\}\right]\right)$$

R.2 Proof of Theorem 5.6

The optimal hyperparameter $\alpha_{\mathrm{TLdeb},\infty}^{\mathrm{opt}}$ in (114) is a constant independent of m. Also, recall that our debiasing is for overparameterized pretrained models and therefore $\gamma_{\mathrm{src}} > 1$; we assume that

γ_{tgt} and γ_{src} are fixed here. Hence, $\lim_{m\to\infty} \frac{m\alpha^{\text{opt}}_{\text{TL}deb,\infty}}{\gamma^2_{\text{src}}} \to \infty$. By Lemma N.1, this implies that as the number m of pretrained models increases, the transfer learning error in (113) approaches to the Bayes optimal error σ^2_ϵ:

$$\lim_{m\to\infty} \bar{\mathcal{E}}_{\text{TL}} = \lim_{m\to\infty} \sigma^2_\epsilon \left(1 + \gamma_{\text{tgt}} \cdot g\left(-\frac{m\alpha^{\text{opt}}_{\text{TL}deb,\infty}}{\gamma^2_{\text{src}}}; \gamma_{\text{tgt}}\right)\right) = \sigma^2_\epsilon \tag{156}$$

This proves the consistency in (16) and Theorem 5.6.

R.3 Proof of Theorem 5.7

By comparing (112) and (40), we get that debiasing is beneficial when

$$\alpha^{\text{opt}}_{\text{TL}} m < \alpha^{\text{opt}}_{\text{TLdeb}} m \frac{\widetilde{n}^2}{d^2} \tag{157}$$

where by Corollary 5.1

$$\alpha^{\text{opt}}_{\text{TL}} = \frac{\sigma^2_\epsilon}{nC + \frac{bn}{d}(m-1)(1-\rho)^2} \tag{158}$$

and $C \triangleq \left(1 - \frac{\widetilde{n}}{d}\right)\frac{b}{d} + \frac{\widetilde{n}}{d}\left(\frac{\sigma^2_\eta}{d} + \frac{\sigma^2_\xi}{d-\widetilde{n}-1}\right)$.

We know from Corollary O.2 that

$$\alpha^{\text{opt}}_{\text{TLdeb}} = \frac{\sigma^2_\epsilon}{nC_{\text{deb}}} \tag{159}$$

where

$$C_{\text{deb}} \triangleq \frac{\widetilde{n}}{d}\left(\left(1 - \frac{\widetilde{n}}{d}\right)\frac{b}{d} + \frac{\sigma^2_\eta}{d} + \frac{\sigma^2_\xi}{d-\widetilde{n}-1}\right). \tag{160}$$

Setting the above in (157) and using the definition of ρ from (11) for overparameterized pretrained models, we get

$$\alpha^{\text{opt}}_{\text{TL}} < \alpha^{\text{opt}}_{\text{TLdeb}} \rho^2 \tag{161}$$

$$\frac{\sigma^2_\epsilon}{nC + \frac{bn}{d}(m-1)(1-\rho)^2} < \frac{\sigma^2_\epsilon}{nC_{\text{deb}}}\rho^2 \tag{162}$$

$$dC_{\text{deb}} < \left(dC + b(m-1)(1-\rho)^2\right)\rho^2 \tag{163}$$

$$(\rho - \rho^2)b + \rho\left(\sigma^2_\eta + \frac{d\sigma^2_\xi}{d-\widetilde{n}-1}\right) < \left((1-\rho)\,b + \rho\left(\sigma^2_\eta + \frac{d\sigma^2_\xi}{d-\widetilde{n}-1}\right) + b(m-1)(1-\rho)^2\right)\rho^2 \tag{164}$$

We will assign $t = \sigma^2_\eta + \frac{d\sigma^2_\xi}{d-\widetilde{n}-1}$ to get

$$(\rho - \rho^2)b + \rho t < ((1-\rho)\,b + \rho t + b(m-1)(1-\rho)^2)\rho^2 \tag{165}$$

$$t\frac{1+\rho}{1-\rho} < ((m-1)\rho - 1)b \tag{166}$$

$$t\frac{1+\rho}{(1-\rho)\rho} < \left(m - 1 - \frac{d}{\widetilde{n}}\right)b \tag{167}$$

$$\left(\sigma^2_\eta + \frac{d\sigma^2_\xi}{d-\widetilde{n}-1}\right)\left(\frac{d}{\widetilde{n}} + \frac{2d}{d-\widetilde{n}}\right) < \left(m - 1 - \frac{d}{\widetilde{n}}\right)b \tag{168}$$

R.4 Discussion on Debiasing Results and Performance

The condition in Theorem 5.7 implies that, for any fixed pretrained overparameterization and noise levels, there is a number m of sufficiently many pretrained models that yields beneficial debiasing. In contrast, for a fixed number of pretrained models, higher task relation noise σ_η^2 and source data noise σ_ξ^2 can make debiasing to degrade performance. This behavior is shown in Fig. 5. In Fig. 5b for high noise levels $\sigma_\xi^2 = \sigma_\eta^2 = 0.5$, only as many as 20 pretrained models can provide beneficial debiasing among the examined options. In Fig. 5a for lower noises $\sigma_\xi^2 = \sigma_\eta^2 = 0.05$, 5 and 10 pretrained models are sufficient to provide beneficial debiasing.

Note that the condition for beneficial debiasing in Theorem 5.7 depends *non-monotonically* on the pretrained overparameterization level $\frac{d}{\widetilde{n}}$. Specifically, for a fixed m, an arbitrarily high pretrained overparameterization level takes the right side of the inequality to below zero and thus hinders beneficial debiasing. This implies that beneficial debiasing requires sufficiently many pretrained models whose overparameterization levels are not too high. This is observed in Fig. 5, as a nonlinear trend of debiasing gains as function of the horizontal axis of pretrained overparameterization level — for sufficiently many pretrained models, the debiasing gains increase (i.e., the shown error difference decreases) along the pretrained overparameterization axis until a point at which the debiasing gains start to decrease and eventually can even make debiasing unbeneficial. We will explain this behavior in the next subsection.

S Anisotropic case

S.1 Debiasing-Factor Tuning Algorithm

In Algorithm 2 we describe our validation-based debiasing that can address transfer learning using pretrained models whose source statistics are anisotropic and unknown. The validation based choice does not necessarily need to know the source dataset sizes $\widetilde{n}_j$, but in case that the source dataset sizes are known they can be used for a more efficient definition of the hyperparameter grid $\mathcal{R}$ for $\tilde{\rho}$.

Algorithm 2 Transfer learning with a validation-tuned **debiasing** factor for unknown anisotropic sources

1: **Inputs**: Target task train data $\mathbf{X}, \mathbf{y}$; target validation dataset in matrix-vector form $\mathbf{X}_{\rm val}, \mathbf{y}_{\rm val}$; m pretrained models $\left\{\widehat{\boldsymbol{\theta}}_j\right\}_{j=1}^m$; hyperparameter grids $\mathcal{A}$ for α and $\mathcal{R}$ for $\tilde{\rho}$.
2: **for** each $\alpha \in \mathcal{A}$ and $\tilde{\rho} \in \mathcal{R}$ **do**
3: Set $\widetilde{\mathbf{H}}_j = \tilde{\rho}\mathbf{I}_d$ for all pretrained models
4: Solve:

$$\widehat{\boldsymbol{\beta}}_{\alpha,\tilde{\rho}} = \underset{\mathbf{b}\in\mathbb{R}^d}{\arg\min} \|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 + n\alpha \sum_{j=1}^m \left\|\widetilde{\mathbf{H}}_j\mathbf{b} - \widehat{\boldsymbol{\theta}}_j\right\|_2^2$$

5: Calculate validation error: $E_{\rm val} = \left\|\mathbf{y}_{\rm val} - \mathbf{X}_{\rm val}\widehat{\boldsymbol{\beta}}_{\alpha,\tilde{\rho}}\right\|_2^2$
6: **end for**
7: **Return** $\widehat{\boldsymbol{\beta}}_{\rm TL}$ corresponding to the pair $(\alpha, \tilde{\rho})$ that minimized $E_{\rm val}$.

S.2 Tikhonov Equivalence

$$J_\lambda(b) \;\triangleq\; \|y - Xb\|_2^2 \;+\; n\alpha \sum_{j=1}^m \|\tilde{H}_j b - \hat{\theta}_j\|_2^2 \;+\; \lambda \sum_{j=1}^m \|\tilde{H}_j b\|_2^2.$$

Expand and drop all terms independent of b (write $\equiv$ for equality up to an additive constant):

$$J_\lambda(b) \equiv \|y - Xb\|_2^2 + \sum_{j=1}^m \Big((n\alpha + \lambda)\, b^\top \tilde{H}_j^\top \tilde{H}_j b - 2n\alpha\, \hat{\theta}_j^\top \tilde{H}_j b \Big).$$

Now consider the objective *without* Tikhonov, but with a scaled matrix $\tilde{H}_j^\star = \nu\,\tilde{H}_j$ and a (possibly different) weight $\alpha^\star$:

$$J_0^\star(b) \;\triangleq\; \|y - Xb\|_2^2 \;+\; n\alpha^\star \sum_{j=1}^{m} \|\tilde{H}_j^\star b - \hat{\theta}_j\|_2^2.$$

Expanding and dropping constants:

$$J_0^\star(b) \equiv \|y - Xb\|_2^2 + \sum_{j=1}^{m} \Big(n\alpha^\star\nu^2\, b^\top \tilde{H}_j^\top \tilde{H}_j b - 2n\alpha^\star\nu\, \hat{\theta}_j^\top \tilde{H}_j b \Big).$$

Choose ν and $\alpha^\star$ such that the coefficients match:

$$n\alpha^\star\nu^2 = n\alpha + \lambda, \qquad n\alpha^\star\nu = n\alpha.$$

From the second equation, $\alpha^\star = \alpha/\nu$. Substituting into the first gives

$$n\alpha\,\nu = n\alpha + \lambda \quad\Longrightarrow\quad \nu = 1 + \frac{\lambda}{n\alpha}, \qquad \alpha^\star = \frac{\alpha}{\nu} = \frac{n\alpha^2}{n\alpha + \lambda}.$$

With this choice,

$$J_0^\star(b) \equiv J_\lambda(b),$$

hence

$$\arg\min_b J_0^\star(b) = \arg\min_b J_\lambda(b),$$

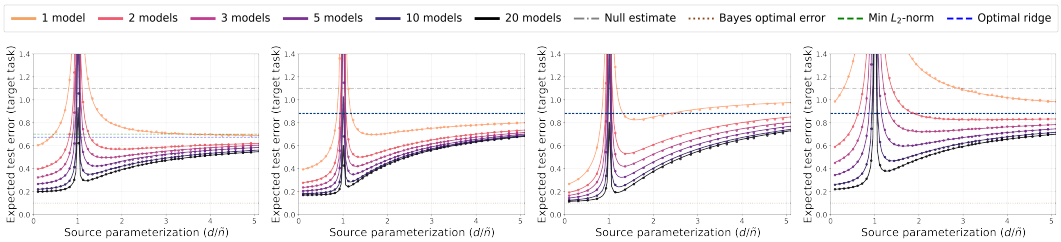
i.e., the optimization with Tikhonov using $\tilde{H}_j$ is equivalent (same minimizers) to the optimization without Tikhonov using $\tilde{H}_j^\star = \nu\,\tilde{H}_j$ and $\alpha^\star = \frac{n\alpha^2}{n\alpha+\lambda}$.

T Additional Experiments

In this section we provide additional experimental results.

T.1 Additional Experiments for the General Case

These results are in addition to Figure 1 from the main paper.

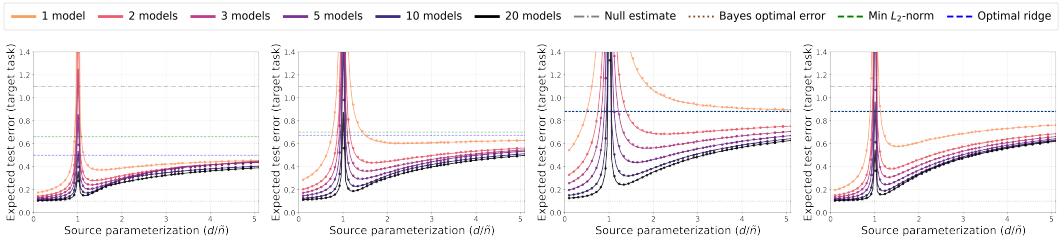


(a) $\gamma_{\text{tgt}} = 2,\ \sigma_\xi^2,\ \sigma_\eta^2 = 0.25$ (b) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.1$ (c) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.1$ (d) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.5$

Figure T.1: The results are shown for: (a) $\mathbf{H}_j$ is an energy-preserving projection onto a subspace of dimension $\frac{3d}{4}$ with $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$; (b) $\mathbf{H}_j$ is for a subspace of dimension $\frac{3d}{4}$ with $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$; (c) $\mathbf{H}_j$ is circulant with $\kappa = 1,000$ in the well-specified case, i.e. $\mathbf{H}_j = \widetilde{\mathbf{H}}_j$; and (d) $\mathbf{H}_j$ is an energy-preserving projection onto a subspace of dimension $\frac{d}{2}$ with $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$.

T.2 Additional Experiments for the Simple Case

These results are in addition to Figure 2 from the main paper.

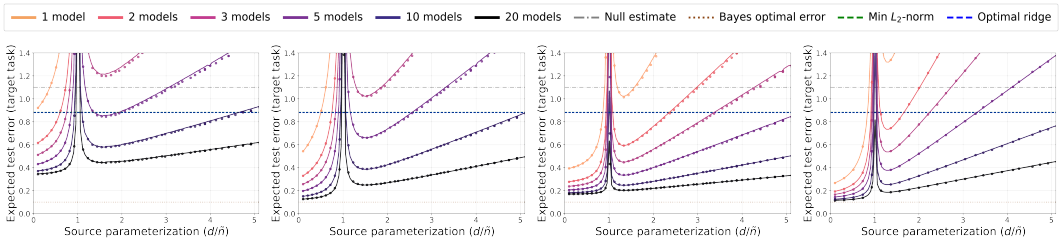


(a) $\gamma_{\text{tgt}} = \frac{4}{3}, \sigma_\xi^2, \sigma_\eta^2 = 0.1$ (b) $\gamma_{\text{tgt}} = 2,\ \sigma_\xi^2,\ \sigma_\eta^2 = 0.25$ (c) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.5$ (d) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.1$

Figure T.2: In all figures $\mathbf{H}_j = \widetilde{\mathbf{H}}_j = \mathbf{I}_d$

T.3 Additional Experiments for Debiasing

These results are in addition to Figure 3 from the main paper.

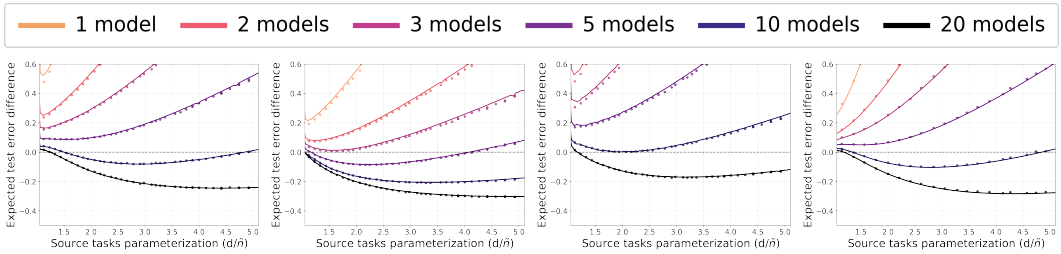


(a) $\gamma_{\text{tgt}} = 2, \sigma_\xi^2, \sigma_\eta^2 = 0.5$ (b) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.1$ (c) $\gamma_{\text{tgt}} = 2, \sigma_\xi^2, \sigma_\eta^2 = 0.1$ (d) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.5$

Figure T.3: In T.3a and T.3b the relation matrices $\mathbf{H}_j$ are projection matrices to a random subspace of dimension $\frac{3}{4}$ of the true data dimension, in T.3c and T.3d the relation matrices are energy-preserving subspace projection of dimension $\frac{d}{2}$ (for further explanation see Appendix H.1.2, H.1.2). The assumed relations are $\widetilde{\mathbf{H}}_j = \rho_j \mathbf{I}_d$ with $\mathbf{\Sigma_x} = \mathbf{I}_d$ except T.3c where the $\mathbf{\Sigma_x}$ is exponential decay covariance matrix (see Appendix H.2.1). The solid lines are the numerical calculation of Theorem 4.4 and the circle markers are the corresponding empirical evaluations.

T.4 Additional Evaluations of Test Error Difference between With and Without Debiasing

These results are in addition to Figure 5 from the main paper.



(a) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.25$ (b) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.1$ (c) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.5$ (d) $\gamma_{\text{tgt}} = 4, \sigma_\xi^2, \sigma_\eta^2 = 0.5$

Figure T.4: The results show the error difference between with and without debiasing, the task relation mentioned is for with out debiasing: T.4a $\mathbf{H}_j = \widetilde{\mathbf{H}}_j = \mathbf{I}_d$ and ; T.4b $\mathbf{H}_j$ is for a subspace of dimension $\frac{d}{2}$ with $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$; T.4c $\mathbf{H}_j$ is an energy-preserving projection onto a subspace of dimension $\frac{d}{2}$ with $\widetilde{\mathbf{H}}_j = \mathbf{I}_d$; and T.4d $\mathbf{H}_j$ is circulant with $\kappa = 1,000$ in the well-specified case, i.e. $\mathbf{H}_j = \widetilde{\mathbf{H}}_j$.

T.5 Additional Experiments for the Anisotropic Debiasing via Cross-Validation Figure 6

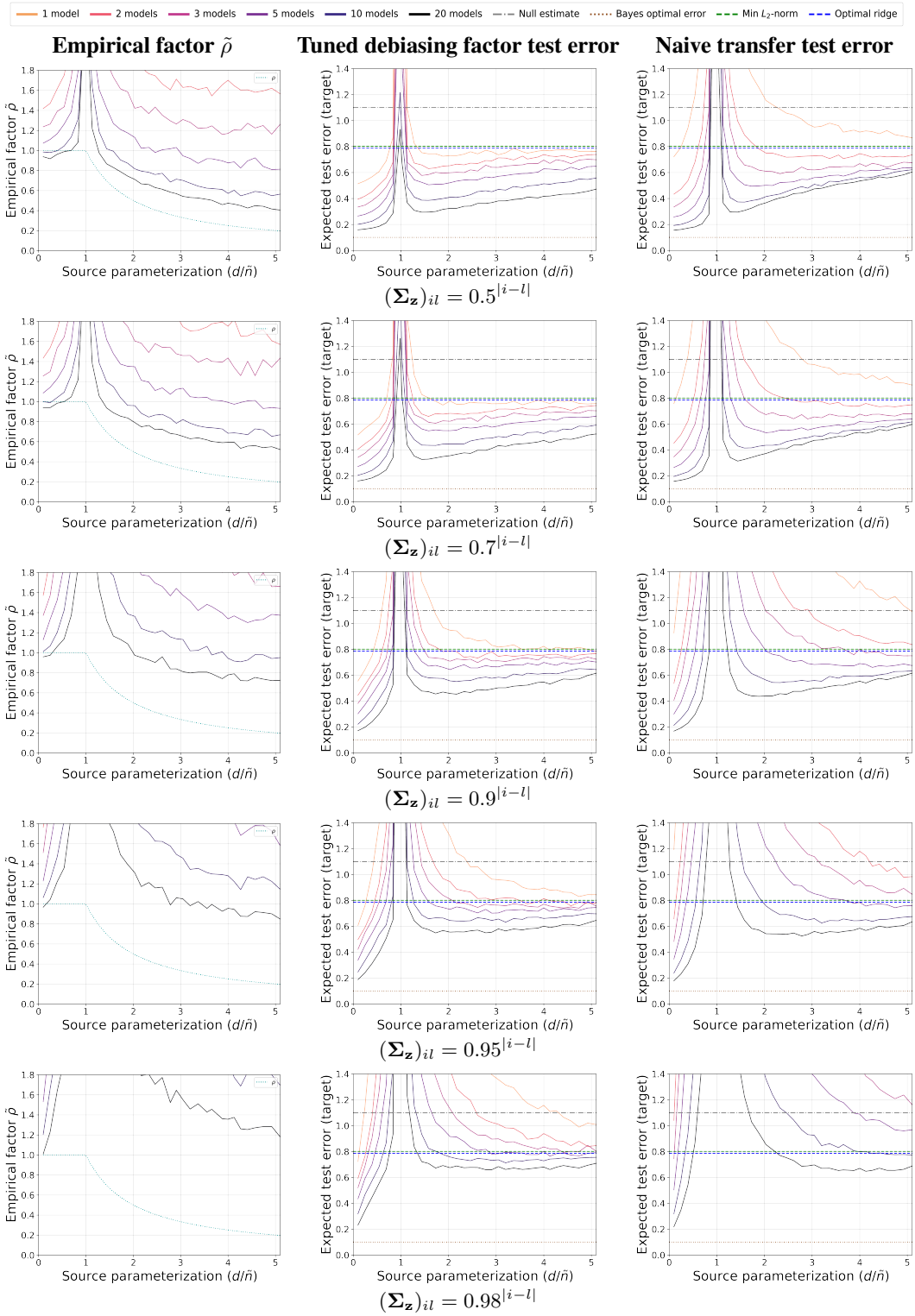
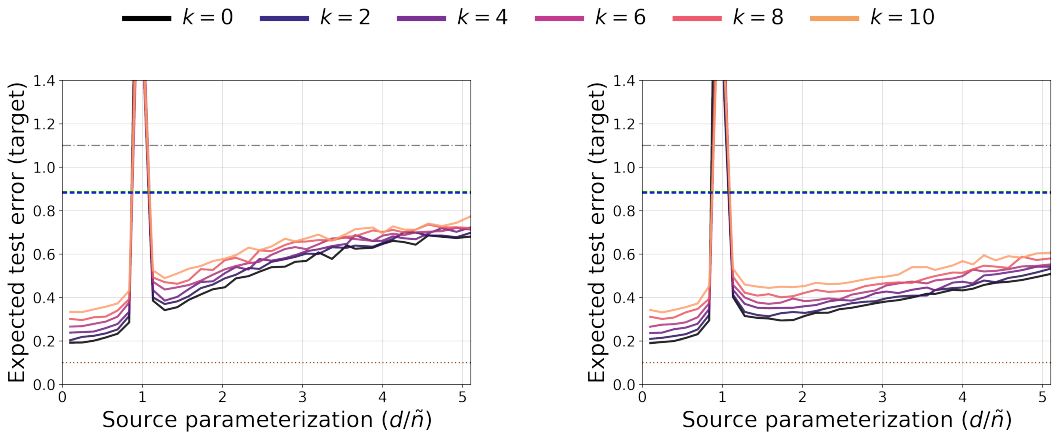


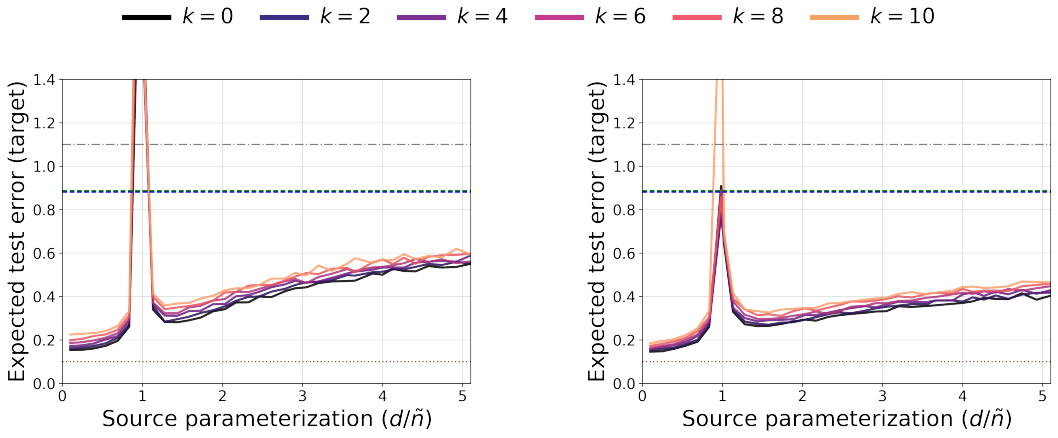
Figure T.5: Additional experiments for anisotropic debiasing via cross-validation under increasingly skewed source covariance. Each row corresponds to a different source covariance $(\mathbf{\Sigma_z})_{il} = q^{|i-l|}$, with $q \in \{0.5, 0.7, 0.9, 0.95, 0.98\}$. The three columns show, respectively, the empirically chosen factor $\tilde{\rho}$, the transfer-learning test error with debiasing, and the naive transfer-learning test error without debiasing.

T.6 Experiments for Heterogeneous Sources



(a) Without debiasing (b) With debiasing

Figure T.6: Transfer performance under heterogeneous source relevance, using the same experimental setting as Figure 3b. For $m = 20$ pretrained models, we vary $k \in \{0, 2, 4, 6, 8, 10\}$, where k sources are weakly related to the target and $m - k$ sources are strongly related. The weakly related sources have relation matrices $\mathbf{H}_j$ given by subspace projections onto $d/4$-dimensional subspaces, while the strongly related sources have relation matrices $\mathbf{H}_j$ given by subspace projections onto $3d/4$-dimensional subspaces.



(a) Without debiasing (b) With debiasing

Figure T.7: Transfer performance under heterogeneous source relevance, using the same experimental setting as Figure 3b. For $m = 20$ pretrained models, we vary $k \in \{0, 2, 4, 6, 8, 10\}$: k sources are weakly related to the target through relation matrices $\mathbf{H}_j$ that are energy-preserving projections onto $d/4$-dimensional subspaces, while the remaining $m - k$ sources are strongly related through relation matrices $\mathbf{H}_j$ that are energy-preserving projections onto $3d/4$-dimensional subspaces.

U Generality of the Theory Beyond the Specialized Case

As our assumptions differ across the theorems, the following list is provided to make this clearer:

- Theorem 4.4 allows source-specific $\mathbf{H}_j$, possibly misspecified $\widetilde{\mathbf{H}}_j$, source-specific sample sizes/noises, and anisotropic target covariance. Misspecification enters through $\mathbf{\Delta}_j = \mathbf{H}_j - \widetilde{\mathbf{H}}_j$ and $\mathbf{A}_j = \widetilde{\mathbf{H}}_j^T \left(\rho_{j,\infty}\mathbf{H}_j - \widetilde{\mathbf{H}}_j\right)$, including pairwise cross-source interactions.
- Theorems 5.2-5.4 impose well-specified orthonormal relations and homogeneous statistics to obtain interpretable formulas for negative transfer and consistency.
- Theorem 5.5 does *not* require orthonormal $\mathbf{H}_j$; for general known task operators satisfying the aggregate rank condition, it identifies the scaling that removes the source minimum-norm attenuation component of predictor bias.
- Theorems 5.6-5.7 use a narrower identity/isotropic setting to prove consistency and a finite-m condition for beneficial debiasing.

There is no universal scalar degradation law in the norm of $\mathbf{H}_j - \widetilde{\mathbf{H}}_j$, because the effect also depends on its orientation relative to $\mathbf{\Sigma}_\mathbf{x}$, the penalty geometry and other sources. Theorem 4.4 gives the corresponding matrix characterization. The projection, ill-conditioned, anisotropic-source, and heterogeneous-source experiments examine departures from the specialized assumptions, but do not replace a general consistency theorem. For unknown $\mathbf{H}_j$ and anisotropic source covariance, our current support for the debiasing benefits is empirical and validation-based for synthetic Gaussian data.

V Potential Implications to Neural Networks

The proposed debiasing correction in this paper is for linear models and cannot be applied directly, layer by layer, to general neural networks. Our analysis relies on the fixed linear source-target relation and the explicit minimum-norm attenuation. For neural networks, several obstacles prevent an immediate analogue: parameters are non-identifiable because of permutation and scaling symmetries; Euclidean distance between weights need not reflect functional similarity; learned representations are nonlinear and data-dependent; and overparameterization does not generally yield a universal scalar attenuation factor analogous to $\rho_j = \frac{\widetilde{n}_j}{d}$. Thus, neither a canonical layer-wise $\mathbf{H}_j$ nor the proposed scalar correction is currently justified for arbitrary networks.

Nevertheless, our theory provides a conceptual lesson that may extend beyond linear regression: increasing the number of pretrained models can average down source-specific variance, but need not remove a systematic bias shared by the transferred models; after correcting such a bias via debiasing, more models can offset the variance cost of the debiasing. Establishing an analogous bias and debiasing for neural networks requires additional theory.

More direct extensions may be possible in restricted settings. For networks with fixed and aligned feature representations, transfer of only the linear last layer reduces to a setting close to ours. Linearized/NTK analyses may permit a function-space or kernel-space analogue, although the debiasing would likely be spectral rather than scalar. Another direction is to formulate relations between aligned representations or model outputs instead of raw layer parameters. These are future directions rather than guarantees established by the present paper.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims made in the introduction are formally supported by the theoretical results in Theorems 5.4 and 5.6.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of this research work in Appendix A.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All assumptions are fully stated and referenced in the theorems and lemmas that require them. Complete proofs for all results are provided in the Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide all necessary information for reproduction in Appendix H.2.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code for reproducing the experimental results is publicly available at https://github.com/danielb1122/multi-pre-trained-models#transfer-learning-with-multiple-pretrained-models.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We describe the details of our experiment settings in the end of Section 4 and Appendix H.2.2 which are sufficient to understand the results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Error bars are omitted as the experiments serve as a numerical verification of our theoretical results. Following established practices in the foundational studies of Hastie et al. [12] and Dobriban and Wager [9], we report empirical results averaged over a sufficient number of independent trials.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In Appendix C we mentioned all experiments were conducted on an internal CPU cluster.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This research conforms with the entire NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the potential broader impacts of this work in Appendix B.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This is a foundational research paper that does not poses misuse risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [N/A]

    Justification: This work does not use existing assets.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not publish new assets except for the experiments code.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This research does not include human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification:

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.