How Important is the Train-Validation Split in Meta-Learning?

Yu Bai[1]   **Minshuo Chen**[2]   **Pan Zhou**[1]   **Tuo Zhao**[2]   **Jason D. Lee**[3]   **Sham Kakade**[4]
Huan Wang[1]   **Caiming Xiong**[1]

Abstract

Meta-learning aims to perform fast adaptation on a new task through learning a "prior" from multiple existing tasks. A common practice in meta-learning is to perform a train-validation split (*train-val method*) where the prior adapts to the task on one split of the data, and the resulting predictor is evaluated on another split. Despite its prevalence, the importance of the train-validation split is not well understood either in theory or in practice, particularly in comparison to the more direct *train-train method*, which uses all the per-task data for both training and evaluation.

We provide a detailed theoretical study on whether and when the train-validation split is helpful in the linear centroid meta-learning problem. In the agnostic case, we show that the expected loss of the train-val method is minimized at the optimal prior for meta testing, and this is not the case for the train-train method in general without structural assumptions on the data. In contrast, in the realizable case where the data are generated from linear models, we show that both the train-val and train-train losses are minimized at the optimal prior in expectation. Further, perhaps surprisingly, our main result shows that the train-train method achieves a *strictly better* excess loss in this realizable case, even when the regularization parameter and split ratio are optimally tuned for both methods. Our results highlight that sample splitting may not always be preferable, especially when the data is realizable by the model. We validate our theories by experimentally showing that the train-train method can indeed outperform the train-val method, on both simulations and real meta-learning tasks.

[1]Salesforce Research [2]Georgia Tech [3]Princeton University [4]University of Washington. Correspondence to: Yu Bai <yu.bai@salesforce.com>.

1. Introduction

Meta-learning, also known as "learning to learn", has recently emerged as a powerful paradigm for learning to adapt to unseen tasks (Schmidhuber, 1987). The high-level methodology in meta-learning is akin to how human beings learn new skills, which is typically done by relating to certain prior experience that makes the learning process easier. More concretely, meta-learning does not train one model for each individual task, but rather learns a "prior" model from multiple existing tasks so that it is able to quickly adapt to unseen new tasks. Meta-learning has been successfully applied to many real problems, including few-shot image classification (Finn et al., 2017; Snell et al., 2017), hyper-parameter optimization (Franceschi et al., 2018), low-resource machine translation (Gu et al., 2018) and short event sequence modeling (Xie et al., 2019).

A common practice in meta-learning algorithms is to perform a *sample splitting*, where the data within each task is divided into a *training split* which the prior uses to adapt to a task-specific predictor, and a *validation split* on which we evaluate the performance of the task-specific predictor (Vinyals et al., 2016; Nichol et al., 2018; Rajeswaran et al., 2019; Fallah et al., 2020; Wang et al., 2020a). For example, in a 5-way k-shot image classification task, standard meta-learning algorithms such as MAML (Finn et al., 2017) use $5k$ examples within each task as training data, and use additional examples (e.g. k images, one for each class) as validation data. This sample splitting is believed to be crucial as it matches the evaluation criterion at meta-test time, where we perform adaptation on training data from a new task but evaluate its performance on unseen data from the same task.

Despite the aforementioned importance, performing the train-validation split has a potential drawback from the data efficiency perspective — Because of the split, neither the training nor the evaluation stage is able to use all the available per-task data. In the few-shot image classification example, each task has a total of $6k$ examples available, but the train-validation split forces us to use these data separately in the two stages. Meanwhile, performing the train-validation split is also not the only option in practice: there exist algorithms such as Reptile (Nichol & Schulman, 2018) and

Meta-MinibatchProx (Zhou et al., 2019) that can instead use all the per-task data for training the task-specific predictor and also perform well empirically on benchmark tasks. These algorithms modify the loss function in the outer loop so that the training loss no longer matches the meta-test loss, but may have the advantage in terms of data efficiency for the overall problem of learning the best prior. So far it is theoretically unclear how these two approaches (with/without train-validation split) compare with each other, which motivates us to ask the following

> **Question**: Is the train-validation split *necessary* and *optimal* in meta-learning?

In this paper, we perform a detailed theoretical study on the importance of the train-validation split. We consider the linear centroid meta-learning problem (Denevi et al., 2018b), where for each task we learn a linear predictor that is close to a common centroid in the inner loop, and find the best centroid in the outer loop (see Section 2 for the detailed problem setup). We compare two meta-learning algorithms: the **train-val method** which performs the standard train-validation split, and the **train-train method** which uses all the per-task data for both training and evaluation.

We summarize our contributions as follows:

- We show that the train-validation split is necessary in the general agnostic setting (Section 3): The expected loss of the train-val method equals the meta test-time loss. In contrast, the train-train method has a different expected loss and is not minimized at the best test-time centroid in general, for which we construct a concrete counter-example.

- In the perhaps more interesting realizable setting, we show the train-validation split is not necessary: When the tasks are generated from noiseless linear models, the expected loss of both the train-val and train-train methods are minimized at the best test-time centroid (Section 4.1).

- Our main theoretical contribution shows that **the train-validation split is non-optimal** in the realizable setting: The MSE (and test loss) of the two methods concentrates sharply around $C^{\{\text{tr-val,tr-tr}\}}/T$ when T (the number of tasks) is large, where the constants depend on the {dimension, per-task sample size, regularization parameter}. A precise comparison of constants further shows that $C^{\text{tr-tr}} < C^{\text{tr-val}}$ when we optimally tune the regularization parameter in both methods (Section 4.2). Thus, in the realizable setting, the train-train method performs strictly better than the train-val method, which is in stark contrast with the agnostic case. This result provides a novel insight into the effect of the train-validation split on the sample complexity of meta-learning.

- We perform meta-learning experiments on simulations and benchmark few-shot image classification tasks, showing that the train-train method consistently outperforms the train-val method (Section 5 & Appendix F). This validates our theories and presents empirical evidence that sample-splitting may not be crucial; methods that utilize the per-task data more efficiently may be preferred.

- On the technical end, our main results in Section 4 build on concentration analyses on a group of ridge-covariance matrices, as well as tools from random matrix theory in the proportional regime, which may be of broader interest. (See Section 4.3 for an overview of techniques.)

1.1. Related work

Meta-learning and representation learning theory Baxter (2000) provided the first theoretical analysis of meta-learning via covering numbers, and Maurer et al. (2016) improved the analysis via Gaussian complexity techniques. Another recent line of theoretical work analyzed gradient-based meta-learning methods (Denevi et al., 2018a; Finn et al., 2019; Khodak et al., 2019; Ji et al., 2020) and showed guarantees for convex losses by using tools from online convex optimization. Saunshi et al. (2020) proved the success of Reptile in a one-dimensional subspace setting. Wang et al. (2020c) compared the performance of train-train and train-val methods for learning the learning rate. Denevi et al. (2018b) proposed the linear centroid model studied in this paper, and provided generalization error bounds for train-val method; the bounds proved also hold for train-train method, so are not sharp enough to compare the two algorithms. Wang et al. (2020b;a) studied the convergence of gradient-based meta-learning by relating to the kernelized approximation. Arnold et al. (2019) observe that MAML adapts better with a deep model architecture both empirically and theoretically.

On the representation learning end, Du et al. (2020); Tripuraneni et al. (2020a;b) showed that ERM can successfully pool data across tasks to learn the representation. Yet the focus is on the accurate estimation of the common representation, not on the fast adaptation of the learned prior. Several recent work compares MAML versus ERM style approches (Gao & Sener, 2020; Collins et al., 2020); these comparisons couple the effect of sample splitting with other factors such as whether the algorithm uses per-task adaptation. Lastly, we remark that there are analyses for other representation learning schemes (McNamara & Balcan, 2017; Galanti et al., 2016; Alquier et al., 2016).

Empirical understandings of meta-learning Raghu et al. (2020) showed that MAML with a full finetuning inner loop mostly learns the top-layer linear classifier and does not change the representation layers much. This re-

sult partly justifies the validity of our linear centroid meta-learning problem in which the features (representations) are fixed and only a linear classifier is learned. Goldblum et al. (2020) investigated the difference of the neural representations learned by classical training (supervised learning) and meta-learning, and showed that the meta-learned representation is better for downstream adaptation and makes classes more separated. Additionally, Setlur et al. (2020); Yao et al. (2020) investigated alternative ways of choosing the support set (training split) in meta-learning.

Multi-task learning Multi-task learning also exploits structures and similarities across multiple tasks. The earliest idea dates back to Caruana (1997); Thrun & Pratt (1998); Baxter (2000), initially in connections to neural network models. They further motivated other approaches using kernel methods (Evgeniou et al., 2005; Argyriou et al., 2007) and multivariate linear regression models with structured sparsity (Liu et al., 2009; 2015). More recent advances on deep multi-task learning focus on learning shared intermediate representations across tasks (Ruder, 2017). These multi-task learning approaches usually minimize the joint empirical risk over all tasks, and the models for different tasks are enforced to share a large amount of parameters. In contrast, meta-learning only requires the models to share the same "prior", and is more flexible than multi-task learning.

2. Preliminaries

In this paper, we consider the standard meta-learning setting, in which we observe data from $T \geq 1$ supervised learning tasks, and the goal is to find a prior (or "initialization") using the combined data, such that the $(T + 1)$-th new task may be solved sample-efficiently using the prior.

Linear centroid meta-learning We instantiate our study on the *linear centroid meta-learning problem* (also known as learning to learn around a common mean, Denevi et al. (2018b)), where we wish to learn a task-specific linear predictor $\mathbf{w}_t \in \mathbb{R}^d$ in the inner loop for each task t, and learn a "centroid" $\mathbf{w}_0$ in the outer loop that enables fast adaptation to $\mathbf{w}_t$ within each task:

> Find the best centroid $\mathbf{w}_0 \in \mathbb{R}^d$ for adapting to a linear predictor $\mathbf{w}_t$ on each task t.

Formally, we assume that we observe training data from $T \geq 1$ tasks, where for each task index t, we sample a task p_t (a distribution over $\mathbb{R}^d \times \mathbb{R}$) from some distribution of tasks Π, and observe n examples $(\mathbf{X}_t, \mathbf{y}_t) \in \mathbb{R}^{n \times d} \times \mathbb{R}^n$ that are drawn i.i.d. from p_t:

$$p_t \sim \Pi, \ (\mathbf{X}_t, \mathbf{y}_t) = \{(\mathbf{x}_{t,i}, y_{t,i})\}_{i=1}^n \text{ where } (\mathbf{x}_{t,i}, y_{t,i}) \overset{\text{iid}}{\sim} p_t. \tag{1}$$

We do not make further assumptions on (n, d); in particular, we allow the underdetermined setting $n \leq d$, in which there exists (one or many) interpolators $\widetilde{\mathbf{w}}_t$ that perfectly fit the data: $\mathbf{X}_t \widetilde{\mathbf{w}}_t = \mathbf{y}_t$.

Inner loop: Ridge solver with biased regularization towards the centroid Our goal in the inner loop is to find a linear predictor $\mathbf{w}_t$ that fits the data in task t while being close to the given "centroid" $\mathbf{w}_0 \in \mathbb{R}^d$. We instantiate this through ridge regression (i.e. linear regression with L_2 regularization) where the regularization biases $\mathbf{w}_t$ towards the centroid. Formally, for any $\mathbf{w}_0 \in \mathbb{R}^d$ and any dataset $(\mathbf{X}, \mathbf{y})$, we consider the algorithm

$$\begin{aligned} & \mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}, \mathbf{y}) \\ & := \arg\min_{\mathbf{w}} \frac{1}{n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{w} - \mathbf{w}_0\|_2^2 \\ & = \mathbf{w}_0 + (\mathbf{X}^\top \mathbf{X} + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}^\top (\mathbf{y} - \mathbf{X}\mathbf{w}_0), \end{aligned}$$

where $\lambda > 0$ is the regularization strength (typically a tunable hyper-parameter). As we regularize by $\|\mathbf{w} - \mathbf{w}_0\|_2^2$, this inner solver encourages the solution to be close to $\mathbf{w}_0$, as we desire. Such a regularizer is widely used in practical meta-learning algorithms such as MetaOptNet (Lee et al., 2019) and Meta-MinibatchProx (Zhou et al., 2019). In addition, as $\lambda \to 0$, this solver recovers gradient descent fine-tuning: we have

$$\begin{aligned} & \mathcal{A}_0(\mathbf{w}_0; \mathbf{X}, \mathbf{y}) := \lim_{\lambda \to 0} \mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}, \mathbf{y}) \\ & = \mathbf{w}_0 + \mathbf{X}^\dagger(\mathbf{y} - \mathbf{X}\mathbf{w}_0) = \arg\min_{\mathbf{X}\mathbf{w}=\mathbf{y}} \|\mathbf{w} - \mathbf{w}_0\|_2^2, \end{aligned}$$

where $\mathbf{X}^\dagger \in \mathbb{R}^{d \times n}$ denotes the pseudo-inverse of $\mathbf{X}$. This is the *minimum-distance* interpolator of $(\mathbf{X}, \mathbf{y})$ and also the solution found by gradient descent [1] on $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2$ initialized at $\mathbf{w}_0$. Therefore our ridge solver with $\lambda > 0$ can be seen as a generalized version of the gradient descent solver used in MAML (Finn et al., 2017).

Outer loop: Learning the best centroid In the outer loop, our goal is to find the best centroid $\mathbf{w}_0$. The standard approach in meta-learning is to perform a *train-validation split*, that is, (1) execute the inner solver on a first split of the task-specific data, and (2) evaluate the loss on a second split, yielding a function of $\mathbf{w}_0$ that we can optimize. This two-stage procedure can be written as

> Compute $\mathbf{w}_t(\mathbf{w}_0) = \mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}_t^{\text{train}}, \mathbf{y}_t^{\text{train}})$, and
>
> Evaluate $\left\|\mathbf{y}_t^{\text{val}} - \mathbf{X}_t^{\text{val}} \mathbf{w}_t(\mathbf{w}_0)\right\|_2^2$.

where $(\mathbf{X}_t^{\text{train}}, \mathbf{y}_t^{\text{train}}) = \{(\mathbf{x}_{t,i}, y_{t,i})\}_{i=1}^{n_1}$ and $(\mathbf{X}_t^{\text{val}}, \mathbf{y}_t^{\text{val}}) = \{(\mathbf{x}_{t,i}, y_{t,i})\}_{i=n_1+1}^n$ are two disjoint splits of the per-task

---

[1] with a small step-size, or gradient flow.

data $(\mathbf{X}_t, \mathbf{y}_t)$ of size (n_1, n_2), with $n_1 + n_2 = n$. This amounts to the

---

Train-val method: Output $\widehat{\mathbf{w}}_{0,T}^{\text{tr-val}}$ that minimizes

$$\widehat{L}_T^{\text{tr-val}}(\mathbf{w}_0) = \frac{1}{T} \sum_{t=1}^{T} \ell_t^{\text{tr-val}}(\mathbf{w}_0)$$

$$:= \frac{1}{T} \sum_{t=1}^{T} \frac{1}{2n_2} \left\| \mathbf{y}_t^{\text{val}} - \mathbf{X}_t^{\text{val}} \mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}_t^{\text{train}}, \mathbf{y}_t^{\text{train}}) \right\|_2^2.$$

(2)

We compare the train-val method to an alternative version, where we do not perform the train-validation split, but instead use *all the per-task data for both training and evaluation*. Formally, this is to consider the

---

Train-train method: Output $\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}}$ that minimizes

$$\widehat{L}_T^{\text{tr-tr}}(\mathbf{w}_0) = \frac{1}{T} \sum_{t=1}^{T} \ell_t^{\text{tr-tr}}(\mathbf{w}_0)$$

$$:= \frac{1}{T} \sum_{t=1}^{T} \frac{1}{2n} \left\| \mathbf{y}_t - \mathbf{X}_t \mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}_t, \mathbf{y}_t) \right\|_2^2.$$

(3)

Let $L^{\{\text{tr-val,tr-tr}\}}(\mathbf{w}_0) = \mathbb{E}\left[\ell_t^{\{\text{tr-val,tr-tr}\}}(\mathbf{w}_0)\right]$ denote the corresponding expected losses. We remark that this expectation is equivalent to observing an infinite amount of tasks, but still with a finite (n, d) within each task.

(Meta-)Test time The meta-test time performance of any meta-learning algorithm is a joint function of the (learned) centroid $\mathbf{w}_0$ and the inner algorithm Alg. Upon receiving a new task $p_{T+1} \sim \Pi$ and training data $(\mathbf{X}_{T+1}, \mathbf{y}_{T+1}) \in \mathbb{R}^{n \times d} \times \mathbb{R}^n$, we run the inner loop Alg with prior $\mathbf{w}_0$ on the training data, and evaluate it on an (unseen) test example $(\mathbf{x}', y') \sim p_{T+1}$:

$$L^{\text{test}}(\mathbf{w}_0; \text{Alg}) := \mathbb{E}\left[\frac{1}{2}\left(\mathbf{x}'^\top \text{Alg}(\mathbf{w}_0; \mathbf{X}_{T+1}, \mathbf{y}_{T+1}) - y'\right)^2\right].$$

Additionally, for both train-val and train-train methods, we need to ensure that the inner loop used for meta-test is exactly the same as that used in meta-training. Therefore, the meta-test performance for the train-val and train-train methods above should be evaluated as

$$L_{\lambda,n_1}^{\text{test}}(\widehat{\mathbf{w}}_{0,T}^{\text{tr-val}}) := L^{\text{test}}(\widehat{\mathbf{w}}_{0,T}^{\text{tr-val}}; \mathcal{A}_{\lambda,n_1}),$$
$$L_{\lambda,n}^{\text{test}}(\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}}) := L^{\text{test}}(\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}}; \mathcal{A}_{\lambda,n}),$$

where $\mathcal{A}_{\lambda,m}$ denotes the ridge solver with regularization strength $\lambda > 0$ on $m \leq n$ data points. Finally, we let

$$\mathbf{w}_{0,\star}(\lambda; n) = \arg\min_{\mathbf{w}_0} L_{\lambda,n}^{\text{test}}(\mathbf{w}_0) \quad (4)$$

denote the best centroid if the inner loop uses $\mathcal{A}_{\lambda,n}$. The performance of the train-val algorithm $\widehat{\mathbf{w}}_{0,T}^{\text{tr-val}}$ should be compared against $\mathbf{w}_{0,\star}(\lambda, n_1)$, whereas the train-train algorithm $\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}}$ should be compared against $\mathbf{w}_{0,\star}(\lambda, n)$.

3. The importance of sample splitting

We begin by analyzing the train-train and train-val methods defined in (2) and (3), in the agnostic setting where we do not make structural assumptions on the data distribution p_t.

In this case, we show that the importance of the sample splitting is clear even at the population level: the expected loss of the train-val method matches the test-time loss, whereas the expected loss of the train-train method does not match the test-time in general and have a different minimizer.

Theorem 1 (Properties of expected losses in the agnostic case). *Suppose the task distributions satisfy $\mathbb{E}_{\mathbf{x} \sim p_t}[\mathbf{x}\mathbf{x}^\top] \succ 0$, $\mathbb{E}_{\mathbf{x} \sim p_t}[\|\mathbf{x}\|_2^4] < \infty$ and $\mathbb{E}_{(\mathbf{x},y) \sim p_t}[\|\mathbf{x}y\|_2] < \infty$ for almost surely all $p_t \sim \Pi$, but can be otherwise arbitrary. Then, we have the following:*

(a) *(Unbiased loss for train-val method) For any $\lambda > 0$ and any (n_1, n_2) such that $n_1 + n_2 = n$, the expected loss of the train-val method is equal to the meta test-time loss, and thus minimized at the best test-time centroid:*

$$L_{\lambda,n_1,n_2}^{\text{tr-val}}(\mathbf{w}_0) = L_{\lambda,n_1}^{\text{test}}(\mathbf{w}_0).$$

(b) *(Biased loss for train-train method) There exists a distribution of tasks Π on $d = 1$ satisfying the above conditions, on which for any $n \geq 1$ and $\lambda > 0$, the expected loss of the train-train method is not equal to the test-time loss, and the minimizers are not equal:*

$$L_{\lambda,n}^{\text{tr-tr}}(\cdot) \neq L_{\lambda,n}^{\text{test}}(\cdot), \quad \text{and}$$
$$\mathbf{w}_{0,\star}^{\text{tr-tr}} := \arg\min_{\mathbf{w}_0} L^{\text{tr-tr}}(\mathbf{w}_0) \neq \arg\min_{\mathbf{w}_0} L_{\lambda,n}^{\text{test}}(\mathbf{w}_0).$$

Further, the excess test loss of $\mathbf{w}_{0,\star}^{\text{tr-tr}}$ is bounded away from zero: $L_{\lambda,n}^{\text{test}}(\mathbf{w}_{0,\star}^{\text{tr-tr}}) - \min_{\mathbf{w}_0} L_{\lambda,n}^{\text{test}}(\mathbf{w}_0) > 0$.

Theorem 1 makes clear the advantage of the train-val method when there is no structural assumption on the data distributions: The expected version of the train-val loss matches the meta test-time, whereas the expected version of the train-train loss has a bias in general. By standard consistency results (Van der Vaart, 2000), this advantage carries on to the sampled versions as well for large T. In other words, the train-val method is a "valid ERM" (empirical risk minimization) procedure for the test-time loss, whereas the train-train method is not a valid ERM.

Proof intuitions The proof of part (a) follows from direct calculations, whereas the proof of part (b) is trickier as we

need to construct a counter-example in which the expected loss of the train-train method is not equal the test-time loss for *any* λ, n. We provide such a construction in $d = 1$, where the distribution p_t has a certain asymmetry that results in a bias the train-train loss function for any λ and n. However, we expect such a bias to be present in general for any dimensions. The proof of Theorem 1 can be found in Appendix A.

4. Is sample splitting always optimal?

Theorem 1 states a negative result for the train-train method, showing that its expected loss and the meta test-time loss does not have the same values and minimizers. However, such a result does not preclude the possibility that there exists a data distribution on which the minimizers coincide (even though the loss values can still be different).

In this section, we construct a simple data distribution on which train-train method is indeed unbiased in terms of the minimizer of the expected loss, and compare its performance against the train-val method more explicitly.

Realizable linear model We consider the following instantiation of the (generic) meta-learning data distribution assumption in (1): We assume that each task p_t is specified by a $\mathbf{w}_t \in \mathbb{R}^d$ sampled from some distribution Π (overloading notation), and the observed data follows the noiseless linear model with ground truth parameter $\mathbf{w}_t$:

$$\mathbf{y}_t = \mathbf{X}_t \mathbf{w}_t. \qquad (5)$$

Note that when $n \geq d$ and inputs are in general position, we are able to perfectly recover $\mathbf{w}_t$ (by solving linear equations), therefore the problem in the inner loop is easy. However, even in this case the outer loop problem is still nontrivial as we wish to learn the best centroid $\mathbf{w}_0$.

In this section, we make the following assumption on the distributions of $\mathbf{X}_t$ and $\mathbf{w}_t$:

Assumption A (Data distributions for realizable linear model). *The inputs are standard Gaussian:* $\mathbf{x}_{t,i} \overset{\text{iid}}{\sim} \mathsf{N}(\mathbf{0}, \mathbf{I}_d)$. *The true coefficient* $\mathbf{w}_t$ *is independent of* $\mathbf{X}_t$ *and satisfies*

$$\text{Cov}(\mathbf{w}_t) = \mathbb{E}_{\mathbf{w}_t}\left[(\mathbf{w}_t - \mathbf{w}_{0,\star})(\mathbf{w}_t - \mathbf{w}_{0,\star})^\top\right] = \frac{R^2}{d}\mathbf{I}_d, \qquad (6)$$

for some fixed $R^2 > 0$, *and that the individual entries* $\{w_{t,i} - w_{0,\star,i}\}_{i \in [d], t \in [T]}$ *are i.i.d. mean-zero and* KR^2/d- *sub-Gaussian for some absolute constant* $K = O(1)$.

The sub-Gaussian assumption on $\mathbf{w}_t$ allows for a sharp concentration of the MSE to its expectation (over $\mathbf{w}_t$). The Gaussian input assumption allows for a precise characterization of certain ridge covariance type random matrices.

4.1. Population minimizers

We first show that on the realizable linear model (5), the test-time best centroids $\mathbf{w}_{0,\star}(\lambda, n) = \arg\min_{\mathbf{w}_0} L_{\lambda,n}^{\text{test}}(\mathbf{w}_0)$ is the same for any (λ, n), and both the train-train and train-val methods are unbiased: Both expected losses are minimized at $\mathbf{w}_{0,\star}$.

Theorem 2 (Population minimizers on the realizable model). *On the realizable linear model* (5), *suppose Assumption A holds. Then the test-time meta loss for all* $\lambda > 0$ *and all* n *is minimized at the same point, that is, the mean of the ground truth parameters:*

$$\mathbf{w}_{0,\star}(\lambda, n) = \arg\min_{\mathbf{w}_0} L_{\lambda,n}^{\text{test}}(\mathbf{w}_0)$$
$$= \mathbf{w}_{0,\star} := \mathbb{E}_{\mathbf{w}_t \sim \Pi}[\mathbf{w}_t], \quad \text{for all } \lambda > 0, \ n.$$

Furthermore, for both the train-val method and the train-train method, the expected loss is minimized at $\mathbf{w}_{0,\star}$ *for any* $\lambda > 0$, n, *and* (n_1, n_2):

$$\arg\min_{\mathbf{w}_0} L_{\lambda,n_1,n_2}^{\text{tr-val}}(\mathbf{w}_0) = \arg\min_{\mathbf{w}_0} L_{\lambda,n}^{\text{tr-tr}}(\mathbf{w}_0) = \mathbf{w}_{0,\star}.$$

Theorem 2 shows that both the train-val and train-train methods are in expectation minimized at the same optimal parameter $\mathbf{w}_{0,\star}$ which is the mean of $\mathbf{w}_t$. This is a consequence of the good structure in our realizable linear model (5): at a high level, $\mathbf{w}_{0,\star}$ is indeed the best centroid since it has (on average) the closest distance to a randomly sampled $\mathbf{w}_t$. The proof of Theorem 2 be found in Appendix B.

4.2. Precise comparison of rates

Theorem 2 suggests that we are now able to compare performance of the two methods based on their parameter estimation error (for estimating $\mathbf{w}_{0,\star}$).

We are now ready to state our two main theorems, which provide a precise comparison of the MSEs of the train-train and train-val methods under the realizable linear model.

Theorem 3 (Concentration of MSEs in the realizable linear model). *In the realizable linear model* (5), *suppose Assumption A holds,* $T = \widetilde{\Omega}(d)$, $d/n = \Theta(1)$, $n_2/n = \Theta(1)$, *and* $\lambda = \Theta(1) > 0$. *Then with probability at least* $1 - Td^{-10}$, *the MSE of the train-train and train-val methods has the following concentrations, respectively:*

$$\left\|\widehat{\mathbf{w}}_0^{\text{tr-tr}} - \mathbf{w}_{0,\star}\right\|_2^2 = \frac{R^2}{T}\left(C_{d,n,\lambda}^{\text{tr-tr}} + \widetilde{O}\left(\sqrt{\frac{d}{T}} + \frac{1}{\sqrt{d}}\right)\right),$$

$$\left\|\widehat{\mathbf{w}}_0^{\text{tr-val}} - \mathbf{w}_{0,\star}\right\|_2^2 = \frac{R^2}{T}\left(C_{d,n_1,n_2,\lambda}^{\text{tr-val}} + \widetilde{O}\left(\sqrt{\frac{d}{T}} + \frac{1}{\sqrt{d}}\right)\right),$$

where $\widetilde{O}(\cdot)$ *hides* $\log(ndT)$ *factor. Further, the constants*

$C^{\text{tr-tr}}, C^{\text{tr-val}} = \Theta(1)$ *and have explicit expressions:*

$$C^{\text{tr-tr}}_{d,n,\lambda} = \frac{\frac{1}{d}\mathbb{E}\left[\operatorname{tr}\left((\widehat{\boldsymbol{\Sigma}}_n + \lambda\mathbf{I}_d)^{-4}\widehat{\boldsymbol{\Sigma}}_n^2\right)\right]}{\left(\frac{1}{d}\mathbb{E}\left[\operatorname{tr}\left((\widehat{\boldsymbol{\Sigma}}_n + \lambda\mathbf{I}_d)^{-2}\widehat{\boldsymbol{\Sigma}}_n\right)\right]\right)^2},$$

$$C^{\text{tr-val}}_{d,n_1,n_2,\lambda} =$$

$$\frac{\frac{1}{dn_2}\mathbb{E}\left[\operatorname{tr}\left((\widehat{\boldsymbol{\Sigma}}_{n_1} + \lambda\mathbf{I}_d)^{-2}\right)^2 + (n_2+1)\operatorname{tr}\left((\widehat{\boldsymbol{\Sigma}}_{n_1} + \lambda\mathbf{I}_d)^{-4}\right)\right]}{\left(\frac{1}{d}\mathbb{E}\left[\operatorname{tr}\left((\widehat{\boldsymbol{\Sigma}}_{n_1} + \lambda\mathbf{I}_d)^{-2}\right)\right]\right)^2}$$

where $\widehat{\boldsymbol{\Sigma}}_n := \mathbf{X}_t^\top \mathbf{X}_t / n$ *denotes the empirical covariance of a standard Gaussian random matrix* $\mathbf{X}_t \in \mathbb{R}^{n \times d}$.

Theorem 3 asserts that the MSEs of both methods concentrate around R^2/T times a $\Theta(1)$ constant, when both T, d are large and $T = \widetilde{\Omega}(d)$ (so that the error terms vanish). This allows us to compare the performances of the train-train and train-val methods based on the constants. For a fair comparison, we look at the constants with optimal choices of λ and the split ratio, which we state in the following

Theorem 4 (Comparison of constants $C^{\text{tr-tr}}$ and $C^{\text{tr-val}}$). *In the high-dimensional limiting regime* $d, n \to \infty$, $d/n \to \gamma \in (0, \infty)$, *the optimal constant of the train-train method obtained by tuning the regularization* $\lambda \in (0, \infty)$ *satisfies*

$$\inf_{\lambda > 0} \lim_{d,n\to\infty, d/n\to\gamma} C^{\text{tr-tr}}_{d,n,\lambda} = \inf_{\lambda > 0} \rho_{\lambda,\gamma}$$

$$\overset{(\star)}{\le} \max\left\{1 + \frac{5}{27}\gamma, \frac{5}{27} + \gamma\right\},$$

where $\rho_{\lambda,\gamma} := 4\gamma^2\left[(\gamma-1)^2 + (\gamma+1)\lambda\right]/(\lambda + \gamma + 1 - \sqrt{(\lambda+\gamma+1)^2 - 4\gamma})^2/((\lambda+\gamma+1)^2 - 4\gamma)^{3/2}$, *and the inequality becomes equality at* $\gamma = 1$. *In contrast, the optimal rate of the train-val method by tuning the regularization* $\lambda \in (0, \infty)$ *and split ratio* $s \in (0, 1)$ *is*

$$\inf_{\lambda > 0, s \in (0,1)} \lim_{d,n\to\infty, d/n\to\gamma} C^{\text{tr-val}}_{d,ns,n(1-s),\lambda} = (1 + \gamma).$$

As $\max\left\{1 + 5\gamma/27, 5/27 + \gamma\right\} < 1 + \gamma$ *for any* $\gamma > 0$, *the train-train method has a strictly better constant than the train-val method when* λ *and* s *are optimally tuned in both methods.*

Implications Theorem 4 shows that, perhaps surprisingly, the train-train method achieves a strictly better MSE (in terms of the constant) than the train-val method in the realizable linear model[2]. (See Figure 1(a) for a visualization of the exact optimal rates and the upper bound $(\star)$.) This suggests that the train-validation split may not be crucial

---

[2]The same conclusion also holds for the excess test loss, as the Hessian of the test loss is a rescaled identity, see Appendix C.2.

when the data has structural assumptions such as realizability by the model. To the best of our knowledge, this is the first theoretical result that offers a disentangled comparison of meta-learning algorithms with and without sample splitting. Note that our result features an *optimal tuning of hyperparameters*: we compare the rates at the (theoretically) optimal λ for the train-train method and the optimal λ, n_1 for the train-val method. A closely related existing result is (Denevi et al., 2018b, Proposition 3) which proved a sample complexity upper bound for the linear centroid meta-learning problem; however, they only consider the train-val method, and their upper bounds do not tell the exact leading constants as in our Theorem 3.

We also remark that, while our theory considers the linear centroid meta-learning problem, our real data experiments in Section 5.2 suggests that the superiority of the train-train method may also hold on real meta-learning tasks with neural networks.

4.3. Overview of techniques

Here we provide an overview of the techniques in proving Theorem 3 and Theorem 4. We defer the full proofs to Appendix C and Appendix D respectively.

Closed-form expressions for $\widehat{\mathbf{w}}^{\text{tr-tr}}_{0,T}$ **and** $\widehat{\mathbf{w}}^{\text{tr-val}}_{0,T}$ Our first step is to obtain the following closed-form expressions for the estimation errors of both methods in the realizable linear model (see Lemma C.1):

$$\widehat{\mathbf{w}}^{\text{tr-tr}}_{0,T} - \mathbf{w}_{0,\star} = \left(\sum_{t=1}^{T} \mathbf{A}_t\right)^{-1} \sum_{t=1}^{T} \mathbf{A}_t(\mathbf{w}_t - \mathbf{w}_{0,\star}),$$

$$\widehat{\mathbf{w}}^{\text{tr-val}}_{0,T} - \mathbf{w}_{0,\star} = \left(\sum_{t=1}^{T} \mathbf{B}_t\right)^{-1} \sum_{t=1}^{T} \mathbf{B}_t(\mathbf{w}_t - \mathbf{w}_{0,\star}),$$

where

$$\mathbf{A}_t := \lambda^2\left(\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda\mathbf{I}_d\right)^{-2}\left(\mathbf{X}_t^\top \mathbf{X}_t/n\right),$$

$$\mathbf{B}_t := \lambda^2\left(\mathbf{X}_t^{\text{train}\top} \mathbf{X}_t^{\text{train}}/n_1 + \lambda\mathbf{I}_d\right)^{-1}\left(\mathbf{X}_t^{\text{val}\top} \mathbf{X}_t^{\text{val}}/n_2\right)$$
$$\cdot \left(\mathbf{X}_t^{\text{train}\top} \mathbf{X}_t^{\text{train}}/n_1 + \lambda\mathbf{I}_d\right)^{-1}.$$

These expressions simplify the estimation errors as the "weighted averages" of the $\{\mathbf{w}_t - \mathbf{w}_0\}$ with weighting matrices $\mathbf{A}_t$ and $\mathbf{B}_t$.

Sharp concentration to exact constants Our next step is to establish the concentration

$$\left\|\widehat{\mathbf{w}}^{\text{tr-tr}}_{0,T} - \mathbf{w}_{0,\star}\right\|_2^2$$

$$\overset{(i)}{\approx} \frac{R^2}{d} \cdot \operatorname{tr}\left(\left(\sum_{t=1}^{T} \mathbf{A}_t\right)^{-2}\left(\sum_{t=1}^{T} \mathbf{A}_t^2\right)\right)$$

(a) Optimal constants $\inf_\lambda C_{d,n,\lambda}^{\text{tr-tr}}$ and $\inf_\lambda C_{d,n_1,n_2,\lambda}^{\text{tr-val}}$

(b) MSE of $\widehat{\mathbf{w}}_{0,T}^{\{\text{tr-tr, tr-val}\}}$ v.s. T

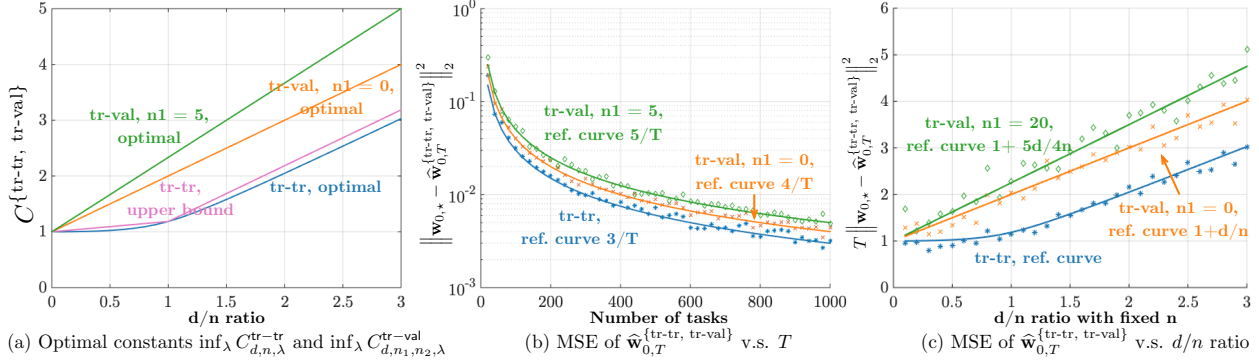(c) MSE of $\widehat{\mathbf{w}}_{0,T}^{\{\text{tr-tr, tr-val}\}}$ v.s. d/n ratio

Figure 1. Panel (a) plots the exact constants in Theorem 4: The optimal train-train constant $\inf_\lambda C_{d,n,\lambda}^{\text{tr-tr}}$ (blue) and its upper bound ($\star$) (magenta), as well as the optimal train-val constant $\inf_\lambda C_{d,n_1,n_2,\lambda}^{\text{tr-val}}$ with $n_1 = 0$ (orange, optimal choice) and $n_1 = 5$ (green). (Optimal $\inf_\lambda C_{d,n_1,n_2,\lambda}^{\text{tr-val}}$ at each n_1 can be found in Lemma D.1.) Curves in panel (a) are used as reference curves in plots (b) and (c). Panel (b) plots the MSE of $\widehat{\mathbf{w}}_{0,T}^{\{\text{tr-tr,tr-val}\}}$ as the total number of tasks increases from 20 to 1000 with an increment of 20. We fix data dimension $d = 60$ and per-task sample size $n = 20$. For the train-val method, we experiment on $n_1 = 0$ and $n_1 = 5$. Panel (c) shows the rescaled MSE of $\widehat{\mathbf{w}}_{0,T}^{\{\text{tr-tr,tr-val}\}}$ as the ratio d/n varies from 0 to 3 (with $n = 100$ and $T = 300$).

$$\overset{(ii)}{\approx} \frac{R^2}{T}\big(\text{tr}(\mathbb{E}[\mathbf{A}_t])/d\big)^{-2}\big(\text{tr}\big(\mathbb{E}[\mathbf{A}_t^2]\big)/d\big) = \frac{R^2}{T}C_{d,n,\lambda}^{\text{tr-tr}}.$$

(and a similar result for $\widehat{\mathbf{w}}_{0,T}^{\text{tr-val}}$ using $\mathbf{B}_t$.) Above, (i) relies on the concentration of a certain quadratic form involving the $(\mathbf{w}_t - \mathbf{w}_0)$'s, following from the Hanson-Wright inequality (cf. Lemma C.5), and (ii) relies on the concentration of the matrices $\sum_{t=1}^T \mathbf{A}_t/T$ and $\sum_{t=1}^T \mathbf{A}_t^2/T$, using standard sub-Gaussian matrix concentration and a truncation argument (cf. Lemma C.4). Further calculating the expectations $\mathbb{E}[\mathbf{A}_t]$ and $\mathbb{E}[\mathbf{A}_t^2]$ gives the exact formula of $C_{d,n,\lambda}^{\text{tr-tr}}$ (cf. Lemma C.2) and finishes the proof of Theorem 3.

Optimizing and comparing $C_{d,n,\lambda}^{\text{tr-tr}}$ and $C_{d,n_1,n_2,\lambda}^{\text{tr-val}}$ The constants $C_{d,n,\lambda}^{\text{tr-tr}}$ and $C_{d,n_1,n_2,\lambda}^{\text{tr-val}}$ involve tunable hyperparameters λ (for both methods) and n_1 (for the train-val method). We use the following strategies to optimize the hyperparameters in each method, which combine to yield Theorem 4.

- For the train-val method, we show that the optimal tunable parameters for any (n,d) is taken at a special case $\lambda = \infty$ and $(n_1,n_2) = (0,n)$, at which the rate only depends on $\frac{1}{n_1}\mathbf{X}_t^{\text{train}\top}\mathbf{X}_t^{\text{train}}$ through its rank (and thus has a simple closed-form). We state this result in Lemma D.1. The proof builds on algebraic manipulations of the quantity $C_{d,n,\lambda_1,\lambda_2}^{\text{tr-val}}$, and can be found in Appendix D.1.

- For the train-train method, we apply random matrix theory to simplify the spectrum of $\frac{1}{n}\mathbf{X}_t^\top \mathbf{X}_t$ in the *proportional limit* where $d,n \to \infty$ and d/n stays as a constant (Bai & Silverstein, 2010; Anderson et al., 2010), and obtain a closed-form expression of the asymptotic MSE for

any $\lambda > 0$, which we can analytically optimize over λ. We state this result in Theorem D.1. The proof builds on the Stieltjes transform and its "derivative trick" (Dobriban et al., 2018), and is deferred to Appendix D.2.

5. Experiments

5.1. Simulations

We experiment on the realizable linear model studied in Section 4. Recall that the observed data of the t-th task are generated as

$$\mathbf{y}_t = \mathbf{X}_t \mathbf{w}_t, \quad \text{with} \quad \mathbf{x}_{t,i} \overset{\text{iid}}{\sim} \mathsf{N}(0, \mathbf{I}_d).$$

We independently generate $\mathbf{w}_t \overset{\text{iid}}{\sim} \mathsf{N}(\mathbf{w}_{0,\star}, \mathbf{I}_d/\sqrt{d})$, where $\mathbf{w}_{0,\star}$ is the linear centroid and the corresponding $R^2 = 1$ here. The goal is to learn the linear centroid $\mathbf{w}_{0,\star}$ using the train-train method and train-val method, i.e., minimizing $\widehat{L}_T^{\text{tr-tr}}$ and $\widehat{L}_T^{\text{tr-val}}$, respectively. Recall that the optimal closed-form solutions $\widehat{\mathbf{w}}_{0,T}^{\{\text{tr-tr,tr-val}\}}$ are given in Section 4.3. We measure the performance of the train-train and train-val methods using the ℓ_2-error $\|\mathbf{w}_{0,\star} - \widehat{\mathbf{w}}_{0,T}^{\{\text{tr-tr,tr-val}\}}\|_2^2$.

Result Figure 1 shows the performance of the train-train and train-val methods on simulated linear centroid meta-learning problems. Across all simulations, we optimally tune the regularization coefficient λ in the train-train method, and use a sufficiently large $\lambda = 2000$ in the train-val method (according to Lemma D.1). Observe that the MSEs of the two methods decay at rate $O(1/T)$ (Figure 1(b)). Further, the performance of the two methods

Table 1. Comparison of train-train and train-val on few-shot image classification (accuracy in %).

	method	1-shot 5-way	5-shot 5-way	1-shot 20-way	5-shot 20-way
miniImage	train-val	48.76 ± 0.87	63.56 ± 0.95	17.52 ± 0.49	21.32 ± 0.54
	train-train	$\mathbf{50.77 \pm 0.90}$	$\mathbf{67.43 \pm 0.89}$	$\mathbf{21.17 \pm 0.38}$	$\mathbf{34.30 \pm 0.41}$
	method	1-shot 5-way	5-shot 5-way	1-shot 10-way	5-shot 10-way
tieredImage	train-val	50.61 ± 1.12	67.30 ± 0.98	29.18 ± 0.57	43.15 ± 0.72
	train-train	$\mathbf{54.37 \pm 0.93}$	$\mathbf{71.45 \pm 0.94}$	$\mathbf{35.56 \pm 0.60}$	$\mathbf{54.50 \pm 0.71}$

in our simulation closely matches the theoretical result in Theorem 4, and the train-train method reliably outperforms the train-val method at all d/n with a moderately large T (Figure 1(c)). In Appendix G, we additionally investigate the effect of averaging the loss over multiple splits in the train-val method (a "cross-validation" type loss).

5.2. Few-shot image classification

We further compare train-train and train-val type methods on the benchmark few-shot image classification tasks miniImageNet (Ravi & Larochelle, 2017) and tieredImageNet (Ren et al., 2018).

Methods We instantiate the train-train and train-val method in the centroid meta-learning setting with a ridge solver. The methods are almost exactly the same as in our theoretical setting in (2) and (3), with the only differences being that the parameters $\mathbf{w}_t$ (and hence $\mathbf{w}_0$) parametrize a deep neural network instead of a linear classifier, and the loss function is the cross-entropy instead of squared loss. Mathematically, we minimize the following two loss functions:

$$L_{\lambda,n_1}^{\text{tr-val}}(\mathbf{w}_0) := \frac{1}{T} \sum_{t=1}^{T} \ell_t^{\text{tr-val}}(\mathbf{w}_0)$$

$$= \frac{1}{T} \sum_{t=1}^{T} \ell \Big(\arg\min_{\mathbf{w}_t} \ell(\mathbf{w}_t; \mathbf{X}_t^{\text{train}}, \mathbf{y}_t^{\text{train}}) + \lambda \|\mathbf{w}_t - \mathbf{w}_0\|_2^2 ; \mathbf{X}_t^{\text{val}}, \mathbf{y}_t^{\text{val}} \Big),$$

$$L_{\lambda}^{\text{tr-tr}}(\mathbf{w}_0) := \frac{1}{T} \sum_{t=1}^{T} \ell_t^{\text{tr-tr}}(\mathbf{w}_0)$$

$$= \frac{1}{T} \sum_{t=1}^{T} \ell \Big(\arg\min_{\mathbf{w}_t} \ell(\mathbf{w}_t; \mathbf{X}_t, \mathbf{y}_t) + \lambda \|\mathbf{w}_t - \mathbf{w}_0\|_2^2 ; \mathbf{X}_t, \mathbf{y}_t \Big),$$

where $(\mathbf{X}_t, \mathbf{y}_t)$ is the data for task t of size n, and $(\mathbf{X}_t^{\text{train}}, \mathbf{y}_t^{\text{train}})$ and $(\mathbf{X}_t^{\text{val}}, \mathbf{y}_t^{\text{val}})$ is a split of the data of size (n_1, n_2). We note that both loss functions above have been considered in prior work ($L^{\text{tr-val}}$ in iMAML (Rajeswaran et al., 2019), and $L^{\text{tr-tr}}$ in Meta-MinibatchProx (Zhou et al., 2019)), though we use slightly different implementation details from these prior work to make sure that the two methods here are exactly the same except for whether the split is used. Additional details about the implementation

can be found in Appendix F.

We experiment on miniImageNet (Ravi & Larochelle, 2017) and tieredImageNet (Ren et al., 2018) datasets. MiniImageNet consists of 100 classes of images from ImageNet (Krizhevsky et al., 2012) and each class has 600 images of resolution $84 \times 84 \times 3$. We use 64 classes for training, 16 classes for validation, and the remaining 20 classes for testing (Ravi & Larochelle, 2017). TieredImageNet consists of 608 classes from the ILSVRC-12 data set (Russakovsky et al., 2015) and each image is also of resolution $84 \times 84 \times 3$.

We adopt the episodic training procedure (Vinyals et al., 2016; Finn et al., 2017; Zhou et al., 2019; Rajeswaran et al., 2019) In each "N-way K-shot setting" (in Table 1), in meta-test, each task provides an N-way K-shot dataset for the model adaptation. In meta-training, for each task we sample an N-way $(K+1)$-shot dataset (and does not allow the algorithm to tune the size of this dataset), so that each task *only* has $n = N(K+1)$ examples, and we allow the algorithm to tune $n_1 \in [0, n]$[3]. Table 1 uses the default choice of an even split $n_1 = n_2 = n/2$ following (Zhou et al., 2019; Rajeswaran et al., 2019). For example, for a 5-way 5-shot classification setting, each task contains $5 \times (5+1) = 30$ total images, and we set $n_1 = n_2 = 15$. (We additionally investigate the optimality of this split ratio in Appendix F.1.) We report the average accuracy over $2,000$ random test episodes with 95% confidence interval.

Results We find that the train-train method consistently outperforms the train-val method (Table 1). Specifically, on miniImageNet, train-train method outperforms train-val by 2.01% and 3.87% on the 1-shot 5-way and 5-shot 5-way tasks respectively; On tieredImageNet, train-train on average improves by about 6.40% on the four testing cases. These results show the advantages of train-train method over train-val and support our theoretical findings in Theorem 4.

---

[3]This setting deviates slightly from our setting (in Section 2 & 3) that meta-train and meta-test needs to have exactly the same (n_1, n_2), but allows a fair comparison of algorithms under the realistic scenario of limited per-task data (fixed n) and fixed data size at meta test time.

6. Conclusion

We study the importance of train-validation split on the linear-centroid meta-learning problem, and show that the necessity and optimality of train-validation split depends greatly on whether the tasks are structured: the sample splitting is necessary in general situations, and not necessary and non-optimal when the tasks are nicely structured. It would be of interest to study whether similar conclusions hold on other meta-learning problems such as learning representations, or how our insights can guide the design of meta-learning algorithms with better empirical performance.

Acknowledgment

References

Alquier, P., Mai, T. T., and Pontil, M. Regret bounds for lifelong learning. *arXiv preprint arXiv:1610.08628*, 2016.

Anderson, G. W., Guionnet, A., and Zeitouni, O. *An introduction to random matrices*, volume 118. Cambridge university press, 2010.

Argyriou, A., Evgeniou, T., and Pontil, M. Multi-task feature learning. In *Advances in neural information processing systems*, pp. 41–48, 2007.

Arnold, S. M., Iqbal, S., and Sha, F. When maml can adapt fast and how to assist when it cannot. *arXiv preprint arXiv:1910.13603*, 2019.

Bai, Y. and Lee, J. D. Beyond linearization: On quadratic and higher-order approximation of wide neural networks. *arXiv preprint arXiv:1910.01619*, 2019.

Bai, Z. and Silverstein, J. W. *Spectral analysis of large dimensional random matrices*, volume 20. Springer, 2010.

Baxter, J. A model of inductive bias learning. *J. Artif. Int. Res.*, 2000.

Caruana, R. Multitask learning. *Machine Learning*, 28 (1):41–75, Jul 1997. ISSN 1573-0565. doi: 10.1023/A:1007379606734. URL https://doi.org/10.1023/A:1007379606734.

Collins, L., Mokhtari, A., and Shakkottai, S. Why does maml outperform erm? an optimization perspective. *arXiv preprint arXiv:2010.14672*, 2020.

Denevi, G., Ciliberto, C., Stamos, D., and Pontil, M. Incremental learning-to-learn with statistical guarantees. *arXiv preprint arXiv:1803.08089*, 2018a.

Denevi, G., Ciliberto, C., Stamos, D., and Pontil, M. Learning to learn around a common mean. In *Advances in Neural Information Processing Systems*, pp. 10169–10179, 2018b.

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

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

Evgeniou, T., Micchelli, C. A., and Pontil, M. Learning multiple tasks with kernel methods. *Journal of machine learning research*, 6(Apr):615–637, 2005.

Fallah, A., Mokhtari, A., and Ozdaglar, A. On the convergence theory of gradient-based model-agnostic meta-learning algorithms. In *International Conference on Artificial Intelligence and Statistics*, pp. 1082–1092, 2020.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1126–1135, 2017.

Finn, C., Rajeswaran, A., Kakade, S., and Levine, S. Online meta-learning. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., and Pontil, M. Bilevel programming for hyperparameter optimization and meta-learning. *arXiv preprint arXiv:1806.04910*, 2018.

Galanti, T., Wolf, L., and Hazan, T. A theoretical framework for deep transfer learning. *Information and Inference: A Journal of the IMA*, 5(2):159–209, 2016.

Gao, K. and Sener, O. Modeling and optimization tradeoff in meta-learning. *arXiv preprint arXiv:2010.12916*, 2020.

Goldblum, M., Reich, S., Fowl, L., Ni, R., Cherepanova, V., and Goldstein, T. Unraveling meta-learning: Understanding feature representations for few-shot tasks. *arXiv preprint arXiv:2002.06753*, 2020.

Gu, J., Wang, Y., Chen, Y., Cho, K., and Li, V. O. Meta-learning for low-resource neural machine translation. *arXiv preprint arXiv:1808.08437*, 2018.

Ji, K., Lee, J. D., Liang, Y., and Poor, H. V. Convergence of meta-learning with task-specific adaptation over partial parameters. *arXiv preprint arXiv:2006.09486*, 2020.

Khodak, M., Balcan, M.-F., and Talwalkar, A. Adaptive gradient-based meta-learning methods. *arXiv preprint arXiv:1906.02717*, 2019.

Krizhevsky, A., Sutskever, I., and Hinton, G. Imagenet classification with deep convolutional neural networks. pp. 1097–1105, 2012.

Lee, K., Maji, S., Ravichandran, A., and Soatto, S. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10657–10665, 2019.

Liu, H., Palatucci, M., and Zhang, J. Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 649–656, 2009.

Liu, H., Wang, L., and Zhao, T. Calibrated multivariate regression with application to neural semantic basis discovery. *Journal of machine learning research: JMLR*, 16: 1579, 2015.

Maurer, A., Pontil, M., and Romera-Paredes, B. The benefit of multitask representation learning. *The Journal of Machine Learning Research*, 17(1):2853–2884, 2016.

McNamara, D. and Balcan, M.-F. Risk bounds for transferring representations with and without fine-tuning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 2373–2381. JMLR. org, 2017.

Nichol, A. and Schulman, J. Reptile: a scalable metalearning algorithm. *arXiv preprint arXiv:1803.02999*, 2, 2018.

Nichol, A., Achiam, J., and Schulman, J. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

Raghu, A., Raghu, M., Bengio, S., and Vinyals, O. Rapid learning or feature reuse? towards understanding the effectiveness of maml. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rkgMkCEtPB.

Rajeswaran, A., Finn, C., Kakade, S. M., and Levine, S. Meta-learning with implicit gradients. In *Advances in Neural Information Processing Systems*, pp. 113–124, 2019.

Ravi, S. and Larochelle, H. Optimization as a model for few-shot learning. 2017.

Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J., Larochelle, H., and Zemel, R. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018.

Ruder, S. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., and Bernstein, M. Imagenet large scale visual recognition challenge. 115(3):211–252, 2015.

Saunshi, N., Zhang, Y., Khodak, M., and Arora, S. A sample complexity separation between non-convex and convex meta-learning. *arXiv preprint arXiv:2002.11172*, 2020.

Schmidhuber, J. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, Technische Universität München, 1987.

Setlur, A., Li, O., and Smith, V. Is support set diversity necessary for meta-learning?, 2020.

Snell, J., Swersky, K., and Zemel, R. Prototypical networks for few-shot learning. pp. 4077–4087, 2017.

Thrun, S. and Pratt, L. *Learning to Learn: Introduction and Overview*, pp. 3–17. Springer US, Boston, MA, 1998. ISBN 978-1-4615-5529-2. doi: 10.1007/978-1-4615-5529-2_1. URL https://doi.org/10.1007/978-1-4615-5529-2_1.

Tripuraneni, N., Jin, C., and Jordan, M. I. Provable meta-learning of linear representations. *arXiv preprint arXiv:2002.11684*, 2020a.

Tripuraneni, N., Jordan, M. I., and Jin, C. On the theory of transfer learning: The importance of task diversity. *arXiv preprint arXiv:2006.11650*, 2020b.

Van der Vaart, A. W. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., and Wierstra, D. Matching networks for one shot learning. *arXiv preprint arXiv:1606.04080*, 2016.

Wang, H., Sun, R., and Li, B. Global convergence and induced kernels of gradient-based meta-learning with neural nets. *arXiv preprint arXiv:2006.14606*, 2020a.

Wang, L., Cai, Q., Yang, Z., and Wang, Z. On the global optimality of model-agnostic meta-learning. In *International Conference on Machine Learning*, pp. 9837–9846. PMLR, 2020b.

Wang, X., Yuan, S., Wu, C., and Ge, R. Guarantees for tuning the step size using a learning-to-learn approach. *arXiv preprint arXiv:2006.16495*, 2020c.

Xie, Y., Jiang, H., Liu, F., Zhao, T., and Zha, H. Meta learning with relational information for short sequences. In *Advances in Neural Information Processing Systems*, pp. 9904–9915, 2019.

Yao, H., Huang, L., Wei, Y., Tian, L., Huang, J., and Li, Z. Don't overlook the support set: Towards improving generalization in meta-learning. *arXiv preprint arXiv:2007.13040*, 2020.

Zhou, P., Yuan, X., Xu, H., Yan, S., and Feng, J. Efficient meta learning via minibatch proximal update. In *Advances in Neural Information Processing Systems*, pp. 1534–1544, 2019.

A. Proof of Theorem 1

A.1. Proof of part (a)

We need to show that

$$L^{\text{tr-val}}(\mathbf{w}_0) = \mathbb{E}[\ell_t^{\text{tr-val}}(\mathbf{w}_0)] = L_{\lambda,n_1}^{\text{test}}(\mathbf{w}_0)$$

for all $\mathbf{w}_0$, that is, the population meta-test loss is exactly the same as the expected loss of the train-val method. This is straightforward: as the tasks are i.i.d. and $\mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}_t^{\text{train}}, \mathbf{y}_t^{\text{train}})$ is independent of the test points $(\mathbf{X}_t^{\text{val}}, \mathbf{y}_t^{\text{val}})$, we have for any $\mathbf{w}_0$ that

$$\mathbb{E}[\ell_t^{\text{tr-val}}(\mathbf{w}_0)] = \mathbb{E}_{p_t \sim \Pi, (\mathbf{X}_t, \mathbf{y}_t) \sim p_t} \left[\frac{1}{2n_2} \left\| \mathbf{y}_t^{\text{val}} - \mathbf{X}_t^{\text{val}} \mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}_t^{\text{train}}, \mathbf{y}_t^{\text{train}}) \right\|_2^2 \right]$$

$$= \mathbb{E}_{p_t \sim \Pi, (\mathbf{X}_t, \mathbf{y}_t) \sim p_t} \left[\frac{1}{2} \left(y_{t,1}^{\text{val}} - \mathbf{x}_{t,1}^{\text{val}\top} \mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}_t^{\text{train}}, \mathbf{y}_t^{\text{train}}) \right)^2 \right]$$

$$= \mathbb{E}_{p_{T+1} \sim \Pi, (\mathbf{X}_{T+1}, \mathbf{y}_{T+1}), (\mathbf{x}', y') \overset{\text{iid}}{\sim} p_t} \left[\frac{1}{2} \left(y' - \mathbf{x}'^\top \mathcal{A}_{\lambda,n_1}(\mathbf{w}_0; \mathbf{X}_{T+1}, \mathbf{y}_{T+1}) \right)^2 \right]$$

$$= L_{\lambda,n_1}^{\text{test}}(\mathbf{w}_0).$$

This finishes the proof of part (a). $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad$ $\square$

We also calculate the minimizer of the test-time loss $L_{\lambda,n}^{\text{test}}$ (notice that here we use n instead of n_1 training samples per task), which will be useful for our proof of part (b). We have

$$L_{\lambda,n}^{\text{test}}(\mathbf{w}_0) = \mathbb{E}_{p_t \sim \Pi, (\mathbf{X}_t, \mathbf{y}_t), (\mathbf{x}', y') \overset{\text{iid}}{\sim} p_t} \left[\frac{1}{2} \left(y' - \mathbf{x}'^\top \mathcal{A}_{\lambda,n}(\mathbf{w}_0; \mathbf{X}_t, \mathbf{y}_t) \right)^2 \right]$$

$$= \mathbb{E}_{p_t \sim \Pi, (\mathbf{X}_t, \mathbf{y}_t), (\mathbf{x}', y') \overset{\text{iid}}{\sim} p_t} \left[\frac{1}{2} \left(y' - \mathbf{x}'^\top \left[\mathbf{w}_0 + (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top (\mathbf{y}_t - \mathbf{X}_t \mathbf{w}_0) \right] \right)^2 \right]$$

$$= \frac{1}{2} \mathbf{w}_0^\top \mathbf{M} \mathbf{w}_0 - \mathbf{w}_0^\top \mathbf{b} + \text{const},$$

where

$$\mathbf{M} := \mathbb{E}_{p_t \sim \Pi, (\mathbf{X}_t, \mathbf{y}_t), (\mathbf{x}', y') \overset{\text{iid}}{\sim} p_t} \left[\left(\mathbf{I}_d - (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top \mathbf{X}_t \right) \mathbf{x}' \mathbf{x}'^\top \left(\mathbf{I}_d - (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top \mathbf{X}_t \right) \right]$$
$$= \mathbb{E}_{p_t, (\mathbf{X}_t, \mathbf{y}_t)} \left[\lambda^2 (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \mathbf{\Sigma}_t (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \right] \succ \mathbf{0}, \tag{7}$$

where $\mathbf{\Sigma}_t := \mathbb{E}_{x \sim p_t}[\mathbf{x}\mathbf{x}^\top] \succ \mathbf{0}$, and

$$\mathbf{b} := \mathbb{E}_{p_t \sim \Pi, (\mathbf{X}_t, \mathbf{y}_t), (\mathbf{x}', y') \overset{\text{iid}}{\sim} p_t} \left[\left(\mathbf{I}_d - (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \right) \mathbf{x}'_t \left(y'_t - \mathbf{x}_t'^\top (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top \mathbf{y}_t \right) \right]$$

$$= \mathbb{E}_{p_t \sim \Pi, (\mathbf{X}_t, \mathbf{y}_t), (\mathbf{x}', y') \overset{\text{iid}}{\sim} p_t} \left[\lambda \left(\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d \right)^{-1} \mathbf{x}'_t \left(y'_t - \mathbf{x}_t'^\top (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top \mathbf{y}_t/n \right) \right] \tag{8}$$

$$= \lambda \mathbb{E} \left[(\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \right] \mathbb{E}_{p_t, (\mathbf{x}', y') \sim p_t}[\mathbf{x}' y'] - \lambda \mathbb{E} \left[(\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \mathbf{\Sigma}_t (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \frac{1}{n} \mathbf{X}_t^\top \mathbf{y}_t \right].$$

Noticing that $(\mathbf{X}_t^{\text{train}\top} \mathbf{X}_t^{\text{train}}/n + \lambda \mathbf{I}_d)^{-1} \preceq \lambda^{-1} \mathbf{I}_d$ and by the assumptions that $\mathbb{E}_{(\mathbf{x},y) \sim p_t}[\|\mathbf{x}\mathbf{x}^\top\|_{\text{op}}] < \infty$, $\mathbb{E}_{(\mathbf{x},y) \sim p_t}[\|\mathbf{x}y\|_2] < \infty$, we have $\|\mathbf{M}\|_{\text{op}} < \infty$ and $\|\mathbf{b}_T\|_2 < \infty$. Therefore, the minimizer of $L_{\lambda,n}^{\text{test}}$ is

$$\mathbf{w}_{0,\star}(\lambda, n) = \arg\min_{\mathbf{w}_0} L_{\lambda,n}^{\text{test}}(\mathbf{w}_0) = \mathbf{M}^{-1} \mathbf{b}, \tag{9}$$

where $\mathbf{M} \in \mathbb{R}^{d \times d}$ and $\mathbf{b} \in \mathbb{R}^d$ are defined in (7) and (8).

A.2. Proof of part (b)

We construct a simple counter-example on which the minimizer of $L^{\text{tr-tr}}$ is not equal to that of $L_{\lambda,n}^{\text{test}}$ for any $\lambda > 0$ and $n \geq 1$. We begin by simplifying the train-train loss. We have

$$\ell_t^{\text{tr-tr}}(\mathbf{w}_0) = \frac{1}{2n} \left\| \mathbf{y}_t - \mathbf{X}_t \mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}_t, \mathbf{y}_t) \right\|_2^2$$

$$= \frac{1}{2n} \left\| \mathbf{y}_t - \mathbf{X}_t \left[\mathbf{w}_0 + (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top (\mathbf{y}_t - \mathbf{X}_t \mathbf{w}_0) \right] \right\|_2^2$$

$$= \frac{1}{2} \left\| \mathbf{A}_t \mathbf{w}_0 - \mathbf{c}_t \right\|_2^2,$$

where

$$\mathbf{A}_t = \frac{1}{\sqrt{n}} n\lambda \mathbf{X}_t (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \quad \text{and} \quad \mathbf{c}_t = \frac{1}{\sqrt{n}} \left(\mathbf{I}_n - \mathbf{X}_t (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top \right) \mathbf{y}_t.$$

Therefore, the minimizer of the expected loss $L^{\text{tr-tr}}$ is

$$\mathbf{w}_{0,\star}^{\text{tr-tr}} = \arg\min_{\mathbf{w}_0} L^{\text{tr-tr}}(\mathbf{w}_0) = \left(\mathbb{E}[\mathbf{A}_t^\top \mathbf{A}_t] \right)^{-1} \mathbb{E}[\mathbf{A}_t^\top \mathbf{c}_t]$$

$$= \mathbb{E}\left[\lambda^2 (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-2} \frac{\mathbf{X}_t^\top \mathbf{X}_t}{n} \right]^{-1} \cdot \mathbb{E}\left[\frac{1}{n} \lambda (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top (\mathbf{I}_n - \mathbf{X}_t (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top) \mathbf{y}_t \right] \quad (10)$$

$$= \mathbb{E}\left[\lambda^2 (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-2} \frac{\mathbf{X}_t^\top \mathbf{X}_t}{n} \right]^{-1} \cdot \mathbb{E}\left[\lambda^2 (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-2} \frac{1}{n} \mathbf{X}_t^\top \mathbf{y}_t \right].$$

On the other hand, recall from (9) that the minimizer of the test-time loss $L_{\lambda,n}^{\text{test}}$ is

$$\mathbf{w}_{0,\star}(\lambda, n) = \arg\min_{\mathbf{w}_0} L_{\lambda,n}^{\text{test}}(\mathbf{w}_0)$$

$$= \mathbb{E}\left[\lambda^2 (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \mathbf{\Sigma}_t (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \right]^{-1} \cdot \left\{ \lambda \mathbb{E}\left[(\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \right] \mathbb{E}_{p_t,(\mathbf{x}',y') \sim p_t}[\mathbf{x}'y'] \right.$$

$$\left. - \lambda \mathbb{E}\left[(\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \mathbf{\Sigma}_t (\mathbf{X}_t^\top \mathbf{X}_t/n + \lambda \mathbf{I}_d)^{-1} \frac{1}{n} \mathbf{X}_t^\top \mathbf{y}_t \right] \right\}. \quad (11)$$

Construction of the counter-example   We now construct a distribution for which (10) is not equal to (11). Let $d = 1$ and let all p_t be the following distribution:

$$p_t : \quad (x_{t,i}, y_{t,i}) = \begin{cases} (1, 3) & \text{with probability } 1/2; \\ (3, -1) & \text{with probability } 1/2. \end{cases}$$

Clearly, we have $\mathbf{\Sigma}_t = 5$, $s_t := \mathbf{X}_t^\top \mathbf{X}_t/n \in [1, 9]$, and $\mathbb{E}_{x',y' \sim p_t}[x'y'] = 0$. Therefore we have

$$\mathbf{w}_{0,\star}^{\text{tr-tr}} = \mathbb{E}\left[(s_t + \lambda)^{-2} s_t \right]^{-1} \cdot \mathbb{E}\left[(s_t + \lambda)^{-2} \frac{1}{n} \sum_{i=1}^n x_{t,i} y_{t,i} \right],$$

and

$$\mathbf{w}_{0,\star}(\lambda, n) = -\mathbb{E}\left[5\lambda^2 (s_t + \lambda)^{-2} \right]^{-1} \cdot \mathbb{E}\left[5\lambda (s_t + \lambda)^{-2} \frac{1}{n} \sum_{i=1}^n x_{t,i} y_{t,i} \right]$$

$$= -\mathbb{E}\left[\lambda (s_t + \lambda)^{-2} \right]^{-1} \cdot \mathbb{E}\left[(s_t + \lambda)^{-2} \frac{1}{n} \sum_{i=1}^n x_{t,i} y_{t,i} \right].$$

We now show that $\mathbf{w}_{0,\star}^{\text{tr-tr}} \neq \mathbf{w}_{0,\star}(\lambda, n)$ by showing that

$$\mathbb{E}\left[(s_t + \lambda)^{-2} \frac{1}{n} \sum_{i=1}^n x_{t,i} y_{t,i} \right] = \mathbb{E}\left[\frac{x_{t,1} y_{t,1}}{(s_t + \lambda)^2} \right] \neq 0$$

for any $\lambda > 0$. Indeed, conditioning on $(x_{t,1}, y_{t,1}) = (1, 3)$, we know that the sum-of-squares in s_t has one term that equals 1, and all others i.i.d. being 1 or 9 with probability one half. On the other hand, if we condition on $(x_{t,1}, y_{t,1}) = (3, -1)$,

then we know the sum in s_t has one term that equals 9 and all others i.i.d.. This means that the negative contribution in the expectation is smaller than the positive contribution, in other words

$$\mathbb{E}\left[\frac{x_{t,1}y_{t,1}}{(s_t+\lambda)^2}\right] = \frac{1}{2} \cdot 3\mathbb{E}\left[\frac{1}{(s_t+\lambda)^2}\bigg|(x_{t,1},y_{t,1})=(1,3)\right]$$
$$+ \frac{1}{2} \cdot -3\mathbb{E}\left[\frac{1}{(s_t+\lambda)^2}\bigg|(x_{t,1},y_{t,1})=(3,-1)\right] > 0.$$

This shows $\mathbf{w}_{0,\star}^{\text{tr-tr}} \neq \mathbf{w}_{0,\star}(\lambda,n)$.

Finally, for this distribution, the test loss $L_{\lambda,n}^{\text{test}}(\mathbf{w}_0)$ is strongly convex (since it has a positive second derivative), this further implies that the excess loss $L_{\lambda,n}^{\text{test}}(\mathbf{w}_{0,\star}^{\text{tr-tr}}) - L_{\lambda,n}^{\text{test}}(\mathbf{w}_{0,\star}(\lambda,n))$ is bounded away from zero. $\qquad\square$

B. Proof of Theorem 2

We first show that $\mathbf{w}_{0,\star} = \mathbb{E}_{\mathbf{w}_t\sim\Pi}[\mathbf{w}_t]$ is a global optimizer for $L_{\lambda,n}^{\text{tr-tr}}$ and $L_{\lambda,n_1,n_2}^{\text{tr-val}}$ with any regularization coefficient $\lambda > 0$, any n, and any split (n_1,n_2). To do this, it suffices to check that the gradient at $\mathbf{w}_{0,\star}$ is zero and the Hessian is positive definite (PD).

Optimality of $\mathbf{w}_{0,\star}$ in both $L_{\lambda,n}^{\text{tr-tr}}$ and $L_{\lambda,n_1,n_2}^{\text{tr-val}}$. We first look at $L^{\text{tr-tr}}$: for any $\mathbf{w}_0 \in \mathbb{R}^d$ we have

$$L_{\lambda,n}^{\text{tr-tr}}(\mathbf{w}_0) = \mathbb{E}[\ell_t^{\text{tr-tr}}(\mathbf{w}_0)]$$
$$= \frac{1}{2n}\mathbb{E}\left[\left\|\mathbf{X}_t\mathbf{w}_t - \mathbf{X}_t\left[\left(\mathbf{X}_t^\top\mathbf{X}_t + n\lambda\mathbf{I}_d\right)^{-1}\mathbf{X}_t^\top\left(\mathbf{X}_t\mathbf{w}_t - \mathbf{X}_t\mathbf{w}_0\right) + \mathbf{w}_0\right]\right\|_2^2\right]$$
$$= \frac{1}{2n}\mathbb{E}\left[\left\|\mathbf{X}_t\left(\mathbf{I}_d - \left(\mathbf{X}_t^\top\mathbf{X}_t + n\lambda\mathbf{I}_d\right)^{-1}\mathbf{X}_t^\top\mathbf{X}_t\right)(\mathbf{w}_t - \mathbf{w}_0)\right\|_2^2\right]. \tag{12}$$

Similarly, $L_{\lambda,n_1,n_2}^{\text{tr-val}}$ can be written as

$$L^{\text{tr-val}}(\mathbf{w}_0) = \mathbb{E}[\ell_t^{\text{tr-val}}(\mathbf{w}_0)]$$
$$= \frac{1}{2n_2}\mathbb{E}\left[\left\|\mathbf{X}_t^{\text{val}}\mathbf{w}_t - \mathbf{X}_t^{\text{val}}\left[\left((\mathbf{X}_t^{\text{train}})^\top\mathbf{X}_t^{\text{train}} + n_1\lambda\mathbf{I}_d\right)^{-1}(\mathbf{X}_t^{\text{train}})^\top\left(\mathbf{X}_t^{\text{train}}\mathbf{w}_t - \mathbf{X}_t^{\text{train}}\mathbf{w}_0\right) + \mathbf{w}_0\right]\right\|_2^2\right]$$
$$= \frac{1}{2n_2}\mathbb{E}\left[\left\|\mathbf{X}_t^{\text{val}}\left(\mathbf{I}_d - \left((\mathbf{X}_t^{\text{train}})^\top\mathbf{X}_t^{\text{train}} + n_1\lambda\mathbf{I}_d\right)^{-1}(\mathbf{X}_t^{\text{train}})^\top\mathbf{X}_t^{\text{train}}\right)(\mathbf{w}_t - \mathbf{w}_0)\right\|_2^2\right]. \tag{13}$$

We denote

$$\mathbf{M}_t^{\text{tr-tr}} = \mathbf{X}_t\left(\mathbf{I}_d - \left(\mathbf{X}_t^\top\mathbf{X}_t + n\lambda\mathbf{I}_d\right)^{-1}\mathbf{X}_t^\top\mathbf{X}_t\right) \quad \text{and}$$
$$\mathbf{M}_t^{\text{tr-val}} = \mathbf{X}_t^{\text{val}}\left(\mathbf{I}_d - \left((\mathbf{X}_t^{\text{train}})^\top\mathbf{X}_t^{\text{train}} + n_1\lambda\mathbf{I}_d\right)^{-1}(\mathbf{X}_t^{\text{train}})^\top\mathbf{X}_t^{\text{train}}\right)$$

to simplify the notations in (12) and (13). We take gradient of $L^{\text{tr-tr}}$ and $L^{\text{tr-val}}$ with respect to $\mathbf{w}_0$:

$$\nabla_{\mathbf{w}_0}L_{\lambda,n}^{\text{tr-tr}}(\mathbf{w}_0) = -\frac{1}{n}\mathbb{E}\left[(\mathbf{M}_t^{\text{tr-tr}})^\top\mathbf{M}_t^{\text{tr-tr}}(\mathbf{w}_t - \mathbf{w}_0)\right], \tag{14}$$

$$\nabla_{\mathbf{w}_0}L_{\lambda,n_1,n_2}^{\text{tr-val}}(\mathbf{w}_0) = -\frac{1}{n_2}\mathbb{E}\left[(\mathbf{M}_t^{\text{tr-val}})^\top\mathbf{M}_t^{\text{tr-val}}(\mathbf{w}_t - \mathbf{w}_0)\right]. \tag{15}$$

Substituting $\mathbf{w}_{0,\star}$ into (14) and taking expectation, we deduce

$$\nabla_{\mathbf{w}_0}L_{\lambda,n}^{\text{tr-tr}}(\mathbf{w}_{0,\star}) = -\frac{1}{n}\mathbb{E}\left[(\mathbf{M}_t^{\text{tr-tr}})^\top\mathbf{M}_t^{\text{tr-tr}}(\mathbf{w}_t - \mathbf{w}_{0,\star})\right] = \mathbf{0}. \tag{16}$$

To see this, observe that by definition $\mathbb{E}[\mathbf{w}_t - \mathbf{w}_{0,\star}] = \mathbf{0}$. Combining with $\mathbf{w}_t$ being generated independently of $\mathbf{X}_t$, we obtain that the RHS of (16) vanish. Following the same argument, we can show

$$\nabla_{\mathbf{w}_0}L_{\lambda,n_1,n_2}^{\text{tr-val}}(\mathbf{w}_{0,\star}) = \mathbf{0},$$

since $\mathbf{X}_t^{\text{val}}$ is also independent of $\mathbf{w}_t$. The reasoning above indicates that $\mathbf{w}_{0,\star}$ is a stationary point of both $L_{\lambda,n}^{\text{tr-tr}}$ and $L_{\lambda,n_1,n_2}^{\text{tr-val}}$. The remaining step is to check $\nabla_{\mathbf{w}_0}^2 L_{\lambda,n}^{\text{tr-tr}}(\mathbf{w}_{0,\star})$ and $\nabla_{\mathbf{w}_0}^2 L_{\lambda,n_1,n_2}^{\text{tr-val}}(\mathbf{w}_{0,\star})$ are positive definite. From (14) and (15), we derive respectively the Hessian of $L_{\lambda,n}^{\text{tr-tr}}$ and $L_{\lambda,n_1,n_2}^{\text{tr-val}}$ as

$$\nabla_{\mathbf{w}_0}^2 L_{\lambda,n}^{\text{tr-tr}}(\mathbf{w}_{0,\star}) = \frac{1}{n}\mathbb{E}[(\mathbf{M}_t^{\text{tr-tr}})^\top \mathbf{M}_t^{\text{tr-tr}}] \quad \text{and}$$

$$\nabla_{\mathbf{w}_0}^2 L_{\lambda,n_1,n_2}^{\text{tr-val}}(\mathbf{w}_{0,\star}) = \frac{1}{n_2}\mathbb{E}[(\mathbf{M}_t^{\text{tr-val}})^\top \mathbf{M}_t^{\text{tr-val}}].$$

Let $\mathbf{v} \in \mathbb{R}^d$ be any nonzero vector, our goal is to check $\mathbf{v}^\top \nabla_{\mathbf{w}_0}^2 L_{\lambda,n}^{\text{tr-tr}}(\mathbf{w}_{0,\star})\mathbf{v} > 0$, and also $\mathbf{v}^\top \nabla_{\mathbf{w}_0}^2 L_{\lambda,n_1,n_2}^{\text{tr-val}}(\mathbf{w}_{0,\star})\mathbf{v} > 0$. Consider $\mathbf{v}^\top \nabla_{\mathbf{w}_0}^2 L_{\lambda,n}^{\text{tr-tr}}(\mathbf{w}_{0,\star})\mathbf{v}$ first. By the isotropicity of Gaussian and some algebraic manipulation, we can show

$$\mathbf{v}^\top \nabla_{\mathbf{w}_0}^2 L_{\lambda,n}^{\text{tr-tr}}(\mathbf{w}_{0,\star})\mathbf{v} = \frac{1}{n}\mathbb{E}\left[\mathbf{v}^\top \left(\mathbf{I}_d - (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{X}_t^\top \mathbf{X}_t\right)\mathbf{X}_t^\top \mathbf{X}_t \left(\mathbf{I}_d - (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{X}_t^\top \mathbf{X}_t\right)\mathbf{v}\right] > 0.$$

See a detailed computation in the proof of Lemma C.2 (term $\mathbb{E}[\mathbf{A}_t]$ in (25)). As a consequence, we have shown that $\mathbf{w}_{0,\star}$ is a global optimum of $L_{\lambda,n}^{\text{tr-tr}}$. The same argument applies to $L_{\lambda,n_1,n_2}^{\text{tr-val}}$ (The Hessian $\nabla_{\mathbf{w}_0}^2 L_{\lambda,n_1,n_2}^{\text{tr-val}}(\mathbf{w}_{0,\star})$ being positive definite is also shown in Lemma C.2. See term $\mathbb{E}[\mathbf{B}_t]$ in (28)). The proof is complete. $\qquad\square$

C. Proof of Theorem 3

The proof is organized as follows. We first derive the closed-form expressions of the train-train and train-val estimators in terms of the matrices $\mathbf{A}_t$, $\mathbf{B}_t$ in Section C.1. We compute the first and second moments of $\mathbf{A}_t$ and $\mathbf{B}_t$ in Section C.2, present some concentration lemmas in Section C.3, and then prove the main theorem in Section C.4.

C.1. Closed-form expressions for the estimators

Lemma C.1 (Closed-form expressions for $\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}}$ and $\widehat{\mathbf{w}}_{0,T}^{\text{tr-val}}$). *For the realizable linear model* (5)*, the train-train method* (3) *and the train-val method* (2) *have closed-form expressions*

$$\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}} = \left(\sum_{t=1}^T \mathbf{A}_t\right)^{-1}\sum_{t=1}^T \mathbf{A}_t \mathbf{w}_t, \tag{17}$$

$$\widehat{\mathbf{w}}_{0,T}^{\text{tr-val}} = \left(\sum_{t=1}^T \mathbf{B}_t\right)^{-1}\sum_{t=1}^T \mathbf{B}_t \mathbf{w}_t, \tag{18}$$

where

$$\mathbf{A}_t := \lambda^2\left(\frac{\mathbf{X}_t^\top \mathbf{X}_t}{n} + \lambda\mathbf{I}_d\right)^{-2}\frac{\mathbf{X}_t^\top \mathbf{X}_t}{n}, \tag{19}$$

$$\mathbf{B}_t := \lambda^2\left(\frac{\mathbf{X}_t^{\text{train}\top}\mathbf{X}_t^{\text{train}}}{n_1} + \lambda\mathbf{I}_d\right)^{-1}\frac{\mathbf{X}_t^{\text{val}\top}\mathbf{X}_t^{\text{val}}}{n_2}\left(\frac{\mathbf{X}_t^{\text{train}\top}\mathbf{X}_t^{\text{train}}}{n_1} + \lambda\mathbf{I}_d\right)^{-1}. \tag{20}$$

Proof. We consider the train-train method first. Substituting $\mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}, \mathbf{y}) = \mathbf{w}_0 + (\mathbf{X}^\top \mathbf{X} + n\lambda\mathbf{I}_d)^{-1}\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\mathbf{w}_0)$ into (3) yields

$$\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}} = \arg\min_{\mathbf{w}_0}\frac{1}{T}\sum_{t=1}^T \frac{1}{2n}\left\|\mathbf{y}_t - \mathbf{X}_t\left(\mathbf{w}_0 + (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{X}_t^\top(\mathbf{y}_t - \mathbf{X}_t\mathbf{w}_0)\right)\right\|_2^2.$$

The optimization problem above is quadratic in $\mathbf{w}_0$. Therefore, by setting the gradient with respect to $\mathbf{w}_0$ equal to zero, we derive

$$\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}} = \arg\min_{\mathbf{w}_0}\frac{1}{T}\sum_{t=1}^T \frac{1}{2n}\left\|\left(\mathbf{I}_d - \mathbf{X}_t(\mathbf{X}_t^\top \mathbf{X}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{X}_t^\top\right)\mathbf{y}_t - \mathbf{X}_t\left(\mathbf{I}_d - (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{X}_t^\top \mathbf{X}_t\right)\mathbf{w}_0\right\|_2^2$$

$$\overset{(i)}{=} \arg\min_{\mathbf{w}_0} \frac{1}{T} \sum_{t=1}^{T} \frac{1}{2n} \left\| \mathbf{X}_t \left(\mathbf{I}_d - (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top \mathbf{X}_t \right) \mathbf{w}_t - \mathbf{X}_t \left(\mathbf{I}_d - (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top \mathbf{X}_t \right) \mathbf{w}_0 \right\|_2^2$$

$$\overset{(ii)}{=} \arg\min_{\mathbf{w}_0} \frac{1}{T} \sum_{t=1}^{T} \frac{1}{2n} \left\| \lambda \mathbf{X}_t \left(\mathbf{X}_t^\top \mathbf{X}_t / n + \lambda \mathbf{I}_d \right)^{-1} (\mathbf{w}_t - \mathbf{w}_0) \right\|_2^2$$

$$= \left(\sum_{t=1}^{T} \lambda^2 \left(\frac{\mathbf{X}_t^\top \mathbf{X}_t}{n} + \lambda \mathbf{I}_d \right)^{-1} \frac{\mathbf{X}_t^\top \mathbf{X}_t}{n} \left(\frac{\mathbf{X}_t^\top \mathbf{X}_t}{n} + \lambda \mathbf{I}_d \right)^{-1} \right)^{-1}$$

$$\cdot \sum_{t=1}^{T} \lambda^2 \left(\frac{\mathbf{X}_t^\top \mathbf{X}_t}{n} + \lambda \mathbf{I}_d \right)^{-1} \frac{\mathbf{X}_t^\top \mathbf{X}_t}{n} \left(\frac{\mathbf{X}_t^\top \mathbf{X}_t}{n} + \lambda \mathbf{I}_d \right)^{-1} \mathbf{w}_t$$

$$\overset{(iii)}{=} \left(\sum_{t=1}^{T} \mathbf{A}_t \right)^{-1} \sum_{t=1}^{T} \mathbf{A}_t \mathbf{w}_t,$$

where step (i) invokes the data generating assumption $\mathbf{y}_t = \mathbf{W}_t \mathbf{w}_t$, step (ii) simplifies $\mathbf{I}_d - (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} \mathbf{X}_t^\top \mathbf{X}_t$ as $\lambda n (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)$ by writing $\mathbf{I}_d = (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1} (\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)$, and step (iii) follows from plugging in the definition of $\mathbf{A}_t$ in (19) and the fact that $(\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d)^{-1}$ and $\mathbf{X}_t^\top \mathbf{X}_t$ commute.

Next we consider the train-val method. The argument is analogous to the train-train method. In particular, we recall from (2):

$$\widehat{\mathbf{w}}_{0,T}^{\text{tr-val}} = \arg\min_{\mathbf{w}_0} \frac{1}{T} \sum_{t=1}^{T} \frac{1}{2n_2} \left\| \mathbf{y}_t^{\text{val}} - \mathbf{X}_t^{\text{val}} \left(\mathbf{w}_0 + ((\mathbf{X}_t^{\text{train}})^\top \mathbf{X}_t^{\text{train}} + n_1 \lambda \mathbf{I}_d)^{-1} (\mathbf{X}_t^{\text{train}})^\top (\mathbf{y}_t^{\text{train}} - \mathbf{X}_t^{\text{train}} \mathbf{w}_0) \right) \right\|_2^2 .$$

The optimization problem above is still quadratic in $\mathbf{w}_0$. Using the same rearrangement technique for the train-train method, we deduce

$$\widehat{\mathbf{w}}_{0,T}^{\text{tr-val}} = \arg\min_{\mathbf{w}_0} \frac{1}{T} \sum_{t=1}^{T} \frac{1}{2n_2} \left\| \lambda \mathbf{X}_t^{\text{val}} \left((\mathbf{X}_t^{\text{train}})^\top \mathbf{X}_t^{\text{train}} / n_1 + \lambda \mathbf{I}_d \right)^{-1} (\mathbf{w}_t - \mathbf{w}_0) \right\|_2^2$$

$$= \left(\sum_{t=1}^{T} \lambda^2 \left(\frac{(\mathbf{X}_t^{\text{train}})^\top \mathbf{X}_t^{\text{train}}}{n_1} + \lambda \mathbf{I}_d \right)^{-1} \frac{(\mathbf{X}_t^{\text{val}})^\top \mathbf{X}_t^{\text{val}}}{n_2} \left(\frac{(\mathbf{X}_t^{\text{train}})^\top \mathbf{X}_t^{\text{train}}}{n_1} + \lambda \mathbf{I}_d \right)^{-1} \right)^{-1}$$

$$\cdot \sum_{t=1}^{T} \lambda^2 \left(\frac{(\mathbf{X}_t^{\text{train}})^\top \mathbf{X}_t^{\text{train}}}{n_1} + \lambda \mathbf{I}_d \right)^{-1} \frac{(\mathbf{X}_t^{\text{val}})^\top \mathbf{X}_t^{\text{val}}}{n_2} \left(\frac{(\mathbf{X}_t^{\text{train}})^\top \mathbf{X}_t^{\text{train}}}{n_1} + \lambda \mathbf{I}_d \right)^{-1} \mathbf{w}_t$$

$$= \left(\sum_{t=1}^{T} \mathbf{B}_t \right)^{-1} \sum_{t=1}^{T} \mathbf{B}_t \mathbf{w}_t,$$

where the last equality follows by substituting the definition of matrix $\mathbf{B}_t$ in (20). The proof is complete. $\square$

C.2. Moments of $\mathbf{A}_t$ and $\mathbf{B}_t$

Lemma C.2 (Moments of $\mathbf{A}_t$ and $\mathbf{B}_t$). *Suppose $\mathbf{x}_{t,i} \overset{\text{iid}}{\sim} \mathsf{N}(0, \mathbf{I}_d)$ and $\mathbf{A}_t, \mathbf{B}_t$ are defined as in (19) and (20). Then, we have*

$$\mathbb{E}[\mathbf{A}_t] = f_A(n, d, \lambda) \cdot \mathbf{I}_d, \quad \mathbb{E}[\mathbf{A}_t^2] = f_{A^2}(n, d, \lambda) \cdot \mathbf{I}_d,$$

$$\mathbb{E}[\mathbf{B}_t] = f_B(n_1, n_2, d, \lambda) \cdot \mathbf{I}_d, \quad \mathbb{E}[\mathbf{B}_t^2] = f_{B^2}(n_1, n_2, d, \lambda) \cdot \mathbf{I}_d,$$

where

$$f_A(n, d, \lambda) := \frac{1}{d} \mathbb{E} \left[\operatorname{tr} \left(\lambda^2 (\widehat{\boldsymbol{\Sigma}}_n + \lambda \mathbf{I}_d)^{-2} \widehat{\boldsymbol{\Sigma}}_n \right) \right] = \frac{1}{d} \mathbb{E} \left[\sum_{i=1}^{d} \sigma_i^{(n)} \lambda^2 / (\sigma_i^{(n)} + \lambda)^2 \right],$$

$$f_{A^2}(n, d, \lambda) := \frac{1}{d} \mathbb{E} \left[\operatorname{tr} \left(\lambda^4 (\widehat{\boldsymbol{\Sigma}}_n + \lambda \mathbf{I}_d)^{-4} \widehat{\boldsymbol{\Sigma}}_n^2 \right) \right] = \frac{1}{d} \mathbb{E} \left[\sum_{i=1}^{d} (\sigma_i^{(n)})^2 \lambda^4 / (\sigma_i^{(n)} + \lambda)^4 \right],$$

$$f_B(n_1, n_2, d, \lambda) := \frac{1}{d}\mathbb{E}\left[\mathrm{tr}\left(\lambda^2(\widehat{\mathbf{\Sigma}}_{n_1} + \lambda\mathbf{I}_d)^{-2}\right)\right] = \frac{1}{d}\mathbb{E}\left[\sum_{i=1}^{d} \lambda^2/(\sigma_i^{(n_1)} + \lambda)^2\right],$$

$$f_{B^2}(n_1, n_2, d, \lambda) := \frac{1}{dn_2}\mathbb{E}\left[\mathrm{tr}\left(\lambda^2(\widehat{\mathbf{\Sigma}}_{n_1} + \lambda\mathbf{I}_d)^{-2}\right)^2 + (n_2+1)\mathrm{tr}\left(\lambda^4(\widehat{\mathbf{\Sigma}}_{n_1} + \lambda\mathbf{I}_d)^{-4}\right)\right]$$

$$= \frac{1}{dn_2}\mathbb{E}\left[\left(\sum_{i=1}^{d} \lambda^2/(\sigma_i^{(n_1)} + \lambda)^2\right)^2 + (n_2+1)\sum_{i=1}^{d} \lambda^4/(\sigma_i^{(n_1)} + \lambda)^4\right],$$

where $\widehat{\mathbf{\Sigma}}_n$ denotes the empirical covariance matrix $\mathbf{X}_t^\top \mathbf{X}_t/n$ where $\mathbf{X}_t \in \mathbb{R}^{n \times d}$ has i.i.d. $\mathsf{N}(0,1)$ entries, and $\sigma_1^{(n)} \geq \cdots \geq \sigma_d^{(n)} \geq 0$ is its eigenvalues.

Proof. The proof manipulates the isotropicity of $\mathbf{X}_t$. We begin with the first moment computation.

• **First moment of $\mathbf{A}_t$ and $\mathbf{B}_t$.** We rewrite $\mathbf{A}_t$ in a symmetric form to ease the analysis:

$$\mathbb{E}[\mathbf{A}_t] = \mathbb{E}\left[\left(\mathbf{I}_d - (\mathbf{X}_t^\top\mathbf{X}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{X}_t^\top\mathbf{X}_t\right)^\top \frac{\mathbf{X}_t^\top\mathbf{X}_t}{n}\left(\mathbf{I}_d - (\mathbf{X}_t^\top\mathbf{X}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{X}_t^\top\mathbf{X}_t\right)\right]$$

$$\overset{(i)}{=} \frac{1}{n}\mathbb{E}\left[\mathbf{V}_t\left(\mathbf{I}_d - (\mathbf{D}_t^\top\mathbf{D}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{D}_t^\top\mathbf{D}_t\right)^\top \mathbf{D}_t^\top\mathbf{D}_t\left(\mathbf{I}_d - (\mathbf{D}_t^\top\mathbf{D}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{D}_t^\top\mathbf{D}_t\right)\mathbf{V}_t^\top\right], \quad (21)$$

where the equality (i) is obtained by plugging in the SVD of $\mathbf{X}_t = \mathbf{U}_t\mathbf{D}_t\mathbf{V}_t^\top$ with $\mathbf{U}_t \in \mathbb{R}^{n \times n}$, $\mathbf{D}_t \in \mathbb{R}^{n \times d}$, and $\mathbf{V}_t \in \mathbb{R}^{d \times d}$. A key observation is that $\mathbf{U}_t$ and $\mathbf{V}_t$ are independent, since $\mathbf{X}_t$ is isotropic, i.e., homogeneous in each orthogonal direction. To see this, for any orthogonal matrices $\mathbf{Q} \in \mathbb{R}^{n \times n}$ and $\mathbf{P} \in \mathbb{R}^{d \times d}$, we know $\mathbf{X}_t$ and $\mathbf{Q}\mathbf{X}_t\mathbf{P}^\top$ share the same distribution. Moreover, we have $\mathbf{Q}\mathbf{X}_t\mathbf{P}^\top = (\mathbf{Q}\mathbf{U}_t)\mathbf{D}_t(\mathbf{P}\mathbf{V}_t)^\top$ as the SVD. This shows that the left and right singular matrices are independent and both uniformly distributed on all the orthogonal matrices of the corresponding dimensions ($\mathbb{R}^{n \times n}$ and $\mathbb{R}^{d \times d}$, respectively).

Recall that we denote $\sigma_1^{(n)} \geq \cdots \geq \sigma_d^{(n)}$ as the eigenvalues of $\frac{1}{n}\mathbf{X}_t^\top\mathbf{X}_t$. Thus, we have $\mathbf{D}_t^\top\mathbf{D}_t = \mathrm{Diag}(n\sigma_1^{(n)}, \ldots, n\sigma_d^{(n)})$. We can further simplify (21) as

$$\frac{1}{n}\mathbb{E}\left[\mathbf{V}_t\left(\mathbf{I}_d - (\mathbf{D}_t^\top\mathbf{D}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{D}_t^\top\mathbf{D}_t\right)^\top \mathbf{D}_t^\top\mathbf{D}_t\left(\mathbf{I}_d - (\mathbf{D}_t^\top\mathbf{D}_t + n\lambda\mathbf{I}_d)^{-1}\mathbf{D}_t^\top\mathbf{D}_t\right)\mathbf{V}_t^\top\right]$$

$$= \frac{1}{n}\mathbb{E}\left[\mathbf{V}_t\mathrm{Diag}\left(\frac{n\lambda^2\sigma_1^{(n)}}{(\sigma_1^{(n)} + \lambda)^2}, \ldots, \frac{n\lambda^2\sigma_d^{(n)}}{(\sigma_d^{(n)} + \lambda)^2}\right)\mathbf{V}_t^\top\right] \quad (22)$$

$$= \mathbb{E}\left[\sum_{i=1}^{d} \frac{\lambda^2\sigma_i^{(n)}}{(\sigma_i^{(n)} + \lambda)^2}\mathbf{v}_{t,i}\mathbf{v}_{t,i}^\top\right]. \quad (23)$$

We will utilize the isotropicity of $\mathbf{X}_t$ to find (23). Recall that we have shown that $\mathbf{V}_t$ is uniform on all the orthogonal matrices. Let $\mathbf{P} \in \mathbb{R}^{d \times d}$ be any permutation matrix, then $\mathbf{V}_t\mathbf{P}$ has the same distribution as $\mathbf{V}_t$. For this permuted data matrix $\mathbf{V}_t\mathbf{P}$, (23) becomes

$$\mathbb{E}\left[\sum_{i=1}^{d} \frac{\lambda^2\sigma_i^{(n)}}{(\sigma_i^{(n)} + \lambda)^2}\mathbf{v}_{t,\tau_p(i)}\mathbf{v}_{t,\tau_p(i)}^\top\right] \quad \text{with } \tau_p(i) \text{ denotes the permutation of the } i\text{-th element in } \mathbf{P}.$$

Summing over all the permutations $\mathbf{P}$ (and there are totally $d!$ instances), we deduce

$$d!\mathbb{E}[\mathbf{A}_t] = \sum_{\text{all permutation } \tau_p} \mathbb{E}\left[\sum_{i=1}^{d} \frac{\lambda^2\sigma_i^{(n)}}{(\sigma_i^{(n)} + \lambda)^2}\mathbf{v}_{t,\tau_p(i)}\mathbf{v}_{t,\tau_p(i)}^\top\right]$$

$$= (d-1)!\mathbb{E}\left[\sum_{j=1}^{d}\left[\sum_{i=1}^{d} \frac{\lambda^2\sigma_i^{(n)}}{(\sigma_i^{(n)} + \lambda)^2}\right]\mathbf{v}_{t,j}\mathbf{v}_{t,j}^\top\right]$$

$$= (d-1)! \mathbb{E}\left[\mathbf{V}_t \mathrm{Diag}\left(\sum_{i=1}^d \frac{\lambda^2 \sigma_i^{(n)}}{(\lambda + \sigma_i^{(n)})^2}, \dots, \sum_{i=1}^d \frac{\lambda^2 \sigma_i^{(n)}}{(\lambda + \sigma_i^{(n)})^2} \right) \mathbf{V}_t^\top \right]$$

$$= (d-1)! \mathbb{E}\left[\sum_{i=1}^d \frac{\lambda^2 \sigma_i^{(n)}}{(\lambda + \sigma_i^{(n)})^2} \mathbf{V}_t \mathbf{V}_t^\top \right]. \tag{24}$$

Dividing $(d-1)!$ on both sides of (24) yields

$$\mathbb{E}[\mathbf{A}_t] = \frac{1}{d} \mathbb{E}\left[\sum_{i=1}^d \frac{\lambda^2 \sigma_i^{(n)}}{(\lambda + \sigma_i^{(n)})^2} \right] \mathbf{I}_d. \tag{25}$$

Similar to the computation of $\mathbf{A}_t$, we compute $\mathbb{E}[\mathbf{B}_t]$ as follows.

$$\mathbb{E}[\mathbf{B}_t] = \mathbb{E}\left[\left(\mathbf{I}_d - ((\mathbf{X}_t^{\mathsf{train}})^\top \mathbf{X}_t^{\mathsf{train}} + n_1 \lambda \mathbf{I}_d)^{-1} (\mathbf{X}_t^{\mathsf{train}})^\top \mathbf{X}_t^{\mathsf{train}} \right)^\top \frac{(\mathbf{X}_t^{\mathsf{val}})^\top \mathbf{X}_t^{\mathsf{val}}}{n_2} \right.$$
$$\left. \cdot \left(\mathbf{I}_d - ((\mathbf{X}^{\mathsf{train}})_t^\top \mathbf{X}_t^{\mathsf{train}} + n_1 \lambda \mathbf{I}_d)^{-1} (\mathbf{X}_t^{\mathsf{train}})^\top \mathbf{X}_t^{\mathsf{train}} \right) \right]$$
$$\overset{(i)}{=} \mathbb{E}\left[\left(\mathbf{I}_d - ((\mathbf{X}_t^{\mathsf{train}})^\top \mathbf{X}_t^{\mathsf{train}} + n_1 \lambda \mathbf{I}_d)^{-1} ((\mathbf{X}_t^{\mathsf{train}})^\top \mathbf{X}_t^{\mathsf{train}} \right)^\top \right.$$
$$\left. \cdot \left(\mathbf{I}_d - ((\mathbf{X}^{\mathsf{train}})_t^\top \mathbf{X}_t^{\mathsf{train}} + n_1 \lambda \mathbf{I}_d)^{-1} (\mathbf{X}_t^{\mathsf{train}})^\top \mathbf{X}_t^{\mathsf{train}} \right) \right]$$
$$\overset{(ii)}{=} \mathbb{E}\left[\mathbf{V}_t^{\mathsf{train}} \left(\mathbf{I}_d - ((\mathbf{D}_t^{\mathsf{train}})^\top \mathbf{D}_t^{\mathsf{train}} + n_1 \lambda \mathbf{I}_d)^{-1} (\mathbf{D}_t^{\mathsf{train}})^\top \mathbf{D}_t^{\mathsf{train}} \right)^2 (\mathbf{V}_t^{\mathsf{train}})^\top \right], \tag{26}$$

where (i) uses the data generating assumption $\mathbb{E}[(\mathbf{X}_t^{\mathsf{val}})^\top \mathbf{X}_t^{\mathsf{val}}] = n_2 \mathbf{I}_d$ and the independence between $\mathbf{X}_t^{\mathsf{train}}$ and $\mathbf{X}_t^{\mathsf{val}}$, and (ii) follows from the SVD of $\mathbf{X}_t^{\mathsf{train}} = \mathbf{U}_t^{\mathsf{train}} \mathbf{D}_t^{\mathsf{train}} (\mathbf{V}_t^{\mathsf{train}})^\top$.

Here we denote $\sigma_1^{(n_1)} \geq \cdots \geq \sigma_d^{(n_1)}$ as the eigenvalues of $\frac{1}{n_1}(\mathbf{X}_t^{\mathsf{train}})^\top \mathbf{X}_t^{\mathsf{train}}$. Thus, we have $(\mathbf{D}_t^{\mathsf{train}})^\top \mathbf{D}_t^{\mathsf{train}} = \mathrm{Diag}(n_1 \sigma_1^{(n_1)}, \dots, n_1 \sigma_d^{(n_1)})$. We can now further simplify (26) as

$$\mathbb{E}\left[\mathbf{V}_t^{\mathsf{train}} \left(\mathbf{I}_d - ((\mathbf{D}_t^{\mathsf{train}})^\top \mathbf{D}_t^{\mathsf{train}} + n_1 \lambda \mathbf{I}_d)^{-1} (\mathbf{D}_t^{\mathsf{train}})^\top \mathbf{D}_t^{\mathsf{train}} \right)^2 (\mathbf{V}_t^{\mathsf{train}})^\top \right]$$
$$\overset{(i)}{=} \mathbb{E}\left[\mathbf{V}_t^{\mathsf{train}} \mathrm{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)} + \lambda)^2}, \dots, \frac{\lambda^2}{(\sigma_d^{(n_1)} + \lambda)^2} \right) (\mathbf{V}_t^{\mathsf{train}})^\top \right] \tag{27}$$
$$\overset{(ii)}{=} \frac{1}{d} \mathbb{E}\left[\sum_{i=1}^d \frac{\lambda^2}{(\lambda + \sigma_i^{(n_1)})^2} \right] \mathbf{I}_d. \tag{28}$$

Step (i) follows from the same computation in (22), and step (ii) uses the permutation trick in (24).

● **Second moment of $\mathbf{A}_t$ and $\mathbf{B}_t$.** Using the SVD of $\mathbf{X}_t$ and the computation in (22), we derive

$$\mathbb{E}[\mathbf{A}_t^2] = \mathbb{E}\left[\mathbf{V}_t \mathrm{Diag}\left(\frac{\lambda^2 \sigma_1^{(n)}}{(\sigma_1^{(n)} + \lambda)^2}, \dots, \frac{\lambda^2 \sigma_d^{(n)}}{(\sigma_d^{(n)} + \lambda)^2} \right) \mathbf{V}_t^\top \cdot \mathbf{V}_t \mathrm{Diag}\left(\frac{\lambda^2 \sigma_1^{(n)}}{(\sigma_1^{(n)} + \lambda)^2}, \dots, \frac{\lambda^2 \sigma_d^{(n)}}{(\sigma_d^{(n)} + \lambda)^2} \right) \mathbf{V}_t^\top \right]$$
$$= \mathbb{E}\left[\mathbf{V}_t \mathrm{Diag}\left(\frac{\lambda^4 (\sigma_1^{(n)})^2}{(\sigma_1^{(n)} + \lambda)^4}, \dots, \frac{\lambda^4 (\sigma_d^{(n)})^2}{(\sigma_d^{(n)} + \lambda)^4} \right) \mathbf{V}_t^\top \right]$$
$$\overset{(i)}{=} \frac{1}{d} \mathbb{E}\left[\mathrm{Diag}\left(\frac{\lambda^4 (\sigma_1^{(n)})^2}{(\sigma_1^{(n)} + \lambda)^4}, \dots, \frac{\lambda^4 (\sigma_d^{(n)})^2}{(\sigma_d^{(n)} + \lambda)^4} \right) \right] \mathbf{I}_d,$$

where step (i) applies the permutation trick in (24) to tackle $\mathbb{E}\left[\mathbf{V}_t \mathrm{Diag}\left(\frac{\lambda^4 (\sigma_1^{(n)})^2}{(\sigma_1^{(n)} + \lambda)^4}, \dots, \frac{\lambda^4 (\sigma_d^{(n)})^2}{(\sigma_d^{(n)} + \lambda)^4} \right) \mathbf{V}_t^\top \right]$.

For $\mathbb{E}[\mathbf{B}_t^2]$, the computation is a bit more complex. Using the SVD of $\mathbf{X}_t^{\text{train}}$ as in (27), we obtain

$$
\mathbb{E}[\mathbf{B}_t^2] = \frac{1}{n_2^2}\mathbb{E}\bigg[\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda}{\sigma_1^{(n_1)}+\lambda},\ldots,\frac{\lambda}{\sigma_d^{(n_1)}+\lambda}\right)(\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top
$$
$$
\cdot \mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)}+\lambda)^2},\ldots,\frac{\lambda^2}{(\sigma_d^{(n_1)}+\lambda)^2}\right)(\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top
$$
$$
\cdot \mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda}{\sigma_1^{(n_1)}+\lambda},\ldots,\frac{\lambda}{\sigma_d^{(n_1)}+\lambda}\right)(\mathbf{V}_t^{\text{train}})^\top\bigg]. \tag{29}
$$

We claim that $\mathbb{E}[\mathbf{B}_t^2]$ is diagonal. To see this, we take expectation with respect to $\mathbf{X}_t^{\text{val}}$ first in (29). Since $\mathbf{V}_t^{\text{train}}$ is an orthogonal matrix, $\mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}$ has the same distribution as $\mathbf{X}_t^{\text{val}}$ and independent of $\mathbf{X}_t$. We verify that any off-diagonal element is zero in the following matrix

$$
\mathbf{T} := \mathbb{E}_{\mathbf{X}_t^{\text{val}}}\bigg[(\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top\mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)}+\lambda)^2},\ldots,\frac{\lambda^2}{(\sigma_d^{(n_1)}+\lambda)^2}\right)
$$
$$
\cdot (\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top\mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\bigg].
$$

We denote $\mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}} = [\mathbf{x}_1,\ldots,\mathbf{x}_n]^\top \in \mathbb{R}^{n_2\times d}$ with $\mathbf{x}_i \overset{\text{iid}}{\sim} \mathsf{N}(\mathbf{0},\mathbf{I}_d)$. For $k \neq \ell$, the (k,ℓ)-th entry $T_{k,\ell}$ of $\mathbf{T}$ is

$$
T_{k,\ell} = \mathbb{E}\left[\sum_j \left(\frac{\lambda^2}{(\sigma_j^{(n_1)}+\lambda)^2}\left(\sum_i x_{k,i}x_{j,i}\right)\left(\sum_i x_{j,i}x_{\ell,i}\right)\right)\right]
$$
$$
= \mathbb{E}\left[\sum_j \frac{\lambda^2}{(\sigma_j^{(n_1)}+\lambda)^2}\left(\sum_{m,n} x_{k,m}x_{j,m}x_{j,n}x_{\ell,n}\right)\right]
$$
$$
\overset{(i)}{=} 0,
$$

where $x_{i,j}$ denotes the j-th element of $\mathbf{x}_i$. Equality (i) holds, since either $x_{k,m}$ or $x_{\ell,n}$ only appears once in each summand. Therefore, we can write $\mathbf{T} = \text{Diag}\left(T_{1,1},\ldots,T_{d,d}\right)$ with $T_{k,k}$ being

$$
T_{k,k} = \mathbb{E}\left[\sum_j \frac{\lambda^2}{(\sigma_j^{(n_1)}+\lambda)^2}\left(\sum_{m,n} x_{k,m}x_{j,m}x_{j,n}x_{\ell,n}\right)\right]
$$
$$
= \mathbb{E}\left[\frac{\lambda^2}{(\sigma_k^{(n_1)}+\lambda)^2}\left(\sum_{m,n} x_{k,m}x_{k,m}x_{k,n}x_{k,n}\right)\right].
$$

Observe that $T_{k,k}$ only depends on $\sigma_k^{(n_1)}$. Plugging back into (29), we have

$$
\mathbb{E}[\mathbf{B}_t^2] = \frac{1}{n_2^2}\mathbb{E}\bigg[\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda}{\sigma_1^{(n_1)}+\lambda},\ldots,\frac{\lambda}{\sigma_d^{(n_1)}+\lambda}\right)(\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top
$$
$$
\cdot \mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)}+\lambda)^2},\ldots,\frac{\lambda^2}{(\sigma_d^{(n_1)}+\lambda)^2}\right)(\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top
$$
$$
\cdot \mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda}{\sigma_1^{(n_1)}+\lambda},\ldots,\frac{\lambda}{\sigma_d^{(n_1)}+\lambda}\right)(\mathbf{V}_t^{\text{train}})^\top\bigg]
$$
$$
= \frac{1}{n_2^2}\mathbb{E}\bigg[\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda}{\sigma_1^{(n_1)}+\lambda},\ldots,\frac{\lambda}{\sigma_d^{(n_1)}+\lambda}\right)\text{Diag}(T_{1,1},\ldots,T_{d,d})
$$

$$\cdot \, \text{Diag}\left(\frac{\lambda}{\sigma_1^{(n_1)} + \lambda}, \dots, \frac{\lambda}{\sigma_d^{(n_1)} + \lambda}\right)(\mathbf{V}_t^{\text{train}})^\top\Bigg]$$

$$= \frac{1}{n_2^2}\mathbb{E}\left[\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda^2 T_{1,1}}{(\sigma_1^{(n_1)} + \lambda)^2}, \dots, \frac{\lambda^2 T_{d,d}}{(\sigma_d^{(n_1)} + \lambda)^2}\right)(\mathbf{V}_t^{\text{train}})^\top\right]$$

$$\overset{(i)}{=} c\mathbf{I}_d, \tag{30}$$

where equality (i) utilizes the permutation trick in (25). To this end, it suffices to find c as

$$c = \frac{1}{d}\mathbb{E}[\mathbf{B}_t^2]$$

$$= \frac{1}{dn_2^2}\text{tr}\Bigg(\mathbb{E}\Bigg[\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda}{\sigma_1^{(n_1)} + \lambda}, \dots, \frac{\lambda}{\sigma_d^{(n_1)} + \lambda}\right)(\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top$$

$$\cdot \, \mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)} + \lambda)^2}, \dots, \frac{\lambda^2}{(\sigma_d^{(n_1)} + \lambda)^2}\right)(\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top$$

$$\cdot \, \mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda}{\sigma_1^{(n_1)} + \lambda}, \dots, \frac{\lambda}{\sigma_d^{(n_1)} + \lambda}\right)(\mathbf{V}_t^{\text{train}})^\top\Bigg]\Bigg)$$

$$= \frac{1}{dn_2^2}\text{tr}\Bigg(\mathbb{E}\Bigg[\mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)} + \lambda)^2}, \dots, \frac{\lambda^2}{(\sigma_d^{(n_1)} + \lambda)^2}\right)(\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top$$

$$\cdot \, \mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}}\text{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)} + \lambda)^2}, \dots, \frac{\lambda^2}{(\sigma_d^{(n_1)} + \lambda)^2}\right)(\mathbf{V}_t^{\text{train}})^\top(\mathbf{X}_t^{\text{val}})^\top\Bigg]\Bigg). \tag{31}$$

Observe again that $\mathbf{X}_t^{\text{val}}\mathbf{V}_t^{\text{train}} \in \mathbb{R}^{n_2 \times d}$ is a Gaussian random matrix. We rewrite (31) as

$$c = \frac{1}{dn_2^2}\mathbb{E}\left[\left(\sum_{i,j=1}^{n_2}\mathbf{v}_i^\top\text{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)} + \lambda)^2}, \dots, \frac{\lambda^2}{(\sigma_d^{(n_1)} + \lambda)^2}\right)\mathbf{v}_j\right)^2\right], \tag{32}$$

where $\mathbf{v}_i \overset{\text{iid}}{\sim} \mathsf{N}(\mathbf{0}, \mathbf{I}_d)$ is i.i.d. Gaussian random vectors for $i = 1, \dots, n_2$. To compute (32), we need the following result.

Claim C.1. *Given any symmetric matrix* $\mathbf{T} \in \mathbb{R}^{d \times d}$ *and i.i.d. standard Gaussian random vectors* $\mathbf{v}, \mathbf{u} \overset{\text{iid}}{\sim} \mathsf{N}(\mathbf{0}, \mathbf{I}_d)$, *we have*

$$\mathbb{E}\left[(\mathbf{v}^\top\mathbf{T}\mathbf{v})^2\right] = 2\|\mathbf{T}\|_{\mathsf{Fr}}^2 + \text{tr}^2(\mathbf{T}) \quad \text{and} \tag{33}$$

$$\mathbb{E}\left[(\mathbf{v}^\top\mathbf{T}\mathbf{u})^2\right] = \|\mathbf{T}\|_{\mathsf{Fr}}^2. \tag{34}$$

Proof of Claim C.1. We show (33) first. We denote $T_{i,j}$ as the (i,j)-th element of $\mathbf{T}$ and v_i as the i-th element of $\mathbf{v}$. Expanding the quadratic form, we have

$$\mathbb{E}\left[(\mathbf{v}^\top\mathbf{T}\mathbf{v})^2\right] = \mathbb{E}\left[\sum_{i,j,k,\ell \leq d} v_i v_j v_k v_\ell T_{i,j} T_{k,\ell}\right]$$

$$= \mathbb{E}\left[\sum_{i \leq d} v_i^4 T_{i,i}^2\right] + \mathbb{E}\left[\sum_{i \neq j} v_i^2 v_j^2 (T_{i,j}^2 + T_{i,i}T_{j,j} + T_{i,j}T_{j,i})\right]$$

$$= 3\sum_{i \leq d} T_{i,i}^2 + \sum_{i \neq j}(T_{i,j}^2 + T_{i,i}T_{j,j} + T_{i,j}T_{j,i})$$

$$= \text{tr}^2(\mathbf{T}) + 2\sum_{i \leq d} T_{i,i}^2 + \sum_{i \neq j}(T_{i,j}^2 + T_{i,j}T_{j,i})$$

$$= \text{tr}^2(\mathbf{T}) + 2\|\mathbf{T}\|_{\mathsf{Fr}}^2.$$

Next, we show (34) by the cyclic property of race.

$$\mathbb{E}\left[(\mathbf{v}^\top \mathbf{T}\mathbf{u})^2\right] = \operatorname{tr}\left(\mathbb{E}\left[\mathbf{u}\mathbf{u}^\top \mathbf{T}\mathbf{v}\mathbf{v}^\top \mathbf{T}\right]\right) = \operatorname{tr}(\mathbf{T}^2) = \|\mathbf{T}\|_{\mathsf{Fr}}^2.$$

$\square$

We back to the computation of (32) using Claim C.1.

$$
\begin{aligned}
c &= \frac{1}{dn_2^2}\mathbb{E}\left[\sum_{i,j=1}^{n_2}\left(\mathbf{v}_i^\top \operatorname{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)}+\lambda)^2},\cdots,\frac{\lambda^2}{(\sigma_d^{(n_1)}+\lambda)^2}\right)\mathbf{v}_j\right)^2\right] \\
&= \frac{1}{dn_2^2}\mathbb{E}\left[\sum_{i=1}^{n_2}\left(\mathbf{v}_i^\top \operatorname{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)}+\lambda)^2},\cdots,\frac{\lambda^2}{(\sigma_d^{(n_1)}+\lambda)^2}\right)\mathbf{v}_i\right)^2\right] \\
&\quad + \frac{1}{dn_2^2}\mathbb{E}\left[\sum_{i\neq j}\left(\mathbf{v}_i^\top \operatorname{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)}+\lambda)^2},\cdots,\frac{\lambda^2}{(\sigma_d^{(n_1)}+\lambda)^2}\right)\mathbf{v}_j\right)^2\right] \\
&= \frac{1}{dn_2}\mathbb{E}\left[\operatorname{tr}^2\left(\operatorname{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)}+\lambda)^2},\cdots,\frac{\lambda^2}{(\sigma_d^{(n_1)}+\lambda)^2}\right)\right)\right] \\
&\quad + \frac{2}{dn_2}\mathbb{E}\left[\left\|\operatorname{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)}+\lambda)^2},\cdots,\frac{\lambda^2}{(\sigma_d^{(n_1)}+\lambda)^2}\right)\right\|_{\mathsf{Fr}}^2\right] \\
&\quad + \frac{n_2-1}{dn_2}\mathbb{E}\left[\left\|\operatorname{Diag}\left(\frac{\lambda^2}{(\sigma_1^{(n_1)}+\lambda)^2},\cdots,\frac{\lambda^2}{(\sigma_d^{(n_1)}+\lambda)^2}\right)\right\|_{\mathsf{Fr}}^2\right] \\
&= \frac{1}{dn_2}\left(\mathbb{E}\left[\sum_{i=1}^d \frac{\lambda^2}{(\sigma_i^{(n_1)}+\lambda)^2}\right]^2 + (n_2+1)\mathbb{E}\left[\sum_{i=1}^d \frac{\lambda^4}{(\sigma_i^{(n_1)}+\lambda)^4}\right]\right).
\end{aligned}
\tag{35}
$$

Substituting the value of c in (35) into (30), we derive the desired result

$$\mathbb{E}[\mathbf{B}_t^2] = \frac{1}{dn_2}\left(\mathbb{E}\left[\sum_{i=1}^d \frac{\lambda^2}{(\sigma_i^{(n_1)}+\lambda)^2}\right]^2 + (n_2+1)\mathbb{E}\left[\sum_{i=1}^d \frac{\lambda^4}{(\sigma_i^{(n_1)}+\lambda)^4}\right]\right).$$

The proof is complete. $\square$

Lemma C.3 (Bounds on moments). *Let $f_{\{A,A^2\}}(n,d,\lambda)$ and $f_{\{B,B^2\}}(n_1,n_2,d,\lambda)$ be defined as in Lemma C.2. Suppose $d/n = \gamma = \Theta(1)$ and $n_1/n = s = \Theta(1)$, $\lambda = \Theta(1) > 0$. Then we have*

$$\underline{c}_A \leq f_A(n,d,\lambda), f_{A^2}(n,d,\lambda) \leq \overline{c}_A,$$

and

$$\underline{c}_B \leq f_B(n_1,n_2,d,\lambda), f_{B^2}(n_1,n_2,d,\lambda) \leq \overline{c}_B,$$

where $\underline{c}_A, \overline{c}_A > 0$ depend only on γ, λ but not d, and $\underline{c}_B, \overline{c}_B > 0$ depend only on γ, s, λ but not d.

Proof. The upper bounds follow straightforwardly from the closed-form expressions established in Lemma C.2: we have

$$f_A(n,d,\lambda) = \frac{1}{d}\mathbb{E}\left[\operatorname{tr}\left(\lambda^2\left(\widehat{\boldsymbol{\Sigma}}_n + \lambda \mathbf{I}_d\right)^{-2}\widehat{\boldsymbol{\Sigma}}_n\right)\right] \leq \frac{1}{d}\mathbb{E}\left[\operatorname{tr}\left(\widehat{\boldsymbol{\Sigma}}_n\right)\right] = 1.$$

$$f_{A^2}(n,d,\lambda) = \frac{1}{d}\mathbb{E}\left[\operatorname{tr}\left(\lambda^4\left(\widehat{\boldsymbol{\Sigma}}_n + \lambda \mathbf{I}_d\right)^{-2}\widehat{\boldsymbol{\Sigma}}_n^2\right)\right] \leq \frac{1}{d}\mathbb{E}\left[\operatorname{tr}\left(\widehat{\boldsymbol{\Sigma}}_n^2\right)\right] \overset{(i)}{=} \frac{1}{d}\cdot\left(\frac{d^2+(n+1)d}{n}\right) = \frac{d+n+1}{n} \leq 2+\gamma,$$

where (i) used the fact that

$$\mathbb{E}\left[\mathrm{tr}\left(\widehat{\mathbf{\Sigma}}_n^2\right)\right] = \mathbb{E}\left[\left\|\widehat{\mathbf{\Sigma}}_n\right\|_{\mathsf{Fr}}^2\right] = \frac{1}{n^2}\cdot\mathbb{E}\left[\sum_{i,j=1}^n (\mathbf{x}_i^\top\mathbf{x}_j)^2\right] = \frac{1}{n^2}\cdot\left[n(n-1)d + n(d^2+2d)\right] = \frac{d^2+(n+1)d}{n}.$$

Therefore, we can take $\bar{c}_A = 2 + \gamma$. Similarly, we have

$$f_B(n_1, n_2, d, \lambda) = \frac{1}{d}\mathbb{E}\left[\sum_{i=1}^d \lambda^2/(\sigma_i^{(n_1)} + \lambda)^2\right] \le 1,$$

$$f_{B^2}(n_1, n_2, d, \lambda) = \frac{1}{dn_2}\mathbb{E}\left[\left(\sum_{i=1}^d \lambda^2/(\sigma_i^{(n_1)} + \lambda)^2\right)^2 + (n_2+1)\sum_{i=1}^d \lambda^4/(\sigma_i^{(n_1)} + \lambda)^4\right]$$

$$\le \frac{1}{dn_2}\left[d^2 + d(n_2+1)\right] = \frac{d + n_2 + 1}{n_2} = \frac{d + n(1-s) + 1}{n(1-s)} \le \frac{\gamma + 2 - s}{1 - s}.$$

Therefore we can take $\bar{c}_B = (2 + \gamma - s)/(1 - s)$.

For the lower bounds, it suffices to prove the lower bounds for $f_A(n, d, \lambda)$ and $f_B(n_1, n_2, d, \lambda)$ (as $\frac{1}{d}\mathrm{tr}(\mathbf{M}^2) \ge (\frac{1}{d}\mathrm{tr}(\mathbf{M}))^2$ always holds for any PSD matrix $\mathbf{M} \in \mathbb{R}^{d\times d}$). For this we apply the same Stieltjes calculation as in the proof of Theorem D.1 to conclude that

$$\lim_{d,n\to\infty, d/n\to\gamma} f_A(n, d, \lambda) = \frac{\lambda^2}{2\gamma}\left(\frac{\lambda + 1 + \gamma}{\sqrt{(\lambda + 1 + \gamma)^2 - 4\gamma}} - 1\right) > 0.$$

Also note that $f_A(\lceil d/\gamma\rceil, d, \lambda) > 0$ for any $d \ge 1$. Therefore, we have

$$\inf_{d\ge 1} f_A(\lceil d/\gamma\rceil, d, \lambda) := \widetilde{c}_A > 0.$$

Taking $\underline{c}_A = \min\left\{\widetilde{c}_A, \widetilde{c}_A^2\right\} > 0$ (which only depends on γ, we get $\min\left\{f_A(n, d, \lambda), f_{A^2}(n, d, \lambda)\right\} \ge \underline{c}_A$, the desired result. Similarly, for f_B, we have

$$\lim_{d,n_1\to\infty, d/n_1\to\gamma/s} \frac{1}{d}\mathbb{E}\left[\mathrm{tr}\left(\lambda^2(\lambda\mathbf{I}_d + \widehat{\mathbf{\Sigma}}_{n_1})^{-2}\right)\right] = \lambda^2\cdot\left[-\frac{d}{d\lambda_1}s(\lambda_1, \lambda_2)|_{\lambda_1=\lambda,\lambda_2=1}\right],$$

$$= \lambda^2\cdot\frac{1}{4\gamma/s\cdot\lambda^2\sqrt{(\lambda + 1 + \gamma/s)^2 - 4\gamma/s}}\left[2(\gamma - 1)\sqrt{(\lambda + 1 + \gamma/s)^2 - 4\gamma/s} + 2\lambda(1 + \gamma) + 2(1 - \gamma)^2\right] > 0,$$

where $s(\lambda_1, \lambda_2)$ is the generalized Stieltjes transform defined in (40). (The detailed calculation can be found in Section C.2.1.) As this limit is strictly positive, ssing a similar argument as the above, we get that there exists some $\underline{c}_B > 0$ which only depends on $\gamma/s = \Theta(1) > 0$, such that $\min\left\{f_B(n_1, n_2, d, \lambda), f_{B^2}(n_1, n_2, d, \lambda)\right\} \ge \underline{c}_B$. $\qquad\square$

C.2.1. CALCULATIONS OF $\frac{d}{d\lambda_1}s(\lambda_1, \lambda_2)$

Recall by (40) that

$$s(\lambda_1, \lambda_2) = \frac{\gamma - 1 - \lambda_1/\lambda_2 + \sqrt{(\lambda_1/\lambda_2 + 1 + \gamma)^2 - 4\gamma}}{2\gamma\lambda_1}.$$

At $\lambda_1 = \lambda$ and $\lambda_2 = 1$, the above can be simplied as

$$s(\lambda, 1) = \frac{\gamma - 1}{2\gamma\lambda} - \frac{1}{2\gamma} + \frac{1}{2\gamma}\sqrt{(1 + (1 + \gamma)/\lambda)^2 - 4\gamma/\lambda^2}.$$

Differentiating with respect to λ, we get

$$-\frac{d}{d\lambda_1}s(\lambda_1, \lambda_2)|_{\lambda_1=\lambda,\lambda_2=1} = -\frac{d}{d\lambda}s(\lambda, 1) = \frac{\gamma - 1}{2\gamma\lambda^2} + \frac{(\lambda + 1 + \gamma)(1 + \gamma) - 4\gamma}{2\gamma\lambda^2\sqrt{(\lambda + 1 + \gamma)^2 - 4\gamma}}$$

$$= \frac{1}{2\gamma\lambda^2\sqrt{(\lambda+1+\gamma)^2-4\gamma}}\Big[(\gamma-1)\sqrt{(\lambda+1+\gamma)^2-4\gamma}+\lambda(1+\gamma)+(\gamma-1)^2\Big].$$

The above is clearly positive at $\gamma \geq 1$. At $\gamma < 1$, we have

$$(1-\gamma)\sqrt{(\lambda+1+\gamma)^2-4\gamma} = \sqrt{\lambda^2+2\lambda(1+\gamma)+(1-\gamma)^2}$$

$$< (1-\gamma)\left(\sqrt{(\lambda(1+\gamma)/(1-\gamma))^2+2\lambda(1+\gamma)+(1-\gamma)^2}\right) = (1-\gamma)(\lambda(1+\gamma)/(1-\gamma)+1-\gamma)$$

$$= \lambda(1+\gamma)+(1-\gamma)^2.$$

Therefore, the quantity inside the bracket in the preceding display is also strictly positive. This shows that $-\frac{d}{d\lambda_1}s(\lambda_1,\lambda_2)|_{\lambda_1=\lambda,\lambda_2=1} > 0$ for all $\gamma > 0$.

C.3. Concentration lemmas

Lemma C.4 (Concentration of $\mathbf{A}_t$ and $\mathbf{B}_t$). *Let $\mathbf{A}_t$ and $\mathbf{B}_t$ be defined as in (19) and (20). Then with probability at least $1 - Td^{-10}$, we have the following bounds: $\mathbf{0} \preceq \mathbf{A}_t \preceq \widetilde{O}(C_a)\mathbf{I}_d$ and $\mathbf{0} \preceq \mathbf{B}_t \preceq \widetilde{O}(C_b)\mathbf{I}_d$, and*

$$\left\|\frac{1}{T}\sum_{t=1}^{T}\mathbf{A}_t - \mathbb{E}[\mathbf{A}_t]\right\|_{\mathrm{op}} \leq \widetilde{O}\left(C_a\sqrt{\frac{d}{T}}+d^{-4}\right) \quad \text{and} \quad \left\|\frac{1}{T}\sum_{t=1}^{T}\mathbf{A}_t^2 - \mathbb{E}\big[\mathbf{A}_t^2\big]\right\|_{\mathrm{op}} \leq \widetilde{O}\left(C_a^2\sqrt{\frac{d}{T}}+d^{-4}\right),$$

$$\left\|\frac{1}{T}\sum_{t=1}^{T}\mathbf{B}_t - \mathbb{E}[\mathbf{B}_t]\right\|_{\mathrm{op}} \leq \widetilde{O}\left(C_b\sqrt{\frac{d}{T}}+d^{-4}\right) \quad \text{and} \quad \left\|\frac{1}{T}\sum_{t=1}^{T}\mathbf{B}_t^2 - \mathbb{E}\big[\mathbf{B}_t^2\big]\right\|_{\mathrm{op}} \leq \widetilde{O}\left(C_b^2\sqrt{\frac{d}{T}}+d^{-4}\right),$$

where $C_a := 1 + \max\left\{d/n, \sqrt{d/n}\right\}$, $C_b := 1 + \max\left\{d/n_2, \sqrt{d/n_2}\right\}$, and $\widetilde{O}(\cdot)$ hides the logarithmic factor $\log(ndT)$.

Proof. We first prove the result for $\mathbf{B}_t$. We use a truncation argument. Recall that by definition of $\mathbf{B}_t$ we have

$$\mathbf{0} \preceq \mathbf{B}_t \preceq \frac{\mathbf{X}_t^{\mathsf{val}\top}\mathbf{X}_t^{\mathsf{val}}}{n_2}.$$

Since $\mathbf{x}_{t,i} \sim \mathsf{N}(\mathbf{0}, \mathbf{I}_d)$, applying the standard sub-Gaussian covariance concentration (Vershynin, 2018, Exercise 4.7.3), we have with probability at least $1 - d^{-10}$ that

$$\mathbf{B}_t \preceq \frac{1}{n_2}\mathbf{X}_t^{\mathsf{val}\top}\mathbf{X}_t^{\mathsf{val}} \preceq \mathbf{I}_d + \left\|\frac{1}{n_2}\mathbf{X}_t^{\mathsf{val}\top}\mathbf{X}_t^{\mathsf{val}} - \mathbf{I}_d\right\|_{\mathrm{op}}\mathbf{I}_d \preceq \left(1 + C\sqrt{\frac{d+\log d}{n_2}} + C\frac{d+\log d}{n_2}\right)\mathbf{I}_d \preceq KC_b\mathbf{I}_d,$$

where $C_b := 1 + \max\left\{d/n_2, \sqrt{d/n_2}\right\}$ and $K = O(1)$ is an absolute constant. Let $\mathbf{E}_t := \{\mathbf{B}_t \preceq KC_b\mathbf{I}_d\}$ denote this event. We have $\mathbb{P}(\mathbf{E}_t) \geq 1 - d^{-10}$. Let $\mathbf{E} := \bigcup_{t=1}^{T}\mathbf{E}_t$ denote the union event. Note that on the event $\mathbf{E}$ we have

$$\frac{1}{T}\sum_{t=1}^{T}\mathbf{B}_t = \frac{1}{T}\sum_{t=1}^{T}\mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\}.$$

Concentration of $\mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\}$ On the event $\mathbf{E}_t$, $\mathbf{B}_t$ are bounded matrices:

$$\mathbf{0} \preceq \mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\} \preceq C_b\mathbf{I}_d.$$

In particular, this means that for any $\mathbf{v} \in \mathbb{R}^d$ with unit norm $\|\mathbf{v}\|_2 = 1$, the random variable

$$\mathbf{v}^\top\mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\}\mathbf{v} - \mathbf{v}^\top\mathbb{E}[\mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\}]\mathbf{v}$$

is mean-zero and C^2-sub-Gaussian. Therefore by the standard sub-Gaussian concentration, we get

$$\mathbb{P}\left(\left|\mathbf{v}^\top\left(\frac{1}{T}\sum_{t=1}^{T}\mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\}\right)\mathbf{v} - \mathbf{v}^\top\mathbb{E}[\mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\}]\mathbf{v}\right| \geq t\right) \leq 2\exp\left(-Tt^2/C_b^2\right).$$

Using the fact that for any symmetric matrix $\mathbf{M}$,

$$\|\mathbf{M}\|_{\text{op}} \leq 2 \sup_{\mathbf{v} \in N_{1/4}(\mathbb{S}^{d-1})} \left|\mathbf{v}^\top \mathbf{M} \mathbf{v}\right|,$$

where $N_{1/4}(\mathbb{S}^{d-1})$ is a $1/4$-covering set of the unit sphere with $|N_{1/4}(\mathbb{S}^{d-1})| \leq 9^d$ (Vershynin, 2018, Exercise 4.4.3), we get

$$\mathbb{P}\left(\left\|\frac{1}{T}\sum_{t=1}^T \mathbf{B}_t \mathbf{1}\{\mathbf{E}_t\} - \mathbb{E}[\mathbf{B}_t \mathbf{1}\{\mathbf{E}_t\}]\right\|_{\text{op}} \geq t\right) \leq |N_{1/4}(\mathbb{S}^{d-1})| \cdot \sup_{\|\mathbf{v}\|_2 = 1} \mathbb{P}\left(\left|\mathbf{v}^\top\left(\frac{1}{T}\sum_{t=1}^T \mathbf{B}_t \mathbf{1}\{\mathbf{E}_t\}\right)\mathbf{v} - \mathbf{v}^\top \mathbb{E}[\mathbf{B}_t \mathbf{1}\{\mathbf{E}_t\}]\mathbf{v}\right| \geq t\right)$$

$$\leq \exp\left(-Tt^2/C_b^2 + 3d\right).$$

Taking $t = O(C_b\sqrt{\frac{d + \log(1/d^{10})}{T}}) = \widetilde{O}(C_b\sqrt{d/T})$, the above probability is upper bounded by $d^{-10}/2$. In other words, with probability at least $1 - d^{-10}/2$, we get

$$\left\|\frac{1}{T}\sum_{t=1}^T \mathbf{B}_t \mathbf{1}\{\mathbf{E}_t\} - \mathbb{E}[\mathbf{B}_t \mathbf{1}\{\mathbf{E}_t\}]\right\|_{\text{op}} \leq \widetilde{O}\left(C_b\sqrt{\frac{d}{T}}\right).$$

Bounding difference betwen $\mathbb{E}[\mathbf{B}_t]$ and $\mathbb{E}[\mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\}]$ We have

$$\|\mathbb{E}[\mathbf{B}_t] - \mathbb{E}[\mathbf{B}_t \mathbf{1}\{\mathbf{E}_t\}]\|_{\text{op}} \leq \mathbb{E}\left[\|\mathbf{B}_t\|_{\text{op}}\mathbf{1}\{\mathbf{E}_t^c\}\right] \leq \left(\mathbb{E}\left[\|\mathbf{B}_t\|_{\text{op}}^2\right]\cdot\mathbb{P}(\mathbf{E}_t^c)\right)^{1/2}$$

$$\leq \sqrt{\mathbb{E}\left[\max_i \|\mathbf{x}_{t,i}\|_2^2\right]\cdot d^{-10}} \leq \sqrt{(d + C\log n_2)\cdot d^{-10}} = \widetilde{O}(d^{-4.5}).$$

where the last inequality is by standard Gaussian norm concentration (e.g. (Bai & Lee, 2019, Appendix A.3)).

Concentration of $\mathbf{B}_t$ Combining the preceding two parts, we get that with probability at least $1 - Td^{-10}$ that

$$\left\|\frac{1}{T}\sum_{t\leq T} \mathbf{B}_t - \mathbb{E}[\mathbf{B}_t]\right\|_{\text{op}} \leq \left\|\frac{1}{T}\sum_{t\leq T} \mathbf{B}_t \mathbf{1}\{\mathbf{E}_t\} - \mathbb{E}[\mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\}]\right\|_{\text{op}} + \|\mathbb{E}[\mathbf{B}_t] - \mathbb{E}[\mathbf{B}_t\mathbf{1}\{\mathbf{E}_t\}]\|_{\text{op}} \leq \widetilde{O}\left(C_b\sqrt{d/T} + d^{-4.5}\right).$$

Concentration for $\mathbf{B}_t^2$, $\mathbf{A}_t$, and $\mathbf{A}_t^2$ For $\mathbf{B}_t^2$, using a similar analysis as the above, we get

$$\left\|\frac{1}{T}\sum_{t=1}^T \mathbf{B}_t^2 - \mathbb{E}\left[\mathbf{B}_t^2\right]\right\|_{\text{op}} \leq \widetilde{O}\left(C_b^2\sqrt{\frac{d}{T}} + d^{-4}\right).$$

For $\mathbf{A}_t$, we note that the bound

$$\mathbf{0} \preceq \mathbf{A}_t \preceq C_a\mathbf{I}_d, \quad \text{where } C_a = 1 + O\left(\max\left\{d/n, \sqrt{d/n}\right\}\right)$$

holds. Therefore using the same argument as above, we get the desired concentration bounds for $\mathbf{A}_t$. $\square$

We also need the following Hanson-Wright inequality.

Lemma C.5 (Restatement of Theorem 6.2.1, (Vershynin, 2018)). *Let $\mathbf{z} \in \mathbb{R}^D$ be a random vector with independent, mean-zero, and $O(K^2)$-sub-Gaussian entries, and let $\mathbf{C} \in \mathbb{R}^{D\times D}$ be a fixed matrix. Then we have with probability at least $1 - \delta$ that*

$$\left|\mathbf{z}^\top \mathbf{C}\mathbf{z} - \mathbb{E}[\mathbf{z}^\top \mathbf{C}\mathbf{z}]\right| \leq O\left(K^2 \max\left\{\|\mathbf{C}\|_{\text{Fr}}\sqrt{\log(2/\delta)}, \|\mathbf{C}\|_{\text{op}}\log(2/\delta)\right\}\right) \leq O\left(K^2\|\mathbf{C}\|_{\text{Fr}}\log(2/\delta)\right).$$

C.4. Proof of main theorem

We are now ready to prove Theorem 3. We first prove the result for $\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}}$. Define the matrix

$$\boldsymbol{\Sigma}_T := \left(\sum_{t=1}^{T} \mathbf{A}_t\right)^{-2} \sum_{t=1}^{T} \mathbf{A}_t^2 = \frac{1}{T} \cdot \left(\frac{\sum_{t=1}^{T} \mathbf{A}_t}{T}\right)^{-2} \frac{\sum_{t=1}^{T} \mathbf{A}_t^2}{T},$$

which will be key to our analysis. Observe that

$$\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}} - \mathbf{w}_{0,\star} = \left(\sum_{t=1}^{T} \mathbf{A}_t\right)^{-1} \sum_{t=1}^{T} \mathbf{A}_t(\mathbf{w}_t - \mathbf{w}_{0,\star}),$$

Therefore, conditioned on $\mathbf{A}_t$ (and only looking at the randomness of $\mathbf{w}_t$), we have

$$\mathbb{E}_{\mathbf{w}_t}\left[\text{MSE}(\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}})\right] = \mathbb{E}_{\mathbf{w}_t}\left[\left\|\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}} - \mathbf{w}_{0,\star}\right\|_2^2\right]$$

$$= \sum_{t=1}^{T} \text{tr}\left(\left(\sum_{t=1}^{T} \mathbf{A}_t\right)^{-1} \mathbf{A}_t \cdot \text{Cov}(\mathbf{w}_t)\mathbf{A}_t^\top \left(\sum_{t=1}^{T} \mathbf{A}_t\right)^{-1}\right) = \frac{R^2}{d}\text{tr}\left(\left(\sum_{t=1}^{T} \mathbf{A}_t\right)^{-2} \sum_{t=1}^{T} \mathbf{A}_t^2\right) = \frac{R^2}{d}\text{tr}(\boldsymbol{\Sigma}_T).$$

Concentration of $\mathbf{A}_t$ By Lemma C.3 and Lemma C.4, we have with probability at least $1 - Td^{-10}$ that

$$\underline{c}_A \mathbf{I}_d \preceq \mathbb{E}[\mathbf{A}_t] \preceq \bar{c}_A \mathbf{I}_d \quad \text{and} \quad \left\|\frac{1}{T}\sum_{t=1}^{T} \mathbf{A}_t - \mathbb{E}[\mathbf{A}_t]\right\|_{\text{op}} \leq \tilde{O}\left(C_a\sqrt{d/T} + d^{-4}\right),$$

where $\underline{c}_A, \bar{c}_A, C_a > 0$ are $\Theta(1)$ constants that depend only on γ. Therefore, taking $d = \tilde{\Omega}(\max\{\underline{c}_A^{-1/2}, 1\})$ and $T \geq \tilde{\Omega}(\max\{4C_a^2/\underline{c}_A^2 d, 4C_a^2/d\}) = \tilde{\Omega}(d)$, we get that $\left\|\frac{1}{T}\sum_{t=1}^{T} \mathbf{A}_t - \mathbb{E}[\mathbf{A}_t]\right\|_{\text{op}} \leq \min\{\underline{c}_A/2, 1/2\} \leq 1/2$ and $\lambda_{\min}(\frac{1}{T}\sum_{t=1}^{T} \mathbf{A}_t) \geq \underline{c}_A/2 > 0$. On the similar concentration event for $\mathbf{A}_t^2$ (in Lemma C.4), for $T \geq \tilde{\Omega}(d)$ we also have $\left\|\frac{1}{T}\sum_{t=1}^{T} \mathbf{A}_t^2 - \mathbb{E}[\mathbf{A}_t^2]\right\|_{\text{op}} \leq 1/2$ and $\lambda_{\min}(\frac{1}{T}\sum_{t=1}^{T} \mathbf{A}_t^2) \geq \underline{c}_A/2 > 0$.

Concentration of MSE around expectation We first bound the concentration between the MSE and its expectation $R^2/d \cdot \text{tr}(\boldsymbol{\Sigma}_T)$. Define

$$\mathbf{z} = \begin{bmatrix} \mathbf{w}_1 - \mathbf{w}_{0,\star} \\ \vdots \\ \mathbf{w}_T - \mathbf{w}_{0,\star} \end{bmatrix} \in \mathbb{R}^{dT} \quad \text{and} \quad \mathbf{U} = \begin{bmatrix} \left(\sum_{t\leq T} \mathbf{A}_t\right)^{-1} \mathbf{A}_1 \\ \vdots \\ \left(\sum_{t\leq T} \mathbf{A}_t\right)^{-1} \mathbf{A}_T \end{bmatrix} \in \mathbb{R}^{dT \times d}.$$

Then we have $\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}} - \mathbf{w}_{0,\star} = \mathbf{U}^\top \mathbf{z}$ and thus $\left\|\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}} - \mathbf{w}_{0,\star}\right\|_2^2 = \mathbf{z}^\top(\mathbf{U}\mathbf{U}^\top)\mathbf{z}$. By Assumption A, $\mathbf{z} \in \mathbb{R}^{dT}$ has i.i.d. mean-zero $O(R^2/d)$-sub-Gaussian entries. Therefore, applying the Hanson-Wright inequality (Lemma C.5) with $\mathbf{C} = \mathbf{U}\mathbf{U}^\top$, we get that with probability at least $1 - \delta$ we have

$$\left|\left\|\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}} - \mathbf{w}_{0,\star}\right\|_2^2 - \frac{R^2}{d}\text{tr}(\boldsymbol{\Sigma}_T)\right| = \left|\mathbf{z}^\top \mathbf{C}\mathbf{z} - \mathbb{E}[\mathbf{z}^\top \mathbf{C}\mathbf{z}]\right|$$

$$\leq \tilde{O}\left(\frac{R^2}{d}\|\mathbf{C}\|_{\text{Fr}}\right) = \tilde{O}\left(\frac{R^2}{d}\|\mathbf{U}^\top \mathbf{U}\|_{\text{Fr}}\right)$$

$$= \tilde{O}\left(\frac{R^2}{d}\left\|\left(\sum_{t\leq T} \mathbf{A}_t\right)^{-2} \sum_{t\leq T} \mathbf{A}_t^2\right\|_{\text{Fr}}\right) \tag{36}$$

$$= \tilde{O}\left(\frac{R^2}{dT}\lambda_{\min}\left(\frac{1}{T}\sum_{t\leq T} \mathbf{A}_t\right)^{-2} \cdot \sqrt{d}\left\|\frac{1}{T}\sum_{t\leq T} \mathbf{A}_t^2\right\|_{\text{op}}\right) = \tilde{O}\left(\frac{R^2}{T} \cdot \frac{1}{\sqrt{d}}\right).$$

Concentration of $\mathrm{tr}(\boldsymbol{\Sigma}_T)$ Recall that $\mathbf{A}_t$ are i.i.d. PSD matrices in $\mathbb{R}^{d \times d}$. We have

$$\frac{R^2}{d} \cdot \mathrm{tr}(\boldsymbol{\Sigma}_T) = \frac{R^2}{Td}\left\langle \left(\frac{\sum_{t=1}^{T} \mathbf{A}_t}{T}\right)^{-2}, \frac{\sum_{t=1}^{T} \mathbf{A}_t^2}{T} \right\rangle$$

$$= \frac{R^2}{T}\left\{ \underbrace{\frac{1}{d}\left\langle \mathbb{E}[\mathbf{A}_1]^{-2}, \mathbb{E}[\mathbf{A}_1^2] \right\rangle}_{\mathrm{I}} + \underbrace{\frac{1}{d}\left\langle \left(\frac{\sum_{t=1}^{T} \mathbf{A}_t}{T}\right)^{-2} - \mathbb{E}[\mathbf{A}_1]^{-2}, \mathbb{E}[\mathbf{A}_1^2] \right\rangle}_{\mathrm{II}} + \underbrace{\frac{1}{d}\left\langle \left(\frac{\sum_{t=1}^{T} \mathbf{A}_t}{T}\right)^{-2}, \frac{\sum_{t=1}^{T} \mathbf{A}_t^2}{T} - \mathbb{E}[\mathbf{A}_1^2] \right\rangle}_{\mathrm{III}} \right\}.$$

By Lemma C.2 and C.3, term I is the main $\Theta(1)$ term:

$$\mathrm{I} = \frac{1}{d}\left\langle f_A(n,d,\lambda)^{-2}\mathbf{I}_d, f_{A^2}(n,d,\lambda)\mathbf{I}_d \right\rangle = f_{A^2}(n,d,\lambda)/f_A(n,d,\lambda)^2$$

$$= \frac{\frac{1}{d}\mathbb{E}\left[\mathrm{tr}\left(\lambda^4(\widehat{\boldsymbol{\Sigma}}_n + \lambda\mathbf{I}_d)^{-4}\widehat{\boldsymbol{\Sigma}}_n^2\right)\right]}{\left(\frac{1}{d}\mathbb{E}\left[\mathrm{tr}\left(\lambda^2(\widehat{\boldsymbol{\Sigma}}_n + \lambda\mathbf{I}_d)^{-2}\widehat{\boldsymbol{\Sigma}}_n\right)\right]\right)^2} =: C_{d,n,\lambda}^{\mathsf{tr\text{-}tr}} = \Theta(1).$$

For term II we have

$$|\mathrm{II}| \le \left\|\left(\sum_{t=1}^{T} \mathbf{A}_t/T\right)^{-2} - \mathbb{E}[\mathbf{A}_1]^{-2}\right\|_{\mathrm{op}} \cdot \left\|\mathbb{E}[\mathbf{A}_1^2]\right\|_{\mathrm{op}}$$

$$\le \lambda_{\min}\left(\sum_{t=1}^{T} \mathbf{A}_t/T\right)^{-2}\left\|\left(\sum_{t=1}^{T} \mathbf{A}_t/T\right)^2 - \mathbb{E}[\mathbf{A}_1]^2\right\|_{\mathrm{op}} \lambda_{\min}(\mathbb{E}[\mathbf{A}_1])^{-2} \cdot \left\|\mathbb{E}[\mathbf{A}_1^2]\right\|_{\mathrm{op}}$$

$$\le \widetilde{O}\left(\left\|\left(\sum_{t=1}^{T} \mathbf{A}_t/T\right)^2 - \mathbb{E}[\mathbf{A}_1]^2\right\|_{\mathrm{op}}\right)$$

$$\le \widetilde{O}\left(\max\left\{\left\|\sum_{t=1}^{T} \mathbf{A}_t/T\right\|_{\mathrm{op}}, \|\mathbb{E}[\mathbf{A}_1]\|_{\mathrm{op}}\right\} \cdot \left\|\left(\sum_{t=1}^{T} \mathbf{A}_t/T\right) - \mathbb{E}[\mathbf{A}_1]\right\|_{\mathrm{op}}\right) \le \widetilde{O}\left(\sqrt{d/T} + d^{-4}\right).$$

Similarly we also have $|\mathrm{III}| \le \widetilde{O}(\sqrt{d/T})$. Combining terms I, II, III, we get that (on the concentration event)

$$\frac{R^2}{d}\mathrm{tr}(\boldsymbol{\Sigma}_T) = \frac{R^2}{T}\left(C_{d,n,\lambda}^{\mathsf{tr\text{-}tr}} + \widetilde{O}\left(\sqrt{d/T} + d^{-4}\right)\right).$$

This further combined with (36) gives

$$\left\|\widehat{\mathbf{w}}_{0,T}^{\mathsf{tr\text{-}tr}} - \mathbf{w}_{0,\star}\right\|_2^2 = \frac{R^2}{d}\mathrm{tr}(\boldsymbol{\Sigma}_T) + \frac{R^2}{T} \cdot \widetilde{O}(1/\sqrt{d}) = \frac{R^2}{T}\left(C_{d,n,\lambda}^{\mathsf{tr\text{-}tr}} + \widetilde{O}\left(\sqrt{d/T}\right) + \widetilde{O}\left(1/\sqrt{d}\right)\right).$$

This proves the desired result for the train-train method.

For the train-val method, observe that all the above analysis still holds if we replace $\mathbf{A}_t$ with $\mathbf{B}_t$ (and using the concentration for $\mathbf{B}_t$ guaranteed in Lemma C.4), we obtain a similar conclusion

$$\left\|\widehat{\mathbf{w}}_{0,T}^{\mathsf{tr\text{-}val}} - \mathbf{w}_{0,\star}\right\|_2^2 = \frac{R^2}{T}\left(C_{d,n_1,n_2,\lambda}^{\mathsf{tr\text{-}val}} + \widetilde{O}\left(\sqrt{d/T}\right) + \widetilde{O}\left(1/\sqrt{d}\right)\right),$$

where

$$C_{d,n_1,n_2,\lambda}^{\mathsf{tr\text{-}val}} := f_{B^2}(d,n_1,n_2,\lambda)/f_B(d,n_1,n_2,\lambda)^2$$

$$= \frac{\frac{1}{dn_2}\mathbb{E}\left[\mathrm{tr}\left(\lambda^2(\widehat{\boldsymbol{\Sigma}}_{n_1} + \lambda\mathbf{I}_d)^{-2}\right)^2 + (n_2+1)\mathrm{tr}\left(\lambda^4(\widehat{\boldsymbol{\Sigma}}_{n_1} + \lambda\mathbf{I}_d)^{-4}\right)\right]}{\left(\frac{1}{d}\mathbb{E}\left[\mathrm{tr}\left(\lambda^2(\widehat{\boldsymbol{\Sigma}}_{n_1} + \lambda\mathbf{I}_d)^{-2}\right)\right]\right)^2}.$$

This is the desired result. $\qquad\square$

D. Proof of Theorem 4

The proof is organized as follows. We optimize the hyperparameter (λ, n_1) for the train-val method in Section D.1. We derive the exact limit of the $C_{d,n,\lambda}^{\text{tr-tr}}$ in the proportional limit $d, n \to \infty$, $d/n \to \gamma$ and optimize over λ in Section D.2. We prove the main theorem in Section D.3.

D.1. Optimizing the hyperparameters for the train-val method

Lemma D.1 (Optimal constant of the train-val method). *For any (n, d) and any split ratio $(n_1, n_2) = (n_1, n - n_1)$, the optimal constant (by tuning the regularization $\lambda > 0$) of the train-val method is achieved at*

$$\inf_{\lambda > 0} C_{d,n_1,n_2,\lambda}^{\text{tr-val}} = \lim_{\lambda \to \infty} C_{d,n_1,n_2,\lambda}^{\text{tr-val}} = \frac{d + n_2 + 1}{n_2}.$$

Further optimizing the rate over n_2, the best rate is taken at $(n_1, n_2) = (0, n)$, in which the rate is

$$\inf_{\lambda > 0, \ n_2 \in [n]} C_{d,n_1,n_2,\lambda}^{\text{tr-val}} = \frac{(d + n + 1)R^2}{n}.$$

Discussion: Using all data as validation Lemma D.1 suggests that the optimal constant of the train-val method is obtained at $\lambda = \infty$ and $(n_1, n_2) = (0, n)$. In other words, the optimal choice for the train-val method is to *use all the data as validation*. In this case, since there is no training data, the inner solver reduces to the identity map: $\mathcal{A}_{\infty,0}(\mathbf{w}_0; \mathbf{X}_t, \mathbf{y}_t) = \mathbf{w}_0$, and the outer loop reduces to learning a single linear model $\mathbf{w}_0$ on all the tasks combined. We remark that while the optimality of such a split ratio is likely an artifact of the data distribution we assumed (noiseless realizable linear model) and may not generalize to other meta-learning problems, we do find experimentally that using more data as validation (than training) can also improve the performance on real meta-learning tasks (see Table 2).

Proof of Lemma D.1 Fix $n_1 \in [n]$ and $n_2 = n - n_1$. Recall from Theorem 3 (with the eigenvalue-based expressions in Lemma C.2) that

$$C_{d,n_1,n_2,\lambda}^{\text{tr-val}} = \frac{d}{n_2} \cdot \frac{\mathbb{E}\left[\left(\sum_{i=1}^{d} \lambda^2/(\sigma_i^{(n_1)} + \lambda)^2\right)^2 + (n_2 + 1)\sum_{i=1}^{d} \lambda^4/(\sigma_i^{(n_1)} + \lambda)^4\right]}{\left(\mathbb{E}\left[\sum_{i=1}^{d} \lambda^2/(\sigma_i^{(n_1)} + \lambda)^2\right]\right)^2}.$$

Clearly, as $\lambda \to \infty$, we have

$$\lim_{\lambda \to \infty} C_{d,n_1,n_2,\lambda}^{\text{tr-val}} = \frac{d}{n_2} \cdot \frac{d^2 + (n_2 + 1)d}{d^2} = \frac{(d + n_2 + 1)}{n_2}.$$

It remains to show that the above quantity is a lower bound for $C_{d,n_1,n_2,\lambda}^{\text{tr-val}}$ for any $\lambda > 0$, which is equivalent to

$$\frac{\mathbb{E}\left[\left(\sum_{i=1}^{d} \lambda^2/(\sigma_i^{(n_1)} + \lambda)^2\right)^2 + (n_2 + 1)\sum_{i=1}^{d} \lambda^4/(\sigma_i^{(n_1)} + \lambda)^4\right]}{\left(\mathbb{E}\left[\sum_{i=1}^{d} \lambda^2/(\sigma_i^{(n_1)} + \lambda)^2\right]\right)^2} \geq \frac{d + n_2 + 1}{d}, \quad \text{for all } \lambda > 0. \tag{37}$$

We now prove (37). For $i \in [n_1]$, define random variables

$$X_i := \frac{\lambda^2}{(\sigma_i^{(n_1)} + \lambda)^2} \in [0, 1] \quad \text{and} \quad Y_i := 1 - X_i \in [0, 1].$$

Then the left-hand side of (37) can be rewritten as

$$\frac{\mathbb{E}\left[\left(d - n_1 + \sum_{i=1}^{n_1} X_i\right)^2 + (n_2 + 1)\left(d - n_1 + \sum_{i=1}^{n_1} X_i^2\right)\right]}{\left(\mathbb{E}[d - n_1 + \sum_{i=1}^{n} X_i]\right)^2}.$$

$$= \frac{\mathbb{E}\left[(d - \sum_{i=1}^{n_1} Y_i)^2 + (n_2 + 1)(d - 2\sum_{i=1}^{n_1} Y_i + \sum_{i=1}^{n_1} Y_i^2)\right]}{(\mathbb{E}[d - \sum_{i=1}^{n_1} Y_i])^2}$$

$$= \frac{d^2 + (n_2 + 1)d - 2(d + n_2 + 1)\mathbb{E}[\sum Y_i] + \mathbb{E}[(\sum Y_i)^2] + (n_2 + 1)\mathbb{E}[\sum Y_i^2]}{d^2 - 2d\mathbb{E}[\sum Y_i] + (\mathbb{E}[\sum Y_i])^2}$$

By algebraic manipulation, inequality (37) is equivalent to showing that

$$\frac{\mathbb{E}[(\sum Y_i)^2] + (n_2 + 1)\mathbb{E}[\sum Y_i^2]}{(\mathbb{E}[\sum Y_i])^2} \geq \frac{d + n_2 + 1}{d}. \tag{38}$$

Clearly, $\mathbb{E}[(\sum Y_i)^2] \geq (\mathbb{E}[\sum Y_i])^2$. By Cauchy-Schwarz we also have

$$\mathbb{E}\left[\sum Y_i^2\right] \geq \frac{1}{n_1}\mathbb{E}\left[\left(\sum Y_i\right)^2\right] \geq \frac{1}{n_1}\left(\mathbb{E}\left[\sum Y_i\right]\right)^2.$$

Therefore we have

$$\frac{\mathbb{E}[(\sum Y_i)^2] + (n_2 + 1)\mathbb{E}[\sum Y_i^2]}{(\mathbb{E}[\sum Y_i])^2} \geq 1 + \frac{n_2 + 1}{n_1} \geq 1 + \frac{n_2 + 1}{d} = \frac{d + n_2 + 1}{d},$$

where we have used that $n_1 \leq n \leq d$. This shows (38) and consequently (37). $\qquad\square$

D.2. Optimizing the hyperparameters for the train-train method (in the proportional limit)

Theorem D.1 (Exact constant of the train-train method in the proportional limit)**.** *In the high-dimensional limiting regime* $d, n \to \infty$, $d/n \to \gamma$ *where* $\gamma \in (0, \infty)$ *is a fixed shape parameter, for any* $\lambda > 0$

$$\lim_{d,n\to\infty,d/n=\gamma} C^{\text{tr-tr}}_{d,n,\lambda} = \rho_{\lambda,\gamma}.$$

where $\rho_{\lambda,\gamma} = 4\gamma^2\left[(\gamma - 1)^2 + (\gamma + 1)\lambda\right]/(\lambda + 1 + \gamma - \sqrt{(\lambda + \gamma + 1)^2 - 4\gamma})^2/\left((\lambda + \gamma + 1)^2 - 4\gamma\right)^{3/2}$.

Proof of Theorem D.1   Let $\widehat{\Sigma}_n := \frac{1}{n}\mathbf{X}_t\mathbf{X}_t^\top$ denote the sample covariance matrix of the inputs in a single task (t). By Theorem 3 (with the eigenvalue-based expressions in Lemma C.2), we have

$$C^{\text{tr-tr}}_{d,n,\lambda} = \frac{\frac{1}{d}\mathbb{E}\left[\sum_{i=1}^d \sigma_i(\widehat{\Sigma}_n)^2/(\sigma_i(\widehat{\Sigma}_n) + \lambda)^4\right]}{\left(\frac{1}{d}\mathbb{E}\left[\sum_{i=1}^d \sigma_i(\widehat{\Sigma}_n)/(\sigma_i(\widehat{\Sigma}_n) + \lambda)^2\right]\right)^2}$$

$$= \underbrace{\frac{1}{d}\mathbb{E}\left[\text{tr}\left((\widehat{\Sigma}_n + \lambda\mathbf{I}_d)^{-4}\widehat{\Sigma}_n^2\right)\right]}_{\text{I}_{n,d}} \bigg/ \left\{\underbrace{\frac{1}{d}\mathbb{E}\left[\text{tr}\left((\widehat{\Sigma}_n + \lambda\mathbf{I}_d)^{-2}\widehat{\Sigma}_n\right)\right]}_{\text{II}_{n,d}}\right\}^2. \tag{39}$$

We now evaluate quantities $\text{I}_{n,d}$ and $\text{II}_{n,d}$ in the high-dimensional limit of $d, n \to \infty$, $d/n \to \gamma \in (0, \infty)$. Consider the (slightly generalized) Stieltjes transform of $\widehat{\Sigma}_n$ defined for all $\lambda_1, \lambda_2 > 0$:

$$s(\lambda_1, \lambda_2) := \lim_{d,n\to\infty,\ d/n\to\gamma} \frac{1}{d}\mathbb{E}\left[\text{tr}\left((\lambda_1\mathbf{I}_d + \lambda_2\widehat{\Sigma}_n)^{-1}\right)\right]. \tag{40}$$

As the entries of $\mathbf{X}_t$ are i.i.d. $\mathsf{N}(0, 1)$, the above limiting Stieltjes transform is the Stieltjes form of the Marchenko-Pastur law, which has a closed form (see, e.g. (Dobriban et al., 2018, Equation (7)))

$$s(\lambda_1, \lambda_2) = \lambda_2^{-1}s(\lambda_1/\lambda_2, 1) = \frac{1}{\lambda_2} \cdot \frac{\gamma - 1 - \lambda_1/\lambda_2 + \sqrt{(\lambda_1/\lambda_2 + 1 + \gamma)^2 - 4\gamma}}{2\gamma\lambda_1/\lambda_2}$$

$$= \frac{\gamma - 1 - \lambda_1/\lambda_2 + \sqrt{(\lambda_1/\lambda_2 + 1 + \gamma)^2 - 4\gamma}}{2\gamma\lambda_1}. \tag{41}$$

Now observe that differentiating $s(\lambda_1, \lambda_2)$ yields quantity II (known as the derivative trick of Stieltjes transforms). Indeed, we have

$$
\begin{aligned}
-\frac{d}{d\lambda_2} s(\lambda_1, \lambda_2) &= -\frac{d}{d\lambda_2} \lim_{d,n\to\infty,\, d/n\to\gamma} \frac{1}{d}\mathbb{E}\left[\mathrm{tr}\left((\lambda_1 \mathbf{I}_d + \lambda_2 \widehat{\boldsymbol{\Sigma}}_n)^{-1}\right)\right] \\
&= \lim_{d,n\to\infty,\, d/n\to\gamma} \frac{1}{d}\mathbb{E}\left[-\frac{d}{d\lambda_2}\mathrm{tr}\left((\lambda_1 \mathbf{I}_d + \lambda_2 \widehat{\boldsymbol{\Sigma}}_n)^{-1}\right)\right] \\
&= \lim_{d,n\to\infty,\, d/n\to\gamma} \frac{1}{d}\mathbb{E}\left[\mathrm{tr}\left((\lambda_1 \mathbf{I}_d + \lambda_2 \widehat{\boldsymbol{\Sigma}}_n)^{-2}\widehat{\boldsymbol{\Sigma}}_n\right)\right].
\end{aligned}
\tag{42}
$$

(Above, the exchange of differentiation and limit is due to the uniform convergence of the derivatives, which holds at any $\lambda_1, \lambda_2 > 0$. See Section D.2.1 for a detailed justification.) Taking $\lambda_1 = \lambda$ and $\lambda_2 = 1$, we get

$$
\lim_{d,n\to\infty,\, d/n\to\gamma} \mathrm{II}_{n,d} = \lim_{d,n\to\infty,\, d/n\to\gamma} \frac{1}{d}\mathbb{E}\left[\mathrm{tr}\left((\lambda \mathbf{I}_d + \widehat{\boldsymbol{\Sigma}}_n)^{-2}\widehat{\boldsymbol{\Sigma}}_n\right)\right] = -\frac{d}{d\lambda_2} s(\lambda_1, \lambda_2)|_{\lambda_1=\lambda,\lambda_2=1}.
$$

Similarly we have

$$
\lim_{d,n\to\infty,\, d/n\to\gamma} \mathrm{I}_{n,d} = \lim_{d,n\to\infty,\, d/n\to\gamma} \frac{1}{d}\mathbb{E}\left[\mathrm{tr}\left((\lambda \mathbf{I}_d + \widehat{\boldsymbol{\Sigma}}_n)^{-4}\widehat{\boldsymbol{\Sigma}}_n^2\right)\right] = -\frac{1}{6}\frac{d}{d\lambda_1}\frac{d^2}{d\lambda_2^2} s(\lambda_1, \lambda_2)|_{\lambda_1=\lambda,\lambda_2=1}.
$$

Evaluating the right-hand sides from differentiating the closed-form expression (41), we get

$$
\lim_{d,n\to\infty,\, d/n\to\gamma} \mathrm{II}_{n,d} = \frac{1}{2\gamma}\cdot\frac{\lambda+1+\gamma}{\sqrt{(\lambda+1+\gamma)^2 - 4\gamma}} - \frac{1}{2\gamma},
$$

$$
\lim_{d,n\to\infty,\, d/n\to\gamma} \mathrm{I}_{n,d} = \frac{(\gamma-1)^2 + (\gamma+1)\lambda}{((\lambda+1+\gamma)^2 - 4\gamma)^{5/2}}.
$$

Substituting back to (39) yields that

$$
\begin{aligned}
\lim_{d,n\to\infty,\, d/n\to\gamma} C^{\mathsf{tr\text{-}tr}}_{d,n,\lambda} &= \lim_{d,n\to\infty,\, d/n\to\gamma} \cdot \mathrm{I}_{n,d}/\mathrm{II}_{n,d}^2 \\
&= \frac{4\gamma^2\left[(\gamma-1)^2 + (\gamma+1)\lambda\right]}{((\lambda+1+\gamma)^2 - 4\gamma)^{5/2}\cdot\left(\frac{\lambda+1+\gamma}{\sqrt{(\lambda+1+\gamma)^2-4\gamma}} - 1\right)^2} \\
&= \frac{4\gamma^2\left[(\gamma-1)^2 + (\gamma+1)\lambda\right]}{((\lambda+1+\gamma)^2 - 4\gamma)^{3/2}\cdot\left(\lambda+1+\gamma - \sqrt{(\lambda+1+\gamma)^2 - 4\gamma}\right)^2}.
\end{aligned}
$$

This proves the desired result. $\qquad\square$

D.2.1. EXCHANGING DERIVATIVE AND EXPECTATION / LIMIT

Here we rigorously establish the exchange of the derivative and the expectation / limit used in (42). For convenience of notation let $\boldsymbol{\Sigma} = \widehat{\boldsymbol{\Sigma}}_n = \mathbf{X}_t^\top \mathbf{X}_t/n$ denote the empirical covariance matrix of $\mathbf{X}_t$. We wish to show that

$$
\frac{d}{d\lambda_2} \lim_{d,n\to\infty,\, d/n\to\gamma} \frac{1}{d}\mathbb{E}\left[\mathrm{tr}((\lambda_1 \mathbf{I}_d + \lambda_2 \boldsymbol{\Sigma})^{-1})\right] = \lim_{d,n\to\infty,\, d/n\to\gamma} \frac{1}{d}\mathbb{E}\left[\frac{d}{d\lambda_2}\mathrm{tr}((\lambda_1 \mathbf{I}_d + \lambda_2 \boldsymbol{\Sigma})^{-1})\right].
$$

This involves the exchange of derivative and limit, and then the exchange of derivative and expectation.

Exchange of derivative and expectation First, we show that for any fixed (d, n),

$$
\frac{d}{d\lambda_2}\mathbb{E}\left[\mathrm{tr}((\lambda_1 \mathbf{I}_d + \lambda_2 \boldsymbol{\Sigma})^{-1})\right] = \mathbb{E}\left[\frac{d}{d\lambda_2}\mathrm{tr}((\lambda_1 \mathbf{I}_d + \lambda_2 \boldsymbol{\Sigma})^{-1})\right].
$$

By definition of the derivative, we have

$$\frac{d}{d\lambda_2}\mathbb{E}\big[\text{tr}\big((\lambda_1\mathbf{I}_d + \lambda_2\boldsymbol{\Sigma})^{-1}\big)\big] = \lim_{t\to 0}\mathbb{E}\left[\frac{\text{tr}\big((\lambda_1\mathbf{I}_d + \lambda_2\boldsymbol{\Sigma} + t\boldsymbol{\Sigma})^{-1}\big) - \text{tr}\big((\lambda_1\mathbf{I}_d + \lambda_2\boldsymbol{\Sigma})^{-1}\big)}{t}\right].$$

For any $\mathbf{A} \succ \mathbf{0}$, the function $t \mapsto \text{tr}((\mathbf{A} + t\mathbf{B})^{-1})$ is continuously differentiable at $t = 0$ with derivative $-\text{tr}(\mathbf{A}^{-2}\mathbf{B})$, and thus locally Lipschitz around $t = 0$ with Lipschitz constant $|\text{tr}(\mathbf{A}^{-2}\mathbf{B})| + 1$. Applying this in the above expectation with $\mathbf{A} = \lambda_1\mathbf{I}_d + \lambda_2\boldsymbol{\Sigma} \succeq \lambda_1\mathbf{I}_d$ and $\mathbf{B} = \boldsymbol{\Sigma}$, we get that for sufficiently small $|t|$, the fraction inside the expectation is upper bounded by $|\text{tr}(\lambda_1^{-2}\boldsymbol{\Sigma})| + 1 < \infty$ uniformly over t. Thus by the Dominated Convergence Theorem, the limit can be passed into the expectation, which yields the expectation of the derivative.

Exchange of derivative and limit   Define $f_{n,d}(\lambda_2) := \frac{1}{d}\mathbb{E}\big[\text{tr}\big((\lambda_1\mathbf{I}_d + \lambda_2\boldsymbol{\Sigma})^{-1}\big)\big]$. It suffices to show that

$$\frac{d}{d\lambda_2}\lim_{d,n\to\infty, d/n\to\gamma} f_{n,d}(\lambda_2) = \lim_{d,n\to\infty, d/n\to\gamma} f'_{n,d}(\lambda_2),$$

where

$$f'_{n,d}(\lambda_2) = \mathbb{E}\left[\frac{d}{d\lambda_2}\frac{1}{d}\text{tr}\big((\lambda_1\mathbf{I}_d + \lambda_2\boldsymbol{\Sigma})^{-1}\big)\right] = -\frac{1}{d}\mathbb{E}\big[\text{tr}\big((\lambda_1\mathbf{I}_d + \lambda_2\boldsymbol{\Sigma})^{-2}\boldsymbol{\Sigma}\big)\big]$$

by the result of the preceding part.

As $f_{n,d}(\lambda_2) \to s(\lambda_1, \lambda_2)$ pointwise over λ_2 by properties of the Wishart matrix (Bai & Silverstein, 2010) and each individual $f_{n,d}$ is differentiable, it suffices to show that the derivatives $f'_{n,d}(\widetilde{\lambda}_2)$ converges uniformly for $\widetilde{\lambda}_2$ in a neighborhood of λ_2. Observe that can rewrite $f'_{n,d}$ as

$$f'_{n,d}(\widetilde{\lambda}_2) = -\mathbb{E}_{\widehat{\mu}_{n,d}}\Big[\mathbb{E}_{\lambda\sim\widehat{\mu}_{n,d}}\big[g_{\widetilde{\lambda}_2}(\lambda)\big]\Big],$$

where $\widehat{\mu}_{n,d}$ is the empirical distribution of the eigenvalues of $\boldsymbol{\Sigma}$, and

$$g_{\widetilde{\lambda}_2}(\lambda) := \frac{\lambda}{(\lambda_1 + \widetilde{\lambda}_2\lambda)^2} \leq \frac{1}{\lambda_1\widetilde{\lambda}_2}\quad\text{for all } \lambda \geq 0.$$

Therefore, as $\widehat{\mu}_{n,d}$ converges weakly to the Marchenko-Pastur distribution with probability one and $g_{\widetilde{\lambda}_2}$ is uniformly bounded for $\widetilde{\lambda}_2$ in a small neighborhood of λ_2, we get that $f'_{n,d}(\widetilde{\lambda}_2)$ does converge uniformly to the expectation of $g_{\widetilde{\lambda}_2}(\lambda)$ under the Marchenko-Pastur distribution. This shows the desired exchange of derivative and limit.

D.3. Proof of main theorem

We are now ready to prove the main theorem (Theorem 4).

Part I: Optimal constant for $L^{\text{tr-tr}}$ By Theorem D.1, we have

$$\inf_{\lambda>0}\lim_{d,n\to\infty, d/n=\gamma} C^{\text{tr-tr}}_{d,n,\lambda}$$

$$= \inf_{\lambda>0}\underbrace{\frac{4\gamma^2\big[(\gamma-1)^2 + (\gamma+1)\lambda\big]}{(\lambda + 1 + \gamma - \sqrt{(\lambda+\gamma+1)^2 - 4\gamma})^2 \cdot ((\lambda+\gamma+1)^2 - 4\gamma)^{3/2}}}_{:=f(\lambda,\gamma)}.$$

In order to bound $\inf_{\lambda>0} f(\lambda, \gamma)$, picking any $\lambda = \lambda(\gamma)$ gives $f(\lambda(\gamma), \gamma)$ as a valid upper bound, and our goal is to choose λ that yields a bound as tight as possible. Here we consider the choice

$$\lambda = \lambda(\gamma) = \max\{1 - \gamma/2, \gamma - 1/2\} = (1 - \gamma/2)\mathbf{1}\{\gamma \leq 1\} + (\gamma - 1/2)\mathbf{1}\{\gamma > 1\}$$

which we now show yields the claimed upper bound.

Case 1: $\gamma \leq 1$ Substituting $\lambda = 1 - \gamma/2$ into $f(\lambda, \gamma)$ and simplifying, we get

$$f(1 - \gamma/2, \gamma) = \frac{2(\gamma^2 - 3\gamma + 4)}{(2 - \gamma/2)^3} =: g_1(\gamma).$$

Clearly, $g_1(0) = 1$ and $g_1(1) = 32/27$. Further differentiating g_1 twice gives

$$g_1''(\gamma) = \frac{\gamma^2 + 7\gamma + 4}{(2 - \gamma/2)^5} > 0 \quad \text{for all } \gamma \in [0, 1].$$

Thus g_1 is convex on $[0, 1]$, from which we conclude that

$$g_1(\gamma) \leq (1 - \gamma) \cdot g_1(0) + \gamma \cdot g_1(1) = 1 + \frac{5}{27}\gamma.$$

Case 2: $\gamma > 1$ Substituting $\lambda = \gamma - 1/2$ into $f(\lambda, \gamma)$ and simplifying, we get

$$f(\gamma - 1/2, \gamma) = \frac{2\gamma^2(4\gamma^2 - 3\gamma + 1)}{(2\gamma - 1/2)^3} =: g_2(\gamma).$$

We have $g_2(1) = g_1(1) = 32/27$. Further differentiating g_2 gives

$$g_2'(\gamma) = -\frac{1}{(4\gamma - 1)^2} - \frac{6}{(4\gamma - 1)^3} - \frac{6}{(4\gamma - 1)^4} + 1 < 1 \quad \text{for all } \gamma > 1.$$

Therefore we have for all $\gamma > 1$ that

$$g_2(\gamma) = g_2(1) + \int_1^\gamma g_2'(t)dt \leq g_2(1) + \gamma - 1 = \gamma + \frac{5}{27}.$$

Combining Case 1 and 2, we get

$$\inf_{\lambda > 0} f(\lambda, \gamma) \leq g_1(\gamma)$$

$$\leq \mathbf{1}\{\gamma \leq 1\} + g_2(\gamma)\mathbf{1}\{\gamma > 1\} \leq \left(1 + \frac{5}{27}\gamma\right)\mathbf{1}\{\gamma \leq 1\} + \left(\frac{5}{27} + \gamma\right)\mathbf{1}\{\gamma > 1\}$$

$$= \max\left\{1 + \frac{5}{27}\gamma, \frac{5}{27} + \gamma\right\}.$$

This is the desired upper bound for $L^{\text{tr-tr}}$.

Equality at $\gamma = 1$ We finally show that the above upper bound becomes an equality when $\gamma = 1$. At $\gamma = 1$, we have

$$f(\lambda, 1) = \frac{8\lambda}{(\lambda + 2 - \sqrt{\lambda^2 + 4\lambda})^2(\lambda^2 + 4\lambda)^{3/2}} = \frac{8\lambda^{-4}}{(1 + 2/\lambda - \sqrt{1 + 4/\lambda})^2(1 + 4/\lambda)^{3/2}}.$$

Make the change of variable $t = \sqrt{1 + 4/\lambda}$ so that $\lambda^{-1} = (t^2 - 1)/4$, minimizing the above expression is equivalent to minimizing

$$\frac{(t^2 - 1)^4/32}{(t^2/2 - t + 1/2)^2 t^3} = \frac{(t + 1)^4}{8t^3}$$

over $t > 1$. It is straightforward to check (by computing the first and second derivatives) that the above quantity is minimized at $t = 3$ with value $32/27$. In other words, we have shown

$$\inf_{\lambda > 0} f(\lambda, 1) = \frac{32}{27} = \max\left\{1 + \frac{5}{27}\gamma, \frac{5}{27} + \gamma\right\}\Bigg|_{\gamma = 1},$$

that is, the equality holds at $\gamma = 1$.

Part II: Optimal constant for $L^{\text{tr-val}}$ We now prove the result on $L^{\text{tr-val}}$, that is,

$$\inf_{\lambda>0,s\in(0,1)} \lim_{d,n\to\infty,d/n=\gamma} C^{\text{tr-val}}_{d,ns,n(1-s),\lambda}$$

$$\overset{(i)}{=} \lim_{d,n\to\infty,d/n=\gamma} \underbrace{\inf_{\lambda>0,n_1+n_2=n} C^{\text{tr-val}}_{d,n_1,n_2,\lambda}}_{\frac{d+n+1}{n}} \overset{(ii)}{=} 1+\gamma.$$

First, equality (ii) follows from Lemma D.1 and the fact that $(d+n+1)/n \to 1+\gamma$. Second, the "$\geq$" direction of equality (i) is trivial (since we always have "inf lim $\geq$ lim inf"). Therefore we get the "$\geq$" direction of the overall equality, and it remains to prove the "$\leq$" direction.

For the "$\leq$" direction, we fix any $\lambda > 0$, and bound $C^{\text{tr-val}}_{d,n_1,n_2,\lambda}$ (and consequently its limit as $d, n \to \infty$.) We have by Theorem 3 (with the eigenvalue-based expressions in Lemma C.2) that

$$C^{\text{tr-val}}_{d,n_1,n_2,\lambda} = \frac{d}{n_2} \cdot \frac{\mathbb{E}\left[\left(\sum_{i=1}^{d} \lambda^2/(\sigma_i^{(n_1)}+\lambda)^2\right)^2 + (n_2+1)\sum_{i=1}^{d}\lambda^4/(\sigma_i^{(n_1)}+\lambda)^4\right]}{\left(\mathbb{E}\left[\sum_{i=1}^{d}\lambda^2/(\sigma_i^{(n_1)}+\lambda)^2\right]\right)^2}$$

$$\leq \frac{d}{n_2} \cdot \frac{d^2+(n_2+1)d}{\left(\mathbb{E}\left[\sum_{i=1}^{d}\lambda^2/(\sigma_i^{(n_1)}+\lambda)^2\right]\right)^2}$$

$$= \frac{d+n_2+1}{n_2} \cdot \frac{1}{\left(\mathbb{E}\left[\frac{1}{d}\sum_{i=1}^{d}\lambda^2/(\sigma_i^{(n_1)}+\lambda)^2\right]\right)^2}$$

Observe that

$$\mathbb{E}\left[\frac{1}{d}\sum_{i=1}^{d}\frac{\lambda^2}{(\sigma_i^{(n_1)}+\lambda)^2}\right] \overset{(i)}{\geq} \mathbb{E}\left[\frac{\lambda^2}{\left(\sum_{i=1}^{d}\sigma_i^{(n_1)}/d+\lambda\right)^2}\right]$$

$$\overset{(ii)}{\geq} \frac{\lambda^2}{\left(\mathbb{E}\left[\sum_{i=1}^{d}\sigma_i^{(n_1)}/d\right]+\lambda\right)^2} \overset{(iii)}{=} \frac{\lambda^2}{(1+\lambda)^2},$$

where (i) follows from the convexity of $t \mapsto \lambda^2/(t+\lambda)^2$ on $t \geq 0$; (ii) follows from the same convexity and Jensen's inequality, and (iii) is since $\mathbb{E}\left[\sum_{i=1}^{d}\sigma_i^{(n_1)}\right] = \mathbb{E}\left[\text{tr}(\frac{1}{n_1}\mathbf{X}_t^\top \mathbf{X}_t)\right] = \mathbb{E}\left[\|\mathbf{X}_t\|_{\text{Fr}}^2/n_1\right] = d$. Applying this in the preceeding bound yields

$$C^{\text{tr-val}}_{d,n_1,n_2,\lambda} \leq \frac{d+n_2+1}{n_2} \cdot \frac{(1+\lambda)^2}{\lambda^2}.$$

Further plugging in $n_1 = ns$ and $n_2 = n(1-s)$ for any $s \in (0,1)$ yields

$$\lim_{d,n\to\infty,d/n\to\gamma} C^{\text{tr-val}}_{d,ns,n(1-s),\lambda} \leq \frac{\gamma+1-s}{1-s} \cdot \frac{(1+\lambda)^2}{\lambda^2}.$$

Finally, the right-hand side is minimized at $\lambda \to \infty$ and $s = 0$, from which we conclude that

$$\inf_{\lambda>0,s\in(0,1)} \lim_{d,n\to\infty, d/n\to\gamma} C^{\text{tr-val}}_{d,ns,n(1-s),\lambda} \leq 1+\gamma,$$

which is the desired "$\leq$" direction. $\qquad\square$

E. Connections to Bayesian estimator

Here we discuss the relationship between our train-train meta-learining estimator using ridge regression solvers and a Bayesian estimator under a somewhat natural hierarchical generative model for the realizable setting in Section 4. We show that these two estimators are not equal in general, albeit they have some similarities in their expressions.

We consider the following hierarchical probabilitistic model:

$$\mathbf{w}_{0,\star} \sim \mathsf{N}\left(0, \frac{\sigma_w^2}{d}\mathbf{I}_d\right), \quad \mathbf{w}_t|\mathbf{w}_{0,\star} \stackrel{\text{iid}}{\sim} \mathsf{N}\left(\mathbf{w}_{0,\star}, \frac{R^2}{d}\mathbf{I}_d\right), \quad \mathbf{y}_t = \mathbf{X}_t\mathbf{w}_t + \sigma\mathbf{z}_t \text{ where } \mathbf{z}_t \stackrel{\text{iid}}{\sim} \mathsf{N}(0, \mathbf{I}_n).$$

This model is similar to our realizable linear model (5), except that $\mathbf{w}_0$ has a prior and that there is observation noise in the data (such that data likelihoods and posteriors are well-defined). We also note that the Gaussian distribution assumption (with R^2/d variance) for $\mathbf{w}_t$ is consistent with our Assumption A.

Bayesian estimator We now derive the Bayesian posterior mean estimator of $\mathbf{w}_{0,\star}$, which requires us to compute the posterior distribution of $\mathbf{w}_{0,\star}$ given the data $\{(\mathbf{X}_t, \mathbf{y}_t)\}_{t=1}^T$[4].

We begin by computing the likelihood of one task by marginalizing over $\mathbf{w}_t$:

$$p(\mathbf{X}_t, \mathbf{y}_t|\mathbf{w}_{0,\star}) \propto \int p(\mathbf{w}_t|\mathbf{w}_{0,\star}) \cdot p(\mathbf{y}_t|\mathbf{X}_t, \mathbf{w}_t)d\mathbf{w}_t$$

$$\propto \int \exp\left(-\frac{\|\mathbf{w}_t - \mathbf{w}_{0,\star}\|_2^2}{2R^2/d}\right) \cdot \exp\left(-\frac{\|\mathbf{y}_t - \mathbf{X}_t\mathbf{w}_t\|_2^2}{2\sigma^2}\right)d\mathbf{w}_t$$

$$\stackrel{(i)}{\propto} \exp\left(-\frac{\|\mathbf{w}_{0,\star}\|_2^2}{2R^2/d} + \frac{1}{2}\left(\frac{\mathbf{w}_{0,\star}}{R^2/d} + \frac{\mathbf{X}_t^\top\mathbf{y}_t}{\sigma^2}\right)^\top \left(\frac{\mathbf{X}_t^\top\mathbf{X}_t}{\sigma^2} + \frac{\mathbf{I}_d}{R^2/d}\right)^{-1}\left(\frac{\mathbf{w}_{0,\star}}{R^2/d} + \frac{\mathbf{X}_t^\top\mathbf{y}_t}{\sigma^2}\right)\right)$$

$$\propto \exp\left(-\frac{1}{2}\mathbf{w}_{0,\star}^\top\left(\left(\mathbf{X}_t^\top\mathbf{X}_t + \frac{d\sigma^2}{R^2}\mathbf{I}_d\right)^{-1}\frac{\mathbf{X}_t^\top\mathbf{X}_t}{R^2/d}\right)\mathbf{w}_{0,\star} + \mathbf{w}_{0,\star}^\top\left(\mathbf{X}_t^\top\mathbf{X}_t + \frac{d\sigma^2}{R^2}\mathbf{I}_d\right)^{-1}\frac{\mathbf{X}_t^\top\mathbf{y}_t}{R^2/d}\right),$$

where (i) is obtained by integrating a multivariate Gaussian density over $\mathbf{w}_t$, and "$\propto$" drops all the terms that do not depend on $\mathbf{w}_{0,\star}$. Therefore, by the Bayes rule, the overall posterior distribution of $\mathbf{w}_{0,\star}$ is given by

$$p\left(\mathbf{w}_{0,\star}|\{(\mathbf{X}_t, \mathbf{y}_t)\}_{t=1}^T\right) \propto p(\mathbf{w}_{0,\star}) \cdot \prod_{t=1}^T p(\mathbf{X}_t, \mathbf{y}_t|\mathbf{w}_{0,\star})$$

$$\propto \exp\left(-\frac{\|\mathbf{w}_{0,\star}\|_2^2}{2\sigma_w^2/d}\right) \cdot$$

$$\prod_{t=1}^T \exp\left(-\frac{1}{2}\mathbf{w}_{0,\star}^\top\left(\left(\mathbf{X}_t^\top\mathbf{X}_t + \frac{d\sigma^2}{R^2}\mathbf{I}_d\right)^{-1}\frac{\mathbf{X}_t^\top\mathbf{X}_t}{R^2/d}\right)\mathbf{w}_{0,\star} + \mathbf{w}_{0,\star}^\top\left(\mathbf{X}_t^\top\mathbf{X}_t + \frac{d\sigma^2}{R^2}\mathbf{I}_d\right)^{-1}\frac{\mathbf{X}_t^\top\mathbf{y}_t}{R^2/d}\right).$$

This means that the posterior distribution of $\mathbf{w}_{0,\star}$ is Gaussian, with mean , i.e. the Bayesian estimator, equal to[5]

$$\widehat{\mathbf{w}}_{0,T}^{\mathsf{Bayes}} := \mathbb{E}\left[\mathbf{w}_{0,\star} \mid \{(\mathbf{X}_t, \mathbf{y}_t)\}_{t=1}^T\right] = (\mathbf{A}_T^{\mathsf{Bayes}})^{-1}\mathbf{c}_T^{\mathsf{Bayes}},$$

where

$$\mathbf{A}_T^{\mathsf{Bayes}} := \frac{d}{\sigma_w^2}\mathbf{I}_d + \sum_{t=1}^T \left(\mathbf{X}_t^\top\mathbf{X}_t + \frac{d\sigma^2}{R^2}\mathbf{I}_d\right)^{-1}\frac{\mathbf{X}_t^\top\mathbf{X}_t}{R^2/d},$$

$$\mathbf{c}_T^{\mathsf{Bayes}} := \sum_{t=1}^T \left(\mathbf{X}_t^\top\mathbf{X}_t + \frac{d\sigma^2}{R^2}\mathbf{I}_d\right)^{-1}\frac{\mathbf{X}_t^\top\mathbf{y}_t}{R^2/d}.$$

---

[4]Hereafter we treat $\mathbf{X}_t$ as fixed, as the density of $\mathbf{X}_t$ won't affect the Bayesian calculation.

[5]Any density $p(\mathbf{w}) \propto \exp(-\mathbf{w}^\top\mathbf{A}\mathbf{w}/2 + \mathbf{w}^\top\mathbf{c})$ specifies a Gaussian distreibution $\mathsf{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\mathbf{A} = \boldsymbol{\Sigma}^{-1}$ and $\mathbf{c} = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$, so that $\boldsymbol{\mu} = \mathbf{A}^{-1}\mathbf{c}$.

We note that $\widehat{\mathbf{w}}_{0,T}^{\text{Bayes}}$ has a similar form as our train-train estimator, but is not exactly the same. Indeed, recall the closed form of our train-train estimator is (cf. (10))

$$\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}} = (\mathbf{A}_T^{\text{tr-tr}})^{-1}\mathbf{c}_T^{\text{tr-tr}},$$

where

$$\mathbf{A}_T^{\text{tr-tr}} = \sum_{t=1}^{T} \left(\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d\right)^{-2} \mathbf{X}_t^\top \mathbf{X}_t,$$

$$\mathbf{c}_T^{\text{tr-tr}} = \sum_{t=1}^{T} \left(\mathbf{X}_t^\top \mathbf{X}_t + n\lambda \mathbf{I}_d\right)^{-2} \mathbf{X}_t^\top \mathbf{y}_t.$$

As $\widehat{\mathbf{w}}_{0,T}^{\text{Bayes}}$ uses the **inverse** and $\widehat{\mathbf{w}}_{0,T}^{\text{tr-tr}}$ uses the **squared inverse**, these two sets of estimators are not the same in general, no matter how we tune the λ in the train-train estimator. This is true even if we set $\sigma_w = \infty$ so that the prior of $\mathbf{w}_{0,\star}$ becomes degenerate (and the Bayesian estimator reduces to the MLE).

F. Details on the few-shot image classification experiment

Here we provide additional details of the few-shot image classification experiment in Section 5.2.

Optimization and architecture For both methods, we run a few gradient steps on the inner argmin problem to obtain (an approximation of) $\mathbf{w}_t$, and plug $\mathbf{w}_t$ into the $\nabla_{\mathbf{w}_0}\ell_t^{\{\text{tr-val,tr-tr}\}}(\mathbf{w}_0)$ (which involves $\mathbf{w}_t$ through implicit function differentiation) for optimizing $\mathbf{w}_0$ in the outer loop.

For both train-train and train-val methods, we use the standard 4-layer convolutional network in (Finn et al., 2017; Zhou et al., 2019) as the backbone (i.e. the architecture for $\mathbf{w}_t$). We further tune their hyper-parameters, such as the regularization constant λ, the learning rate (initial learning rate and its decay strategy), and the gradient clipping threshold.

Dataset and evaluation MiniImageNet consists of 100 classes of images from ImageNet (Krizhevsky et al., 2012) and each class has 600 images of resolution $84 \times 84 \times 3$. We use 64 classes for training, 16 classes for validation, and the remaining 20 classes for testing (Ravi & Larochelle, 2017). TieredImageNet consists of 608 classes from the ILSVRC-12 data set (Russakovsky et al., 2015) and each image is also of resolution $84 \times 84 \times 3$. TieredImageNet groups classes into broader hierarchy categories corresponding to higher-level nodes in the ImageNet. Specifically, its top hierarchy has 20 training categories (351 classes), 6 validation categories (97 classes) and 8 test categories (160 classes). This structure ensures that all training classes are distinct from the testing classes, providing a more realistic few-shot learning scenario.

We evaluate both methods under the transduction setting where the information is shared between the test data via batch normalization.

F.1. Effect of the split ratio for the train-val method

We further tune the split (n_1, n_2) in the train-val method and report the results in Table 2. As can be seen, as the number of test samples n_2 increases, the percent classification accuracy on both the miniImageNet and tieredImageNet datasets becomes higher. This testifies our theoretical affirmation in Lemma D.1. However, note that even if we take the best split $(n_1, n_2) = (5, 25)$ (and compare again with Table 1), the train-val method still performs worse than the train-train method.

We remark that our theoretical results on train-train performing better than train-val (in Section 4) rely on the assumptions that the data can be exactly realized by the representation and contains no label noise. Our experimental results here may suggest that the miniImageNet and tieredImageNet few-shot tasks may have a similar structure (there exists a NN representation that almost perfectly realizes the label with no noise) that allows the train-train method to perform better than the train-val method.

G. Comparison with Cross-Validation on Synthetic Data

We test the effect of using cross-validation for the train-val method on the same synthetic data (realizable linear centroid meta-learning) as in Section 5.1.

Table 2. Investigation of the effects of training/validation splitting ratio in the train-val method (iMAML) to the few-shot classification accuracy (%) on miniImageNet and tieredImageNet.

datasets	$n_1 = 25, n_2 = 5$	$n_1 = 15, n_2 = 15$	$n_1 = 5, n_2 = 25$
miniImageNet	62.09 ± 0.97	63.56 ± 0.95	63.92 ± 1.04
tieredImageNet	66.45 ± 1.05	67.30 ± 0.98	67.50 ± 0.94

Method   We fix the number of per-task data $n = 20$, and use 4-fold cross validation in the following two settings: $(n_1, n_2) = (5, 15)$, and $(n_1, n_2) = (15, 5)$. In both cases, we partition the data into 4 parts each with 5 data points, and we roulette over 4 possible partitions of which one as train and which one as validation. The estimated optimal $\widehat{\mathbf{w}}_0^{\text{cv}}$ is obtained by minimize the averaged train-val loss over the 4 partitions:

$$\ell_t^{\text{cv}}(\mathbf{w}_0) := \frac{1}{4} \sum_{j=1}^{4} \frac{1}{2n_{\text{val}}} \left\| \mathbf{y}_t^{\text{val},j} - \mathbf{X}_t^{\text{val},j} \mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}_t^{\text{train},j}, \mathbf{y}_t^{\text{train},j}) \right\|_2^2,$$

$$\widehat{\mathbf{w}}_0^{\text{cv}} = \arg\min_{\mathbf{w}_0} \frac{1}{T} \sum_{t=1}^{T} \ell_t^{\text{cv}}(\mathbf{w}_0),$$

where superscript j denotes the index of the cross-validation. The performance is depicted in Figure 2.
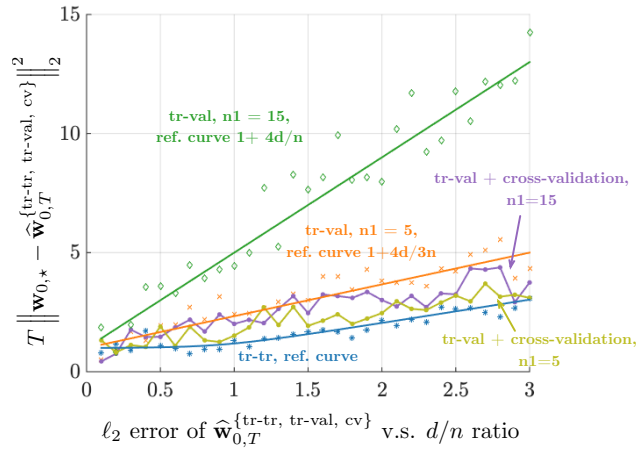

Figure 2. The scaled (by T) ℓ_2-error of $\widehat{\mathbf{w}}_{0,T}^{\{\text{tr-tr,tr-val,cv}\}}$ as the ratio d/n varies from 0 to 3 ($n = 20$ and $T = 1000$ are fixed). For the cross-validation method, the regularization coefficient $\lambda = 0.5$ is tuned.

Result   As showin in Figure 2, for both $(n_1, n_2) = (15, 5)$ and $(n_1, n_2) = (5, 15)$, using cross-validation consistently beats the performance of the train-val method. This demonstrates the variance-reduction effect of cross-validation. Note that the best performance (among the cross-validation methods) is still achieved at $n_1 = 5$, similar as for the vanilla train-val method. However, numerically, the best cross-validation performance is still not as good as the train-train method.

Leave-one-out cross-validation   Figure 3 left further tests with an increased number of per-task samples $n = 40$, and incorporates the train-val method with the leave-one-out cross-validation, i.e., $(n_1, n_2) = (39, 1)$ and $(n_1, n_2) = (1, 39)$. We repeat the experiment 10 times for plotting the error bar (shaded area). We see that the train-train method still outperforms the train-val method with leave-one-out validation.

We further increase the per-task sample size n to 200, and test the leave-one-out method with a sample split of $(n_1, n_2) = (1, 199)$. We adopt a matrix inverse trick to mitigate the computational overhead of finding $\mathcal{A}_\lambda(\mathbf{w}_0; \mathbf{X}_t^{\text{train},j}, \mathbf{y}_t^{\text{train},j})$. To ease the computation, we also vary d from 0 to 400 on a coarse grid (with an increment of 80). From Figure 3 right, we see that the leave-one-out method can slightly beat the train-train method for some d/n values. Compared to $n = 20$ and

$n = 40$ experiments, this is the first time of seeing leave-one-out method outperforms the train-train method. We suspect that the per-task sample size n plays a vital role in the power of the leave-one-out method: a large n tends to have a strong variance reduction effect in the leave-one-out method, so that the performance can be improved. Yet using the leave-one-out method with a large n invokes a high computational burden.
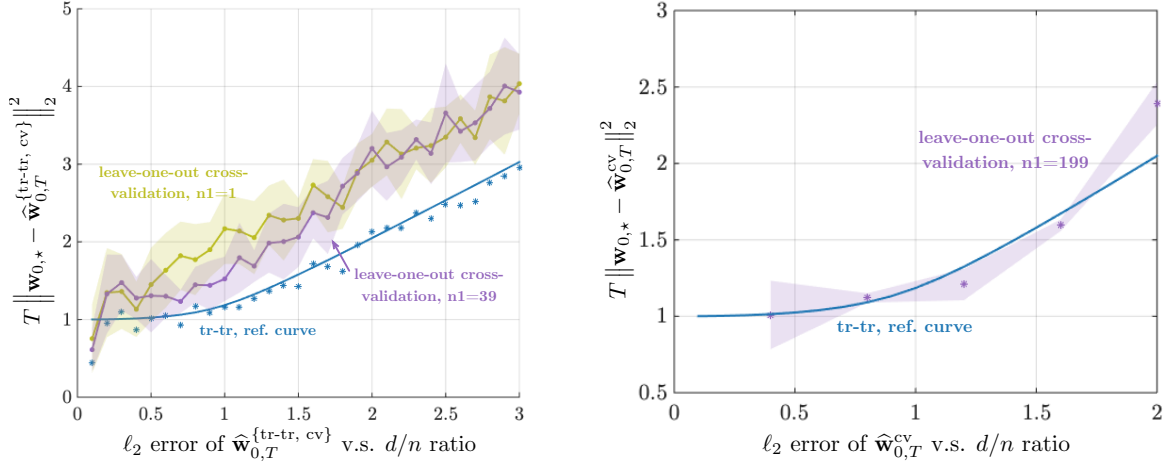

Figure 3. The scaled (by T) ℓ_2-error of $\widehat{\mathbf{w}}_{0,T}^{\{\text{tr-tr,cv}\}}$ as the ratio d/n varies from 0 to 3 ($n \in \{40, 200\}$ and $T = 1000$ are fixed). For the cross-validation method, the regularization coefficient $\lambda = 0.5$. Left: $n = 40$. Leave-out-out CV performs worse than the train-train method. Right: $n = 200$. Leave-one-out CV appears better than the train-train method for $d/n \in \{1.2, 1.6\}$.