Towards Sample-efficient Overparameterized Meta-learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

An overarching goal in machine learning is to build a generalizable model with few samples. To this end, overparameterization has been the subject of immense interest to explain the generalization ability of deep nets even when the size of the dataset is smaller than that of the model. While the prior literature focuses on the classical supervised setting, this paper aims to demystify overparameterization for meta-learning. Here we have a sequence of linear-regression tasks and we ask: (1) Given earlier tasks, what is the optimal linear representation of features for a new downstream task? and (2) How many samples do we need to build this representation? This work shows that surprisingly, overparameterization arises as a natural answer to these fundamental meta-learning questions. Specifically, for (1), we first show that learning the optimal representation coincides with the problem of designing a task-aware regularization to promote inductive bias. We leverage this inductive bias to explain how the downstream task actually benefits from overparameterization, in contrast to prior works on few-shot learning. For (2), we develop a theory to explain how feature covariance can implicitly help reduce the sample complexity well below the degrees of freedom and lead to small estimation error. We then integrate these findings to obtain an overall performance guarantee for our meta-learning algorithm. Numerical experiments on real and synthetic data verify our insights on overparameterized meta-learning.

1 Introduction

In a multitude of machine learning (ML) tasks with limited data, it is crucial to build accurate models in a sample-efficient way. Constructing a simple yet informative representation of features is a critical component of learning a model that generalizes well to an unseen test set. The field of meta-learning dates back to [9, 5] and addresses this challenge by transferring insights across distinct but related tasks. Usually, the meta-learner first (1) learns a feature-representation from previously seen tasks and then (2) uses this representation to succeed at an unseen task. The first phase is called representation learning and the second is called few-shot learning. Such information transfer between tasks is the backbone of modern transfer and multitask learning and finds ubiquitous applications in image classification [14], machine translation [7] and reinforcement learning [17].

Recent literature in ML theory has posited that overparameterization can be beneficial to generalization in traditional single-task setups for both regression [28, 38, 4, 32, 29] and classification [31, 30] problems. Empirical literature in deep learning suggests that overparameterization is of interest for both phases of meta-learning as well. Deep networks are stellar representation learners despite containing many more parameters than the sample size. Additionally, overparameterization is observed to be beneficial in the few-shot phase for transfer-learning in Figure 1(a). A pretrained ResNet50 network on Imagenet was utilized to obtain a representation of R features for classification

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

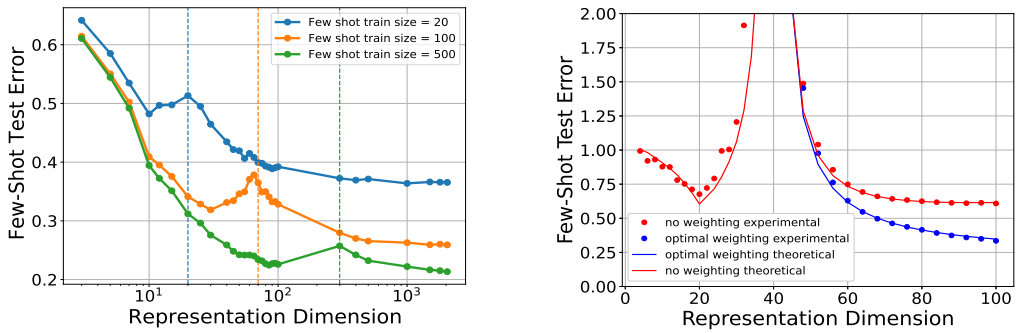

Figure 1: **Illustration of the benefit of overparameterization in the few-shot phase.** (a) Double-descent in transfer learning: dashed lines indicate the location where the number of features R exceed the number of training points; i.e., the transition from under to over-parameterization. The experimental details are contained in the supplement. (b) Illustration of the benefit of using Weighted minL2-interpolation in Definition 3. $\boldsymbol{\Sigma}_F = \boldsymbol{I}_{100}$, $\boldsymbol{\Sigma}_T = \text{diag}(\boldsymbol{I}_{20}, 0.1\boldsymbol{I}_{80})$, $n_2 = 40$. Dots are experimental results, curves are from theoretical calculation.

on CIFAR-10. The figure plots the effect of increasing R on the test error on CIFAR-10, for different choices of training size k. For each choice of k, increasing R beyond k is seen to reduce the rate of test-error. These findings are corroborated by [17] (MAML) and [36], who successfully use a transfer learning method that adapts a pre-trained model, with 112980 parameters, to downstream tasks with only 1-5 new training samples.

In Figure 1(b), we consider a sequence of linear-regression tasks and plot the few-shot error of our proposed projection and eigen-weighting based meta-learning algorithm for a fixed few-shot training size, but varying dimensionality of features. The resulting curve looks similar to Figure 1(a) and suggests that the observations regarding overparameterization for meta-learning in neural networks can, to a good extent, be captured by linear models, thus motivating their detailed study. This aligns with trends in recent literature: while deep nets are nonlinear, recent advances show that linearized problems such as kernel regression (e.g., via neural tangent kernel [20, 16, 2]) provide a good proxy to understand some of the theoretical properties of practical, highly overparameterized deep nets.

However, existing analysis of subspace-based meta-learning algorithms for both the representation learning and few-shot phases of linear models have typically focused on the classical *underparameterized regime*. These works (see Paragraphs 2-3 of Sec. 1.2) consider the case where representation learning involves projection onto a lower-dimensional subspace. On the other hand, recent works on double descent shows that an *overparameterized* interpolator beats PCA-based method. This motivates us to build upon these results to develop a theoretical understanding of overparameterized meta-learning.

1.1 Our contributions

This paper studies meta-learning when each task is a linear regression problem, similar in spirit to [35, 22]. In the representation learning phase, the learner is provided with training data from T distinct tasks, with n_1 training samples per task. The goal is to learn a *representation matrix* $\boldsymbol{\Lambda} \in \mathbb{R}^{d \times R}$ with arbitrary R. In the few-shot learning phase, given a task with n_2 samples, $\boldsymbol{\Lambda}$ is used to weight the features as $\boldsymbol{\Lambda}^\top \boldsymbol{x}$. $\boldsymbol{\Lambda}$ throws away $d - R$ features if $R < d$ ($R = d$ is allowed), and decides the importance of features obtained from representation learning. Such information is incorporated in calculating the least norm interpolator in few-shot learning (see Def. 3). We ask the following fundamental questions:

Q1: What is the optimal representation $\boldsymbol{\Lambda} \in \mathbb{R}^{d \times R}$ to minimize risk at the few-shot phase?
Q2: How to meta-learn in the overparameterized regime with $N \ll Rd$ and $n_2 \ll R$?

Towards answering these, we make several contributions to the finite-sample understanding of *linear* meta-learning, under assumptions discussed in Section 2. Our results are obtained for a general data/task model with *arbitrary task covariance* $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}$ *and feature covariance* $\boldsymbol{\Sigma}_F$ which allows for a rich set of observations.

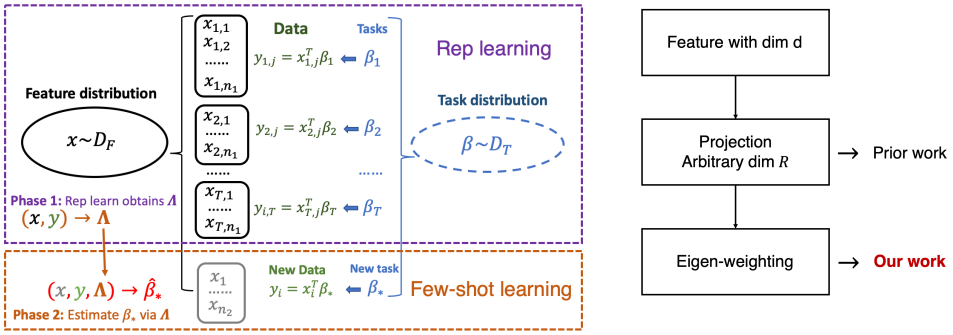

Figure 2: (a) Steps of the meta-learning algorithm. (b) Our representation-learning algorithm has two steps: projection and eigen-weighting. We focus on the use of overparameterization+weighting matrix (Def. 3), and compare this with overparameterization with simple projection (no weighting), and underparameterization. [35, 22, 21, 15] study underparameterized projection only.

Optimal representation for few-shot learning. We first explore the design of optimal representation matrix $\mathbf{\Lambda}^*$ when tasks and features are sampled randomly from known distributions with covariances $\mathbf{\Sigma}_T$ and $\mathbf{\Sigma}_F$ respectively (Algorithm 1). Compared to subspace-based representations, our scheme applies *eigen-weighting* to incentivize the optimizer to place higher weight on promising eigen-directions. Figure 1(b) depicts the importance of this additional step: in the overparameterized regime, the blue eigen-weighted learner achieves markedly lower test-error than its unweighted counterpart. This eigen-weighting procedure can introduce inductive bias during few-shot learning and helps explain how optimal representation minimizing the few-shot risk can be overparameterized with $R \gg n_2$. Our analysis also reveals that, the optimal subspace and representation is governed by a *canonical task covariance* $\tilde{\mathbf{\Sigma}}_T = \mathbf{\Sigma}_F^{1/2}\mathbf{\Sigma}_T\mathbf{\Sigma}_F^{1/2}$ which captures the feature saliency by summarizing the feature and task distributions.

Representation learning. When we have access to finite samples, we need to estimate the principal subspace of the canonical task covariance $\tilde{\mathbf{\Sigma}}_T$ which has a DoF of $\Omega(Rd)$. To answer Q2, we first present empirical evidence that feature covariance $\mathbf{\Sigma}_F$ is "positively correlated" with $\tilde{\mathbf{\Sigma}}_T$. Following this observation, we propose an efficient algorithm –based on Method-of-Moments (MoM)– to explain how feature covariance can provably provide inductive bias to dramatically reduce sample complexity of representation learning well below $\mathcal{O}(Rd)$ samples. Our sample complexity bound depends on insightful quantities such as *effective ranks* $\mathbf{\Sigma}_F, \tilde{\mathbf{\Sigma}}_T$, yet it still improves over prior art (e.g., [22, 35]) that specialized to low-rank $\tilde{\mathbf{\Sigma}}_T$ and identity $\mathbf{\Sigma}_F$ (see Table 2).

Integrated meta-learning guarantee. The eigen-weighting scheme is sophisticated and requires the exact knowledge of $\tilde{\mathbf{\Sigma}}_T$. We establish the robustness of optimal representation finite number (denoted by N) of representation learning samples and inexact $\tilde{\mathbf{\Sigma}}_T$. This leads to an overall meta-learning guarantee in terms of $\mathbf{\Lambda}^*$, N and n_2. This also uncovers a bias-variance tradeoff on when to use overparameterized representations: As N decreases, it becomes more preferable to use a smaller R (more bias, less variance) due to inaccurate estimate of the weak eigen-directions of $\tilde{\mathbf{\Sigma}}_T$. This explains why, in practice, increasing representation dimension may not help beyond a certain point (see Fig. 5 for details).

$\mathbf{\Sigma}_F$	Feature covariance
$\mathbf{\Sigma}_T$	Task covariance
$\tilde{\mathbf{\Sigma}}_T$	Canonical task covariance
n_1	Samples per each earlier task
T	Number of earlier tasks
N	Total sample size $T \times n_1$
n_2	Samples for new task
$\mathbf{\Lambda}$	Representation matrix

Table 1: Key notations

1.2 Related work

Overparameterized ML and double-descent The phenomenon of double-descent was first discovered by [6]. This paper and future works on this topic [4, 32, 31, 29, 11] emphasize the importance of the right prior (sometimes referred to as inductive bias or regularization) to avail the benefits of

overparameterization. However, an important question that arises is: where does this prior come from? Our work shows that the prior can come from the insights learned from related previously-seen tasks. Section 3 extends the ideas in [33, 38] to depict how the optimal representation described can be learned from imperfect covariance estimates as well.

Theory for representation learning Recent papers [22, 21, 35, 15] propose the theoretical bounds of representation learning when the tasks lie in an exactly r dimensional subspace. The first three discuss method of moment estimators and the last two discuss matrix factorized formulations. [35] shows that the number of samples that enable meaningful representation learning is $\mathcal{O}(dr^2)$. [22, 21, 35] assume the features follow a standard normal distribution. We define a canonical covariance which handles arbitrary feature and task covariances. We also show that our estimator succeeds with $\mathcal{O}(dr)$ samples when $n_1 \sim r$, and extend the bound to general covariances with effective rank defined.

Subspace-based meta learning With tasks being low rank, [22, 21, 35, 15] does few-shot learning in a low dimensional space. [39, 40] study meta-learning for linear bandits. [25] gives information theoretic lower and upper bounds. [8] proposes subspace-based methods for nonlinear problems such as classification. Other slightly different settings involve learning low rank matrices [12] and learning functionals [18] from a family of functions with covering number depicting its complexity. We investigate a representation with arbitrary dimension, specifically interested in overparameterized case and show it yield a smaller error with general task/feature covariances.

Mixed Linear Regression (MLR) In MLR [41, 23, 12], multiple linear regression are executed, similar to representation learning. The difference is that, the tasks are drawn from a finite set, and number of tasks can be larger than d and not necessarily low rank. [24, 10, 27] propose sample complexity bounds of representation learning for mixed linear regression. They can be combined with other structures such as binary task vectors [3] and sparse task vectors [1].

2 Problem Setup

The problem we consider consists of two phases:

1. Representation learning: Prior tasks are used to learn a suitable representation to process features.
2. Few-shot learning: A new task is learned with a few samples thanks to the suitable representation.

To aid in the theoretical analysis and understanding the generic behavior of the sample efficiency of general meta learning algorithms (see Fig. 1(a)), we study linear regression tasks, and focus on a setup where the task vectors and the features associated to the tasks are generated randomly.

2.1 Data generation

Definition 1 (Task and feature distributions) *Throughout, $\mathcal{D}_T$ and $\mathcal{D}_F$ denote the distributions of tasks $\boldsymbol{\beta}_i$ and features $\boldsymbol{x}_{ij}$ respectively. These distributions are zero-mean with corresponding covariance matrices $\boldsymbol{\Sigma}_T$ and $\boldsymbol{\Sigma}_F$. Additionally, the whitened vectors $\boldsymbol{\Sigma}_T^{-1/2}\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}_F^{-1/2}\boldsymbol{x}$ have subgaussian norms bounded by a constant.*

In [22, 21, 35, 15], the task covariance is assumed to be exactly low-rank leading to a low-dimensional subspace projection-based representation. A key aspect of this work is exploring a rich setting where both feature and task covariances are allowed to be arbitrary. This will reveal how the interplay between these matrices will influence the representation learning process.

Definition 2 (Data distribution for a single task) *Given a specific realization of task vector $\boldsymbol{\beta} \sim \mathcal{D}_T$, the corresponding label/input distribution $(y, \boldsymbol{x}) \sim \mathcal{D}_{\boldsymbol{\beta}}$ is obtained via $y = \boldsymbol{x}^\top \boldsymbol{\beta} + \varepsilon$ where $\boldsymbol{x} \sim \mathcal{D}_F$ and ε is zero-mean subgaussian noise with variance σ^2.*

Data for Representation Learning (Phase 1). We have T tasks, each with n_1 training examples. The task vectors $(\boldsymbol{\beta}_i)_{i=1}^T \subset \mathbb{R}^d$ are drawn i.i.d. from the distribution $\mathcal{D}_T$. The data for ith task is given by $(y_{ij}, \boldsymbol{x}_{ij})_{j=1}^{n_1} \overset{\text{i.i.d.}}{\sim} \mathcal{D}_{\boldsymbol{\beta}_i}$. In total, there are $N = T \times n_1$ examples.

Data for Few-Shot Learning (Phase 2). Sample task $\boldsymbol{\beta}_\star \sim \mathcal{D}_T$. Few-shot dataset has n_2 examples $(y_i, \boldsymbol{x}_i)_{j=1}^{n_2} \overset{\text{i.i.d.}}{\sim} \mathcal{D}_{\boldsymbol{\beta}_\star}$.

We investigate the training error and the sample efficiency for the few-shot learning (Phase 2) with respect to the representation learned from Phase 1. As described in the next section, our approach

uses a weighted representation of features (say R dimensional) and, thanks to intelligent weighting, the new task can potentially be learned with much fewer than R samples.

2.2 Training in Phase 2.

Denote $\boldsymbol{X} \in \mathbb{R}^{n_2 \times d}$ whose i_{th} row is $\boldsymbol{x}_i$, and $\boldsymbol{y} = [y_1, ..., y_m]^\top$. We are interested in studying the weighted 2-norm interpolator defined below for overparameterization regime $R \geq n_2$.

Definition 3 (Weighted 2-norm interpolator)

$$\hat{\boldsymbol{\beta}}_{\boldsymbol{\Lambda}} = \arg\min_{\boldsymbol{\beta}'} \|\boldsymbol{\Lambda}^\dagger \boldsymbol{\beta}'\|_2 \quad \textit{s.t. } \boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}',\ \boldsymbol{\beta} \in \text{column space of } \boldsymbol{\Lambda}.$$

We can easily see that $\hat{\boldsymbol{\beta}} = \boldsymbol{\Lambda}(\boldsymbol{X}\boldsymbol{\Lambda})^\dagger \boldsymbol{y}$. In the above, $\boldsymbol{\Lambda} \in \mathbb{R}^{d \times R}$ imposes an unequal penalty on features, and we apply the smallest norm interpolator on top of this penalty. A similar formulation was considered in [32, 31].

Further, we observe that Definition 3 is a special case of the weighted ridge regression discussed in [38], as stated in Observation 1. An alternative equivalence between min-norm interpolation and ridge regression can be found in [32].

Observation 1 *Let $\boldsymbol{X} \in \mathbb{R}^{n_2 \times d}$ and $\boldsymbol{y} \in \mathbb{R}^{n_2}$, define*

$$\hat{\boldsymbol{\beta}}_1 = \lim_{t \to 0} \text{argmin}_{\boldsymbol{\beta}} \|\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}\|^2 + t\boldsymbol{\beta}^\top (\boldsymbol{\Lambda}\boldsymbol{\Lambda}^\top)^\dagger \boldsymbol{\beta},\ \boldsymbol{\beta} \in \text{column space of } \boldsymbol{\Lambda}. \tag{2.1}$$

We have that $\hat{\boldsymbol{\beta}}_1 = \hat{\boldsymbol{\beta}}$.

3 Canonical Covariance and Optimal Representation

The natural choice of optimization for choosing $\boldsymbol{\Lambda}$ would be to choose the weighting that minimizes the eventual risk of the learned interpolator.

$$\boldsymbol{\Lambda}^* = \arg \min_{\boldsymbol{\Lambda}' \in \mathbb{R}^{d \times R}} \text{risk}(\boldsymbol{\Lambda}', \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F) \tag{3.1}$$

In the above, the risk of $\hat{\boldsymbol{\beta}}_{\boldsymbol{\Lambda}}$ is given by

$$\text{risk}(\boldsymbol{\Lambda}, \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F) = \boldsymbol{E}_{\boldsymbol{x},y,\boldsymbol{\beta}}(y - \boldsymbol{x}^\top \hat{\boldsymbol{\beta}}_{\boldsymbol{\Lambda}})^2 = \boldsymbol{E}_{\boldsymbol{\beta}}(\hat{\boldsymbol{\beta}}_{\boldsymbol{\Lambda}} - \boldsymbol{\beta})^\top \boldsymbol{\Sigma}_F (\hat{\boldsymbol{\beta}}_{\boldsymbol{\Lambda}} - \boldsymbol{\beta}) + \sigma^2. \tag{3.2}$$

This risk definition motivates **canonical covariance** to capture the joint behavior of feature and task covariances $\boldsymbol{\Sigma}_F, \boldsymbol{\Sigma}_T$. We introduce whitened features $\tilde{\boldsymbol{x}} = \boldsymbol{\Sigma}_F^{-1/2}\boldsymbol{x}$ and associated task vector $\tilde{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_F^{1/2}\boldsymbol{\beta}$. This change of variables ensures $\boldsymbol{x}^T\boldsymbol{\beta} = \tilde{\boldsymbol{x}}^T\tilde{\boldsymbol{\beta}}$ and leads to the canonical task covariance

$$\tilde{\boldsymbol{\Sigma}}_T = \boldsymbol{\Sigma}_F^{1/2}\boldsymbol{\Sigma}_T\boldsymbol{\Sigma}_F^{1/2}.$$

The risk of meta-learning depends on the canonical task covariance, i.e., does not change when $\boldsymbol{\Sigma}_F^{1/2}\boldsymbol{\Sigma}_T\boldsymbol{\Sigma}_F^{1/2}$ is fixed and we vary $\boldsymbol{\Sigma}_F$ and $\boldsymbol{\Sigma}_T$.

Observation 2 (Equivalence to problem with whitened features) *Let data be generated as in Phase 1. Denote $\tilde{\boldsymbol{\Sigma}}_T = \boldsymbol{\Sigma}_F^{1/2}\boldsymbol{\Sigma}_T\boldsymbol{\Sigma}_F^{1/2}$. Then risk$(\boldsymbol{\Sigma}_F^{1/2}\boldsymbol{\Lambda}, \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F) =$ risk$(\boldsymbol{\Lambda}, \tilde{\boldsymbol{\Sigma}}_T, \boldsymbol{I})$.*

This will dictate the representation learning phase as we need to learn the top eigenspace of $\tilde{\boldsymbol{\Sigma}}_T$. $\tilde{\boldsymbol{\Sigma}}_T$ also plays a key role in determining the optimal eigen-weighting which is the topic of this section.

3.1 Optimal representation

Prior work [22, 21, 35] typically projects the features onto the subspace for few shot learning. In **Q1** we ask, what can be said about the performance of a general linear representation with arbitrary dimension? We propose an expression for computing the representation matrix $\boldsymbol{\Lambda}$ in Algo. 1.

For a given $\boldsymbol{\Lambda}$, the following theorem characterizes the exact asymptotic risk that helps us differentiate the performance of different linear representations.

Algorithm 1 Optimal representation

Require: $R,\ d,\ n_2, \sigma$, the canonical covariance $\tilde{\boldsymbol{\Sigma}}_T$, the task covariance $\boldsymbol{\Sigma}_F$.

1: *Eigendecomposition:* Spectrally decompose $\tilde{\boldsymbol{\Sigma}}_T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}$,

2: *Subspace:* $\boldsymbol{U}_1 \in \mathbb{R}^{d\times R}$ = the first R columns of $\boldsymbol{U}$.

3: *Get eigenvalues:* $\boldsymbol{\Sigma}_F^R, \boldsymbol{\Sigma}_T^R \in \mathbb{R}^{R\times R}$ be diagonal matrices. The diagonal values are the first to R-th eigenvalue of $\boldsymbol{\Sigma}_F$ and $\boldsymbol{\Sigma}_T$.

4: *Get eigenvalues:* Let $\boldsymbol{\Sigma}_F^\perp, \boldsymbol{\Sigma}_T^\perp \in \mathbb{R}^{(d-R)\times(d-R)}$ be diagonal matrices. The diagonal values are the $R+1$ to d-th eigenvalue of $\boldsymbol{\Sigma}_F$ and $\boldsymbol{\Sigma}_T$.

5: *Equivalent noise level:* $\sigma_R^2 \leftarrow \sigma^2 + \mathbf{tr}(\boldsymbol{\Sigma}_F^\perp \boldsymbol{\Sigma}_T^\perp)$.

6: *Optimization:* Get $\boldsymbol{\theta}^*$ from (3.5).

7: *Map to eigenvalues:* Let $\boldsymbol{\Lambda}_R^* \in \mathbb{R}^{R\times R}$ be a diagonal matrix, $\boldsymbol{\Lambda}_{R_i}^* = ((1/\boldsymbol{\theta}_i^* - 1))^{-2}$.

8: *Projection:* $\boldsymbol{\Lambda}^* \leftarrow \boldsymbol{U}_1(\boldsymbol{\Sigma}_F^R)^{1/2}\boldsymbol{\Lambda}_R^*$.

Theorem 1 (Equivalent formulation of optimal representation problem (3.1)**)** *Assume that $\boldsymbol{\Sigma}_F$ and $\boldsymbol{\Sigma}_T$ are jointly diagonalizable(This is Not assumed in representation learning.). Let $\boldsymbol{\Lambda}_R$ and $\tilde{\boldsymbol{\Sigma}}_T^R$ be defined in Algorithm 1, they are $R\times R$ diagonal matrices. Fix $\kappa > 1$, suppose $\frac{R}{n_2} = \kappa$ is fixed and $n_2 \to \infty$, noise level σ is fixed. Let $\boldsymbol{\Lambda}_{R_i}, \tilde{\boldsymbol{\Sigma}}_{T_i}^R$ be the i-th diagonal entry of $\boldsymbol{\Lambda}_R, \tilde{\boldsymbol{\Sigma}}_T^R$, the joint empirical distribution $\frac{1}{p}\sum_{i=1}^p \delta_{\boldsymbol{\Lambda}_{R_i}, \tilde{\boldsymbol{\Sigma}}_{T_i}^R}$ converges to a fixed distribution. Then we define $\xi > 0$ such that $n_2 = \sum_{i=1}^R \left(1 + (\xi\boldsymbol{\Lambda}_i^2)^{-1}\right)^{-1}$. Define $\boldsymbol{\theta} \in \mathbb{R}^R$ as $\boldsymbol{\theta}_i = \frac{\xi\boldsymbol{\Lambda}_i^2}{1+\xi\boldsymbol{\Lambda}_i^2}$ and*

$$f(\boldsymbol{\theta}, \tilde{\boldsymbol{\Sigma}}_T^R, n_2) = \frac{1}{R(n_2 - \|\boldsymbol{\theta}\|^2)}\left(n_2 \sum_{i=1}^R (1-\boldsymbol{\theta}_i)^2 \tilde{\boldsymbol{\Sigma}}_{T_i}^R + \|\boldsymbol{\theta}\|^2\sigma_R^2\right). \tag{3.3}$$

we have

$$\lim_{n_2\to\infty} f(\boldsymbol{\theta}, \tilde{\boldsymbol{\Sigma}}_T^R, n_2) = \lim_{n_2\to\infty} \frac{\textit{risk}(\boldsymbol{\Sigma}_F^{1/2}\boldsymbol{\Lambda}, \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F)}{R} \tag{3.4}$$

The proof applies the convex Gaussian Min-max Theorem (CGMT) in [34] and can be found in Appendix[1]. We use CGMT to show that the distribution of the estimator $\hat{\beta}$ converges to a Gaussian distribution as the dimension tends to infinity, so we can treat the risk as a function of Gaussian random vector and compute its expectation.

Theorem 1 provides us with a closed-form risk for any linear representation. Thus, one can solve for the optimal representation by minimizing this risk using $\boldsymbol{\theta}$. We propose an algorithm for the optimization problem in the appendix, which is derived from its KKT condition[2].

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}}\ f(\boldsymbol{\theta}, \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F),\ \text{s.t. } 0 \le \boldsymbol{\theta} < 1, \sum_{i=1}^R \boldsymbol{\theta}_i = n_2 \tag{3.5}$$

We can then find the optimal representation by[3] $\boldsymbol{\Lambda}_{R_i} = ((1/\boldsymbol{\theta}_i^* - 1)\xi)^{-2}$.

How overparameterization arises and inductive bias of representation learning: Thm. 1 leads to an optimal eigen-weigting strategy via asymptotic analysis. As shown in Figure 1(b), eigen-weighting becomes critical when the few-shot problem is overparameterized (for underparameterized problem, weighting has no impact). Low dimensional representation zeros out features and cause estimation bias. By contrast, when $\tilde{\boldsymbol{\Sigma}}_T \in \mathbb{R}^{d\times d}$ is not low rank, every feature contributes to learning. The eigen-weighting provides up/down-voting of features, allowing better bias-variance tradeoff than missing features. This viewpoint is in similar spirit to that of [19] as well as others, where the authors devise a misspecified linear regression to demonstrate the benefits of overparameterization. Our algorithm allows both PCA and eigen-weighting. In the next section, we will acquire prior information by learning $\tilde{\boldsymbol{\Sigma}}_T$. Thm. 1 and (3.5) infuses this information to design the best eigen-weighting.

[1] CGMT typically requires Gaussian distribution, but the optimal distribution is same as [38] which does not require Gaussian.

[2] For implementation of (3.5) we let $0 < \underline{\theta} < n_2/d$ and require $\underline{\theta} \le \boldsymbol{\theta} \le 1 - \frac{d-n_2}{n_2}\underline{\theta}$ instead of $0 \le \boldsymbol{\theta} \le 1$ for robustness concerns. This will be discussed in Sec. 5.

[3] In the algorithm, $\xi = 1$ and $\boldsymbol{\Lambda}_{R_i} = (1/\boldsymbol{\theta}_i^* - 1)^{-2}$, because $c\boldsymbol{\Lambda}^*$ for any constant c gives the same $\hat{\boldsymbol{\beta}}$.

Risk with respect to spike level. We note that, an R dimensional representation can be recovered by a d dimensional rank R representation matrix, thus the underparameterized case can never beat d dimensional case in theory. In Fig. 3, we take $n_2 = 40$, $d = 100$, $\boldsymbol{\Sigma}_F = \boldsymbol{I}_{100}$, $\boldsymbol{\Sigma}_T = \text{diag}(\boldsymbol{I}_{20}, \iota \boldsymbol{I}_{80})$ with different ι. When $\iota = 0.1$, $\tilde{\boldsymbol{\Sigma}}_T$ has spiked spectrum and optimal overparameterized weighting is far better. When $\iota = 0.05$, $\tilde{\boldsymbol{\Sigma}}_T$ is approximately low rank, the truncation error caused by small R is not obvious, so the risk with optimal eigen-weighting has small gap with underparameterized case (while overparameterized projection-only representation has large risk). When $\iota = 0.5$, $\tilde{\boldsymbol{\Sigma}}_T$ is not spiked and no representation does good job.

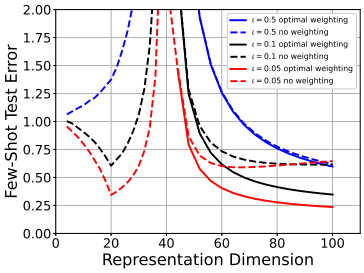

Figure 3: Theoretical risk of optimal representation. $\boldsymbol{\Sigma}_F = \boldsymbol{I}_{100}$, $\boldsymbol{\Sigma}_T = \text{diag}(\boldsymbol{I}_{20}, \iota \boldsymbol{I}_{80})$, $n_2 = 40$.

4 Representation Learning

In the previous section, we derived the optimal choice of $\boldsymbol{\Lambda}$ if we were provided with $\boldsymbol{\Sigma}_T$ and $\boldsymbol{\Sigma}_F$. In practice, the learner does not have access to these covariance matrices. Recall from Observation 2 that the risk depends on the canonical covariance $\tilde{\boldsymbol{\Sigma}}_T = \boldsymbol{\Sigma}_F^{1/2} \boldsymbol{\Sigma}_T \boldsymbol{\Sigma}_F^{1/2}$, so we only[4] need to estimate the R-dimensional principal subspace of $\tilde{\boldsymbol{\Sigma}}_T$. Let us denote this subspace by $\boldsymbol{S}$.

Sample complexity of subspace learning: Recall that the representation has Rd degrees of freedom. When $\tilde{\boldsymbol{\Sigma}}_T$ is exactly rank R and features are whitened, [35] provides a lower bound of $\Omega(Rd)$ examples and gives an algorithm achieving $\mathcal{O}(R^2 d)$ samples. However, in practice, deep nets learn good representations despite overparameterization. In this section, we argue how inductive bias of the feature distribution can implicitly accelerate learning the canonical covariance. Inductive bias is found in practical instances. As an example, we trained $45 = \binom{10}{2}$ SVM classifiers on MNIST each a binary classifier differentiating a pair of digits. The features are images and the task vectors are the support vectors. Here, we assessed correlation coefficient between two covariances via canonical-feature alignment defined as $\frac{\text{trace}(\boldsymbol{\Sigma}_F{}^{1/2} \tilde{\boldsymbol{\Sigma}}_T \boldsymbol{\Sigma}_F{}^{1/2})}{\text{trace}(\boldsymbol{\Sigma}_F)\text{trace}(\tilde{\boldsymbol{\Sigma}}_T)}$ which generalizes the correlation coefficient for vectors. As reference, we compared it to canonical-identity alignment defined as $\frac{\text{trace}(\tilde{\boldsymbol{\Sigma}}_T)}{d \cdot \text{trace}(\tilde{\boldsymbol{\Sigma}}_T)}$ (replacing $\boldsymbol{\Sigma}_F$ with $\boldsymbol{I}$). The canonical-feature alignment score is 0.32 whereas the canonical-identity alignment score is 0.15. This significant score difference exemplifies how $\boldsymbol{\Sigma}_F$ and $\tilde{\boldsymbol{\Sigma}}_T$ can synergistically align with each other (inductive bias). This alignment helps our MoM estimator defined below, illustrated by Example 1 (spiked covariance).

To develop a theory for representation learning, we propose the method-of-moment estimator towards this end. Our work differs from prior works such as [22, 21, 35] in two aspects:

1. Prior works learn a *low dimensional* subspace, whereas we estimate $\tilde{\boldsymbol{\Sigma}}_T$ for overparameterized representation.
2. Prior works assume that the feature covariance $\boldsymbol{\Sigma}_F = \boldsymbol{I}$. Let r be task rank, the sample complexity is dr^2. Our learning guarantee holds for arbitrary covariance matrices $\boldsymbol{\Sigma}_F, \boldsymbol{\Sigma}_T$. The sample complexity depends on *effective rank* and can be smaller than DoF. We use spiked covariance as an example below.

Definition 4 (Method of Moment (MoM) Estimator) *Define* $\hat{\boldsymbol{b}}_{i,1} = 2n_1^{-1} \sum_{j=1}^{n_1/2} y_{ij} \boldsymbol{x}_{ij}$, $\hat{\boldsymbol{b}}_{i,2} = 2n_1^{-1} \sum_{j=n_1/2+1}^{n_1} y_{ij} \boldsymbol{x}_{ij}$. *Define*

$$\hat{\boldsymbol{M}} = n_1^{-1} \sum_{i=1}^{T} (\boldsymbol{b}_{i,1} \boldsymbol{b}_{i,2}^\top + \boldsymbol{b}_{i,2} \boldsymbol{b}_{i,1}^\top),$$

The expectation of $\hat{\boldsymbol{M}}$ *is* $\boldsymbol{M} = \boldsymbol{\Sigma}_F \boldsymbol{\Sigma}_T \boldsymbol{\Sigma}_F$.

[4] We also need to estimate $\boldsymbol{\Sigma}_F$ for whitening. Estimating $\boldsymbol{\Sigma}_F$ is easy and has smaller error compared with $\tilde{\boldsymbol{\Sigma}}_T$. The analysis is in first part of Appendix B.

feature cov	$\boldsymbol{\Sigma}_F = \boldsymbol{I}, \boldsymbol{\Sigma}_T = \text{diag}(\boldsymbol{I}_{s_T}, \boldsymbol{0})$			$\boldsymbol{\Sigma}_F = \text{diag}(\boldsymbol{I}_{s_F}, \iota_F \boldsymbol{I}_{d-s_F})$, $\boldsymbol{\Sigma}_T = \text{diag}(\boldsymbol{I}_{s_T}, \iota_T \boldsymbol{I}_{d-s_T})$		
estimator	sample N	sample n_1	error	sample N	sample n_1	error
MoM	ds_T^2	1	$(ds_T^2/N)^{1/2}$	$r_F r_T^2$	1	$(r_F r_T^2/N)^{1/2}$
MoM	ds_T	s_T	$(s_T/n_1)^{1/2}$	$r_F r_T$	r_T	$(r_T/n_1)^{1/2}$

Table 2: Sample complexity and error of MoM estimators. s_F, s_T stand for feature, task approximate rank. $r_F = s_F + \iota_F(d - s_F)$, $r_T = s_T + \iota_T(d - s_T)$. Additionally, for the well-studied setting of identity feature and low-rank task covariance (left side), our bound in the second row is the first result to achieve optimal sample complexity of $\mathcal{O}(ds_T)$ (cf. [35, 22]).

In our definition, we assume $n_1 \geq 2$, that there are at least 2 samples per task. In [35], $n_1 = 1$ is allowed. We give the analysis of their estimator in appendix. The error is similar to $\hat{\boldsymbol{M}}$.

In the following subsections, let $N = n_1 T$ refer to the total tasks in representation-learning phase. Let $r_F = \mathbf{tr}(\boldsymbol{\Sigma}_F)$, $r_T = \mathbf{tr}(\boldsymbol{\Sigma}_T)$, and $r = \mathbf{tr}(\tilde{\boldsymbol{\Sigma}}_T)$. We have two results for this estimator.

1. Generally, we can estimate $\boldsymbol{M}$ with $\mathcal{O}(r_F r^2)$ samples.
2. With a definition of effective feature rank s_F, let $n_1 \geq s_T$, we can estimate $\boldsymbol{M}$ with $\mathcal{O}(s_F s_T)$ samples.

[35] has sample complexity $\mathcal{O}(dr^2)$ (r is exact rank). If we replace d, r by our effective ranks r_F, r, we map to our result $\mathcal{O}(r_F r^2)$. So r_F, r can be seen as effective ranks and our bounds are always smaller than d, r of the setting in [35]. We will discuss later in Example 1. Our second result says when $n_1 \geq s_T$, our sample complexity achieves the $\mathcal{O}(dr)$ lower bound in [35].

Theorem 2 *Let data be generated as in Phase 1. Assume $\|\boldsymbol{\Sigma}_F\|, \|\boldsymbol{\Sigma}_T\| = 1$ for normalization*[5]

1. Let n_1 be a even number. Then there is a constant $c > 10$, with probability $1 - N^{-c}$,

$$\|\hat{\boldsymbol{M}} - \boldsymbol{M}\| \lesssim (r + \sigma^2)\sqrt{\frac{r_F}{N}} + \sqrt{\frac{r_T}{T}}.$$

2. Let $T \gtrsim \min\ s_F : s_F \geq d\lambda_{s_F+1}(\boldsymbol{\Sigma}_F)$. If $n_1 \gtrsim (r + \sigma^2)\mathcal{S}$, then with probability $1 - \mathcal{O}(T^{-C_1} + (Tr)^{-C_2} + \exp(-C_3 T^2))$, where C_i are positive constants,

$$\|\hat{\boldsymbol{M}} - \boldsymbol{M}\| \lesssim \left((r + \sigma^2)/n_1\right)^{1/2}.$$

Suppose the top R principal subspace of $\boldsymbol{M}, \hat{\boldsymbol{M}}$ are $\boldsymbol{W}, \hat{\boldsymbol{W}}$, and $\lambda_R(\boldsymbol{M}) - \lambda_{R+1}(\boldsymbol{M}) > \|\hat{\boldsymbol{M}} - \boldsymbol{M}\|$, then a direct use of [13] says the angle between them satisfy

$$\sin(\angle \boldsymbol{W}, \hat{\boldsymbol{W}}) \lesssim (\lambda_R(\boldsymbol{M}) - \lambda_{R+1}(\boldsymbol{M}) - \|\hat{\boldsymbol{M}} - \boldsymbol{M}\|)^{-1}\|\hat{\boldsymbol{M}} - \boldsymbol{M}\|.$$

Estimating eigenspace of canonical covariance. Note that if $\boldsymbol{\Sigma}_F$ and $\boldsymbol{\Sigma}_T$ are aligned, (e.g. Example 1 below), then $\boldsymbol{W} = \boldsymbol{S}$ is exactly the principal subspace of $\tilde{\boldsymbol{\Sigma}}_T$. Thm. 2 indeed gives estimation error for the principal subspace of $\tilde{\boldsymbol{\Sigma}}_T$. Note that, such alignment is strictly weaker and more general requirement compared to related works which require whitened features [35, 22].

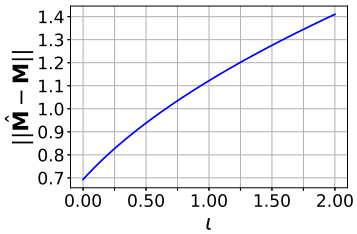

Figure 4: Error of MoM estimator

We use spiked $\tilde{\boldsymbol{\Sigma}}_T$ for the example below. This characterization was inspired by recent interest in spiked covariance models [37, 26] and the double-descent literature.

Example 1 (Spiked $\tilde{\boldsymbol{\Sigma}}_T$, Aligned principal subspace)
Consider the specific case of when the spectra of $\boldsymbol{\Sigma}_F$ and $\tilde{\boldsymbol{\Sigma}}_T$ follow the bilevel decay $\boldsymbol{\Sigma}_F = diag(\boldsymbol{I}_{s_F}, \iota_F \boldsymbol{I}_{d-s_F})$, $\boldsymbol{\Sigma}_T = diag(\boldsymbol{I}_{s_T}, \iota_T \boldsymbol{I}_{d-s_T})$. When $s_F > s_T$, the error in recovery of $\boldsymbol{M}$ is bounded as [6]

$$\|\hat{\boldsymbol{M}} - \boldsymbol{M}\| \lesssim \sqrt{\frac{r_T^2 r_F}{N}} + \sqrt{\frac{r_T}{T}}.$$

[5] This is simply equivalent to scaling y_{ij}, which does not affect the normalized error $\|\hat{\boldsymbol{M}} - \boldsymbol{M}\|/\|\boldsymbol{M}\|$. In the appendix we define $\mathcal{S} = \max\{\|\boldsymbol{\Sigma}_F\|, \|\boldsymbol{\Sigma}_T\|\}$ and prove the theorem for general $\mathcal{S}$.

[6] The second term can also be written as $\sqrt{r_T n_1/N}$. Typically $s_T \gtrsim n_1$ so the first term is larger.

In Fig. 4 we plot the error of estimating $\boldsymbol{M}$ (whose principal subspace coincides with $\tilde{\boldsymbol{\Sigma}}_T$). $\boldsymbol{\Sigma}_F = \text{diag}(\boldsymbol{I}_{30}, \iota \boldsymbol{I}_{70})$, $\boldsymbol{\Sigma}_T = \text{diag}(\boldsymbol{I}_{30}, \boldsymbol{0}_{70})$. $T = N = 100$. We can see that the error increase with ι.

Implicit bias in representation learning: Note that, in population limit, MoM estimator returns

$$\boldsymbol{M} = \boldsymbol{\Sigma}_F \boldsymbol{\Sigma}_T \boldsymbol{\Sigma}_F = \boldsymbol{\Sigma}_F^{1/2} \tilde{\boldsymbol{\Sigma}}_T \boldsymbol{\Sigma}_F^{1/2}.$$

Here, canonical covariance is the attribute of interest and feature covariance $\boldsymbol{\Sigma}_F^{1/2}$ terms implicitly modulates the estimation procedure. If our goal is estimating the principle canonical subspace, the degree of alignment between $\boldsymbol{\Sigma}_F$ and $\tilde{\boldsymbol{\Sigma}}_T$ can make or break the estimation procedure.

5 Robustness of Optimal Representation and Overall Meta-Learning Bound

In Sec. 3, we described the algorithm for computing the optimal representation with *known* distributions of features and tasks. In Sec. 4, we proposed the MoM estimator in representation learning phase. Armed with these provably reliable estimators, we can replace $\tilde{\boldsymbol{\Sigma}}_T$ and $\boldsymbol{\Sigma}_F$ in Algorithm 1 with our estimator. In this section, we inquire: how does the estimation error in covariance-estimation affect the downstream risk? That says, we are interested in[7] $\text{risk}(\boldsymbol{\Lambda}, \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F) - \text{risk}(\boldsymbol{\Lambda}^*, \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F)$.

Below, to specify the dependence with R, we use $\boldsymbol{\Lambda}_{\underline{\theta}}(R)$, $\boldsymbol{\Lambda}^*_{\underline{\theta}}(R)$ to denote R dimensional optimal representation when its smallest singular value is lower bounded by $\underline{\theta}$, obtained from $\hat{\boldsymbol{M}}$ and true $\tilde{\boldsymbol{\Sigma}}_T$We will propose the bound of the whole meta-learning algorithm in Thm. 3. It shows the robustness of few-shot learning algorithm caused by the error in representation learning.

Theorem 3 *Suppose the data is generated as Phase 2, $\boldsymbol{\Lambda}$ and $\underline{\theta}$ are defined in Def. 1 and the estimated task is obtained as* (3)*. The risk of meta-learning algorithm satisfies*[8]

$$\textit{risk}(\boldsymbol{\Lambda}_{\underline{\theta}}(R), \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F) - \textit{risk}(\boldsymbol{\Lambda}^*_{\underline{\theta}}(R), \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F) \lesssim \frac{n_2^2}{(R-n_2)(2n_2 - R\underline{\theta})\underline{\theta}} \left[(r+\sigma^2)\sqrt{\frac{r_F}{N}} + \sqrt{\frac{r_T}{T}} \right]$$

Risk with respect to PCA level R. In Fig. 5, we plot the error of the whole meta-learning algorithm. We simulate representation learning and get $\hat{\boldsymbol{M}}$, use it to compute $\boldsymbol{\Lambda}$ and plot the theoretical downstream risk (experiments match, see Fig. 1). Mainly, we compare the behavior of Thm. 3 with different R. When R grows, we search the $\boldsymbol{\Lambda}$ in a larger space, thus risk decreases with R. At the same time, representation learning error increases with R. In essence, this result provides a theoretical justification on a sweet-spot for the optimal representation. $d = R$ is optimal when $N = \infty$. As N decreases, there is a tradeoff between learning error and truncating small eigenvalues. Thus a large dimensional representation –that uses this noisy covariance– may result in noisier features. Thus choosing R adaptively with N can strike the right bias-variance tradeoff between the excess risk (variance) and the risk due to suboptimal representation.

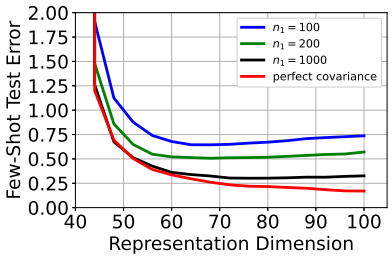

Figure 5: End to end learning guarantees. $d = 100, n_2 = 40, T = 200$, $\boldsymbol{\Sigma}_T = (\boldsymbol{I}_{20}, 0.05 \cdot \boldsymbol{I}_{80})$, $\boldsymbol{\Sigma}_F = \boldsymbol{I}_{100}$.

6 Conclusion

In this paper, we study the sample efficiency of meta-learning with linear representations. We show that the optimal representation is typically overparameterized and outperforms subspace-based representations for general data distributions. We refine the sample complexity analysis and provide an end-to-end bound for the meta-learning algorithm.

Limitations. Our optimal representation works with jointly diagonalizable covariances and is asymptotic (although this is also the case in literature such as [11, 38]). The setting is limited to mixed linear regression; nonlinear meta-learning can be the next step. Linearized settings (such as neural tangent kernel [20, 16, 2]) are helpful for understanding nonlinear models.
We do not see potential negative societal impacts of this work.

[7] Note that Sec.6 of [38] gives the exact value of $\text{risk}(\boldsymbol{\Lambda}^*, \boldsymbol{\Sigma}_T, \boldsymbol{\Sigma}_F)$ so we have an end to end error guarantee.
[8] The bracketed expression applies first conclusion of Thm. 3. One can plug in the second as well.

References

[1] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Convex multi-task feature learning. *Machine learning*, 73(3):243–272, 2008.

[2] Sanjeev Arora, Simon Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *International Conference on Machine Learning*, pages 322–332, 2019.

[3] Maria-Florina Balcan, Avrim Blum, and Santosh Vempala. Efficient representations for lifelong learning and autoencoding. In *Conference on Learning Theory*, pages 191–210, 2015.

[4] Peter L Bartlett, Philip M Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 2020.

[5] Jonathan Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198, 2000.

[6] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

[7] Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the ninth workshop on statistical machine translation*, pages 12–58, 2014.

[8] Quentin Bouniot, Ievgen Redko, Romaric Audigier, Angélique Loesch, Yevhenii Zotkin, and Amaury Habrard. Towards better understanding meta-learning methods through multi-task representation learning theory. *arXiv preprint arXiv:2010.01992*, 2020.

[9] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[10] Giovanni Cavallanti, Nicolo Cesa-Bianchi, and Claudio Gentile. Linear algorithms for online multitask classification. *The Journal of Machine Learning Research*, 11:2901–2934, 2010.

[11] Xiangyu Chang, Yingcong Li, Samet Oymak, and Christos Thrampoulidis. Provable benefits of overparameterization in model compression: From double descent to pruning neural networks. *arXiv preprint arXiv:2012.08749*, 2020.

[12] Sitan Chen, Jerry Li, and Zhao Song. Learning mixtures of linear regressions in subexponential time via fourier moments. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 587–600, 2020.

[13] Chandler Davis and William Morton Kahan. The rotation of eigenvectors by a perturbation. iii. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[15] Simon S Du, Wei Hu, Sham M Kakade, Jason D Lee, and Qi Lei. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*, 2020.

[16] Simon S Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. In *International Conference on Learning Representations*, 2018.

[17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135, 2017.

[18] Siddhant Garg and Yingyu Liang. Functional regularization for representation learning: A unified theoretical perspective. *arXiv preprint arXiv:2008.02447*, 2020.

[19] Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *arXiv preprint arXiv:1903.08560*, 2019.

[20] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pages 8571–8580, 2018.

[21] Weihao Kong, Raghav Somani, Sham Kakade, and Sewoong Oh. Robust meta-learning for mixed linear regression with small batches. *arXiv preprint arXiv:2006.09702*, 2020.

[22] Weihao Kong, Raghav Somani, Zhao Song, Sham Kakade, and Sewoong Oh. Meta-learning for mixed linear regression. *arXiv preprint arXiv:2002.08936*, 2020.

[23] Yuanzhi Li and Yingyu Liang. Learning mixtures of linear regressions with nearly optimal complexity. In *Conference On Learning Theory*, pages 1125–1144, 2018.

[24] Karim Lounici, Massimiliano Pontil, Sara Van De Geer, Alexandre B Tsybakov, et al. Oracle inequalities and optimal inference under group sparsity. *The annals of statistics*, 39(4):2164–2204, 2011.

[25] James Lucas, Mengye Ren, Irene Kameni, Toniann Pitassi, and Richard Zemel. Theoretical bounds on estimation error for meta-learning. *arXiv preprint arXiv:2010.07140*, 2020.

[26] Yasaman Mahdaviyeh and Zacharie Naulet. Risk of the least squares minimum norm estimator under the spike covariance model, 2020.

[27] Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. The benefit of multitask representation learning. *The Journal of Machine Learning Research*, 17(1):2853–2884, 2016.

[28] Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and double descent curve. *arXiv preprint arXiv:1908.05355*, 2019.

[29] Andrea Montanari, Feng Ruan, Youngtak Sohn, and Jun Yan. The generalization error of max-margin linear classifiers: High-dimensional asymptotics in the overparametrized regime. *arXiv preprint arXiv:1911.01544*, 2019.

[30] Andrea Montanari, Feng Ruan, Youngtak Sohn, and Jun Yan. The generalization error of max-margin linear classifiers: High-dimensional asymptotics in the overparametrized regime, 2020.

[31] Vidya Muthukumar, Adhyyan Narang, Vignesh Subramanian, Mikhail Belkin, Daniel Hsu, and Anant Sahai. Classification vs regression in overparameterized regimes: Does the loss function matter? *arXiv preprint arXiv:2005.08054*, 2020.

[32] Vidya Muthukumar, Kailas Vodrahalli, and Anant Sahai. Harmless interpolation of noisy data in regression. *CoRR*, abs/1903.09139, 2019.

[33] Preetum Nakkiran, Prayaag Venkat, Sham Kakade, and Tengyu Ma. Optimal regularization can mitigate double descent. *arXiv preprint arXiv:2003.01897*, 2020.

[34] Christos Thrampoulidis, Ehsan Abbasi, and Babak Hassibi. Lasso with non-linear measurements is equivalent to one with linear measurements. *Advances in Neural Information Processing Systems*, 28:3420–3428, 2015.

[35] Nilesh Tripuraneni, Chi Jin, and Michael I Jordan. Provable meta-learning of linear representations. *arXiv preprint arXiv:2002.11684*, 2020.

[36] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[37] Weichen Wang and Jianqing Fan. Asymptotics of empirical eigenstructure for high dimensional spiked covariance. *The Annals of Statistics*, 45(3):1342 – 1374, 2017.

[38] Denny Wu and Ji Xu. On the optimal weighted ℓ_2 regularization in overparameterized linear regression, 2020.

[39] Jiaqi Yang, Wei Hu, Jason D Lee, and Simon S Du. Provable benefits of representation learning in linear bandits. *arXiv preprint arXiv:2010.06531*, 2020.

[40] Jiaqi Yang, Wei Hu, Jason D Lee, and Simon S Du. Impact of representation learning in linear bandits. In *International Conference on Learning Representations*, 2021.

[41] Kai Zhong, Prateek Jain, and Inderjit S Dhillon. Mixed linear regression with multiple components. In *Advances in neural information processing systems*, pages 2190–2198, 2016.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] In supplementary file
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]