A Distribution Dependent Analysis of Meta-Learning

by

Mikhail Konobeev

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

Abstract

A key problem in the theory of meta-learning is to understand how the task distributions influence transfer risk, the expected error of a meta-learner on a new task drawn from the unknown task distribution. In this work, focusing on fixed design linear regression with Gaussian noise and a Gaussian task (or parameter) distribution, we give distribution-dependent lower bounds on the transfer risk of *any* algorithm, while we also show that a *novel, weighted* version of the so-called biased regularized regression method is able to match these lower bounds up to a fixed constant factor. Notably, the weighting is derived from the covariance of the Gaussian task distribution. Altogether, our results provide a precise characterization of the difficulty of meta-learning in this Gaussian setting. While this problem setting may appear simple, we show that it is rich enough to unify the "parameter sharing" and "representation learning" streams of meta-learning; in particular, representation learning is obtained as the special case when the covariance matrix of the task distribution is unknown. For this case we propose to adopt the EM method, which is shown to enjoy efficient updates in our case. The work is completed by an empirical study of EM. In particular, our experimental results show that the EM algorithm can attain the lower bound as the number of tasks grows, while the algorithm is also successful in competing with its alternatives when used in a representation learning context.

Preface

This work is based on the paper [24] accepted for ICML 2021 and done in collaboration with Ilja Kuzborskij and my supervisor Csaba Szepesvári. My contributions include proving the lower and upper bounds for the general case, showing equivalence with weighted biased regularized least squares, proposing to use EM algorithm for the studied problem and for the subspace estimation task, deriving equations for both steps of the EM algorithm, and implementing all of the experiments.

Acknowledgements

I acknowledge the help of my supervisors Csaba Szepesvári and Martha White, as well as the help of my collaborator Ilja Kuzborskij.

Contents

List of Figures

Acronyms

DA Domain Adaptation

EM Expectation-Maximization

ERM Empirical Risk Minimization

GD Gradient Descent

KL Kullback-Liebler

MGF Moment-Generating Function

MLE Maximum Likelihood Estimator

MoM Method of Moments

OGD Online Gradient Descent

OLS Ordinary Least Squares

PAC Probably Approximately Correct

PSD positive semi-definite

RKHS Reproducing kernel Hilbert space

RLS Regularized Least Squares

SGD Stochastic Gradient Descent

SGLD Stochastic Gradient Langevin Dynamics

SVD Singular Value Decomposition

WBRLS Weighted Biased Regularized Least Squares

Chapter 1

Introduction

In meta-learning, a learner uses past tasks in an attempt to learn faster on a new task. Whether this will be possible depends on whether the new task is similar to the previous ones. In the formal framework of statistical *meta-learning* [5], the learner is given a sequence of training sets. The data in each set is independently sampled from an unknown distribution specific to the set, or task, while each such *task distribution* is independently sampled from an unknown meta-distribution, which we shall just call the *environment*. Define the learner's *transfer risk* as the expected prediction loss of a learner on a *target* task freshly sampled from the environment. *Can a learner achieve smaller transfer risk while using data from the possibly unrelated tasks? What are the limits of reducing transfer risk?* These are the questions we seek answers to in this thesis.

As an instructive example, consider a popular approach where each of the n tasks is associated with ground truth parameters $\boldsymbol{\theta}_i \in \mathbb{R}^d$, each of which is assumed to lie close to an unknown vector $\boldsymbol{\alpha}$ that characterizes the environment. To estimate the unknown parameter vector of the last task, one possibility is to employ a *biased regularization* [25], [32], [39], solving

$$\min_{\boldsymbol{\theta}} \widehat{\mathcal{L}}_n(\boldsymbol{\theta}) + \frac{\lambda}{2}\|\boldsymbol{\theta} - \widehat{\boldsymbol{\alpha}}\|^2,$$

where $\widehat{\mathcal{L}}_n(\cdot)$ is a convex empirical loss on task n, $\lambda > 0$ is a regularization parameter the governs the strength of the regularization term that biases the solution towards $\widehat{\boldsymbol{\alpha}}$, an estimate of $\boldsymbol{\alpha}$, which could be obtained, for example, by averaging parameters estimated on previous tasks [10]. This procedure

implements the maxim "learn on a new task, but stay close to what is already learned", which forms the basis of many successful meta-learning algorithms, including the above mentioned ones, or MAML [15].

Early theoretical work in the area focused on studying the *generalization gap*, which is the difference between the transfer risk and its empirical counterpart. Maurer [30] gives an upper bound on the generalization gap for a concrete algorithm which is similar to the biased regularization approach discussed above. In particular, the author shows that a simple modification of a regularized least-squares estimator will perform well on a future task, as long as empirical loss can be made small, while the number of tasks and the number of examples per each task is large. However, this result, as also discussed by others (e.g. [11]), does not show any benefits to meta-learning; it merely shows that using data from the other tasks does not hurt, which is unsurprising given the worst-case nature of the bound. Numerous other works have shown bounds on the generalization gap when using biased regularization, in one-shot learning [26], meta-learning [32], and sequential learning of tasks [9], [10], [16], [21], [22]. While some of these works introduced a dependence on the environment distribution, or on the "regularity" of the sequence of environments in the sequential setting, *they still leave open the question whether the shown dependence is best possible.*

In summary, the main weakness of the cited literature is the *lack of (problem dependent) lower bounds*: To be able to separate good meta-learning methods from poor ones, one needs to know the best achievable performance in a given problem setting. In learning theory, the most often used lower bounds are *distribution-free* or *problem independent.* In the context of meta learning, the distribution refers to the distribution over the tasks, or the environment. The major limitation of a distribution-free approach is that if the class of environments is sufficiently rich, all that the bound will tell us is that the best standard learner (which ignores the meta-learning aspect of the problem) will be competitive with the best meta-learner since the worst-case environment will be one where the tasks are completely unrelated. As an example, for a linear regression setting with d-dimensional parameter vectors, [29] gives the

worst-case lower bound $\tilde{\Omega}(d/(r^2(2r)^{-d}M + m))$ for parameter identification, where errors are measured in the squared Euclidean distance, and M is the total number of data points in the identically-sized training sets, m is the number of data points in the training set of the target task, and $r \geq 1$ is the radius of the ball that contains the parameter vectors.[1] It follows that as $r \to \infty$, the lower bound reduces to that of linear regression and we see that any method that ignores the tasks is competitive with the best meta-learning method.

This pathological limit can be avoided by restricting the set of environments. This approach is taken by Du *et al.* [11] and Tripuraneni *et al.* [38] who consider linear regression where the tasks share a common low-dimensional representation. Their main results show that natural algorithms can indeed take advantage of this. In addition, Tripuraneni *et al.* also shows a lower bound on the transfer risk which is shown to be matched by their method's transfer risk up to some logarithmic factors and some problem dependent, conditioning constants.

1.1 My contributions

In the present thesis we revisit the framework underlying biased regularized regression. In particular, we propose to study the case when the unknown parameter vectors for the tasks are generated from a normal distribution with some mean and covariance matrix. First, we consider the case when the mean is unknown and the covariance matrix of this distribution is known. For this case, in the context of fixed design linear regression, we prove essentially matching, distribution-dependent lower and upper bounds. The lower bound is a direct lower limit on the transfer risk of *any* meta-learning method. The upper bound is proven for a version of a *weighted* biased regularized least-squares regression. Here, the parameters are biased towards the maximum likelihood estimate of the unknown common mean of the task parameter vectors, and the

[1] This result is stated in Theorem 5 in their paper and the setting is meta linear regression. For readability, we dropped some constants, such as label noise variance and slightly generalized the cited result by introducing r, which is taken to be $r = 1$ in their paper. The analysis in the paper is modified to get the dependence shown on r in Section B.1.

weighting is done with respect to the inverse covariance matrix of the distribution over the task parameter vectors. We show that the maximum likelihood estimator can be efficiently computed, which implies that the entire procedure is efficient.

As opposed to the work of Tripuraneni *et al.* [38], the gap between the lower and upper bounds is a universal constant, regardless of the other parameters of the meta-learning task. The matching lower and upper bounds together provide a *precise and fine-grained characterization of the benefits of meta-learning*. In particular, these bounds show that meta-learning *can outperform standard supervise learning*. Our algorithm shows how one should combine datasets of different cardinalities and suggest specific ways of tuning biased regularized regression based on the noise characteristics of the data and the task structure. Our lower bounds are based on a technique that we have not seen before applied in learning theory and which may be of independent interest on its own.

Second, we consider the case when the covariance matrix of the task parameter vector distribution is unknown. Note that this case can be seen as a way of *unifying* the *representation learning* approach, in which the parameters are assumed to lie in a lower-dimensional subspace, with the approach of regularizing towards a common parameter. In particular, if the covariance matrix is such that $d - s$ of its eigenvalues tend to zero, while the other eigenvalues s are allowed to take on arbitrarily large values, the problem becomes essentially the same as the representation learning problem stuided by Du *et al.* [11] and Tripuraneni *et al.* [38].

While we provide no theoretical analysis for this case, we give a detailed description of how the Expectation-Maximization (EM) algorithm can be used to tackle this problem. In particular, we show that in this special case the EM algorithm enjoys an *efficient* implementation: we show how to implement the iterative steps in the loop of the EM algorithm in an efficient way. The steps of this algorithm are given as closed-form expressions, which are both intuitive and straightforward to implement.

We demonstrate the effectiveness of the resulting procedure on a number

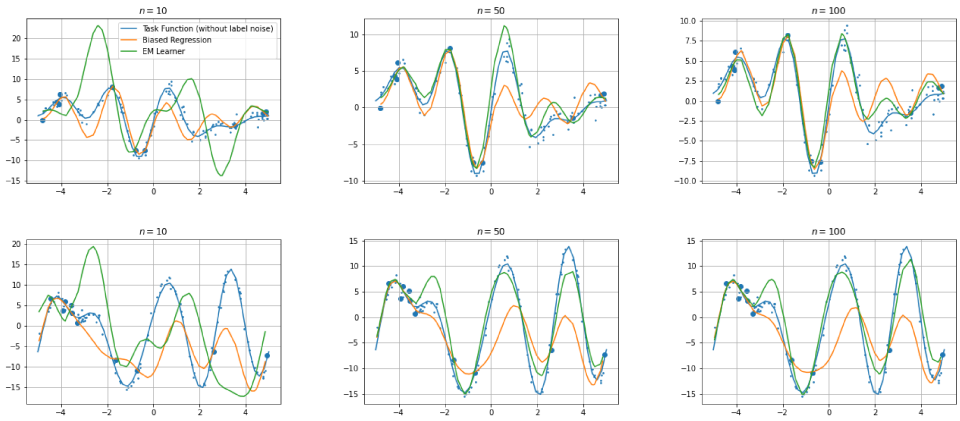

Figure 1.1: Two sets of examples of predictions on the synthetic, 'Fourier' meta-learning problem. Top and bottom rows correspond to different (random) instances. Training data is shown in bold, small dots show test data. We also show the predictions for two learners (at every input) and the target function. The column correspond to outputs obtained training on $n \in \{10, 50, 100\}$ tasks.

of synthetic and real benchmarks; Figure 1.1 shows a preview on a synthetic benchmark problem, comparing our EM algorithm with the earlier cited biased regression procedure that uses a straightforward least-squares approach to find the common parameter. As can be seen from the figure, the EM based learner is significantly more effective. Further experiments suggest that the EM learner is almost as effective as the optimal biased weighted regularized regression procedure that is given the unknown parameters. We also found that the EM learner is also competitive as a representation learning algorithm by comparing it to the method studied by Tripuraneni *et al.* [38].

To summarize, my contributions are as follows:

- Derive up to a universal constant matching lower and upper bounds for the studied problem;
- Show that the upper bound holds for the *weighted* version of biased regularized regression;
- Show how to use EM algorithm for the case of unknown covariance matrices, by deriving equations for the two steps of this algorithm;

- Experimentally show that EM attains the lower bound given a sufficient number of tasks and that it could successfully be applied in representation learning approach to meta-learning, i.e. that it can learn the lower-dimensional subspace to which the parameter vectors belong.

Chapter 2

Related Work

Early generalization analysis in meta-learning [5], [30] focused on bounds on the generalization gap in a problem-independent setting showcasing the interaction between the number of tasks, the sample size, and the capacity of the parameter class. Later on, the community started to pay attention to the problem-dependent setting where the generalization gap is also controlled by the quantities capturing task relatedness, such as τ^2 or eigenvalues of $\mathbf{\Sigma}$ featuring in this work. To name a few, [2], [28], [32] gave a PAC-Bayesian analysis of the generalization gap while [25] revisited an algorithmic stability analysis for the case of $n = 2$.

However, as pointed out earlier, the notion of generalization gap might fall short to fully explain learning abilities of algorithms and to compare them. To this end, the literature on meta-learning also considered the excess risk (the gap between the risk and the risk of the best-in-the-class): in particular, [31] proved upper bounds on the excess risk in multi-task and meta-learning scenarios when learning with dictionary representations. The rates in the same setting were recently improved by [11] to $\mathcal{O}(1/(nm) + 1/m)$ for excess risk in expectation. In this work we try to address less studied topic of lower bounds on the excess risk in a problem-dependent setting.

In the recent years, the meta-learning community dedicated considerable effort to the design and analysis of meta-learning algorithms able to learn in a sequential setting. An early notable line of research here is meta-leaning for non-i.i.d. sequence of tasks investigated in [33], [34]. With the advent

of the tremendous success of Gradient Descent (GD)-based meta-learning in deep neural networks [15], the field focused largely on the design of GD-type algorithms and their analysis, inspired by the online learning with individual sequences [1], [14], [16], [17], and recently demonstrated optimality of such algorithms [21], [36]. Although online meta-learning goes beyond the scope of this work, we note that estimators studied in this work can easily be extended to sequential setting through rank-one updates of matrix inverses.

Biased regularization revisited in this work has been a topic of interest in transfer learning for a long time in applications [23], [39] and theory [6], [26]. To the best of our knowledge, in this work we show the first result regarding optimality of the *weighted* biased regularization. Biased regularization and the regression model discussed in this work can be naturally described from the Bayesian point of view [18]. Indeed, it is well-known that ridge regression with regularization biased to $\boldsymbol{\alpha}$ arises as Maximum Likelihood Estimator (MLE) of a regression model with Gaussian-distributed parameters centered at $\boldsymbol{\alpha}$. The Bayesian point of view has been, for a long time (see discussion in [5]), a source of inspiration for design of transfer learning algorithms [35], [37] and even recent interpretations of GD-based techniques such as MAML [18]. While algorithms discussed in this work are Bayesian, the analysis we follow is fully frequentist.

Several works investigated a principled way of setting the bias in biased regularization [3], [9], [10], [12], such as by averaging parameter estimates of previously observed or held-out tasks [10]. In this work we propose an optimal setting of the bias in the considered regression problem based on MLE, and demonstrate that in order to achieve optimality one has to use a *weighted* biased regularization.

Finally, very recently [29] established universal *minimax* (that is by taking sup over task distributions) lower bounds for meta learning. In this work we explore a rather different, *problem-dependent* characterization of meta-learning which demands the dependence on task distributions. Beyond linear regression setting, no-free-lunch results in meta-learning and multi-task learning have also recently received attention in nonparametric prediction [20].

Chapter 3

Setup and Preview of the Results

In the statistical approach to *meta-learning* [4], [5] the learner observes a sequence of *training tuples* $\mathcal{D} = (D_i)_i^n$, distributed according to a *random* sequence of *task* distributions $(P_i)_i^n$, i.e. $D_i \sim P_i$, and furthermore task distributions are sampled independently from each other from a fixed and unknown *environment* distribution $\mathcal{P}$. The focus of this work is *linear regression with a fixed design* and therefore each training tuple $D_i = \big((\boldsymbol{x}_{i,1}, Y_{i,1}), \ldots, (\boldsymbol{x}_{i,m_i}, Y_{i,m_i})\big)$ consists of m_i fixed training *inputs* from $\mathbb{R}^d$ and corresponding random, real-valued *targets* satisfying

$$\begin{aligned} Y_{i,j} &= \boldsymbol{\theta}_i^\top \boldsymbol{x}_{i,j} + \varepsilon_{i,j}, \\ \text{where} \quad \varepsilon_{i,j} &\overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2),\ \boldsymbol{\theta}_i \overset{\text{iid}}{\sim} \mathcal{N}(\boldsymbol{\alpha}, \boldsymbol{\Sigma}), \end{aligned} \tag{3.1}$$

where $(\varepsilon_{i,j})_{i,j}$ and $(\boldsymbol{\theta}_i)_i$ are also independent of each other. Technically, P_i consists of Dirac deltas $\delta_{\boldsymbol{x}_{i,j}}$ on inputs and the normal distribution $\mathcal{N}(\theta_i^\top \boldsymbol{x}_{i,j}, \sigma^2)$ on the labels: $P_i(\boldsymbol{x}'_{i,j}, Y_{i,j}) = \delta_{\boldsymbol{x}_{i,j}}(\boldsymbol{x}'_{i,j})\mathcal{N}(Y_{i,j}|\boldsymbol{\theta}_i^\top \boldsymbol{x}_{i,j}, \sigma^2)$. A meta-learning environment in this setting is thus given by $\boldsymbol{\alpha}$ and the noise parameters $(\sigma^2, \boldsymbol{\Sigma})$. Initially, we will assume that $(\sigma^2, \boldsymbol{\Sigma})$ is known, while $\boldsymbol{\alpha}$ (just like $(\boldsymbol{\theta}_i)_i$) is unknown. The learner observes $\mathcal{D}$ and needs to produce a prediction of the value

$$Y = \boldsymbol{\theta}_n^\top \boldsymbol{x} + \varepsilon$$

where $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ and where $\boldsymbol{x} \in \mathbb{R}^d$ is a fixed (non-random) point. Our theoretical results will trivially extend to the case when the learner needs to

produce predictions for a sequence of input points or a fixed distribution over these. The (random) transfer risk of the learner is defined as

$$\mathcal{L}(\boldsymbol{x}) = \mathbb{E}\left[\left(Y - \hat{Y}\right)^2 \,\middle|\, \mathcal{D}\right] .$$

The setting described above coincides with the standard fixed-design linear regression model for $n = 1$ and $\boldsymbol{\Sigma} \to \mathbf{0}$, for which the behavior of risk is well understood.

In contrast, the question that meta-learning poses is whether having $n > 1$, one can design an predictor which achieves lower risk compared to the standard model, by exploiting the structure in the environment distribution. Naturally, this is of a particular interest in the small sample regime when for all tasks, $m_i \ll n$, that is when facing scarcity of the training data but having many tasks. Broadly speaking, this reduces to understanding the behavior of the risk in terms of the interaction between the number of tasks n, their sample sizes $(m_1, \ldots, m_n)$, and the task structure given by the noise parametrization $(\sigma^2, \boldsymbol{\Sigma})$.

3.1 Discussion of the Setup

The considered setup of eq. (3.1) is simple yet we do not see the linearity, or the normality of the distributions appearing in our model as strong limitations. The linear model could be generalized to any feature map, and possibly beyond, to the use of kernels. In fact, because we show equivalence to biased regularized regression and ridge regression is known to be capable of operating with kernels, we hypothesize that the proposed algorithm could also be made to work with kernels. Practical adaptation via EM operates with similar formulas (equations for the posterior distribution) at E-step and might also be used with the kernel trick. Our lower bound in the unbiased posterior mean estimator case could be generalized to any distributions $(P_i)_{i=1}^n$ and $\mathcal{P}$ such that the posterior mean $\mathbb{E}[Y|\mathcal{D}]$ and variance $\mathbb{V}[Y|\mathcal{D}]$ are computable and the marginal distribution over the labels $(y_{i,j})_{i,j=1}^{n,m_i}$ and the estimator of the posterior mean satisfy the two weak regularity conditions of the Cramér-Rao bound.

If the marginal distribution over the labels belongs to exponential family and maximum likelihood estimator is unbiased, then we immediately get matching lower and upper bounds for the of unbiased posterior mean estimation-based algorithms. For EM algorithm we require the computation of the posterior distribution over the labels which is easily computable if $\mathcal{P}$ is conjugate prior for the likelihood functions $(P_i)_{i=1}^n$ or if $\mathcal{P}$ is finite and not very big.

Finally, we emphasize that the setting we study *is* practical: Fixed design linear regression is a relevant setting on its own, in statistics, economics and other fields. We compare different algorithms on the real-world dataset of high school exam scores where each P_i is a distribution associated with a different school and $(\boldsymbol{x}_{i,j}, Y_{i,j})$ are feature map and the score of the student j in the i-th school respectively. Similar dataset could be obtained, for example, by considering treatments of patients in different hospitals.

Our setting is also similar to the setting of multi-task learning. The primary distinction, however, is in the fact that we are looking for how well algorithms can perform when adapting to a new task as opposed to how well an algorithm could perform on all n tasks.

3.2 Result 1: Special Cases of the Lower Bounds

Lower bounds. Our first contribution is a problem-dependent lower bound on the risk of **any** estimator for the considered regression problem (Theorem 5.1.1), which we elucidate here through a number of special cases.

To make the interpretation of the results easier, assume for now that inputs are isotropic, meaning that the input covariance matrix of task i is $\frac{m_i}{d}\boldsymbol{I}$. Furthermore, assume a *spherical* task structure: $\boldsymbol{\Sigma} = \tau^2\boldsymbol{I}$. Thus, the coordinates of the parameter vectors $\boldsymbol{\theta}_i$ are uncorrelated and share the same variance τ^2. For this specific case, our lower bound implies that for all estimators and $\boldsymbol{x}$ on the unit sphere,

$$\frac{\mathbb{E}[\mathcal{L}(\boldsymbol{x})] - \sigma^2}{\sigma^2} \geq \frac{H_{\tau^2}}{16\sqrt{e}} \cdot \frac{d^2\sigma^2}{n\left(\tau^2 m_n + d\sigma^2\right)^2} + \frac{d\tau^2}{\tau^2 m_n + d\sigma^2}$$

where H_z is the harmonic mean of the sequence $\{z + \frac{d\sigma^2}{m_i}\}_{i=1}^n$. The first term

decreases with adding more tasks (n growing) while to decrease the second, m_n needs to increase. In particular, as $n \to \infty$, we get

$$\frac{\mathbb{E}[\mathcal{L}(\boldsymbol{x})] - \sigma^2}{\sigma^2} \geq \left(\frac{m_n}{d} + \frac{\sigma^2}{\tau^2}\right)^{-1} \tag{3.2}$$

where $\frac{m_n}{d} + \frac{\sigma^2}{\tau^2}$ can be interpreted as an *effective sample size*. Thus, while having infinitely many previous tasks has the potential to reduce the loss, the size of this effect is fixed and is related to the noise variance ratios. If $\tau^2 \to 0$, having infinitely many tasks will allow perfect prediction, but for any $\tau^2 > 0$, there is a limit on how much the data of previous tasks can help. Finally, for the case $n = 1$ and $\tau^2 = 0$ we recover the standard lower bound for linear setting $\mathbb{E}[\mathcal{L}(\boldsymbol{x})] - \sigma^2 = \Omega(d\sigma^2/m_1)$.

Now, when $\boldsymbol{\Sigma}$ is an arbitrary positive semi-definite (PSD) matrix of rank $s \leq d$, letting λ_1 be its largest and λ_s to be its sth largest eigenvalue, a slightly loosened version of our bound gives

$$\frac{\mathbb{E}[\mathcal{L}(\boldsymbol{x})] - \sigma^2}{\sigma^2} \geq \frac{H_{\lambda_s}}{16\sqrt{e}} \cdot \frac{sd\sigma^2}{n\left(\lambda_1 m_n + d\sigma^2\right)^2} + \frac{s\lambda_s}{\lambda_s m_n + d\sigma^2} \ .$$

The first term scales with sd/n, where sd is the number of parameters in a matrix that would give the low-dimensional representation and the second term scales with s/m_n for $m_n \gg d\sigma^2/\lambda_s$. Somewhat surprisingly (given that here $\boldsymbol{\Sigma}$ is known), these essentially match earlier discovered upper bounds of Du *et al.* [11] and Tripuraneni *et al.* [38] implying that their results are unimprovable.

3.3 Result 2: Optimality of weighted biased regularization.

In the second contribution we show two results. First, we show the maximum likelihood estimator $\widehat{\boldsymbol{\alpha}}^{\mathrm{MLE}}$ of $\boldsymbol{\alpha}$ can be efficiently computed. Second, we show that that the predictor that predicts Y using $\hat{Y} = \boldsymbol{x}^\top \widehat{\boldsymbol{\theta}}_n$ where $\widehat{\boldsymbol{\theta}}_n$ is the minimizer of the biased, $\boldsymbol{\Sigma}$-weighted regularized least-squares problem

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^d} \sum_{j=1}^{m_n} (Y_{n,j} - \boldsymbol{\theta}^\top \boldsymbol{x}_{n,j})^2 + \frac{\sigma^2}{2} \|\boldsymbol{\theta} - \widehat{\boldsymbol{\alpha}}\|^2_{\boldsymbol{\Sigma}^{-1}},$$

is **near-optimal** when we set $\widehat{\boldsymbol{\alpha}} = \widehat{\boldsymbol{\alpha}}^{\text{MLE}}$ in the sense that its transfer risk matches the lower bound that we prove up to a universal constant factor. Note that this result is established without making specific assumptions on the data beyond those that are mentioned in the setup.

3.4 Result 3: EM algorithm for unknown covariance structure

The lower and matching upper bounds assumed that the covariance structure $(\sigma^2, \boldsymbol{\Sigma})$ is known. As a first step towards addressing the setting when these parameters are unknown, here we consider the instantiation of the EM algorithm and show that this leads to a natural algorithm that alternates between refining the covariance structure and using an assumed covariance structure to refine parameter estimates. We empirically test the new algorithm on a number of synthetic and real-world problems. The experimental results suggest that the EM algorithm, in line with our prior expectation, performs reasonably well. In particular, under a number of scenarios we find that it is competitive with the oracle algorithm which is given the task covariance structure. Finally, we reiterate that the setting of unknown covariance matrices is appealing as it unifies the common parameter vector approach with the common lower dimensional subspace approach to meta-learning.

Chapter 4

Sufficency of Meta-mean Prediction

In this section we show that there is no loss of generality in considering predictors of a special form that predict first the unknown meta-mean. We also show that biased regularization belongs to this family. We start with some general remarks and notation. Throughout the rest of the text, for real symmetric matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ notation $\boldsymbol{A} \succeq \boldsymbol{B}$ indicates that the matrix $\boldsymbol{A} - \boldsymbol{B}$ is PSD. For $\boldsymbol{x} \in \mathbb{R}^d$ and PSD matrix $\boldsymbol{A}$, a weighted Euclidean norm is defined as $\|\boldsymbol{x}\|_{\boldsymbol{A}} = \sqrt{\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x}}$. In what follows, without the loss of generality, we assume that $\boldsymbol{x} \neq \boldsymbol{0}$. We will use matrix notation aggregating inputs, targets, and parameters over multiple tasks. In particular, let the cumulative sample size of all tasks be $M = m_1 + \cdots + m_n$ and introduce aggregates for inputs and targets as follows:

$$\boldsymbol{X}_i = \underbrace{\begin{bmatrix} \boldsymbol{x}_{i,1}^\top \\ \vdots \\ \boldsymbol{x}_{i,m_i}^\top \end{bmatrix}}_{m_i \times d}, \quad \boldsymbol{\Psi} = \underbrace{\begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_n \end{bmatrix}}_{M \times d}, \quad \boldsymbol{Y}_i = \underbrace{\begin{bmatrix} Y_{i,1} \\ \vdots \\ Y_{i,m_i} \end{bmatrix}}_{m_i \times 1}, \quad \boldsymbol{Y} = \underbrace{\begin{bmatrix} \boldsymbol{Y}_1 \\ \vdots \\ \boldsymbol{Y}_n \end{bmatrix}}_{M \times 1}$$

$$\boldsymbol{X} = \underbrace{\begin{bmatrix} \boldsymbol{X}_1 & \dots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \dots & \boldsymbol{X}_n \end{bmatrix}}_{M \times nd}, \quad \boldsymbol{\Theta} = \underbrace{\begin{bmatrix} \boldsymbol{\theta}_1 \\ \vdots \\ \boldsymbol{\theta}_n \end{bmatrix}}_{nd \times 1}.$$

4.1 Marginal Distribution over the Labels

The matrix representation allows us to compactly state the regression model simultaneously over all tasks. In particular, for the M-dimensional noise vector $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{I})$:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\Theta} + \boldsymbol{\varepsilon} \quad \Leftrightarrow \quad \boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{\Psi}\boldsymbol{\alpha}, \boldsymbol{K}) \tag{4.1}$$

where $\boldsymbol{\alpha}$ is a meta-mean of model (3.1) and $\boldsymbol{K}$ is the *marginal* covariance matrix defined as $\boldsymbol{K} = \boldsymbol{X}(\boldsymbol{I} \otimes \boldsymbol{\Sigma})\boldsymbol{X}^\top + \sigma^2 \boldsymbol{I}$ where $\otimes$ stands for a Kronecker product. Note that the above equivalence comes from a straightforward observation that a linear map $\mathbf{X}_i$ applied to the Gaussian r.v. $\boldsymbol{\theta}_i$ is itself Gaussian with mean $\mathbb{E}[\boldsymbol{Y}_i] = \boldsymbol{X}_i \mathbb{E}[\boldsymbol{\theta}_i] = \boldsymbol{X}_i \boldsymbol{\alpha}$ and covariance $\boldsymbol{X}_i \boldsymbol{\Sigma} \boldsymbol{X}_i^T + \sigma^2 \mathbf{I}$ which follows from the property that for any random vector $\boldsymbol{\xi}$ with covariance matrix $\boldsymbol{C}$, and matrix $\boldsymbol{A}$ of appropriate dimensions, covariance matrix of $\boldsymbol{A}\boldsymbol{\xi}$ is $\boldsymbol{A}\boldsymbol{C}\boldsymbol{A}^\top$, ultimately giving eq. (4.1).

4.2 Estimators for the New Task

Both our lower and upper bounds will be derived from analyzing a family of estimators that aim to estimate $\boldsymbol{\theta}_n$ through estimating $\boldsymbol{\alpha}$. As we shall see, weighted biased regularization is also member of this family.

The said family is motivated by applying the well-known bias-variance decomposition to the risk of an arbitrary predictor $A : \mathrm{supp}(P_1) \times \cdots \times \mathrm{supp}(P_n) \times \mathbb{R}^d \to \mathbb{R}$. Namely, for a given $\boldsymbol{x}$, random $Y = \boldsymbol{x}^\top \boldsymbol{\theta}_n + \varepsilon$ and random dataset $\mathcal{D}$

$$\begin{aligned} \mathcal{L}(\boldsymbol{x}) &= \mathbb{E}\left[(Y - A(\mathcal{D}, \boldsymbol{x}))^2\right] \\ &= \mathbb{E}\left[\left(\mathbb{E}[Y \mid \mathcal{D}] - A(\mathcal{D}, \boldsymbol{x})\right)^2 + \mathbb{V}[Y \mid \mathcal{D}]\right] \end{aligned}$$

where we used the law of total expectation and that for any r.v. ξ, $\mathbb{E}[\xi^2] = \mathbb{E}[\xi]^2 + \mathbb{V}[\xi]$. Since the variance term does not depend on A, it follows that the prediction problem reduces to predicting the posterior mean $\mathbb{E}[Y \mid \mathcal{D}]$, which in our setting (recall that we assume $\boldsymbol{\theta}_i \sim \mathcal{N}(\boldsymbol{\alpha}, \boldsymbol{\Sigma})$), can be given in closed form:

Proposition 4.2.1. *Let $Y = \boldsymbol{\theta}_n^\top \boldsymbol{x} + \varepsilon$ for $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ and some $\boldsymbol{x} \in \mathbb{R}^d$. Then, $\mathbb{E}[Y \,|\, \mathcal{D}] = \boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \left(\boldsymbol{\Sigma}^{-1} \boldsymbol{\alpha} + \frac{1}{\sigma^2} \boldsymbol{X}_n^\top \boldsymbol{Y}_n \right)$ and $\mathbb{V}[Y \,|\, \mathcal{D}] = \boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \boldsymbol{x} + \sigma^2$, where $\boldsymbol{\mathcal{T}}$ is defined as $\boldsymbol{\mathcal{T}} = \left(\boldsymbol{\Sigma}^{-1} + \frac{1}{\sigma^2} \boldsymbol{X}_n^\top \boldsymbol{X}_n \right)^{-1}$.*

Proof. Recall that from Bayes rule it follows that

$$p_n(\boldsymbol{\theta}_n \,|\, \mathcal{D}) \propto \mathcal{N}(\boldsymbol{Y}_n \,|\, \boldsymbol{X}_n \boldsymbol{\theta}_n, \sigma^2 \boldsymbol{I}) \mathcal{N}(\boldsymbol{\theta}_n \,|\, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \ .$$

Using this, we will derive the following expression for the log-density of the posterior distribution:

$$\begin{aligned}
\ln p_n(\boldsymbol{\theta}_n) &= -\frac{\sum_{j=1}^{m_n} (\boldsymbol{x}_{n,j}^\top \boldsymbol{\theta}_n - Y_{n,j})^2}{2\sigma^2} - \frac{1}{2} (\boldsymbol{\theta}_n - \boldsymbol{\alpha})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{\theta}_n - \boldsymbol{\alpha}) + \text{const}(\boldsymbol{\theta}_n) \\
&= -\frac{\sum_{j=1}^{m_n} (\boldsymbol{\theta}_n^\top \boldsymbol{x}_{n,j} \boldsymbol{x}_{n,j}^\top \boldsymbol{\theta}_n - 2 Y_{n,j} \boldsymbol{x}_{n,j}^\top \boldsymbol{\theta}_n)}{2\sigma^2} \\
&= -\frac{1}{2} (\boldsymbol{\theta}_n^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\theta}_n - 2 \boldsymbol{\alpha}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\theta}_n) + \text{const}(\boldsymbol{\theta}_n) \\
&= -\frac{1}{2} \left(\boldsymbol{\theta}_n^\top \boldsymbol{\mathcal{T}}^{-1} \boldsymbol{\theta}_n - 2 \left(\boldsymbol{\Sigma}^{-1} \boldsymbol{\alpha} + \frac{1}{\sigma^2} \boldsymbol{X}_n^\top \boldsymbol{Y}_n \right)^\top \boldsymbol{\theta}_n \right) + \text{const}(\boldsymbol{\theta}_n) \\
&= -\frac{1}{2} (\boldsymbol{\theta}_n - \boldsymbol{\mu}) \boldsymbol{\mathcal{T}}^{-1} (\boldsymbol{\theta}_n - \boldsymbol{\mu}) + \text{const}(\boldsymbol{\theta}_n) \ .
\end{aligned}$$

where we introduced the notation:

$$\begin{aligned}
\boldsymbol{\mathcal{T}} &= \left(\boldsymbol{\Sigma}^{-1} + \frac{1}{\sigma^2} \boldsymbol{X}_n^\top \boldsymbol{X}_n \right)^{-1} , \\
\boldsymbol{\mu} &= \boldsymbol{\mathcal{T}} \left(\boldsymbol{\Sigma}^{-1} \boldsymbol{\alpha} + \frac{1}{\sigma^2} \boldsymbol{X}_n^\top \boldsymbol{Y}_n \right) .
\end{aligned}$$

Since log of the density takes quadratic form, we know that the posterior distribution is normal with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\mathcal{T}}$. From this the desired result follows by using the definition of Y together with standard properties of expectation and variance. □

Since the only unknown parameter here is the meta-mean $\boldsymbol{\alpha}$, we expect that good predictors will just estimate the meta-mean and use the above formula. That is, these predictors take the form $(\mathcal{D}, \boldsymbol{x}) \mapsto \boldsymbol{x}^\top \widehat{\boldsymbol{\theta}}_n(\boldsymbol{\alpha}(D, \boldsymbol{x}))$, where

$$\widehat{\boldsymbol{\theta}}_n(\boldsymbol{a}) = \boldsymbol{\mathcal{T}} \left(\boldsymbol{\Sigma}^{-1} \boldsymbol{a} + \frac{1}{\sigma^2} \boldsymbol{X}_n^\top \boldsymbol{Y}_n \right) \quad \boldsymbol{a} \in \mathbb{R}^d \ , \tag{4.2}$$

giving our family of predictors. In fact, it also holds that *there is no loss in generality by considering only predictors of the above form.* Indeed, given some predictor A, we can solve $A(D, \boldsymbol{x}) = \boldsymbol{x}^\top \widehat{\boldsymbol{\theta}}_n(\boldsymbol{\alpha})$ for $\boldsymbol{\alpha}$. One solution is given by $\boldsymbol{\alpha}(\mathcal{D}, \boldsymbol{x}) = \frac{\boldsymbol{x}}{\boldsymbol{x}^T \boldsymbol{\Gamma} \boldsymbol{\Sigma}^{-1} \boldsymbol{x}} A(\mathcal{D}, \boldsymbol{x}) - \sigma^{-2} \boldsymbol{\Sigma} \boldsymbol{X}_n^\top \boldsymbol{Y}_n$ which could easily be verified by plugging it into the equation $\boldsymbol{x}^\top \widehat{\boldsymbol{\theta}}_n(\alpha(\mathcal{D}, \boldsymbol{x}))$. Hence, to prove a lower bound for any regressor A, it will be enough to prove it for algorithms that estimate $\boldsymbol{\alpha}$ and then plug in into the above formula.

One special estimator of $\boldsymbol{\alpha}$ is the MLE estimator:

$$\widehat{\boldsymbol{\alpha}}^{\text{MLE}} = (\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1} \boldsymbol{\Psi})^{-1} \boldsymbol{\Psi}^\top \boldsymbol{K}^{-1} \boldsymbol{Y} \ . \tag{4.3}$$

The formula for the estimator is obtained analytically as a solution to the generalized least squares problem thanks to the established equivalence (4.1), that is $\widehat{\boldsymbol{\alpha}}^{\text{MLE}} = \arg\max_{\boldsymbol{a} \in \mathbb{R}^d} \ln \mathcal{N}(\boldsymbol{Y} \mid \boldsymbol{\Psi} \boldsymbol{a}, \boldsymbol{K}) =$ eq. (4.3). Finally, note the dependence of the estimator in eq. (4.2) on the target noise σ^2 and the task covariance matrix $\boldsymbol{\Sigma}$.

4.3 Biased Regularization

Biased regularization is a popular transfer learning technique which commonly appears in the regularized formulations of the empirical risk minimization problems, where one aims at minimizing the empirical risk (such as the mean squared error) while forcing the solution to stay close to some bias variable $\boldsymbol{b}$. Here we propose the Weighted Biased Regularized Least Squares (WBRLS) formulation defined w.r.t. bias $\boldsymbol{b}$ and some PSD matrix $\boldsymbol{\Gamma}$:

$$\widehat{\boldsymbol{\theta}}_n^{\text{BRLS}} = \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^d} \left\{ \widehat{\mathcal{L}}_n(\boldsymbol{\theta}) + \frac{\lambda}{2} \|\boldsymbol{\theta} - \boldsymbol{b}\|_{\boldsymbol{\Gamma}}^2 \right\}$$
$$\text{where} \quad \widehat{\mathcal{L}}_n(\boldsymbol{\theta}) = \sum_{j=1}^{m_n} \left(Y_{n,j} - \boldsymbol{\theta}^\top \boldsymbol{x}_{n,j} \right)^2 \ .$$

Remarkably, an estimate $\widehat{\boldsymbol{\theta}}_n^{\text{BRLS}}$ produced by WBRLS is *equivalent* to estimator $\widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}})$ of eq. (4.2) for the choice of $\boldsymbol{b} = \widehat{\boldsymbol{\alpha}}, \boldsymbol{\Gamma} = \boldsymbol{\Sigma}^{-1}$, and $\lambda = \sigma^2$. Thus, WBRLS is a special member of the family chosen in the previous section.

To see the equivalence, owing to the convenient least-squares formulation, we observe that

$$\widehat{\boldsymbol{\theta}}_n^{\mathrm{BRLS}} = \left(\boldsymbol{X}_n^\top \boldsymbol{X}_n + \lambda \boldsymbol{\Gamma}\right)^{-1} \left(\boldsymbol{X}_n^\top \boldsymbol{Y}_n + \lambda \boldsymbol{\Gamma} \boldsymbol{b}\right)$$

and from here the equivalence follows by substitution. A natural question commonly arising in such formulations is how to set the bias term $\boldsymbol{b}$. The reasoning above suggests that a good value is $\boldsymbol{b} = \widehat{\boldsymbol{\alpha}}^{\mathrm{MLE}}$.

Chapter 5

Problem-Dependent Bounds

We now present our main results, which are essentially matching lower and upper bounds. The proofs of the main results are provided at the end of this chapter after we state and explain the results. The upper bounds concern the parameter estimator that uses the MLE estimate of $\boldsymbol{\alpha}$, while the lower bounds apply to *any* method. We also present a (stronger) lower bound that applies to estimators that are built on *unbiased* meta-mean estimators $\widehat{\boldsymbol{\alpha}}$. Notably, the general lower bounds, which apply to any method, differ from this lower bound only by a universal constant. We also give a high-probability variant of the same general lower bound.

5.1 Lower Bounds

The following theorem gives a lower bound for the expected loss of of the meta-learner that predicts $\hat{Y} = \boldsymbol{x}^\top \widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}})$. As was noted in the previous chapter, such meta-learner is general enough in that it can agree with the prediction of any other algorithm $A(\mathcal{D}, \boldsymbol{x})$ for the right choice of $\widehat{\boldsymbol{\alpha}}$.

Theorem 5.1.1. *Let $\boldsymbol{x} \in \mathbb{R}^d$ and consider the linear regression model* (3.1)*. Let $\widehat{\boldsymbol{\alpha}}$ be any unbiased estimator of $\boldsymbol{\alpha}$ based on $\mathcal{D}$. Then the transfer risk $\mathcal{L}(\boldsymbol{x})$ of the predictor that predicts $\hat{Y} = \boldsymbol{x}^\top \widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}})$ satisfies*

$$\mathbb{E}[\mathcal{L}(\boldsymbol{x})] \geq \boldsymbol{x}^\top \boldsymbol{M} \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \boldsymbol{x} + \sigma^2 \tag{5.1}$$

$$\textit{where} \quad \boldsymbol{M} = \boldsymbol{\mathcal{T}} \boldsymbol{\Sigma}^{-1} (\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1} \boldsymbol{\Psi})^{-1} \boldsymbol{\Sigma}^{-1} \boldsymbol{\mathcal{T}} \ .$$

Moreover, for all predictors we have

$$\mathbb{E}[\mathcal{L}(\boldsymbol{x})] \geq \frac{\boldsymbol{x}^\top \boldsymbol{M} \boldsymbol{x}}{16\sqrt{e}} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \boldsymbol{x} + \sigma^2. \tag{5.2}$$

Finally, with probability at least $1-\delta, \delta \in (0,1)$ *for all predictors we have*

$$\mathcal{L}(\boldsymbol{x}) \geq \frac{1}{2} \log\left(\frac{1}{4(1-\delta)}\right) \boldsymbol{x}^\top \boldsymbol{M} \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \boldsymbol{x} + \sigma^2.$$

Proof. See section 5.4. □

Note that the presented bounds are problem-dependent since they depend on a concerete task structure of the environment characterized by $(\boldsymbol{\Sigma}, \sigma^2)$. The following proposition specializes the lower bound and is the basis of the summary that was given earlier in section 3.2.[1]

Proposition 5.1.2. *Assume the same as in case of eq.* (5.2)*. In addition, let* $\boldsymbol{\Sigma} = \tau^2 \boldsymbol{I}$*, suppose that* $\boldsymbol{X}_i^\top \boldsymbol{X}_i = \frac{m_i}{d} \boldsymbol{I}$*, and let* $\|\boldsymbol{x}\| = 1$*. Then,*

$$\mathbb{E}[\mathcal{L}(\boldsymbol{x})] \geq \frac{H_{\tau^2}}{16\sqrt{e}n} \cdot \frac{d^2\sigma^4}{\left(\tau^2 m_n + d\sigma^2\right)^2} + \frac{d\sigma^2\tau^2}{\tau^2 m_n + d\sigma^2} + \sigma^2$$

where H_z *is a harmonic mean of a sequence* $(z + \frac{d\sigma^2}{m_i})_{i=1}^n$*.*

Moreover, let $\boldsymbol{\Sigma}$ *be a PSD matrix of rank* $s \leq d$ *with eigenvalues* $\lambda_1 \geq \ldots \geq \lambda_s > 0$,[2] *and suppose that* $\|\boldsymbol{x}\|^2_{\boldsymbol{P}_s^\top \boldsymbol{P}_s} = s/d$ *where* $\boldsymbol{P}_s = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_s]^\top$ *and* $(\boldsymbol{u}_j)_{j=1}^s$ *are unit length eigenvectors of* $\boldsymbol{\Sigma}$*. Then,*

$$\mathbb{E}[\mathcal{L}(\boldsymbol{x})] \geq \frac{H_{\lambda_s}}{16\sqrt{e}n} \cdot \frac{sd\sigma^4}{\left(\lambda_1 m_n + d\sigma^2\right)^2} + \frac{s\sigma^2\lambda_s}{\lambda_s m_n + d\sigma^2} + \sigma^2 \ .$$

Proof. See appendix B.2. □

This result was discussed in detail in section 3.2.

5.2 Upper Bounds for the Maximum Likelihood-based Estimator

Next we present a risk *identity* for $\widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}}^{\text{MLE}})$, that is the one which employs *unbiased* MLE meta-mean estimator $\widehat{\boldsymbol{\alpha}}^{\text{MLE}}$ defined in eq. (4.3). We also give a high probability upper bound.

[1] proposition 5.1.2 is for the setting of eq. (5.2), however it is straightforward to give analogous bounds for other cases.

[2] When $s < d$, we replace $\boldsymbol{\Sigma}^{-1}$ with its pseudo-inverse $\boldsymbol{\Sigma}^\dagger$.

Theorem 5.2.1. *For the estimator $\widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}}^{\mathrm{MLE}})$ and for any $\boldsymbol{x} \in \mathbb{R}^d$ we have $\mathbb{E}[\mathcal{L}(\boldsymbol{x})] = \boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{x} + \sigma^2$. Moreover for the same estimator, with probability at least $1-\delta, \delta \in (0,1)$ we have*

$$\mathcal{L}(\boldsymbol{x}) \leq 2\log\left(\frac{2}{\delta}\right) \boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{x} + \sigma^2.$$

Proof. See section 5.5 □

This result, together with our lower bound shows that *(i)* the predictors based on $\widehat{\boldsymbol{\alpha}}^{\mathrm{MLE}}$ is *optimal, with matching constant* within the set of predictors that is based on unbiased estimators of $\boldsymbol{\alpha}$. It also follows that *(ii)* apart from a constant factor of $16\sqrt{e}$ of the transfer risk, this predictor is also optimal among *all* predictors.

5.3 Proof of Problem-Dependent Bounds for the General Case

In this section we provide proofs for the general case — that is when there are no additional assumptions on the feature covariance matrices $\boldsymbol{X}_i^\top \boldsymbol{X}_i$ and covariance matrix $\boldsymbol{\Sigma}$. The proofs of the proposition 5.1.2 is deferred to section B.2 as it is the proof which is done by my collaborator Ilja Kuzborskij.

We start by establishing the following corollary of the bias-variance decomposition, proposition 4.2.1 and the form of estimator $\widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}})$ defined in eq. (4.2).

Corollary 5.3.1. *For $\widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}})$ defined in eq. (4.2), any task mean estimator $\widehat{\boldsymbol{\alpha}}$, and any $\boldsymbol{x} \in \mathbb{R}^d$ we have $\mathbb{E}[\mathcal{L}(\boldsymbol{x})] = \mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}})\right)^2\right] + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{x} + \sigma^2$.*

Proof. Using the law of total expectation and that for a r.v. ξ we have $\mathbb{E}[\xi^2] = \mathbb{E}[\xi]^2 + \mathbb{V}[\xi]$,

$$\begin{aligned}
\mathcal{L}(\boldsymbol{x}) &= \mathbb{E}\left[(Y - \widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}})^\top \boldsymbol{x})^2\right] \\
&= \mathbb{E}\left[\left(\mathbb{E}[Y \mid \mathcal{D}] - \widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}})^\top \boldsymbol{x}\right)^2 + \mathbb{V}[Y \mid \mathcal{D}]\right] \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}}\right)\right)^2\right] + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{x} + \sigma^2
\end{aligned}$$

where identities for $\mathbb{E}[Y \mid \mathcal{D}]$ and $\mathbb{V}[Y \mid \mathcal{D}]$ come from proposition 4.2.1 and identity for $\widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}})$ is due to (4.2). □

5.4 Proof of the Lower Bounds

By corollary 5.3.1 our task reduces to establishing lower bounds on

$$\mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \boldsymbol{\Sigma}^{-1}(\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}})\right)^2\right] \tag{5.3}$$

for any choice of estimator $\widehat{\boldsymbol{\alpha}}$, which in combination with corollary 5.3.1 will prove theorem 5.1.1. In the next section we first prove a lower bound for any unbiased estimator relying on the Cramér-Rao inequality. In what follows, in section 5.4.2, we will show a general bound in lemma 5.4.4 valid for any estimator (possibly biased) using a *hypothesis testing* technique (see, e.g. [27, Chap. 13]). Finally, in lemma 5.4.5 we prove a high-probability lower bound on eq. (5.3).

5.4.1 Lower Bound for Unbiased Estimator $\widehat{\boldsymbol{\alpha}}$

Theorem 5.4.1 (Cramér-Rao inequality)**.** *Suppose that $\boldsymbol{\alpha} \in \mathbb{R}^d$ is an unknown deterministic parameter with a probability density function $f(x \mid \boldsymbol{\alpha})$ and that $\widehat{\boldsymbol{\alpha}}$ is an unbiased estimator of $\boldsymbol{\alpha}$. Moreover assume that for all $i, j \in [d]$, $x : f(x \mid \boldsymbol{\alpha}) > 0$, $\frac{\partial^2}{\partial\alpha_i \partial\alpha_j} \ln f(x \mid \boldsymbol{\alpha})$ exists and is finite, and $\frac{\partial^2}{\partial\alpha_i \partial\alpha_j} \int \widehat{\boldsymbol{\alpha}} f(x \mid \boldsymbol{\alpha}) \,\mathrm{d}x = \int \widehat{\boldsymbol{\alpha}} \left(\frac{\partial^2}{\partial\alpha_i \partial\alpha_j} f(x \mid \boldsymbol{\alpha})\right) \mathrm{d}x$.*

Then, for the Fisher information matrix defined as

$$\boldsymbol{F} = -\mathbb{E}\left[\nabla_{\boldsymbol{\alpha}} \ln f(X \mid \boldsymbol{\alpha}) \nabla_{\boldsymbol{\alpha}} \ln f(X \mid \boldsymbol{\alpha})^\top\right]$$

we have

$$\mathbb{E}\left[(\widehat{\boldsymbol{\alpha}} - \mathbb{E}[\widehat{\boldsymbol{\alpha}}])(\widehat{\boldsymbol{\alpha}} - \mathbb{E}[\widehat{\boldsymbol{\alpha}}])^\top\right] \succeq \boldsymbol{F}^{-1} \ .$$

Lemma 5.4.2. *For any unbiased estimator $\widehat{\boldsymbol{\alpha}}$ of $\boldsymbol{\alpha}$ in eq. (4.1) we have*

$$\mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \boldsymbol{\Sigma}^{-1}(\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}})\right)^2\right] \geq \boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \boldsymbol{\Sigma}^{-1} (\boldsymbol{\Psi}^\top \boldsymbol{K} \boldsymbol{\Psi})^{-1} \boldsymbol{\Sigma}^{-1} \boldsymbol{\mathcal{T}} \boldsymbol{x}. \tag{5.4}$$

Proof. Recall that according to the equivalence (4.1) $\boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{\Psi\alpha}, \boldsymbol{K})$ and the unknown parameter is $\boldsymbol{\alpha}$. To compute the Fisher information matrix we first observe that

$$\nabla_{\boldsymbol{\alpha}} \ln \mathcal{N}\left(\boldsymbol{Y} \mid \boldsymbol{\Psi\alpha}, \boldsymbol{K}\right) = \boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}(\boldsymbol{Y} - \boldsymbol{\Psi\alpha})$$

and so

$$\begin{aligned}
\boldsymbol{F} &= \mathbb{E}\left[\nabla_{\boldsymbol{\alpha}} \ln \mathcal{N}\left(\boldsymbol{Y} \mid \boldsymbol{\Psi\alpha}, \boldsymbol{K}\right) \nabla_{\boldsymbol{\alpha}} \ln \mathcal{N}\left(\boldsymbol{Y} \mid \boldsymbol{\Psi\alpha}, \boldsymbol{K}\right)^\top\right] \\
&= \boldsymbol{\Psi}^\top \boldsymbol{K}^{-1} \mathbb{E}\left[(\boldsymbol{Y} - \boldsymbol{\Psi\alpha})(\boldsymbol{Y} - \boldsymbol{\Psi\alpha})^\top\right] \boldsymbol{K}^{-1}\boldsymbol{\Psi} \\
&= \boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi}.
\end{aligned}$$

Thus, by theorem 5.4.1 we have

$$\mathbb{E}\left[(\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}})(\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}})^\top\right] \succeq (\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1} \ .$$

Finally, left-multiplying by $\boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \boldsymbol{\Sigma}^{-1}$ and right-multiplying the above by $\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{T}}\boldsymbol{x}$ gives us the statement. □

5.4.2 Lower Bound for Any Estimator $\widehat{\boldsymbol{\alpha}}$

The proof of is based on the following lemma.

Lemma 5.4.3 ([7])**.** *Let P and Q be probability measures on the same measurable space $(\Omega, \mathcal{F})$, and let $A \in \mathcal{F}$ be an arbitrary event. Then,*

$$P(A) + Q(A^c) \geq \frac{1}{2} \exp(-\mathrm{D}_{\mathrm{KL}}(P, Q)), \tag{5.5}$$

where $\mathrm{D}_{\mathrm{KL}}(P, Q) = \int_\Omega \ln\left(P(\omega)/Q(\omega)\right) \mathrm{d}P(\omega)$ denotes Kullback-Leibler divergence between P and Q and $A^c = \Omega \setminus A$ is the complement of A.

Lemma 5.4.4. *For any estimator $\widehat{\boldsymbol{\alpha}}$ of $\boldsymbol{\alpha}$ in eq. (4.1) we have*

$$\mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{\mathcal{T}} \boldsymbol{\Sigma}^{-1}(\widehat{\boldsymbol{\alpha}} - \boldsymbol{\alpha})\right)^2\right] \geq \frac{\boldsymbol{x}^\top \boldsymbol{M} \boldsymbol{x}}{16\sqrt{e}}.$$

where $\boldsymbol{M} = \boldsymbol{\mathcal{T}} \boldsymbol{\Sigma}^{-1}(\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{T}}$.

Proof. Throughout the proof let $\boldsymbol{q} = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{T}}\boldsymbol{x}$. Consider two meta-learning problems with target distributions $\mathbb{P}$ and $\mathbb{Q}$ characterized by two means: $\boldsymbol{\alpha}_{\mathbb{P}} = \mathbf{0}$ and $\boldsymbol{\alpha}_{\mathbb{Q}} = \Delta(\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{q}$ where $\Delta > 0$ is a free parameter to be tuned later on. Thus, according to our established equivalence (4.1), in these two cases targets are generated by respective models $\mathbb{P} = \mathcal{N}(\mathbf{0}, \boldsymbol{K})$ and $\mathbb{Q} = \mathcal{N}(\Delta\boldsymbol{\Psi}(\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{q}, \boldsymbol{K})$.

Recall our abbreviation $\boldsymbol{M} = \boldsymbol{\mathcal{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{T}}$. By Markov's inequality gives us

$$\begin{aligned}
\mathbb{E}_{\mathbb{P}}\left[(\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q} - \boldsymbol{\alpha}_{\mathbb{P}}^\top \boldsymbol{q})^2\right] = \mathbb{E}_{\mathbb{P}}\left[(\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q})^2\right] &\geq \frac{\Delta^2}{4}\left(\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right)^2 \mathbb{P}\left(|\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}| \geq \frac{\Delta}{2}\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right) \\
\mathbb{E}_{\mathbb{Q}}\left[(\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q} - \boldsymbol{\alpha}_{\mathbb{Q}}^\top \boldsymbol{q})^2\right] &\geq \frac{\Delta^2}{4}\left(\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right)^2 \mathbb{Q}\left(|\boldsymbol{\alpha}_{\mathbb{Q}}^\top \boldsymbol{q} - \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}| \geq \frac{\Delta}{2}\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right) \\
&\geq \frac{\Delta^2}{4}\left(\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right)^2 \mathbb{Q}\left(|\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}| < \frac{\Delta}{2}\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right)
\end{aligned}$$

where the last inequality comes using the fact that $|a - b| > |a| - |b|$ for $a, b \in \mathbb{R}$ and observing that $\boldsymbol{\alpha}_{\mathbb{Q}}^\top \boldsymbol{q} = \boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}$. Summing both inequalities above and applying lemma 5.4.3 we get

$$\begin{aligned}
\mathbb{E}_{\mathbb{P}}\left[(\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q} - \boldsymbol{\alpha}_{\mathbb{P}}^\top \boldsymbol{q})^2\right] + \mathbb{E}_{\mathbb{Q}}\left[(\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q} - \boldsymbol{\alpha}_{\mathbb{Q}}^\top \boldsymbol{q})^2\right] &\geq \frac{\Delta^2}{8}\left(\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right)^2 \cdot \exp\left(-\mathrm{D}_{\mathrm{KL}}(\mathbb{P}, \mathbb{Q})\right) \\
&\overset{(a)}{=} \frac{\Delta^2}{8}\left(\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right)^2 \cdot \exp\left(-\frac{\Delta^2}{2}\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right)
\end{aligned}$$

where step (a) follows from KL-divergence between multivariate Gaussians with the same covariance matrix. Now, using a basic fact that $2\max\{a, b\} \geq a + b$, we get that for any measure $\mathbb{P}$ given by parameter $\boldsymbol{\alpha}$ we have

$$\mathbb{E}\left[(\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q} - \boldsymbol{\alpha}^\top \boldsymbol{q})^2\right] \geq \frac{\Delta^2}{16}\left(\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right)^2 \cdot \exp\left(-\frac{\Delta^2}{2}\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right) \ .$$

The statement then follows by tuning $\Delta^2 = (\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x})^{-1}$. □

Now we prove a high-probability version of the just given inequality.

Lemma 5.4.5. *For any estimator $\widehat{\boldsymbol{\alpha}}$ of $\boldsymbol{\alpha}$ in eq. (4.1) and any $\delta \in (0, 1)$ we have*

$$\mathbb{P}\left(\left(\boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{\Sigma}^{-1}(\widehat{\boldsymbol{\alpha}} - \boldsymbol{\alpha})\right)^2 \geq \ln\left(\frac{1}{4}\cdot\frac{1}{1-\delta}\right)\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right) \geq 1 - \delta \ .$$

Proof. The proof is very similar to the proof of lemma 5.4.4 except we will not apply Markov's inequality and focus directly on giving a lower bound the deviation probabilities rather than expectations. Thus, similarly as before introduce mean parameters $\boldsymbol{\alpha}_{\mathbb{P}} = \mathbf{0}$ and $\boldsymbol{\alpha}_{\mathbb{Q}} = \Delta(\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{q}/\left(\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right)$ and their associated probability measures $\mathbb{P} = \mathcal{N}(\mathbf{0}, \boldsymbol{K})$ and $\mathbb{Q} = \mathcal{N}\left(\frac{\Delta\boldsymbol{\Psi}(\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{q}}{\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}}, \boldsymbol{K}\right)$.

Note that

$$\mathbb{P}\left(|\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}| \geq \frac{\Delta}{2}\right) = \mathbb{P}\left(|\boldsymbol{\alpha}_{\mathbb{P}}^\top \boldsymbol{q} - \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}| \geq \frac{\Delta}{2}\right) \ ,$$
$$\mathbb{Q}\left(|\boldsymbol{\alpha}_{\mathbb{Q}}^\top \boldsymbol{q} - \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}| \geq \frac{\Delta}{2}\right) \geq \mathbb{Q}\left(|\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}| < \frac{\Delta}{2}\right)$$

and so by using lemma 5.4.3 we obtain an exponential tail bound

$$\mathbb{P}\left(|\boldsymbol{\alpha}_{\mathbb{P}}^\top \boldsymbol{q} - \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}| \geq \frac{\Delta}{2}\right) + \mathbb{Q}\left(|\boldsymbol{\alpha}_{\mathbb{Q}}^\top \boldsymbol{q} - \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}| \geq \frac{\Delta}{2}\right) \geq$$
$$\geq \exp(-\mathrm{D}_{\mathrm{KL}}(\mathbb{P}\,||\,\mathbb{Q})) = \frac{1}{2}\exp\left(-\frac{\Delta^2}{\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}}\right) \ .$$

Setting the r.h.s. in the above to $2(1-\delta)$ where δ is an error probability, and solving for Δ gives us tuning

$$\Delta^2 = 2\ln\left(\frac{1}{4}\cdot\frac{1}{1-\delta}\right)\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x} \ .$$

Thus, we get

$$\mathbb{P}\left(\left(\boldsymbol{\alpha}_{\mathbb{P}}^\top \boldsymbol{q} - \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}\right)^2 \geq \frac{1}{2}\ln\left(\frac{1}{4}\cdot\frac{1}{1-\delta}\right)\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right) +$$
$$+\,\mathbb{Q}\left(\left(\boldsymbol{\alpha}_{\mathbb{Q}}^\top \boldsymbol{q} - \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}\right)^2 \geq \frac{1}{2}\ln\left(\frac{1}{4}\cdot\frac{1}{1-\delta}\right)\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right) \geq 2(1-\delta)$$

and using the fact that $2\max(a,b) \geq a+b$ we get that for any probability measure $\mathbb{P}$ given by parameter $\boldsymbol{\alpha}$ we have

$$\mathbb{P}\left(\left(\boldsymbol{\alpha}_{\mathbb{P}}^\top \boldsymbol{q} - \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{q}\right)^2 \geq \ln\left(\frac{1}{4}\cdot\frac{1}{1-\delta}\right)\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x}\right) \geq 1-\delta \ .$$

□

5.5 Proof of the Upper Bounds

Theorem 5.2.1 (restated). *For the estimator $\widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}}^{\mathrm{MLE}})$ and for any $\boldsymbol{x} \in \mathbb{R}^d$ we have*

$$\mathcal{L}(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{x} + \sigma^2.$$

Moreover for the same estimator, with probability at least $1-\delta, \delta \in (0,1)$ *we have*

$$\mathcal{L}(\boldsymbol{x}) \leq 2\ln\left(\frac{2}{\delta}\right)\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{x} + \sigma^2.$$

Proof. Recall that

$$\widehat{\boldsymbol{\alpha}}^{\text{MLE}} = (\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{Y} \ .$$

The first result follows from corollary 5.3.1 where we have to give an identity for

$$\mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}}^{\text{MLE}})\right)^2\right] \tag{5.6}$$

and the missing piece is a covariance of the estimator $\widehat{\boldsymbol{\alpha}}^{\text{MLE}}$

$$\begin{aligned}
&\mathbb{E}\left[(\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}}^{\text{MLE}})(\boldsymbol{\alpha} - \widehat{\boldsymbol{\alpha}}^{\text{MLE}})^\top\right] \\
&= (\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\text{Cov}(\boldsymbol{Y},\boldsymbol{Y})\boldsymbol{K}^{-1}\boldsymbol{\Psi}(\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1} \\
&= (\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1} \ .
\end{aligned} \tag{5.7}$$

To prove the second result we have to give a high probability upper bound on eq. (5.6).

Let $\boldsymbol{q} = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mathcal{T}}\boldsymbol{x}$ and observe that $\boldsymbol{q}^\top \widehat{\boldsymbol{\alpha}}^{\text{MLE}}$ is Gaussian (since $\boldsymbol{Y}$ is composed of Gaussian entries) with mean $\boldsymbol{q}^\top \boldsymbol{\alpha}$ by equivalence (4.1), and covariance $(\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}$ by eq. (5.7). Then, by Gaussian concentration for any error probability $\delta \in (0,1)$ we have

$$\mathbb{P}\left((\boldsymbol{q}^\top \boldsymbol{\alpha} - \boldsymbol{q}^\top \widehat{\boldsymbol{\alpha}}^{\text{MLE}})^2 \geq 2\boldsymbol{q}^\top (\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{q}\ln\left(\frac{2}{\delta}\right)\right) \leq \delta$$

which completes the proof. □

Chapter 6

Learning with Unknown Task Structure

So far we have assumed that parameters $(\sigma^2, \mathbf{\Sigma})$ characterizing the structure of environment are known, which limits the applicability of the predictor (though does not limit the lower bound). Staying within our framework, a natural idea is to estimate all the environment parameters $\mathcal{E} = (\boldsymbol{\alpha}, \sigma^2, \mathbf{\Sigma})$ by maximizing the data marginal log-likelihood

$$J(\mathcal{D}, \mathcal{E}') = \ln \int_{\mathbb{R}^{nd}} p(\mathcal{D} \,|\, \boldsymbol{\vartheta}) \, \mathrm{d}p(\boldsymbol{\vartheta} \,|\, \mathcal{E}')$$

over $\mathcal{E}'$, where $p(\mathcal{D}, \mathbf{\Theta}, \mathcal{E})$ stands for the joint distribution in the model (3.1). The above problem is non-convex. Furthermore, while the marginal distribution is available in analytic form by eq. (4.1), in preliminary experiments direct optimization proved to be numerically unstable. As such, we propose to use EM procedure [8], which is known to be a reasonable algorithm for similar settings. In this chapter we introduce EM algorithm for our setting and subsequently provide derivation of equations for the E- and M-steps of it.

6.1 EM Algorithm for Meta-Learning

EM can be derived as a procedure that maximizes a lower bound on $J(\mathcal{D}, \mathcal{E}')$: Jensen's inequality gives us that for any probability measure q on $\mathbb{R}^{nd}$,

$$J(\mathcal{D}, \mathcal{E}') \geq \int \ln \left(\frac{p(\boldsymbol{\vartheta}, \mathcal{D} \,|\, \mathcal{E}')}{q(\boldsymbol{\vartheta})} \right) \mathrm{d}q(\boldsymbol{\vartheta}).$$

This is then maximized in $\mathcal{E}'$ and q in an alternating fashion: Letting $\widehat{\mathcal{E}}_t$ to be a parameter estimate at step t, we maximize the lower bound in q for a fixed $\mathcal{E}' = \widehat{\mathcal{E}}_t$, and then obtain $\widehat{\mathcal{E}}_{t+1}$ by maximizing the lower bound in $\mathcal{E}'$ for a fixed previously obtained solution in q. Maximization in q gives us $q(\boldsymbol{\vartheta}) = p(\boldsymbol{\vartheta} \mid \mathcal{D}, \widehat{\mathcal{E}}_t)$, while maximization in $\mathcal{E}'$ yields

$$\widehat{\mathcal{E}}_{t+1} \in \arg\max_{\mathcal{E}'} \int \ln\left(p(\boldsymbol{\vartheta}, \mathcal{D} \mid \mathcal{E}')\right) \mathrm{d}p(\boldsymbol{\vartheta} \mid \mathcal{D}, \widehat{\mathcal{E}}_t) \ . \tag{6.1}$$

After some calculations (cf. section 6.2), this gives algorithm 1. During the E-step (lines 4-5), the algorithm computes the parameters of the posterior distribution $\mathcal{N}(\boldsymbol{\theta}_i \mid \widehat{\boldsymbol{\mu}}_{t,i}, \widehat{\boldsymbol{\mathcal{T}}}_{t,i})$ relying on $\widehat{\mathcal{E}}_t$, and during the M-step (lines 7–9) it estimates $\widehat{\mathcal{E}}_{t+1}$ based on $(\widehat{\boldsymbol{\mu}}_{t,i}, \widehat{\boldsymbol{\mathcal{T}}}_{t,i})$. We propose to detect convergence (not shown) by checking the relative difference between successive parameter values.

Algorithm 1 EM procedure to estimate $(\boldsymbol{\alpha}, \sigma^2, \boldsymbol{\Sigma})$

Input: Initial parameter estimates $\widehat{\mathcal{E}}_1 = (\widehat{\boldsymbol{\alpha}}_1, \widehat{\sigma}_1^2, \widehat{\boldsymbol{\Sigma}}_1)$
Output: Final parameter estimates $\widehat{\mathcal{E}}_t = (\widehat{\boldsymbol{\alpha}}_t, \widehat{\sigma}_t^2, \widehat{\boldsymbol{\Sigma}}_t)$

1: $\widehat{\boldsymbol{\mathcal{T}}}_{1,i} \leftarrow \mathbf{0},\ \widehat{\boldsymbol{\mu}}_{1,i} \leftarrow \mathbf{0} \quad i \in [n]$
2: **repeat**
3: **for** $i = 1, \ldots, n$ **do** ▷ E-step
4: $\widehat{\boldsymbol{\mathcal{T}}}_{t,i} \leftarrow \left(\widehat{\boldsymbol{\Sigma}}_t^{-1} + \widehat{\sigma}_t^{-2} \boldsymbol{X}_i^\top \boldsymbol{X}_i\right)^{-1}$
5: $\widehat{\boldsymbol{\mu}}_{t,i} \leftarrow \widehat{\boldsymbol{\mathcal{T}}}_{t,i} \left(\widehat{\boldsymbol{\Sigma}}_t^{-1} \widehat{\boldsymbol{\alpha}}_t + \widehat{\sigma}_t^{-2} \boldsymbol{X}_i^\top \boldsymbol{Y}_i\right)$
6: **end for**
7: $\widehat{\boldsymbol{\alpha}}_t \leftarrow \frac{1}{n} \sum_{i=1}^n \widehat{\boldsymbol{\mu}}_{t,i}$ ▷ M-step
8: $\widehat{\boldsymbol{\Sigma}}_t \leftarrow \frac{1}{n} \sum_{i=1}^n \left(\widehat{\boldsymbol{\mathcal{T}}}_{t,i} + (\widehat{\boldsymbol{\mu}}_{t,i} - \widehat{\boldsymbol{\alpha}}_t)(\widehat{\boldsymbol{\mu}}_{t,i} - \widehat{\boldsymbol{\alpha}}_t)^\top\right)$
9: $\widehat{\sigma}_t^2 \leftarrow \frac{1}{n} \sum_{i=1}^n \frac{1}{m_i} \left(\widehat{\mathcal{L}}_i(\widehat{\boldsymbol{\mu}}_{t,i}) + \mathrm{tr}\left(\boldsymbol{X}_i \widehat{\boldsymbol{\mathcal{T}}}_{t,i} \boldsymbol{X}_i^\top\right)\right)$
10: $t \leftarrow t + 1$
11: **until** Convergence (see discussion)

6.2 Derivation of EM Steps

Recall that our goal is to solve

$$\max_{\mathcal{E}'} \int \ln\left(p(\boldsymbol{\vartheta}, \mathcal{D} \mid \mathcal{E}')\right) \mathrm{d}p(\boldsymbol{\vartheta} \mid \mathcal{D}, \widehat{\mathcal{E}}_t) \ .$$

First, we will focus on the integral. The chain rule readily gives

$$\ln p(\boldsymbol{\Theta}, \mathcal{D} \,|\, \mathcal{E}') = \ln p(\boldsymbol{\Theta} \,|\, \mathcal{D}, \mathcal{E}') + \ln p(\boldsymbol{\Theta} \,|\, \mathcal{E}')$$

Using the same reasoning and notation as in the proof of proposition 4.2.1 we get

$$\begin{aligned}
&\int \ln p(\boldsymbol{\vartheta} \,|\, \mathcal{D}, \mathcal{E}') \,\mathrm{d}p(\boldsymbol{\vartheta} \,|\, \mathcal{D}, \widehat{\mathcal{E}}_t) = \\
&= \sum_{i=1}^{n} \sum_{j=1}^{m_i} \left(\frac{1}{2} \ln \left(\frac{1}{\sigma^2} \right) - \frac{1}{2\sigma^2} \int (Y_{i,j} - \boldsymbol{x}_{i,j}^\top \boldsymbol{\vartheta}_i)^2 \,\mathrm{d}p(\boldsymbol{\vartheta}_i \,|\, \mathcal{D}, \widehat{\mathcal{E}}_t) \right) + \mathrm{const}(\mathcal{E}') \\
&= \sum_{i=1}^{n} \sum_{j=1}^{m_i} \left(\frac{1}{2} \ln \left(\frac{1}{\sigma^2} \right) - \frac{1}{2\sigma^2} (Y_{i,j} - \boldsymbol{x}_{i,j}^\top \boldsymbol{\mu}_i)^2 - \boldsymbol{x}_{i,j}^\top \boldsymbol{\mathcal{T}}_i \boldsymbol{x}_{i,j} \right) + \mathrm{const}(\mathcal{E}')
\end{aligned}$$

using the fact that $\int (Y_{i,j} - \boldsymbol{x}_{i,j}^\top \boldsymbol{\vartheta}_i)^2 \,\mathrm{d}p(\boldsymbol{\vartheta}_i \,|\, \mathcal{D}, \widehat{\mathcal{E}}_t) = (Y_{i,j} - \boldsymbol{x}_{i,j}^\top \boldsymbol{\mu}_i)^2 + \boldsymbol{x}_{i,j}^\top \boldsymbol{\mathcal{T}}_i \boldsymbol{x}_{i,j}$ where we took $\boldsymbol{\theta}_i \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\mathcal{T}}_i)$ according the proof of proposition 4.2.1.

Now we compute the expected log-likelihood of the vector of task parameters:

$$\begin{aligned}
&\int \ln p(\boldsymbol{\vartheta} \,|\, \mathcal{E}') \,\mathrm{d}p(\boldsymbol{\vartheta} \,|\, \mathcal{D}, \widehat{\mathcal{E}}_t) = \\
&= \frac{n}{2} \ln \det \boldsymbol{\Sigma}^{-1} - \frac{1}{2} \sum_{i=1}^{n} \int (\boldsymbol{\vartheta}_i - \boldsymbol{\alpha})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{\vartheta}_i - \boldsymbol{\alpha}) \,\mathrm{d}p(\boldsymbol{\vartheta}_i \,|\, \mathcal{D}, \widehat{\mathcal{E}}_t) + \mathrm{const}(\mathcal{E}') \,.
\end{aligned}$$

M-step for σ^2. Now, note that since the likelihood of the vector of task variables $\boldsymbol{\Theta}$ does not depend on the parameter σ^2 we can solve for σ^2 based on the first order condition of the problem above. Differentiating the above equation with respect to σ^{-2} (and ignoring the constant) gives

$$\sum_{i=1}^{n} \sum_{j=1}^{m_i} \left(\sigma^2 - \left((Y_{i,j} - \boldsymbol{x}_{i,j}^\top \boldsymbol{\mu}_i)^2 + \boldsymbol{x}_{i,j}^\top \boldsymbol{\mathcal{T}}_i \boldsymbol{x}_{i,j} \right) \right) . \tag{6.2}$$

while setting the derivative to zero gives

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{m_i} \sum_{j=1}^{m_i} \left((Y_{i,j} - \boldsymbol{x}_{i,j}^\top \boldsymbol{\mu}_i)^2 + \boldsymbol{x}_{i,j}^\top \boldsymbol{\mathcal{T}}_i \boldsymbol{x}_{i,j} \right) . \tag{6.3}$$

M-step for $\boldsymbol{\alpha}$. Differentiating the objective w.r.t. $\boldsymbol{\alpha}$ (and ignoring the constant) gives $\sum_{i=1}^{n} \boldsymbol{\Sigma}^{-1} (\mathbb{E}[\boldsymbol{\theta}_i] - \boldsymbol{\alpha})$ from which we get

$$\boldsymbol{\alpha} = \sum_{i=1}^{n} \boldsymbol{\mu}_i \,. \tag{6.4}$$

M-step for $\boldsymbol{\Sigma}$. Differentiating the expected log-likelihood of the vector of task parameters with respect to $\boldsymbol{A} = \boldsymbol{\Sigma}^{-1}$ gives

$$\sum_{i=1}^{n} \operatorname{tr}(\boldsymbol{\Sigma} d\boldsymbol{A}) - \operatorname{tr} \int \left((\boldsymbol{\vartheta}_i - \boldsymbol{\alpha})(\boldsymbol{\vartheta}_i - \boldsymbol{\alpha})^\top d\boldsymbol{A} \right) \mathrm{d}p(\boldsymbol{\vartheta}_i \,|\, \mathcal{D}, \widehat{\mathcal{E}}_t) \tag{6.5}$$

from which we get

$$\boldsymbol{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[(\boldsymbol{\theta}_i - \boldsymbol{\alpha})(\boldsymbol{\theta}_i - \boldsymbol{\alpha})^\top]. \tag{6.6}$$

Finally, computing the expectation

$$\sum_{i=1}^{n} \left(\mathbb{E}[\boldsymbol{\theta}_i \boldsymbol{\theta}_i^\top] - 2\boldsymbol{\mu}_i \boldsymbol{\alpha}^\top + \boldsymbol{\alpha}\boldsymbol{\alpha}^\top \right) = \sum_{i=1}^{n} \left((\boldsymbol{\mu}_i - \boldsymbol{\alpha})(\boldsymbol{\mu}_i - \boldsymbol{\alpha})^\top + \boldsymbol{\mathcal{T}}_i \right) \tag{6.7}$$

shows the update for $\boldsymbol{\Sigma}$.

Chapter 7

Experiments

In this section we present experiments designed to verify three hypotheses: *(i)* Under ideal circumstances, the predictor $\boldsymbol{x}^\top \widehat{\boldsymbol{\theta}}_n(\widehat{\boldsymbol{\alpha}}^{\mathrm{MLE}})$ is superior to its alternatives, including biased, but unweighted regression; *(ii)* The EM-algorithm reliably recovers unknown parameters of the environment and is also suitable for representation learning; *(iii)* our distribution-dependent lower bound eq. (5.1) is numerically sharp. In addition, we briefly report on experiments with a real-world dataset. For all of the experiments we show averages and standard deviations of the mean test errors computed over 30 independent runs of that experiment.

7.1 Baselines

We consider two non-meta-learning baselines, that is `Linear Regression (All)` — Ordinary Least Squares (OLS) fitted on $\mathcal{D}^{\setminus n} = (D_i)_{i=1}^{n-1}$, which excludes the newly observed task, and `Linear Regression (Task)` — OLS fitted on a newly encountered task D_n. Next, we consider meta-learning algorithms. We report performance of the *unweighted* `Biased Regression` procedure with bias set to the least squares solution $(\sum_{i\neq n} \boldsymbol{X}_i^\top \boldsymbol{X}_i)^{-1} \sum_{i\neq n} \boldsymbol{X}_i^\top \boldsymbol{Y}_i$ and λ found by cross-validation (cf. section A.1). Note that the bias and the regularization coefficient are found on $\mathcal{D}^{\setminus n}$, while D_n is used for the final fitting. `EM Learner` is estimator (4.2) with all environment parameters found by algorithm 1 on $\mathcal{D}^{\setminus n}$. The convergence threshold was set to 10^{-6} while the maximum number of iterations was set to 10^3. Finally, we report numerical

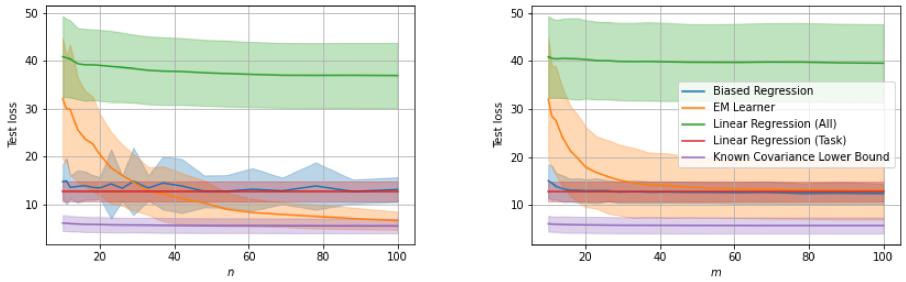

Figure 7.1: Test errors on Fourier synthetic experiment with changing number of tasks n (with $m = 10$) and number of samples per task m (with $n = 10$).

values of eq. (5.1) as `Known Covariance Lower Bound`.

7.2 Synthetic Experiments

We conduct synthetic experiments on datasets with Fourier generated features and features sampled from a d-dimensional unit sphere. In all of the synthetic experiments we have $\boldsymbol{\alpha} = \mathbf{0}, \sigma^2 = 1$ and $\boldsymbol{\Sigma}$ generated by computing $\boldsymbol{\Sigma} = \mathbf{L}\mathbf{L}^\top + \eta\boldsymbol{I}$ where $L_{ij} = \mathbb{I}\{i \geq j\} Z_{ij}$ with Z_{ij} and η sampled from the standard normal distribution. Test error is computed on 100 test tasks using 10 examples for training and 100 examples for testing. For the Fourier-features, we sample a value $u \sim \mathcal{U}(-5, 5)$ and compute features by evaluating $d = 11$ Fourier basis functions at u: $x_j = \mathbb{I}\{1 \leq j \leq 5\} \sin(\frac{j}{5}\pi u) + \mathbb{I}\{6 \leq j \leq 10\} \cos(\frac{j-5}{5}\pi u) + \mathbb{I}\{j = 11\}$, where $\mathbb{I}\{E\} = 1$ if E is true and $\mathbb{I}\{E\} = 0$ otherwise. Examples of these tasks and results of meta-learning on some of these were shown in fig. 1.1.

In fig. 7.1 we show the test errors for various meta-learners while varying the number of tasks n and task sizes m. For the 'spherical' data, the same is shown in fig. 7.2. Here, we generate $\boldsymbol{x}$ from a $d = 42$ dimensional unit sphere. In both experiments for sufficiently large number of training tasks the EM-based learner approaches the optimal estimator even when the number of examples per task is less than the dimensionality of that task.

In the context of the 'Fourier' dataset, we also experimented with generating low-rank $\boldsymbol{\Sigma}$, corresponding to the challenge of learning a low-dimensional representation, shared across the tasks. We found that the EM-based meta

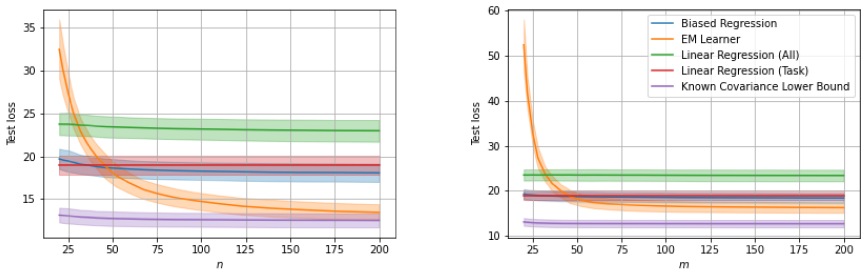

Figure 7.2: Test error on spherical synthetic experiment with changing number of tasks n (with $m = 40$) and number of samples per task m (with $n = 40$).

learner stays competitive in this setting (see section 7.4).

7.3 Real Dataset Experiment

We also conducted experiments on a real world dataset containing information about students in 139 schools in years 1985-1987 [13]. We adapt the dataset to a meta-learning problem with the goal to predict the exam score of students based on the student-specific and school-specific features. After one-hot encoding of the categorical values there are $d = 27$ features for each student. We randomly split schools into two subsets: The first, consisting of 100 schools, forms $\mathcal{D}^{\setminus n}$ (used for training the bias, λ selection, and EM). The second subset consists of 39 schools, where each school is further split into 80%/20% for the final training and testing of the meta-learners.

Results are given in fig. 7.3. We can see that while both Biased Regression and EM Learner outperform regression, their performance is very similar. This could be attributed to the fact that the features mostly contain weakly relevant information, which is confirmed by inspecting the coefficient vector.

7.4 Learning Low-Rank Representations

In this section we provide additional results for the case of low-rank structure.

7.4.1 Low Rank Structure with Fourier Features

Figure (fig. 7.4) shows the outcomes of experiments for the Fourier task but when $\mathbf{\Sigma}$ is low-rank. As can be seen, the EM based learner excels in exploiting

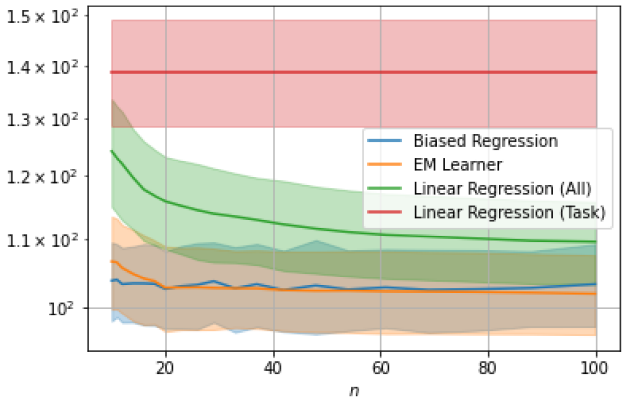

Figure 7.3: Test error on the School Dataset. Up to 100 schools are used for fitting environment-related parameters (see text for details) and the remaining 39 are used as the target task.

the low-rank structure. For this experiment we have the same setup as for the Fourier experiment, but the covariance matrix $\mathbf{\Sigma}$ is generated by computing $\mathbf{\Sigma} = \mathbf{L}\mathbf{L}^\top$ where $\mathbf{L}$ is a $d \times r$ matrix with $r = \lfloor d/2 \rfloor = 5$ and elements $L_{i,j} \sim \mathcal{N}(0, 1)$. Note that in this case we can write

$$\boldsymbol{\theta}_i = \mathbf{B}\boldsymbol{w}_i$$

for some matrix $\mathbf{B}$ of size $d \times r$ and vector $\mathbf{w}_i$ of size r sampled from multivariate normal distribution. Thus, if the matrix $\mathbf{B}$ is known or estimated during training, one can project the features $\boldsymbol{x}_{i,j}$ onto a lower-dimensional space by computing $\mathbf{B}^\top \boldsymbol{x}_{i,j}$ to speed up the adaptation to new tasks by running least-squares regression to estimate $\boldsymbol{w}_i$ instead of $\boldsymbol{\theta}_i$.

In addition to the baselines described in the main text, we compared `EM Learner` with two additional baselines: one is based on the MoM estimator [38] (not shown on the figure), and another which we refer to as `Oracle Representation`. We omit displaying the error of the method of moments estimator since for the features generated as in this experiment it is not able to perform estimation of the subspace and leads to test erros of around 60. At the same time, as shown on fig. 7.4 (left) we observe that `EM Learner` can outperform `Oracle Representation` which assumes the knowledge of the covariance matrix $\mathbf{\Sigma}$ from which it computes the subspace matrix $\mathbf{B}$ and uses it to obtain lower-dimensional representation of the features when adapting to a

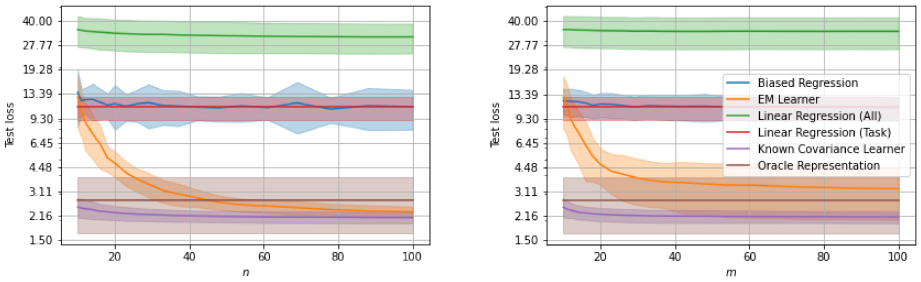

Figure 7.4: Test error when the task covariance matrix is low-rank. As usual, on the left the number of tasks is changed, on the right, the number of training datapoints (per task). When one parameter is varied, the other is set to the value of 10. Notice that the y-axis is in logarithmic scale.

new task via least-squares, as described above. It is possible for `EM Learner` to outperform this baseline because the coefficients estimated by EM are biased toward $\boldsymbol{\alpha}$ which does not happen with least squares regression in the lower dimensional subspace and this is beneficial, especially when the number of test-task training examples is small.

7.4.2 Subspace Estimation

To validate our implementation of the MoM estimator of Tripuraneni *et al.* [38] and to investigate more whether EM is preferable to the MoM estimator beyond the setting that is ideal for the EM method we considered the experimental setup of Tripuraneni *et al.*

To explain the setup, we recall that the MoM estimator computes an estimate $\hat{\boldsymbol{B}}$ of the ground truth matrix $\boldsymbol{B}$. We give the pseudocode of MoM in algorithm 2.

Algorithm 2 MoM Estimator for Learning Linear Features [38]

Input: $((\boldsymbol{x}_{1,j}, y_{1,j}))_{j=1}^{m_1}, \ldots, \left((\boldsymbol{x}_{m_{n-1},j}, y_{m_{n-1},j})\right)_{j=1}^{m_{i-1}}$ — training examples from $n-1$ past tasks, s — problem rank.

$\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top \leftarrow \text{SVD}\left(\frac{1}{M-m_n}\sum_{i=1}^{n-1}\sum_{j=1}^{m_n} y_{i,j}^2 \boldsymbol{x}_{i,j}\boldsymbol{x}_{i,j}^\top\right)$

$\hat{\boldsymbol{B}} \leftarrow [D_{1,1}\boldsymbol{u}_1, \ldots, D_{s,s}\boldsymbol{u}_s]$

return $\hat{\boldsymbol{B}}$

Tripuraneni *et al.* proves results for the *max-correlation* between $\hat{\boldsymbol{B}}$ and $\boldsymbol{B}$,

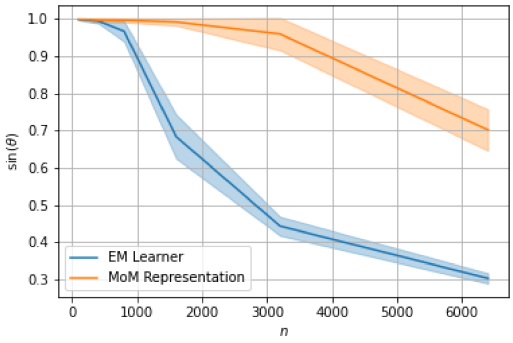

Figure 7.5: Max-correlation $d_{\max}(\hat{\boldsymbol{B}}, \boldsymbol{B})$ between the estimated matrix $\hat{\boldsymbol{B}}$ (by the respective algorithm) and the ground truth matrix $\boldsymbol{B}$ while increasing number of tasks n. The experimental protocol follows the one of previous work [38], while "MoM Representation" is found by algorithm 2. "EM Learner" is algorithm 1 with $d - s$ smallest eigenvalues of the estimated $\widehat{\boldsymbol{\Sigma}}$ clipped to 0.

and also reports experimentally measured max-correlation values between the ground truth and the MoM computed matrix. The max-correlation between matrices $\boldsymbol{A}$ and $\boldsymbol{A}'$ is based on the definition of *principal angles* between the range spaces of these matrices:

Definition 7.4.1 (Principal angles)**.** Let $\boldsymbol{A}, \boldsymbol{A}' \in \mathbb{R}^{d\times s}$ be two arbitrary rank s matrices. The principal angles $0 \leq \theta_1 \leq \ldots \leq \theta_s \leq \pi/2$ between two subspaces $\operatorname{span}(\boldsymbol{A})$ and $\operatorname{span}(\boldsymbol{A}')$, are defined recursively by

$$\cos(\theta_k) = \max_{\boldsymbol{u}_k \in \operatorname{span}(\boldsymbol{A})} \max_{\boldsymbol{v}_k \in \operatorname{span}(\boldsymbol{A}')} \boldsymbol{u}_k^\top \boldsymbol{v}_k$$
$$\text{s.t.} \quad \|\boldsymbol{u}_k\|_2 = \|\boldsymbol{v}_k\|_2 = 1 \text{ and } \boldsymbol{u}_k^\top \boldsymbol{u}_i = \boldsymbol{v}_k^\top \boldsymbol{v}_i = 0 \text{ for all } i \in [k-1], \quad k \in [s]\,.$$

Then, the max-correlation (see, e.g. [19]) between $\boldsymbol{A}, \boldsymbol{A}'$ is then defined as

$$d_{\max}(\boldsymbol{A}, \boldsymbol{A}') = \sqrt{1 - \cos^2(\theta_1)} = \sin(\theta_1)$$

where θ_1 is the largest principal angle. Intuitively, max-correlation captures how well the subspaces spanned by matrices $\boldsymbol{A}$ and $\boldsymbol{A}'$ are aligned.

To compare our EM estimator to MoM we run the EM estimator as described in algorithm 1, and once the final estimate $\widehat{\boldsymbol{\Sigma}}$ is obtained, we reduce its rank by clipping eigenvalues $\lambda_{s+1} \geq \ldots \geq \lambda_d$ to 0.

We follow the experimental setup of Tripuraneni *et al.* [38], that is, inputs are generated as $\boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{I}_d)$, while the regression model is given by eq. (3.1)

with $(\sigma^2, \boldsymbol{\Sigma}) = (1, \frac{1}{s}\boldsymbol{B}\boldsymbol{B}^\top)$. Here, columns of $\boldsymbol{B} \in \mathbb{R}^{d\times s}$ are sampled from a uniform distribution on a unit d-sphere. Finally, the number of examples per previously observed task is set as $m_1 = ... = m_{n-1} = 5$, the representation rank is $s = 5$, the input dimension is $d = 100$, and the experiment is repeated 30 times. Since we only estimate the subspace matrix we do not use the data from the test task $(\boldsymbol{X}_n, \boldsymbol{y}_n)$.

We report our results in fig. 7.5, plotting the max-correlation between $\hat{\boldsymbol{B}}$ found by the respective algorithm and $\boldsymbol{B}$, while increasing the number of tasks. We see that `EM learner` considerably outperforms `MoM Representation` in terms of the subspace estimation to the degree captured by max-correlation. While we suspect that the improvement is due to the joint optimization over the covariance of environment and the mean of the environment (the bias in biased regularization), the detailed understanding of this effect is left for the future work.

Chapter 8

Conclusion

We gave a precise, distribution-dependent characterization of the transfer risk (matching lower and upper bounds) for meta-learning in the context of linear regression when the linear regression parameter vectors for the different tasks are randomly chosen from a normal distribution. While simple, this is a natural problem to consider. In fact, we found that the variant of this when the covariance is low-rank, generalizes the "common parameter" and the "common representation" approaches to meta-learning. For the unknown covariance setting we proposed to use the EM algorithm. Encouraging experimental results confirmed that this the EM algorithm is highly effective.

While ours is the first work to derive matching, distribution-dependent lower and upper bounds, much works remains to be done: our approach to derive meta-learning algorithms based on a probabilistic model should be applicable more broadly and could lead to further interesting developments in meta-learning. The most interesting narrower question is to theoretically analyze the EM algorithm. It is known that optimization of the likelihood via EM algorithm achieves local maximum of the likelihood but unknown whether it achieves global likelihood in our setting. Additionally, one might want to study the convergence rate of EM algorithm in our setting. Doing this in the low-rank setting looks particularly interesting. An additional question for future work would be to study the same problem but in random design setting. Another direction for future work is extending the results from our work to more broader settings. In section 3.1 it was mentioned that our results could

potentially be generalized to kernels and different distributions. We hope that our work will inspire other researchers to do further work in this area.

References

[1] P. Alquier, T. T. Mai, and M. Pontil, "Regret bounds for lifelong learning," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017, pp. 261–269.

[2] R. Amit and R. Meir, "Meta-learning by adjusting priors based on extended pac-bayes theory," in *International Conference on Machine Learing (ICML)*, 2018, pp. 205–214.

[3] Y. Bai, M. Chen, P. Zhou, T. Zhao, J. D. Lee, S. Kakade, H. Wang, and C. Xiong, *How important is the train-validation split in meta-learning?* arXiv:2010.05843, 2020.

[4] J. Baxter, "Theoretical models of learning to learn," in *Learning to learn*, L. P. S. Thrun, Ed., Springer, 1998, pp. 71–94.

[5] J. Baxter, "A model of inductive bias learning," *Journal of artificial intelligence research*, vol. 12, pp. 149–198, 2000.

[6] S. Ben-David and R. Urner, "Domain adaptation as learning with auxiliary information," in *New directions in transfer and multi-task-workshop at NIPS*, 2013.

[7] J. Bretagnolle and C. Huber, "Estimation des densités: Risque minimax," *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, vol. 47, no. 2, pp. 119–137, 1979.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[9] G. Denevi, C. Ciliberto, R. Grazzi, and M. Pontil, "Learning-to-learn stochastic gradient descent with biased regularization," in *International Conference on Machine Learing (ICML)*, 2019.

[10] G. Denevi, C. Ciliberto, D. Stamos, and M. Pontil, "Learning to learn around a common mean," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2018, pp. 10 169–10 179.

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

[12] S. S. Du, J. Koushik, A. Singh, and B. Póczos, "Hypothesis transfer learning via transformation functions," in *Conference on Neural Information Processing Systems (NIPS)*, 2017, pp. 574–584.

[13] D. Dua and C. Graff, *UCI machine learning repository*, 2017. [Online]. Available: `http://archive.ics.uci.edu/ml`.

[14] A. Fallah, A. Mokhtari, and A. Ozdaglar, "On the convergence theory of gradient-based model-agnostic meta-learning algorithms," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020, pp. 1082–1092.

[15] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International Conference on Machine Learing (ICML)*, 2017.

[16] C. Finn, A. Rajeswaran, S. Kakade, and S. Levine, "Online meta-learning," *International Conference on Machine Learing (ICML)*, 2019.

[17] L. Franceschi, P. Frasconi, S. Salzo, R. Grazzi, and M. Pontil, "Bilevel programming for hyperparameter optimization and meta-learning," in *International Conference on Machine Learing (ICML)*, 2018.

[18] E. Grant, C. Finn, S. Levine, T. Darrell, and T. Griffiths, "Recasting gradient-based meta-learning as hierarchical bayes," *International Conference on Learning Representations (ICLR)*, 2018.

[19] J. Hamm and D. D. Lee, "Grassmann discriminant analysis: A unifying view on subspace-based learning," in *International Conference on Machine Learing (ICML)*, 2008.

[20] S. Hanneke and S. Kpotufe, *A no-free-lunch theorem for multi-task learning*, arXiv:2006.15785, 2020.

[21] M. Khodak, M.-F. Balcan, and A. Talwalkar, "Provable guarantees for gradient-based meta-learning," in *International Conference on Machine Learing (ICML)*, 2019, pp. 424–433.

[22] M. Khodak, M.-F. F. Balcan, and A. S. Talwalkar, "Adaptive gradient-based meta-learning methods," *Advances in Neural Information Processing Systems*, vol. 32, pp. 5917–5928, 2019.

[23] W. Kienzle and K. Chellapilla, "Personalized handwriting recognition via biased regularization," in *International Conference on Machine Learing (ICML)*, 2006, pp. 457–464.

[24] M. Konobeev, I. Kuzborskij, and C. Szepesvári, "On optimality of meta-learning in fixed-design regression with weighted biased regularization," *arXiv preprint arXiv:2011.00344*, 2020.

[25] I. Kuzborskij and F. Orabona, "Stability and Hypothesis Transfer Learning," in *International Conference on Machine Learing (ICML)*, 2013, pp. 942–950.

[26] I. Kuzborskij and F. Orabona, "Fast rates by transferring from auxiliary hypotheses," *Machine Learning*, vol. 106, no. 2, pp. 171–195, 2017.

[27] T. Lattimore and C. Szepesvári, *Bandit algorithms*. Cambridge University Press, 2018.

[28] X. Li and J. Bilmes, "A bayesian divergence prior for classiffier adaptation," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2007, pp. 275–282.

[29] J. Lucas, M. Ren, I. Kameni, T. Pitassi, and R. Zemel, "Theoretical bounds on estimation error for meta-learning," in *International Conference on Learning Representations*, 2021.

[30] A. Maurer, "Algorithmic stability and meta-learning," *Journal of Machine Learning Research*, vol. 6, no. Jun, pp. 967–994, 2005.

[31] A. Maurer, M. Pontil, and B. Romera-Paredes, "The benefit of multitask representation learning," *Journal of Machine Learning Research*, 2016.

[32] A. Pentina and C. Lampert, "A pac-bayesian bound for lifelong learning," in *International Conference on Machine Learing (ICML)*, 2014, pp. 991–999.

[33] A. Pentina and C. H. Lampert, "Lifelong learning with non-iid tasks," in *Conference on Neural Information Processing Systems (NIPS)*, 2015, pp. 1540–1548.

[34] A. Pentina and R. Urner, "Lifelong learning with weighted majority votes," in *Conference on Neural Information Processing Systems (NIPS)*, 2016, pp. 3612–3620.

[35] R. Salakhutdinov, J. Tenenbaum, and A. Torralba, "One-shot learning with a hierarchical nonparametric bayesian model," in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, 2012, pp. 195–206.

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

[37] T. Tommasi, F. Orabona, and B. Caputo, "Safety in numbers: Learning categories from few examples with multi model knowledge transfer," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

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

[39] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive svms," in *Proceedings of the 15th ACM international conference on Multimedia*, 2007, pp. 188–197.

Appendix A

Experimental Details

A.1 Selecting λ in Biased Regression

The parameter λ is selected via random search in the following way. For each of the 50 samples of λ from log-uniform distribution on interval $[0; 100]$ we perform the following procedure to estimate the risk $\widehat{L}$. Firstly, we split the training tasks into $K = 10$ groups $\mathcal{S}_1, \ldots, \mathcal{S}_K$ of (approximately) equal size and compute the estimates $\widehat{\boldsymbol{\alpha}}_k$ using the data $\mathcal{S}^{\setminus k}$ from all of the groups excluding the group k: $\mathcal{S}^{\setminus k} := \cup_{i \neq k} \mathcal{S}_i$. For each of the estimated values $\widehat{\boldsymbol{\alpha}}_k$ we perform adaptation to and testing on the tasks in the group $\mathcal{S}_k$ using the given value of λ. We split the samples of each task data $D_i \in \mathcal{S}_k$ randomly into adaptation and test sets 10 times each time such that the size of adaptation set is close to the size of adaptation sets used with the actual test data. For each of the splits we compute an estimate of the parameter vector $\widehat{\boldsymbol{\theta}}_{k,i,l}$ where k is the index of the group which was not used to estimate $\widehat{\boldsymbol{\alpha}}_k$, i is the index of a task data $D_i \in \mathcal{S}_k$, l is the index of a random split of the samples in that task into adaptation and test sets. With this parameter vector and using the test set of the task $D_i \in \mathcal{S}_k$ we can also estimate the loss $\widehat{L}_{k,i,l}$ after which all the loss values are averaged:

$$\widehat{L} = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{|\mathcal{S}^{\setminus k}|} \sum_{i: D_i \in \mathcal{S}^{\setminus k}} \frac{1}{10} \sum_{l=1}^{10} \widehat{L}_{k,i,l}.$$

At the end we select the value of λ which lead to the smallest value of $\widehat{L}$ using this cross-validation procedure.

Appendix B

Supplementary Statements

B.1 Generalization of the Lower Bound of Lucas *et al.* [29]

Consider $\mathcal{P}_{LR}$ to be the set of distributions over the set of labels $\mathcal{Y}$ with the space of input-output pairs $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$:

$$\mathcal{P}_{LR} = \{\mathcal{N}(\boldsymbol{X}\boldsymbol{\theta}, \sigma^2 \boldsymbol{I}) : \boldsymbol{\theta} \in \mathbb{B}_2(r), \boldsymbol{X} \in \mathbb{R}^{m \times d}\},$$

where $\mathbb{B}_2(r) = \{x \in \mathbb{R}^d : \|x\|_2 \leq r\}$ is the 2-norm ball of radius r and define the minimax risk $R^*_{\mathcal{P}_{LR}}$ to be

$$R^*_{\mathcal{P}_{LR}} = \inf_{\hat{\theta}} \sup_{P_1, \ldots, P_n \in \mathcal{P}_{LR}} \mathbb{E}_{\substack{S_{1:n-1} \sim P_{1:n}^m, \\ S_n \sim P_n^{m'}}} \left[\|\hat{\theta}_{S_{1:n-1}}(S_n) - \theta_{P_n}\|_2^2 \right],$$

where θ_{P_n} is the result of a mapping $\mathcal{P}_{LR} \to \Omega$ where Ω is some metric space and $\hat{\theta}$ is a two-stage estimator of θ_{P_n} that maps $S_{1:n-1} \mapsto \hat{\theta}_{S_{1:n-1}}$ which is itself a mapping from $\mathcal{Z}^{m'}$ to Ω.

Proposition B.1.1. *With the definitions as above and for $d \geq 4, r \geq 1$ we have*

$$R^*_{\mathcal{P}_{LR}} = \tilde{\Omega}\left(\frac{d}{r^2(2r)^{-d}nm + m'}\right).$$

Proof. The proof consists of two steps, we first construct a 2δ-packing of $\mathcal{P}_{LR}$. Then we upper bound the two KL divergence between two distributions of this packing and use Corollary 2 from [29].

The maximal packing number $J(\delta)$ with packing radius δ for the 2-norm ball of radius r could be bounded by

$$\left(\frac{r}{\delta}\right)^d \leq J(\delta) \leq \left(1+\frac{2r}{\delta}\right)^d.$$

Through this bound, by setting $\delta = 1/2$ we can get $(2r)^d \leq J(\delta) \leq (1+4r)^d$. Let the maximal packing set with packing radius $1/2$ be denoted by $\mathcal{V}$ and define $\boldsymbol{\theta}_i = 4\delta \boldsymbol{v}_i$ for all $\boldsymbol{v}_i \in \mathcal{V}$. Then for all $i \neq j$ we have

$$\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\|_2 = 4\delta\|\boldsymbol{v}_i - \boldsymbol{v}_j\|_2 \geq 2\delta.$$

Next, we bound the KL divergences:

$$\begin{aligned}
\mathrm{D}_{\mathrm{KL}}(P_i, P_j) &= \frac{1}{2\sigma^2}\|\boldsymbol{X}_i\boldsymbol{\theta}_i - \boldsymbol{X}_j\boldsymbol{\theta}_j\|_2^2 \\
&= \frac{1}{2\sigma^2}\left(\boldsymbol{\theta}_i^\top \boldsymbol{X}_i^\top \boldsymbol{X}_i\boldsymbol{\theta}_i + \boldsymbol{\theta}_j^\top \boldsymbol{X}_j^\top \boldsymbol{X}_j\boldsymbol{\theta}_j - 2\boldsymbol{\theta}_i^\top \boldsymbol{X}_i\boldsymbol{X}_j\boldsymbol{\theta}_j\right) \\
&\leq \frac{1}{2\sigma^2}\left(n_i\gamma_i^2\|\boldsymbol{\theta}_i\|_2^2 + n_j\gamma_j^2\|\boldsymbol{\theta}_j\|_2^2 - 2\boldsymbol{\theta}_i^\top \boldsymbol{X}_i^\top \boldsymbol{X}_j\boldsymbol{\theta}_j\right),
\end{aligned}$$

where $\gamma_i = \sup_\theta \frac{\|\boldsymbol{X}_i\boldsymbol{\theta}_i\|_2}{\sqrt{n_i}\|\boldsymbol{\theta}_i\|_2}$. Denote $n = \max_k n_k$ and $\gamma = \max_k \gamma_k$, then

$$\begin{aligned}
\mathrm{D}_{\mathrm{KL}}(P_i, P_j) &\leq \frac{n\gamma^2}{2\sigma^2}\left(\|\boldsymbol{\theta}_i\|_2^2 + \|\boldsymbol{\theta}_j\|_2^2 - \frac{2}{n\gamma^2}\boldsymbol{\theta}_i^\top \boldsymbol{X}_i^\top \boldsymbol{X}_j\boldsymbol{\theta}_i\right) \\
&\leq \frac{n\gamma^2}{2\sigma^2}\left(\|\boldsymbol{\theta}_i\|_2^2 + \|\boldsymbol{\theta}_j\|_2^2 + 2\|\boldsymbol{\theta}_i\|\|\boldsymbol{\theta}_j\|\right) \\
&= \frac{n\gamma^2}{2\sigma^2}(\|\boldsymbol{\theta}_i\|_2 + \|\boldsymbol{\theta}_j\|_2)^2 \leq \frac{32n\gamma^2\delta^2r^2}{\sigma^2},
\end{aligned}$$

where the second line is derived using the Cauchy-Schwarz inequality and the final inequality uses $\|\boldsymbol{\theta}_i\| = 4\delta\|\boldsymbol{v}_i\| \leq 4\delta r$.

Next, using Corollary 2 from [29] we get

$$R^*_{\mathcal{P}_{LR}} \geq \delta^2\left(1 - \frac{mn((2r)^d-1)^{-1} + m')32\gamma^2\delta^2r^2/\sigma^2 + 1}{d(1+\log_2(r))}\right),$$

and choosing

$$\delta^2 = \frac{d(1+\log_2(r))\sigma^2}{64\gamma^2r^2(mn(2r)^d-1)^{-1} + m'}$$

leads to

$$\begin{aligned}
R^*_{\mathcal{P}_{LR}} &\geq \frac{d(1+\log_2(r))\sigma^2}{64\gamma^2\delta^2r^2((2r)^d-1)^{-1} + m'}\left(1 - \frac{d(1+\log_2(r))/2 + 1}{d(1+\log_2(r))}\right) \\
&\geq \frac{d(1+\log_2(r))\sigma^2}{256\gamma^2\delta^2r^2((2r)^d-1)^{-1} + 4m'},
\end{aligned}$$

where to get the last inequality we used the facts that $d \geq 4$ and $r \geq 1$. $\square$

B.2 Special Cases of Our Lower Bounds

Proposition B.2.1. *For $\boldsymbol{M}$ (see eq. (5.1)) we have*

$$\boldsymbol{M} = \sigma^4 \cdot \left(\boldsymbol{\Sigma}\boldsymbol{X}_n^\top \boldsymbol{X}_n + \sigma^2 \boldsymbol{I}\right)^{-1} \boldsymbol{A}^{-1} \left(\boldsymbol{\Sigma}\boldsymbol{X}_n^\top \boldsymbol{X}_n + \sigma^2 \boldsymbol{I}\right)^{-1}$$

where we denote

$$\boldsymbol{A} = \sum_{i=1}^{n} \boldsymbol{X}_i^\top (\boldsymbol{X}_i \boldsymbol{\Sigma} \boldsymbol{X}_i^\top + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{X}_i \ .$$

Proof. Recall that

$$\boldsymbol{M} = \boldsymbol{\mathcal{T}} \boldsymbol{\Sigma}^{-1} \left(\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1} \boldsymbol{\Psi}\right)^{-1} \boldsymbol{\Sigma}^{-1} \boldsymbol{\mathcal{T}}$$

and observe that

$$\boldsymbol{K}^{-1} = \begin{bmatrix} (\boldsymbol{X}_1 \boldsymbol{\Sigma} \boldsymbol{X}_1^\top + \sigma^2 \boldsymbol{I})^{-1} & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{0} & (\boldsymbol{X}_2 \boldsymbol{\Sigma} \boldsymbol{X}_2^\top + \sigma^2 \boldsymbol{I})^{-1} & \dots & \boldsymbol{0} \\ \vdots & & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \dots & (\boldsymbol{X}_n \boldsymbol{\Sigma} \boldsymbol{X}_n^\top + \sigma^2 \boldsymbol{I})^{-1} \end{bmatrix}$$

which in turn implies

$$\boldsymbol{\Psi}^\top \boldsymbol{K}^{-1} \boldsymbol{\Psi} = \sum_{i=1}^{n} \boldsymbol{X}_i^\top (\boldsymbol{X}_i \boldsymbol{\Sigma} \boldsymbol{X}_i^\top + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{X}_i \ .$$

On the other hand,

$$\begin{aligned} \boldsymbol{\mathcal{T}} \boldsymbol{\Sigma}^{-1} &= \left(\boldsymbol{\Sigma}^{-1} + \frac{1}{\sigma^2} \boldsymbol{X}_n^\top \boldsymbol{X}_n\right)^{-1} \boldsymbol{\Sigma}^{-1} \\ &= \sigma^2 \left(\sigma^2 \boldsymbol{I} + \boldsymbol{\Sigma} \boldsymbol{X}_n^\top \boldsymbol{X}_n\right)^{-1} \ . \end{aligned}$$

Combining the above gives the statement. □

Lemma B.2.2. *In the following assume that $\boldsymbol{X}_i^\top \boldsymbol{X}_i = \frac{m_i}{d} \boldsymbol{I}$ for all i. Let $\lambda_j(\boldsymbol{\Sigma})$ be the jth eigenvalue of $\boldsymbol{\Sigma}$. Then,*

$$\lambda_j(\boldsymbol{M}) = \sigma^4 \cdot \frac{d^2}{\left(m_n \lambda_j(\boldsymbol{\Sigma}) + d\sigma^2\right)^2} \cdot \frac{HM\left(\lambda_j(\boldsymbol{\Sigma}) + \frac{d\sigma^2}{m_i}\right)_{i=1}^{n}}{n}$$

where $HM(z_i)_{i=1}^n$ denotes the harmonic mean of sequence $(z_i)_{i=1}^n$. Moreover,

$$\lambda_j(\boldsymbol{\mathcal{T}}) = \frac{d\sigma^2 \lambda_j(\boldsymbol{\Sigma})}{d\sigma^2 + m_n \lambda_j(\boldsymbol{\Sigma})} \ .$$

Finally, the eigenvectors of $\boldsymbol{M}$ and $\boldsymbol{\mathcal{T}}$ coincide with the eigenvectors of $\boldsymbol{\Sigma}$.

Proof. We first characterize eigenvalues of matrix $\boldsymbol{M}$. By proposition B.2.1,

$$\boldsymbol{M} = \sigma^4 \cdot \left(\boldsymbol{\Sigma}\boldsymbol{X}_n^\top \boldsymbol{X}_n + \sigma^2 \boldsymbol{I}\right)^{-1} \boldsymbol{A}^{-1} \left(\boldsymbol{\Sigma}\boldsymbol{X}_n^\top \boldsymbol{X}_n + \sigma^2 \boldsymbol{I}\right)^{-1} .$$

We start with $\boldsymbol{A}^{-1}$, and by the spectral theorem, $\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top$ for some unitary $\boldsymbol{U}$ and diagonal $\boldsymbol{\Lambda}$:

$$\begin{aligned}
\boldsymbol{A}^{-1} &= \left(\sum_{i=1}^n \boldsymbol{X}_i^\top (\boldsymbol{X}_i \boldsymbol{\Sigma} \boldsymbol{X}_i^\top + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{X}_i\right)^{-1} \\
&= \left(\sum_{i=1}^n (\boldsymbol{\Sigma}\boldsymbol{X}_i^\top \boldsymbol{X}_i + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{X}_i^\top \boldsymbol{X}_i\right)^{-1} \\
&= \left(\sum_{i=1}^n \left(\boldsymbol{\Sigma} \cdot \frac{m_i}{d} + \sigma^2 \boldsymbol{I}\right)^{-1} \frac{m_i}{d}\right)^{-1} \\
&= \left(\sum_{i=1}^n \left(\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top \cdot \frac{m_i}{d} + \sigma^2 \boldsymbol{I}\right)^{-1} \frac{m_i}{d}\right)^{-1} \\
&= \boldsymbol{U} \left(\sum_{i=1}^n \left(\boldsymbol{\Lambda} + \frac{d\sigma^2}{m_i} \cdot \boldsymbol{I}\right)^{-1}\right)^{-1} \boldsymbol{U}^\top .
\end{aligned}$$

Now,

$$\begin{aligned}
\left(\boldsymbol{\Sigma}\boldsymbol{X}_n^\top \boldsymbol{X}_n + \sigma^2 \boldsymbol{I}\right)^{-1} &= \left(\boldsymbol{\Sigma} \cdot \frac{m_n}{d} + \sigma^2 \boldsymbol{I}\right)^{-1} \\
&= \left(\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top \cdot \frac{m_n}{d} + \sigma^2 \boldsymbol{I}\right)^{-1} \\
&= \boldsymbol{U} \left(\boldsymbol{\Lambda} \cdot \frac{m_n}{d} + \sigma^2 \boldsymbol{I}\right)^{-1} \boldsymbol{U}^\top .
\end{aligned}$$

Thus,

$$\boldsymbol{M} = \boldsymbol{U} \left(\left(\boldsymbol{\Lambda} \cdot \frac{m_n}{d} + \sigma^2 \boldsymbol{I}\right)^2 \sum_{i=1}^n \left(\boldsymbol{\Lambda} + \frac{d\sigma^2}{m_i}\right)^{-1}\right)^{-1} \boldsymbol{U}^\top .$$

and moreover the jth eigenvalue of $\boldsymbol{M}$ is

$$\begin{aligned}
\lambda_j(\boldsymbol{M}) &= \frac{1}{\left(\frac{m_n}{d}\lambda_j(\boldsymbol{\Sigma}) + \sigma^2\right)^2} \cdot \frac{1}{\sum_{i=1}^n \frac{1}{\lambda_j(\boldsymbol{\Sigma}) + \frac{d\sigma^2}{m_i}}} \\
&= \frac{1}{\left(\frac{m_n}{d}\lambda_j(\boldsymbol{\Sigma}) + \sigma^2\right)^2} \cdot \frac{\mathrm{HM}\left(\lambda_j(\boldsymbol{\Sigma}) + \frac{d\sigma^2}{m_i}\right)_{i=1}^n}{n}
\end{aligned}$$

where recall that $\mathrm{HM}(z_i)_{i=1}^n$ denotes the harmonic mean of sequence $(z_i)_{i=1}^n$.

Using the same arguments as above

$$\begin{aligned}\boldsymbol{\mathcal{T}} &= \left(\boldsymbol{\Sigma}^{-1} + \frac{1}{\sigma^2}\boldsymbol{X}_n^\top \boldsymbol{X}_n\right)^{-1} \\ &= \left(\boldsymbol{U}\boldsymbol{\Lambda}^{-1}\boldsymbol{U}^\top + \frac{m_n}{d\sigma^2}\right)^{-1}\end{aligned}$$

and so

$$\lambda_j(\boldsymbol{\mathcal{T}}) = \frac{1}{\frac{1}{\lambda_j(\boldsymbol{\Sigma})} + \frac{m_n}{d\sigma^2}} = \frac{d\sigma^2\lambda_j(\boldsymbol{\Sigma})}{d\sigma^2 + m_n\lambda_j(\boldsymbol{\Sigma})} \ .$$

Finally, in both cases of $\boldsymbol{M}$ and $\boldsymbol{\mathcal{T}}$ we observe that their eigenvectors are eigenvectors of $\boldsymbol{\Sigma}$. □

Proposition 5.1.2 (restated). *In the following assume that $\boldsymbol{X}_i^\top \boldsymbol{X}_i = \frac{m_i}{d}\boldsymbol{I}$ for all i. For $\boldsymbol{\Sigma} = \tau^2\boldsymbol{I}$, any $\boldsymbol{x} \in \mathbb{R}^d$, and any $c > 0$,*

$$c\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{x} + \sigma^2 = c \cdot \frac{H_{\tau^2}}{n} \cdot \frac{d^2\sigma^4}{\left(\tau^2 m_n + d\sigma^2\right)^2} \cdot \|\boldsymbol{x}\|^2 + \frac{d\sigma^2\tau^2}{\tau^2 m_n + d\sigma^2} \cdot \|\boldsymbol{x}\|^2 + \sigma^2 \ .$$

where H_{τ^2} is a harmonic mean of the sequence $\left(\tau^2 + \frac{d\sigma^2}{m_i}\right)_{i=1}^n$.

Moreover, let $\boldsymbol{\Sigma}$ be a PSD matrix of rank $s \le d$ with eigenvalues $\lambda_1 \ge \ldots \ge \lambda_s > 0$. Then for any $\boldsymbol{x} \in \mathbb{R}^d$ and any $c > 0$,

$$\begin{aligned}c\boldsymbol{x}^\top \boldsymbol{M}\boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}\boldsymbol{x} + \sigma^2 \ge c \cdot \frac{H_{\lambda_s}}{n} \cdot \frac{d^2\sigma^4}{\left(\lambda_1 m_n + d\sigma^2\right)^2} \cdot \|\boldsymbol{x}\|^2_{\boldsymbol{P}_s^\top \boldsymbol{P}_s} \\ + \frac{d\sigma^2\lambda_s}{\lambda_s m_n + d\sigma^2} \cdot \|\boldsymbol{x}\|^2_{\boldsymbol{P}_s^\top \boldsymbol{P}_s} + \sigma^2\end{aligned}$$

where $\boldsymbol{P}_s = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_s]^\top$ and $(\boldsymbol{u}_j)_{j=1}^s$ are eigenvectors of $\boldsymbol{\Sigma}$.

Proof. Recalling that by proposition B.2.1,

$$\boldsymbol{M} = \sigma^4 \cdot \left(\boldsymbol{\Sigma}\boldsymbol{X}_n^\top \boldsymbol{X}_n + \sigma^2\boldsymbol{I}\right)^{-1} \boldsymbol{A}^{-1} \left(\boldsymbol{\Sigma}\boldsymbol{X}_n^\top \boldsymbol{X}_n + \sigma^2\boldsymbol{I}\right)^{-1} \ .$$

and using lemma B.2.2 with $\boldsymbol{\Sigma} = \tau^2\boldsymbol{I}$ we get the first result.

Now we turn to the low-rank case. We start by considering a PSD matrix $\boldsymbol{\Sigma}_\varepsilon$ with s eigenvalues $\lambda_1 \ge \ldots \ge \lambda_s > 0$ and remaining $d - s$ are $\varepsilon > 0$. Denote also by $\boldsymbol{M}_\varepsilon$, $\boldsymbol{\mathcal{T}}_\varepsilon$ matrices w.r.t. $\boldsymbol{\Sigma}_\varepsilon$. The idea is to lower bound $\boldsymbol{x}^\top \boldsymbol{M}_\varepsilon \boldsymbol{x}$ and $\boldsymbol{x}^\top \boldsymbol{\mathcal{T}}_\varepsilon \boldsymbol{x}$ and then analyze a limiting behavior as $\varepsilon \to 0$.

By lemma B.2.2, $\boldsymbol{M}_\varepsilon$, $\boldsymbol{\mathcal{T}}_\varepsilon$, and $\boldsymbol{\Sigma}_\varepsilon$ share the same eigenvectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_s$, and so

$$
\begin{aligned}
&c\boldsymbol{x}^\top \boldsymbol{M}_\varepsilon \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}_\varepsilon \boldsymbol{x} = \\
&= c \sum_{j=1}^{d} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 \lambda_j(\boldsymbol{M}_\varepsilon) + \sum_{j=1}^{d} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 \lambda_j(\boldsymbol{\mathcal{T}}_\varepsilon) \\
&= c \cdot \sum_{j=1}^{s} \frac{H_{\lambda_j}}{n} \cdot \frac{\sigma^4}{\left(\lambda_j \frac{m_n}{d} + \sigma^2\right)^2} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 + c \cdot \underbrace{\frac{H_\varepsilon}{n} \cdot \frac{\sigma^4}{\left(\varepsilon \frac{m_n}{d} + \sigma^2\right)^2}}_{(a)} \left(\sum_{j=s+1}^{d} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 \right) \\
&\quad + \sum_{j=1}^{s} \frac{\sigma^2 \lambda_j}{\lambda_j \frac{m_n}{d} + \sigma^2} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 + \frac{\sigma^2 \varepsilon}{\varepsilon \frac{m_n}{d} + \sigma^2} \left(\sum_{j=s+1}^{d} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 \right) \ .
\end{aligned}
$$

Now,

$$
\begin{aligned}
&\lim_{\varepsilon \to 0} \left(c\boldsymbol{x}^\top \boldsymbol{M}_\varepsilon \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{\mathcal{T}}_\varepsilon \boldsymbol{x}\right) \\
&= c \cdot \sum_{j=1}^{s} \frac{H_{\lambda_j}}{n} \cdot \frac{\sigma^4}{\left(\lambda_j \frac{m_n}{d} + \sigma^2\right)^2} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 \\
&\quad + \frac{d\sigma^2}{M} \sum_{j=s+1}^{d} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 + \sum_{j=1}^{s} \frac{\sigma^2 \lambda_j}{\lambda_j \frac{m_n}{d} + \sigma^2} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 \\
&\geq c \cdot \frac{H_{\lambda_s}}{n} \cdot \frac{\sigma^4}{\left(\lambda_1 \frac{m_n}{d} + \sigma^2\right)^2} \sum_{j=1}^{s} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2 + \frac{\sigma^2 \lambda_s}{\lambda_s \frac{m_n}{d} + \sigma^2} \sum_{j=1}^{s} \left(\boldsymbol{u}_j^\top \boldsymbol{x}\right)^2
\end{aligned}
$$

where we note that the limit of term (a) is handled as

$$
\lim_{\varepsilon \to 0} \frac{1}{\sum_{i=1}^{n} \frac{1}{\varepsilon + \frac{d\sigma^2}{m_i}}} \cdot \frac{\sigma^4}{\left(\varepsilon \frac{m_n}{d} + \sigma^2\right)^2} = \frac{d\sigma^2}{M} \geq 0 \ .
$$

□