Learning-to-Learn Stochastic Gradient Descent with Biased Regularization

Giulia Denevi [1 2]   **Carlo Ciliberto** [3 4]   **Riccardo Grazzi** [1 4]   **Massimiliano Pontil** [1 4]

Abstract

We study the problem of learning-to-learn: infer-ring a learning algorithm that works well on a fam-ily of tasks sampled from an unknown distribution. As class of algorithms we consider Stochastic Gra-dient Descent (SGD) on the true risk regularized by the square euclidean distance from a bias vec-tor. We present an average excess risk bound for such a learning algorithm that quantifies the po-tential benefit of using a bias vector with respect to the unbiased case. We then propose a novel meta-algorithm to estimate the bias term online from a sequence of observed tasks. The small memory footprint and low time complexity of our approach makes it appealing in practice while our theoretical analysis provides guarantees on the generalization properties of the meta-algorithm on new tasks. A key feature of our results is that, when the number of tasks grows and their vari-ance is relatively small, our learning-to-learn ap-proach has a significant advantage over learning each task in isolation by standard SGD without a bias term. Numerical experiments demonstrate the effectiveness of our approach in practice.

1. Introduction

The problem of learning-to-learn (LTL) (Baxter, 2000; Thrun & Pratt, 1998) is receiving increasing attention in recent years, due to its practical importance (Finn et al., 2017; Franceschi et al., 2018; Ravi & Larochelle, 2017) and the theoretical challenge of statistically principled and efficient solutions (Alquier et al., 2017; Balcan et al., 2015; Maurer et al., 2016; Pentina & Lampert, 2014; Denevi et al., 2018a;b; Gupta & Roughgarden, 2017). The principal aim of LTL is to design a meta-learning algorithm to select a supervised learning algorithm that is well suited to learn tasks from a prescribed family. To highlight the difference between the meta-learning algorithm and the learning algo-rithm, throughout the paper we will refer to the latter as the *inner* or *within-task* algorithm.

The meta-algorithm is trained from a sequence of datasets, associated with different learning tasks sampled from a meta-distribution (also called *environment* in the literature). The performance of the selected inner algorithm is measured by the *transfer risk* (Baxter, 2000; Maurer, 2005), that is, the average risk of the algorithm, trained on a random dataset from the same environment. A key insight is that, when the learning tasks share specific similarities, the LTL framework provides a means to leverage such similarities and select an inner algorithm of low transfer risk.

In this work, we consider environments of linear regres-sion or binary classification tasks and we assume that the associated weight vectors are all close to a common vec-tor. Because of the increasing interest in low computational complexity procedures, we focus on the family of within-task algorithms given by Stochastic Gradient Descent (SGD) working on the regularized true risk. Specifically, motivated by the above assumption on the environment, we consider as regularizer the square distance of the weight vector to a bias vector, playing the role of a common mean among the tasks. Knowledge of this common mean can substantially facilitate the inner algorithm and the main goal of this paper is to design a meta-algorithm to learn a good bias that is supported by both computational and statistical guarantees.

Contributions. The first contribution of this work is to show that, when the variance of the weight tasks' vectors sampled from the environment is small, SGD regularized with the "right" bias yields a model with smaller error than its unbiased counterpart. The latter approach does not ex-ploit the relatedness among the tasks, and it corresponds to learning the tasks in isolation – also known as independent task learning (ITL). The second and principal contribution of this work is to propose a meta-algorithm that estimates the bias term, so that the transfer risk of the corresponding SGD algorithm is as small as possible. We consider the setting in which we receive in input a sequence of datasets and we propose an online meta-algorithm which efficiently updates the bias term used by the inner SGD algorithm. Our meta-algorithm consists in applying a (meta) SGD algo-

[1]Istituto Italiano di Tecnologia, Genoa, Italy [2]University of Genoa, Genoa, Italy [3]Imperial College of London, London, United Kingdom [4]University College London, London, United Kingdom. Correspondence to: Giulia Denevi <giulia.denevi@iit.it>.

rithm to a proxy of the transfer risk, given by the expected minimum regularized empirical risk of a task. We provide a bound on the statistical performance of the biased inner SGD algorithm found by our procedure. It establishes that, when the number of observed tasks grows and the variance of the tasks' weight vectors is significantly smaller than their second moment, then, running the inner SGD algorithm with the estimated bias brings an improvement in comparison to learning the tasks in isolation with no bias. The bound is coherent with the state-of-the-art LTL analysis for other families of algorithms, but it applies for the first time to a fully online meta-algorithm. Our results holds for Lipschitz loss functions both in the regression and binary classification setting.

Our proof techniques combines ideas from online learning, stochastic and convex optimization, with tools from LTL. A key insight in our approach is to exploit the inner SGD algorithm to compute an approximate subgradient of the surrogate objective, in a such way that the degree of approximation can be controlled, without affecting the overall performance or the computational cost of the meta-algorithm.

Paper Organization. We start by recalling in Sec. 2 the basic concepts of LTL. In Sec. 3 we cast the problem of choosing a right bias term in SGD on the regularized objective in the LTL framework. Thanks to this formulation, in Sec. 4 we characterize the situations in which SGD with the right bias term is beneficial in comparison to SGD with no bias. In Sec. 5 we propose an online meta-algorithm to estimate the bias vector from a sequence of datasets and we analyze its statistical properties. In Sec. 6 we report on the empirical performance of the proposed approach while in Sec. 7 we discuss future research directions.

Previous Work. Online LTL (Alquier et al., 2017; Denevi et al., 2018a;b; Pentina & Urner, 2016) has received limited attention and is less developed than standard LTL approaches, in which the data are processed in one batch as opposed to incrementally, see for instance (Baxter, 2000; Maurer, 2009; Maurer et al., 2013; 2016; Pentina & Lampert, 2014). The idea of introducing a bias in the learning algorithm is not new, see e.g. (Denevi et al., 2018b; Kuzborskij & Orabona, 2017; Pentina & Lampert, 2014) and Sec. 3. In this work, we consider the family of inner SGD algorithms with biased regularization and we develop a theoretically grounded meta-learning algorithm to find the bias. Differently from others online methods (Alquier et al., 2017; Denevi et al., 2018a), our approach does not need to keep previous training points in memory and it runs online both across and within the tasks. As a result, both the low space and time complexity are the strengths of our method. We finally point out the recent related work by Khodak et al. (2019) and Finn et al. (2019), that was brought to our attention after completion of the present work.

2. Preliminaries

In this section, we recall the standard supervised (i.e. single-task) learning setting and the learning-to-learn setting.

We first introduce some notation used throughout this work. Let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ be the data space, where $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{Y} \subseteq \mathbb{R}$ (regression) or $\mathcal{Y} = \{-1, +1\}$ (binary classification). We consider linear supervised learning tasks μ, namely distributions over $\mathcal{Z}$, parametrized by a weight vector $w \in \mathbb{R}^d$. We measure the performance by a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ such that, for any $y \in \mathcal{Y}$, $\ell(\cdot, y)$ is convex and closed. Finally, for any positive $k \in \mathbb{N}$, we let $[k] = \{1, \dots, k\}$ and, we denote by $\langle \cdot, \cdot \rangle$ and $\| \cdot \|$ the standard inner product and euclidean norm. In the rest of this work, when specified, we make the following assumptions.

Assumption 1 (Bounded Inputs). *Let* $\mathcal{X} \subseteq \mathcal{B}(0, R)$, *where* $\mathcal{B}(0, R) = \{x \in \mathbb{R}^d : \|x\| \leq R\}$, *for some radius* $R \geq 0$.

Assumption 2 (Lipschitz Loss). *Let* $\ell(\cdot, y)$ *be* L-Lipschitz *for any* $y \in \mathcal{Y}$.

For example, for any $y, \hat{y} \in \mathcal{Y}$, the absolute loss $\ell(\hat{y}, y) = |\hat{y} - y|$ and the hinge loss $\ell(\hat{y}, y) = \max\{0, 1 - y\hat{y}\}$ are both 1-Lipschitz. We now briefly recall the main notion of single-task learning.

2.1. Single-Task Learning

In standard linear supervised learning, the goal is to learn a linear functional relation $f_w : \mathcal{X} \to \mathcal{Y}$, $f_w(\cdot) = \langle \cdot, w \rangle$ between the input space $\mathcal{X}$ and the output space $\mathcal{Y}$. This target can be reformulated as that of finding a weight vector w_μ minimizing the *expected risk* (or true risk)

$$\mathcal{R}_\mu(w) = \mathbb{E}_{(x,y) \sim \mu} \, \ell(\langle x, w \rangle, y) \qquad (1)$$

over the *entire* space $\mathbb{R}^d$. The expected risk measures the prediction error that the weight vector w incurs on average with respect to points sampled from the distribution μ. In practice, the task μ is unknown and only partially observed by a corresponding dataset of n i.i.d. points $Z_n = (z_i)_{i=1}^n \sim \mu^n$, where, for every $i \in [n]$, $z_i = (x_i, y_i) \in \mathcal{Z}$. In the sequel, we often use the more compact notation $Z_n = (X_n, \mathbf{y}_n)$, where $X_n \in \mathbb{R}^{n \times d}$ is the matrix containing the input vectors x_i as rows and $\mathbf{y}_n \in \mathbb{R}^n$ is the vector with entries given by the labels y_i. A *learning algorithm* is a function $A : \cup_{n \in \mathbb{N}} \mathcal{Z}^n \to \mathbb{R}^d$ that, given such a *training* dataset $Z_n \in \mathcal{Z}^n$, returns a "good" estimator, that is, in our case, a weight vector $A(Z_n) \in \mathbb{R}^d$, whose expected risk is small and tends to the minimum of Eq. (1) as n increases.

2.2. Learning-to-Learn (LTL)

In the LTL framework, we assume that each learning task μ we observe is sampled from an *environment* ρ, that is a (meta-)distribution on the set of probability distributions on

$\mathcal{Z}$. The goal is to select a learning algorithm (hence the name *learning-to-learn*) that is well suited to the environment.

Specifically, we consider the following setting. We receive a stream of tasks $\mu_1, \ldots, \mu_T$, which are independently sampled from the ρ and only partially observed by corresponding i.i.d. datasets $Z_n^{(1)}, \ldots, Z_n^{(T)}, \ldots$ each formed by n datapoints. Starting from these datasets, we wish to learn an algorithm A, such that, when we apply it on a new dataset (composed by n points) sampled from a new task $\mu \sim \rho$, the corresponding true risk is low. We reformulate this target into requiring that algorithm A trained with n points[1] over the environment ρ, has small *transfer risk*

$$\mathcal{E}_n(A) = \mathbb{E}_{\mu \sim \rho} \mathbb{E}_{Z_n \sim \mu^n} \mathcal{R}_\mu(A(Z_n)). \tag{2}$$

The transfer risk measures the expected true risk that the inner algorithm A, trained on the dataset Z_n, incurs *on average with respect to the distribution of tasks μ sampled from ρ*. Therefore, the process of learning a learning algorithm is a meta-learning one, in that the inner learning algorithm is applied to tasks from the environment and then chosen from a sequence of training tasks (datasets) in attempt to minimize the transfer risk.

3. SGD on the Biased Regularized Risk

In this section, we introduce the LTL framework for the family of within-task algorithms we analyze in this work.

We consider a family of learning algorithms A_h parametrized by a bias vector $h \in \mathbb{R}^d$. The idea of introducing a bias in a specific family of learning algorithms is not new in the LTL literature, see e.g. (Denevi et al., 2018b; Kuzborskij & Orabona, 2017; Pentina & Lampert, 2014) and references therein. A natural choice is given by regularized empirical risk minimization, in which we introduce a bias vector in the square norm regularizer – which we simply refer to as ERM throughout – namely

$$A_h^{\mathrm{ERM}}(Z_n) \equiv w_h(Z_n) = \underset{w \in \mathbb{R}^d}{\operatorname{argmin}} \ \mathcal{R}_{Z_n, h}(w), \tag{3}$$

where, for any $w, h \in \mathbb{R}^d$, $\lambda > 0$, we have defined the empirical error and its biased regularized version as

$$\mathcal{R}_{Z_n}(w) = \frac{1}{n} \sum_{k=1}^n \ell_k(\langle x_k, w \rangle)$$

$$\mathcal{R}_{Z_n, h}(w) = \mathcal{R}_{Z_n}(w) + \frac{\lambda}{2} \|w - h\|^2. \tag{4}$$

Intuitively, if the weight vectors w_μ of the tasks sampled from ρ are close to each other, then running ERM with

[1]In order to simplify the presentation, we assume that all datasets are composed by the same number of points n. The general setting can be addressed by introducing the slightly different notion of transfer risk $\mathcal{E}(A) = \mathbb{E}_{(n,\mu) \sim \rho} \mathbb{E}_{Z_n \sim \mu^n} \mathcal{R}_\mu(A(Z_n))$.

Algorithm 1 Within-Task Algorithm: SGD on the Biased Regularized True Risk

Input $\lambda > 0$ regularization parameter, h bias, μ task

Initialization $w_h^{(1)} = h$

For $k = 1$ to n

    Receive $(x_k, y_k) \sim \mu$

    Build $\ell_{k,h}(\cdot) = \ell_k(\langle x_k, \cdot \rangle) + \frac{\lambda}{2}\| \cdot - h\|^2$

    Define $\gamma_k = 1/(k\lambda)$

    Compute $u'_k \in \partial \ell_k(\langle x_k, w_h^{(k)} \rangle)$

    Define $s_k = x_k u'_k + \lambda(w_h^{(k)} - h) \in \partial \ell_{k,h}(w_h^{(k)})$

    Update $w_h^{(k+1)} = w_h^{(k)} - \gamma_k s_k$

Return $(w_h^{(k)})_{k=1}^{n+1}, \ \bar{w}_h = \frac{1}{n} \sum_{i=1}^n w_h^{(i)}$

$h = \mathsf{m} \equiv \mathbb{E}_{\mu \sim \rho} w_\mu$ should have a smaller transfer risk than running ERM with, for instance, $h = 0$. We make this statement precise in Sec. 4. Recently, a number of papers have considered how to learn a good bias h in a LTL setting, see e.g. (Pentina & Lampert, 2014; Denevi et al., 2018b). However, one drawback of these works is that they assume the ERM solution to be known exactly, without leveraging the interplay between the optimization and the generalization error. Furthermore, in LTL settings, data naturally arise in an online manner, both *between* and *within* tasks. Hence, an ideal LTL approach should focus on inner algorithms processing one single data point at the time.

Motivated by the above reasoning, in this work, we propose to analyze an online learning algorithm that is computationally and memory efficient while retaining (on average with respect to the sampling of the data) the *same statistical guarantees* of the more expensive ERM estimator. Specifically, for a training dataset $Z_n \sim \mu^n$, a regularization parameter $\lambda > 0$ and a bias vector $h \in \mathbb{R}^d$, we consider the learning algorithm defined as

$$A_h^{\mathrm{SGD}}(Z_n) \equiv \bar{w}_h(Z_n), \tag{5}$$

where, $\bar{w}_h(Z_n)$ is the average of the first n iterations of Alg. 1, in which, for any $k \in [n]$, we have introduced the notation $\ell_k(\cdot) = \ell(\cdot, y_k)$.

Alg. 1 coincides with online subgradient algorithm applied to the strongly convex function $\mathcal{R}_{Z_n, h}$. Moreover, thanks to the assumption that $Z_n \sim \mu^n$, Alg. 1 is equivalent to SGD applied to the regularized true risk

$$\mathcal{R}_{\mu, h}(w) = \mathcal{R}_\mu(w) + \frac{\lambda}{2} \|w - h\|^2. \tag{6}$$

Relying on a standard online-to-batch analysis, see e.g. (Cesa-Bianchi et al., 2004; Hazan, 2016) and references therein, it is easy to link the true error of such an algorithm with the minimum of the regularized empirical risk, that is, $\mathcal{R}_{Z_n,h}(w_h(Z_n))$. This fact is reported in the proposition below and it will be often used in our subsequent statistical analysis. We give a proof in App. F for completeness.

Proposition 1. *Let Asm. 1 and Asm. 2 hold and let $\bar{w}_h$ be the output of Alg. 1. Then, we have that*

$$\mathbb{E}_{Z_n \sim \mu^n} \left[\mathcal{R}_\mu\big(\bar{w}_h(Z_n)\big) - \mathcal{R}_{Z_n,h}(w_h(Z_n)) \right] \leq c_{n,\lambda}$$

(7)

$$c_{n,\lambda} = \frac{2R^2 L^2 \big(\log(n) + 1 \big)}{\lambda n}.$$

We remark that at this level of the analysis, one may avoid the logarithmic factor in the above bound, see e.g. (Shamir & Zhang, 2013; Rakhlin et al., 2012; Lacoste-Julien et al., 2012). However, in order to not complicate our presentation and proofs, we avoid this refinement of the analysis.

In the next section we study the impact on the bias vector on the statistical performance of the inner algorithm. Specifically, we investigate circumstances under which there is an advantage in perturbing the regularization in the objective used by the algorithm with an appropriate ideal bias term h, as opposed to fix $h = 0$. Throughout the paper, we refer to the choice $h = 0$ as independent task learning (ITL), although strictly speaking, when h is fixed in advanced, then, SGD is applied on each task independently regardless of the value of h. Then, in Sec. 5 we address the question of estimating this appropriate bias from the data.

4. The Advantage of the Right Bias Term

In this section, we study the statistical performance of the model $\bar{w}_h$ returned by Alg. 1, on average with respect to the tasks sampled from the environment ρ, for different choices of the bias vector h. To present our observations, we require, for any $\mu \sim \rho$, that the corresponding true risk admits minimizers and we denote by w_μ the minimum norm minimizer[2]. With these ingredients, we introduce the oracle

$$\mathcal{E}_\rho = \mathbb{E}_{\mu \sim \rho} \, \mathcal{R}_\mu(w_\mu),$$

representing the averaged minimum error over the environment of tasks, and, for a candidate bias h, we give a bound on the quantity $\mathcal{E}(\bar{w}_h) - \mathcal{E}_\rho$. This gap coincides with the averaged excess risk of algorithm Alg. 1 with bias h over the environment of tasks, that is

$$\mathcal{E}_n(\bar{w}_h) - \mathcal{E}_\rho = \mathbb{E}_{\mu \sim \rho} \, \mathbb{E}_{Z_n \sim \mu^n} \left[\mathcal{R}_\mu\big(\bar{w}_h(Z_n)\big) - \mathcal{R}_\mu(w_\mu) \right].$$

---

[2]This choice is made in order to simplify our presentation. However, our analysis holds for different choices of a minimizer w_μ, which may potentially improve our bounds.

Hence, this quantity is an indicator of the performance of the bias h with respect to our environment. In the rest of this section, we study the above gap for a bias h which is fixed and does not depend on the data. For this purpose, we introduce the notation

$$\text{Var}_h^2 = \frac{1}{2} \, \mathbb{E}_{\mu \sim \rho} \left\| w_\mu - h \right\|^2$$

(8)

and we observe that

$$\mathsf{m} \equiv \mathbb{E}_{\mu \sim \rho} w_\mu = \underset{h \in \mathbb{R}^d}{\operatorname{argmin}} \, \text{Var}_h^2.$$

(9)

Theorem 2 (Excess Transfer Risk Bound for a Fixed Bias h)**.** *Let Asm. 1 and Asm. 2 hold and let $\bar{w}_h$ be the output of Alg. 1 with regularization parameter*

$$\lambda = \frac{RL}{\text{Var}_h} \sqrt{\frac{2 \big(\log(n) + 1 \big)}{n}}.$$

(10)

Then, the following bound holds

$$\mathcal{E}_n(\bar{w}_h) - \mathcal{E}_\rho \leq \text{Var}_h \, 2RL \sqrt{\frac{2 \big(\log(n) + 1 \big)}{n}}.$$

(11)

Proof. For $\mu \sim \rho$, consider the following decomposition

$$\mathbb{E}_{Z_n \sim \mu^n} \left[\mathcal{R}_\mu(\bar{w}_h(Z_n)) - \mathcal{R}_\mu(w_\mu) \right] \leq \text{A} + \text{B},$$

(12)

where A and B are respectively defined by

$$\begin{aligned}
\text{A} &= \mathbb{E}_{Z_n \sim \mu^n} \left[\mathcal{R}_\mu(\bar{w}_h(Z_n)) - \mathcal{R}_{Z_n,h}(w_h(Z_n)) \right] \\
\text{B} &= \mathbb{E}_{Z_n \sim \mu^n} \left[\mathcal{R}_{Z_n,h}(w_h(Z_n)) - \mathcal{R}_\mu(w_\mu) \right].
\end{aligned}$$

(13)

In order to bound the term A, we use Prop. 1. Regarding the term B, we exploit the definition of the ERM algorithm and the fact that, since w_μ does not depend on Z_n, then $\mathcal{R}_{\mu,h}(w_\mu) = \mathbb{E}_{Z_n \sim \mu^n} \mathcal{R}_{Z_n,h}(w_\mu)$. Consequently, we can upper bound the term B as

$$\begin{aligned}
& \mathbb{E}_{Z_n \sim \mu^n} \left[\mathcal{R}_{Z_n,h}(w_h(Z_n)) - \mathcal{R}_{\mu,h}(w_\mu) \right] + \frac{\lambda}{2} \left\| w_\mu - h \right\|^2 \\
&= \mathbb{E}_{Z_n \sim \mu^n} \left[\mathcal{R}_{Z_n,h}(w_h(Z_n)) - \mathcal{R}_{Z_n,h}(w_\mu) \right] + \frac{\lambda}{2} \left\| w_\mu - h \right\|^2 \\
&\leq \frac{\lambda}{2} \left\| w_\mu - h \right\|^2.
\end{aligned}$$

(14)

The desired statement follows by combining the above bounds on the two terms, taking the average with respect to $\mu \sim \rho$ and optimizing over λ. ∎

Thm. 2 shows that the strength of the regularization that one should use in the within-task algorithm Alg. 1 is inversely

proportional to both the variance of the bias h and the number of points in the datasets. This is exactly in line with the LTL aim: when solving each task is difficult, knowing a priori a good bias can bring a substantial benefit over learning with no bias. To further investigate this point, in the following corollary, we specialize Thm. 2 to two particular choices of the bias h. The first choice we make is $h = 0$, which coincides, as remarked earlier, with learning each task independently, while the second choice considers an ideal bias, namely, assuming that the transfer risk admits minimizer, we set $h = h_n \in \operatorname{argmin}_{h \in \mathbb{R}^d} \mathcal{E}_n(\bar{w}_h)$.

Corollary 3 (Excess Transfer Risk Bound for ITL and the Oracle). *Let Asm. 1 and Asm. 2 hold.*

1. **Independent Task Learning.** *Let $\bar{w}_0$ be the output of Alg. 1 with bias $h = 0$ and regularization parameter as in Eq. (10) with $h = 0$. Then,*

$$\mathcal{E}_n(\bar{w}_0) - \mathcal{E}_\rho \leq \operatorname{Var}_0 2RL \sqrt{\frac{2\big(\log(n)+1\big)}{n}}.$$

2. **The Oracle.** *Let $\bar{w}_{h_n}$ be the output of Alg. 1 with bias $h = h_n$ and regularization parameter as in Eq. (10) with $h = \mathsf{m}$. Then,*

$$\mathcal{E}_n(\bar{w}_{h_n}) - \mathcal{E}_\rho \leq \operatorname{Var}_{\mathsf{m}} 2RL \sqrt{\frac{2\big(\log(n)+1\big)}{n}}.$$

Proof. The proof of the first statement follows directly from the application of Thm. 2 with $h = 0$. The second statement is a direct consequence of the definition of h_n implying $\mathcal{E}_n(\bar{w}_{h_n}) - \mathcal{E}_\rho \leq \mathcal{E}_n(\bar{w}_{\mathsf{m}}) - \mathcal{E}_\rho$ and the application of Thm. 2 with $h = \mathsf{m}$ on the second term. ∎

From the previous bounds we can observe that, using the bias $h = h_n$ in the regularizer brings a substantial benefit with respect to the unbiased case when the number of points n in each dataset in not very large (hence learning each task is quite difficult) and the variance of the weight tasks' vectors sampled from the environment is much smaller than their second moment, i.e. when

$$\operatorname{Var}_{\mathsf{m}}^2 = \frac{1}{2} \mathbb{E}_{\mu \sim \rho} \|w_\mu - \mathsf{m}\|^2 \ll \frac{1}{2} \mathbb{E}_{\mu \sim \rho} \|w_\mu\|^2 = \operatorname{Var}_0^2.$$

Driven by this observation, when the environment of tasks satisfies the above characteristics, we would like to take advantage of this tasks' similarity. But, since in practice we are not able to explicitly compute h_n, in the following we propose an efficient online LTL approach to estimate the bias directly from the observed sequence of datasets.

5. Estimating the Bias

In this section, we study the problem of designing an estimator for the bias vector that is computed *incrementally* from a set of observed T tasks.

5.1. The Meta-Objective

Since direct optimization of the transfer risk is not feasible, a standard strategy used in LTL consists in introducing a proxy objective that is easier to handle, see e.g. (Maurer, 2005; 2009; Maurer et al., 2013; 2016; Denevi et al., 2018a;b). In this paper, motivated by Prop. 1, according to which

$$\mathbb{E}_{Z_n \sim \mu^n}\big[\mathcal{R}_\mu\big(\bar{w}_h(Z_n)\big)\big] \leq$$
$$\mathbb{E}_{Z_n \sim \mu^n}\big[\mathcal{R}_{Z_n,h}(w_h(Z_n))\big] + \frac{2R^2L^2\big(\log(n)+1\big)}{\lambda n},$$

we substitute in the definition of the transfer risk the true risk of the algorithm $\mathcal{R}_\mu\big(\bar{w}_h(Z_n)\big)$ with the minimum of the regularized empirical risk

$$\mathcal{L}_{Z_n}(h) = \min_{w \in \mathbb{R}^d} \mathcal{R}_{Z_n,h}(w) = \mathcal{R}_{Z_n,h}(w_h(Z_n)). \quad (15)$$

This leads us to the following proxy for the transfer risk

$$\hat{\mathcal{E}}_n(h) = \mathbb{E}_{\mu \sim \rho} \mathbb{E}_{Z_n \sim \mu^n} \mathcal{L}_{Z_n}(h). \quad (16)$$

Some remarks about this choice are in order. First, convexity is usually a rare property in LTL. In our case, as described in the following proposition, the definition of the function $\mathcal{L}_{Z_n}$ as the partial minimum of a jointly convex function, ensures convexity and other nice properties, such as differentiability and a closed expression of its gradient.

Proposition 4 (Properties of $\mathcal{L}_{Z_n}$). *The function $\mathcal{L}_{Z_n}$ in Eq. (15) is convex and λ-smooth over $\mathbb{R}^d$. Moreover, for any $h \in \mathbb{R}^d$, its gradient is given by the formula*

$$\nabla \mathcal{L}_{Z_n}(h) = -\lambda\big(w_h(Z_n) - h\big), \quad (17)$$

where $w_h(Z_n)$ is the ERM algorithm in Eq. (3). Finally, when Asm. 1 and Asm. 2 hold, $\mathcal{L}_{Z_n}$ is LR-Lipschitz.

The above statement is a known result in the optimization community, see e.g. (Bauschke & Combettes, 2011, Prop. 12.29) and App. C for more details. In order to minimize the proxy objective in Eq. (16), one standard choice done in stochastic optimization, and also adopted in this work, is to use first-order methods, requiring the computation of an unbiased estimate of the gradient of the stochastic objective. In our case, according to the above proposition, this step would require computing the minimizer of the regularized empirical problem in Eq. (15) exactly. A key observation of our work is to show below that we can easily design a "satisfactory" approximation (see the last paragraph

in Sec. 5) of its gradient, just substituting the minimizer $w_h(Z_n)$ in the expression of the gradient in Eq. (17) with the last iterate $w_h^{(n+1)}(Z_n)$ of Alg. 1. An important aspect to stress here is the fact that this strategy does not require any additional computational effort. Formally, this reasoning is linked to the concept of ϵ-subgradient of a function. We recall that, for a given convex, proper and closed function f and for a given point $\hat{h} \in \mathrm{Dom}(f)$ in its domain, u is an ϵ-subgradient of f at $\hat{h}$, if, for any h, $f(h) \geq f(\hat{h}) + \langle u, h - \hat{h} \rangle - \epsilon$.

Proposition 5 (An ϵ-Subgradient for $\mathcal{L}_{Z_n}$). *Let $w_h^{(n+1)}(Z_n)$ be the last iterate of Alg. 1. Then, under Asm. 1 and Asm. 2, the vector*

$$\hat{\nabla}\mathcal{L}_{Z_n}(h) = -\lambda\big(w_h^{(n+1)}(Z_n) - h\big) \qquad (18)$$

is an ϵ-subgradient of $\mathcal{L}_{Z_n}$ at point h, where ϵ is such that

$$\mathbb{E}_{Z_n \sim \mu^n}\left[\epsilon\right] \leq \frac{2R^2 L^2\big(\log(n)+1\big)}{\lambda n}. \qquad (19)$$

Moreover, introducing $\Delta_{Z_n}(h) = \nabla\mathcal{L}_{Z_n}(h) - \hat{\nabla}\mathcal{L}_{Z_n}(h)$,

$$\mathbb{E}_{Z_n \sim \mu^n}\left\|\Delta_{Z_n}(h)\right\|^2 \leq \frac{4R^2 L^2\big(\log(n)+1\big)}{n}. \qquad (20)$$

The above result is a key tool in our analysis. The proof requires some preliminaries on the ϵ-subdifferential of a function (see App. A) and introducing the dual formulation of both the within-task learning problem and Alg. 1 (see App. B and App. E, respectively). With these two ingredients, the proof of the statement is deduced in App. E.3 by the application of a more general result reported in App. D, describing how an ϵ-minimizer of the dual of the within-task learning problem can be exploited in order to build an ϵ-subgradient of the meta-objective function $\mathcal{L}_{Z_n}$. We stress that this result could be applied to more general class of algorithms, going beyond Alg. 1 considered here.

5.2. The Meta-Algorithm to Estimate the Bias h

In order to estimate the bias h from the data, we apply SGD to the stochastic function $\hat{\mathcal{E}}_n$ introduced in Eq. (16). More precisely, in our setting, the sampling of a "meta-point" corresponds to the incremental sampling of a dataset from the environment[3]. We refer to Alg. 2 for more details. In particular, we propose to take the estimator $\bar{h}_T$ obtained by averaging the iterations returned by Alg. 2. An important feature to stress here is the fact that the meta-algorithm uses ϵ-subgradients of the function $\mathcal{L}_{Z_n}$ which are computed as described above. Specifically, for any $t \in [T]$, we define

$$\hat{\nabla}\mathcal{L}_{Z_n^{(t)}}(h^{(t)}) = -\lambda\big(w_{h^{(t)}}^{(n+1)}(Z_n^{(t)}) - h^{(t)}\big), \qquad (21)$$

---

[3]More precisely we first sample a distribution μ from ρ and then a dataset $Z_n \sim \mu^n$.

Algorithm 2 Meta-Algorithm, SGD on $\hat{\mathcal{E}}$ with ϵ-Subgradients

Input $\gamma > 0$ step size, $\lambda > 0$ inner regularization parameter, ρ meta-distribution

Initialization $h^{(1)} = 0 \in \mathbb{R}^d$

For $t = 1$ to T

    Receive $\mu_t \sim \rho$, $Z_n^{(t)} \sim \mu_t^n$

    Run the inner algorithm Alg. 1 and approximate the gradient $\hat{\nabla}^{(t)} \approx \nabla^{(t)}$ by Eq. (21)

    Update $h^{(t+1)} = h^{(t)} - \gamma\hat{\nabla}^{(t)}$

Return $(h^{(t)})_{t=1}^{T+1}$ and $\bar{h}_T = \dfrac{1}{T}\displaystyle\sum_{t=1}^{T} h^{(t)}$

---

where $w_{h^{(t)}}^{(n+1)}$ is the last iterate of Alg. 1 applied with the current bias $h^{(t)}$ and the dataset $Z_n^{(t)}$. To simplify the presentation, throughout this work, we use the short-hand notation

$$\mathcal{L}_t(\cdot) = \mathcal{L}_{Z_n^{(t)}}(\cdot), \quad \nabla^{(t)} = \nabla\mathcal{L}_t(h^{(t)}), \quad \hat{\nabla}^{(t)} = \hat{\nabla}\mathcal{L}_t(h^{(t)}).$$

Some technical observations follows. First, we stress that Alg. 2 processes one single instance at the time, without the need to store previously encountered data points, neither across the tasks nor within them. Second, the implementation of Alg. 2 does not require computing the meta-objective $\mathcal{L}_{Z_n}$, which would increase the computational effort of the entire scheme. The rest of this section is devoted to the statistical analysis of Alg. 2.

5.3. Statistical Analysis of the Meta-Algorithm

In the following theorem we study the statistical performance of the bias $\bar{h}_T$ returned by Alg. 2. More precisely we bound the excess transfer risk of the inner SGD algorithm run with this biased term learned by the meta-algorithm.

Theorem 6 (Excess Transfer Risk Bound for the Bias $\bar{h}_T$ Estimated by Alg. 2). *Let Asm. 1 and Asm. 2 hold and let $\bar{h}_T$ be the output of Alg. 2 with step size*

$$\gamma = \frac{\sqrt{2}\|\mathsf{m}\|}{LR}\sqrt{\left(T\Big(1 + \frac{4\big(\log(n)+1\big)}{n}\Big)\right)^{-1}}. \qquad (22)$$

Let $\bar{w}_{\bar{h}_T}$ be the output of Alg. 1 with bias $h = \bar{h}_T$ and regularization parameter

$$\lambda = \frac{2RL}{\mathrm{Var}_{\mathsf{m}}}\sqrt{\frac{\log(n)+1}{n}}. \qquad (23)$$

Then, the following bound holds

$$\mathbb{E}\left[\mathcal{E}_n(\bar{w}_{\bar{h}_T})\right] - \mathcal{E}_\rho \leq \text{Var}_{\mathsf{m}}\, 4RL\sqrt{\frac{\log(n)+1}{n}}$$
$$+ \|\mathsf{m}\|RL\sqrt{2\Big(1+\frac{4\big(\log(n)+1\big)}{n}\Big)\frac{1}{T}}$$

where the expectation above is with respect to the sampling of the datasets $Z_n^{(1)}, \dots, Z_n^{(T)}$ from the environment ρ.

Proof. We consider the following decomposition

$$\mathbb{E}\left[\mathcal{E}_n(\bar{w}_{\bar{h}_T})\right] - \mathcal{E}_\rho \leq A + B + C, \qquad (24)$$

where

$$\begin{aligned}
A &= \mathcal{E}_n(\bar{w}_{\bar{h}_T}) - \hat{\mathcal{E}}_n(\bar{h}_T) \\
B &= \mathbb{E}\,\hat{\mathcal{E}}_n(\bar{h}_T) - \hat{\mathcal{E}}_n(\mathsf{m}) \qquad (25) \\
C &= \hat{\mathcal{E}}_n(\mathsf{m}) - \mathcal{E}_\rho.
\end{aligned}$$

Now, in order to bound the term A, noting that

$$A = \mathbb{E}_{\mu\sim\rho}\,\mathbb{E}_{Z_n\sim\mu^n}\left[\mathcal{R}_\mu\big(\bar{w}_{\bar{h}_T}(Z_n)\big) - \mathcal{R}_{Z_n,\bar{h}_T}(w_{\bar{h}_T}(Z_n))\right],$$

we use Prop. 1 with $h = \bar{h}_T$ and, then, we take the average on $\mu \sim \rho$. As regards the term C, we apply the inequality given in Eq. (14) with $h = \mathsf{m}$ and we again average with respect to $\mu \sim \rho$. Finally, the term B is the convergence rate of Alg. 2 and its study requires analyzing the error that we introduce in the meta-gradients by Prop. 5. The bound we use for this term described in Prop. 22 (see App. G) with $\hat{h} = \mathsf{m}$. The result now follows by combining the bounds on the three terms and optimizing over λ. ∎

The bound stated in Thm. 6 with respect to the mean m holds also for a generic bias vector $h \in \mathbb{R}^d$. In particular, the choice of $h = 0$ and the corresponding step-size γ describe the setting in which the meta-algorithm returns the ITL estimator $h = 0$. In such a case, we recover the rate in Cor. 3 for ITL (up to a contant 2).

In addition, the above bound is coherent with the state-of-the-art LTL bounds given in other papers studying other variants of Ivanov or Tikhonov regularized empirical risk minimization algorithms, see e.g. (Maurer, 2005; 2009; Maurer et al., 2013; 2016). Specifically, in our case, the bound has the form

$$\mathcal{O}\Big(\frac{\text{Var}_{\mathsf{m}}}{\sqrt{n}}\Big) + \mathcal{O}\Big(\frac{1}{\sqrt{T}}\Big), \qquad (26)$$

where Var_{m} reflects the advantage in exploiting the relatedness among the tasks sampled from the environment ρ. More precisely, in Sec. 4 we noticed that, if the variance

of the weight vectors of the tasks sampled from our environment is significantly smaller than their second moment, running Alg. 1 with the ideal bias $h = h_n$ on a future task brings a significant improvement in comparison to the unbiased case. One natural question arising at this point of the presentation is whether, under the same conditions on the environment, the same improvement is obtained by running Alg. 1 with the bias vector $h = \bar{h}_T$ returned by our online meta-algorithm in Alg. 2. Looking at the bound in Thm. 6, we can say that, when the number of training tasks T used to estimate the bias $\bar{h}_T$ is sufficiently large, the above question has a positive answer and our LTL approach is effective.

In order to have also a more precise benchmark for the biased setting considered in this work, in App. H we have repeated the statistical study described in the paper also for the more expensive ERM algorithm described in Eq. (3). In this case, we assume to have an oracle providing us with this exact estimator, ignoring any computational costs. As before, we have performed the analysis both for a fixed bias and the one estimated from the data via Alg. 2 (in this case, Alg. 2 is assumed to run with exact meta-gradients). Looking at the results reported in App. H, we immediately see that, up to constants and logarithmic factors, the LTL bounds we have stated in the paper for the low-complexity SGD family are equivalent to those in App. H for the more expensive ERM family.

All the above facts justify the informal statement given before Prop. 5 according to which the trick used to compute the approximation of the meta-gradient by using the last iterate of the inner algorithm, not only, does not require additional effort, but it is also accurate enough from the statistical view point, matching a state-of-the-art bound for more expensive within-task algorithms based on ERM.

6. Experiments

In this section, we test the effectiveness of the LTL approach proposed in this paper on synthetic and real data[4]. In all experiments, the regularization parameter λ and the step-size γ were tuned by validation, see App. I for more details.

Synthetic Data. We considered two different settings, regression with the absolute loss and binary classification with the hinge loss. In both cases, we generated an environment of tasks in which SGD with the right bias is expected to bring a substantial benefit in comparison to the unbiased case. Motivated by our observations in Sec. 4, we generated linear tasks with weight vectors characterized by a variance which is significantly smaller than their second moment. Specifically, for each task μ, we created a weight vector w_μ from a Gaussian distribution with mean m given by the

---
[4]Code available at
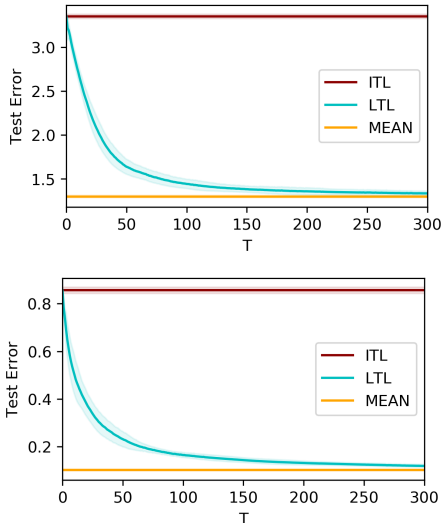https://github.com/prolearner/onlineLTL

Figure 1. **Synthetic Data.** Test performance of different bias with respect to an increasing number of tasks. (Top) Regression with absolute loss. (Bottom) Classification with hinge loss. The results are averaged over 10 independent runs (datasets generations).

Figure 2. **Real Data.** Test performance of different bias with respect to an increasing number of tasks. (Top) Lenk Dataset Regression. (Bottom) Lenk Dataset Classification. The results are averaged over 30 independent runs (datasets generations).

vector in $\mathbb{R}^d$ with all components equal to 4 and standard deviation $\mathrm{Var_m} = 1$. Each task corresponds to a dataset $(x_i, y_i)_{i=1}^n$, $x_i \in \mathbb{R}^d$ with $n = 10$ and $d = 30$. In the regression case, the inputs were uniformly sampled on the unit sphere and the labels were generated as $y = \langle x, w_\mu \rangle + \epsilon$, with ϵ sampled from a zero-mean Gaussian distribution, with standard deviation chosen to have signal-to-noise ratio equal to 10 for each task. In the classification case, the inputs were uniformly sampled on the unit sphere, excluding those points with margin $|\langle x, w_\mu \rangle|$ smaller than 0.5 and the binary labels were generated as a logistic model, $\mathbb{P}(y = 1) = \big(1 + 10 \exp(-\langle x, w_\mu \rangle)\big)^{-1}$. In Fig. 1 we report the performance of Alg. 1 with different choices of the bias: $h = \bar{h}_T$ (our LTL estimator resulting from Alg. 2), $h = 0$ (ITL) and $h = \mathrm{m}$, a reasonable approximation of the oracle minimizing the transfer risk. The plots confirm our theoretical findings: estimating the bias with our LTL approach leads to a substantial benefits with respect to the unbiased case, as the number of the observed training tasks increases.

Real Data. We run experiments on the computer survey data from (Lenk et al., 1996), in which 180 people (tasks) rated the likelihood of purchasing one of 20 different personal computers ($n = 8$). The input represents 13 different computer characteristics (price, CPU, RAM, etc.) while the output is an integer rating from 0 to 10. Similarly to the synthetic data experiments, we consider a regression setting with the absolute loss and a classification setting. In the latter case each task is to predict whether the rating is above 5. We compare the LTL bias with ITL. The results are reported in Fig. 2. The figures above are in line with the results obtained on synthetic experiments, indicating that the bias
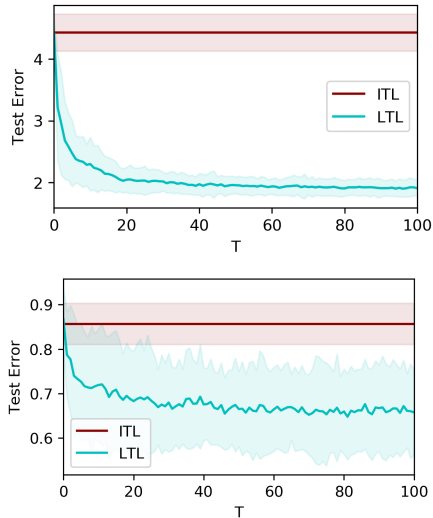
LTL framework proposed in this work is effective for this dataset. Moreover, the results for regression are also in line with what observed in the multitask setting with variance regularization (McDonald et al., 2016). The classification setting has not been used before and has been created ad-hoc for our purpose. In this case we have an increased variance probably due to the datasets being highly unbalanced. In order to investigate the impact of passing through the data only once in the different steps in our method, we conducted additional experiments. The results, presented in App. J, indicate that the single pass strategy is competitive with respect to the more expensive ERM.

7. Conclusion and Future Work

We have studied the performance of Stochastic Gradient Descent on the true risk regularized by the square euclidean distance to a bias vector, over a class of tasks. Drawing upon a learning-to-learn framework, we have shown that, when the variance of the tasks is relatively small, the introduction of an appropriate bias vector may bring a substantial benefit in comparison to the standard unbiased version, corresponding to learning the tasks independently. Then, we have proposed an efficient online meta-learning algorithm to estimate this bias and we have theoretically shown that the bias returned by our method can bring a comparable benefit. In the future, it would be interesting to investigate other kinds of relatedness among the tasks and to extend our analysis to other classes of loss functions, as well as to a Hilbert space setting. Finally, another valuable research direction is to derive fully dependent bounds, in which the hyperparameters are self-tuned during the learning process, see e.g. (Zhuang et al., 2019).

References

Alquier, P., Mai, T. T., and Pontil, M. Regret bounds for life-long learning. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 261–269, 2017.

Balcan, M.-F., Blum, A., and Vempala, S. Efficient representations for lifelong learning and autoencoding. In *Conference on Learning Theory*, pp. 191–210, 2015.

Bauschke, H. H. and Combettes, P. L. *Convex Analysis and Monotone Operator theory in Hilbert Spaces*, volume 408. Springer, 2011.

Baxter, J. A model of inductive bias learning. *J. Artif. Intell. Res.*, 12(149–198):3, 2000.

Beck, A. and Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

Borwein, J. and Zhu, Q. Techniques of variational analysis, ser, 2005.

Bousquet, O. and Elisseeff, A. Stability and generalization. *Journal of machine learning research*, 2(Mar):499–526, 2002.

Cesa-Bianchi, N., Conconi, A., and Gentile, C. On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9):2050–2057, 2004.

Denevi, G., Ciliberto, C., Stamos, D., and Pontil, M. Incremental learning-to-learn with statistical guarantees. In *Proc. 34th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2018a.

Denevi, G., Ciliberto, C., Stamos, D., and Pontil, M. Learning to learn around a common mean. In *Advances in Neural Information Processing Systems*, pp. 10190–10200, 2018b.

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

Finn, C., Rajeswaran, A., Kakade, S., and Levine, S. Online meta-learning. *arXiv preprint arXiv:1902.08438*, 2019.

Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., and Pontil, M. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning*, PMLR 80, pp. 568–1577, 2018.

Gupta, R. and Roughgarden, T. A pac approach to application-specific algorithm selection. *SIAM Journal on Computing*, 46(3):992–1017, 2017.

Hazan, E. Introduction to online convex optimization. *Foundations and Trends in Optimization*, 2016.

Jean-Baptiste, H.-U. *Convex analysis and minimization algorithms: advanced theory and bundle methods*. SPRINGER, 2010.

Khodak, M., Balcan, M.-F., and Talwalkar, A. Provable guarantees for gradient-based meta-learning. *arXiv preprint arXiv:1902.10644*, 2019.

Kuzborskij, I. and Orabona, F. Fast rates by transferring from auxiliary hypotheses. *Machine Learning*, 106(2): 171–195, 2017.

Lacoste-Julien, S., Schmidt, M., and Bach, F. A simpler approach to obtaining an o (1/t) convergence rate for the projected stochastic subgradient method. *arXiv preprint arXiv:1212.2002*, 2012.

Lenk, P. J., DeSarbo, W. S., Green, P. E., and Young, M. R. Hierarchical bayes conjoint analysis: Recovery of part-worth heterogeneity from reduced experimental designs. *Marketing Science*, 15(2):173–191, 1996.

Maurer, A. Algorithmic stability and meta-learning. *Journal of Machine Learning Research*, 6:967–994, 2005.

Maurer, A. Transfer bounds for linear feature learning. *Machine Learning*, 75(3):327–350, 2009.

Maurer, A., Pontil, M., and Romera-Paredes, B. Sparse coding for multitask and transfer learning. In *International Conference on Machine Learning*, 2013.

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

McDonald, A. M., Pontil, M., and Stamos, D. New perspectives on k-support and cluster norms. *Journal of Machine Learning Research*, 17(155):1–38, 2016.

Pentina, A. and Lampert, C. A PAC-Bayesian bound for lifelong learning. In *International Conference on Machine Learning*, pp. 991–999, 2014.

Pentina, A. and Urner, R. Lifelong learning with weighted majority votes. In *Advances in Neural Information Processing Systems*, pp. 3612–3620, 2016.

Rakhlin, A., Shamir, O., Sridharan, K., et al. Making gradient descent optimal for strongly convex stochastic optimization. In *ICML*, volume 12, pp. 1571–1578. Citeseer, 2012.

Ravi, S. and Larochelle, H. Optimization as a model for few-shot learning. In *I5th International Conference on Learning Representations*, 2017.

Shalev-Shwartz, S. and Ben-David, S. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

Shalev-Shwartz, S. and Kakade, S. M. Mind the duality gap: Logarithmic regret algorithms for online optimization. In *Advances in Neural Information Processing Systems*, pp. 1457–1464, 2009.

Shamir, O. and Zhang, T. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *International Conference on Machine Learning*, pp. 71–79, 2013.

Thrun, S. and Pratt, L. *Learning to Learn*. Springer, 1998.

Zhuang, Z., Cutkosky, A., and Orabona, F. Surrogate losses for online learning of stepsizes in stochastic non-convex optimization. *arXiv preprint arXiv:1901.09068*, 2019.

Appendix

The appendix is organized as follows. In App. A we report some basic facts regarding the ϵ-subdifferential of a function which are used in the subsequent analysis. In App. B we give the primal-dual formulation of the biased regularized empirical risk minimization problem for each single task and, in App. C, we recall some well-known properties of our meta-objective function. In App. D, we show how an ϵ-minimizer of the dual problem can be exploited in order to build an ϵ-subgradient of our meta-objective function. As described in App. E, interpreting our within-task algorithm as a coordinate descent algorithm on the dual problem, we can adapt this result to our setting and prove, in this way, Prop. 5. In App. F, we report the proof of Prop. 1 and, in App. G, we give the convergence rate of Alg. 2 which is used in the paper, during the proof of Thm. 6. In App. H, we repeat the statistical study described in the paper also for the family of ERM algorithms introduced in Eq. (3) and, in App. I, we describe how to perform the validation procedure in our LTL setting. Finally, in App. J we report additional experiments comparing our method to ERM variants.

A. Basic Facts on ϵ-Subgradients

In this section, we report some basic concepts about the ϵ-subdifferential which are then used in the subsequent analysis. This material is based on (Jean-Baptiste, 2010, Chap. XI). Throughout this section we consider a convex closed and proper function $f : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ with domain $\mathrm{Dom}(f)$ and we always let $\epsilon \geq 0$.

Definition 7 (ϵ-Subgradient, (Jean-Baptiste, 2010, Chap. XI, Def. 1.1.1)). *Given $\hat{h} \in \mathrm{Dom}(f)$, the vector $u \in \mathbb{R}^d$ is called ϵ-subgradient of f at $\hat{h}$ when the following property holds for any $h \in \mathbb{R}^d$*

$$f(h) \geq f(\hat{h}) + \langle u, h - \hat{h} \rangle - \epsilon. \tag{27}$$

The set of all ϵ-subgradients of f at $\hat{h}$ is the ϵ-subdifferential of f at $\hat{h}$, denoted by $\partial_\epsilon f(\hat{h})$.

The standard subifferential $\partial f(\hat{h})$ is retrieved with $\epsilon = 0$. The following lemma, which is a direct consequence of Def. 7, points out the link between $\partial_\epsilon f$ and an ϵ-minimizer of f.

Lemma 8 (Jean-Baptiste (2010, Chap. XI, Thm. 1.1.5)). *The following two properties are equivalent.*

$$0 \in \partial_\epsilon f(\hat{h}) \quad \Longleftrightarrow \quad f(\hat{h}) \leq f(h) + \epsilon \quad \text{for any } h \in \mathbb{R}^d. \tag{28}$$

The subsequent lemma describes the behavior of the ϵ-subdifferential with respect to the duality.

Lemma 9 (Jean-Baptiste (2010, Chap. XI, Prop. 1.2.1)). *Let $f^* : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be the Fenchel conjugate of f, namely, $f^*(\cdot) = \sup_{h \in \mathbb{R}^d} \langle \cdot, h \rangle - f(h)$. Then, given $\hat{h} \in \mathrm{Dom}(f)$, the vector $u \in \mathbb{R}^d$ is an ϵ-subgradient of f at $\hat{h}$ iff*

$$f^*(u) + f(\hat{h}) - \langle u, \hat{h} \rangle \leq \epsilon. \tag{29}$$

As a result,

$$u \in \partial_\epsilon f(\hat{h}) \quad \Longleftrightarrow \quad \hat{h} \in \partial_\epsilon f^*(u). \tag{30}$$

We now describe some properties of the ϵ-subdifferential which are used in the following analysis.

Lemma 10 (Jean-Baptiste (2010, Chap. XI, Thm. 3.1.1)). *Let f_1 and f_2 be two convex closed and proper functions. Then, given $\hat{h} \in \mathrm{Dom}(f_1 + f_2) = \mathrm{Dom}(f_1) \cap \mathrm{Dom}(f_2)$, we have that*

$$\bigcup_{0 \leq \epsilon_1 + \epsilon_2 \leq \epsilon} \partial_{\epsilon_1} f_1(\hat{h}) + \partial_{\epsilon_2} f_2(\hat{h}) \subset \partial_\epsilon (f_1 + f_2)(\hat{h}). \tag{31}$$

Moreover, denoting by $\mathrm{ri}(A)$ the relative interior of a set A, when $\mathrm{ri}(\mathrm{Dom}(f_1)) \cap \mathrm{ri}(\mathrm{Dom}(f_2)) \neq \emptyset$, equality holds.

Lemma 11 (Jean-Baptiste (2010, Chap. XI, Prop. 1.3.1)). *Let $a \neq 0$ be a scalar. Then, for a given $\hat{h} \in \mathrm{Dom}(f \circ a)$, we have that*

$$\partial_\epsilon (f \circ a)(\hat{h}) = a \, \partial_\epsilon f(a\hat{h}). \tag{32}$$

Lemma 12. *Let $X \in \mathbb{R}^{n \times d}$ be a matrix. Then, for a given $\hat{h} \in \mathbb{R}^d$ such that $X\hat{h} \in \mathrm{Dom}(f)$, we have that*

$$X^\top \partial_\epsilon f(X\hat{h}) \subset \partial_\epsilon (f \circ X)(\hat{h}). \tag{33}$$

Proof. Let be $u \in X^\top \partial_\epsilon f(X\hat{h})$. Then, by definition, there exist $v \in \partial_\epsilon f(X\hat{h})$ such that $u = X^\top v$. Consequenlty, for any $h \in \mathbb{R}^d$, we can write

$$\langle u, h - \hat{h} \rangle = \langle X^\top v, h - \hat{h} \rangle = \langle v, Xh - X\hat{h} \rangle \le f(Xh) - f(X\hat{h}) + \epsilon = (f \circ X)(h) - (f \circ X)(\hat{h}) + \epsilon, \quad (34)$$

where, in the inequality we have used the fact that $v \in \partial_\epsilon f(X\hat{h})$. This gives the desired statement. ∎

The next two results characterize the ϵ-subdifferential of two functions, which are useful in our subsequent analysis. In the following we denote by $\mathbb{S}_+^d$ the set of the $d \times d$ symmetric positive semi-definite matrices.

Example 1 (Quadratic Functions, (Jean-Baptiste, 2010, Chap. XI, Ex. 1.2.2)). *For a given matrix $Q \in \mathbb{S}_+^d$ and a given vector $b \in \mathbb{R}^d$, consider the function*

$$f : h \in \mathbb{R}^d \mapsto \frac{1}{2}\langle Qh, h \rangle + \langle b, h \rangle. \quad (35)$$

Then, given $\hat{h} \in \mathrm{Dom}(f) = \mathbb{R}^d$, we can express the ϵ-subdifferential of f at $\hat{h}$ with respect to the gradient $\nabla f(\hat{h}) = Q\hat{h} + b$ as follows

$$\partial_\epsilon f(\hat{h}) = \left\{ \nabla f(\hat{h}) + Qs : \frac{1}{2}\langle Qs, s \rangle \le \epsilon \right\}. \quad (36)$$

Example 2 (Moreau Envelope (Jean-Baptiste, 2010, Chap. XI, Ex. 3.4.4)). *For $\lambda > 0$ and a fixed vector $h \in \mathbb{R}^d$, consider the Moreau envelope of f at the point h with parameter λ, given by*

$$\mathcal{L}(h) = \min_{w \in \mathbb{R}^d} f(w) + \frac{\lambda}{2}\|w - h\|^2. \quad (37)$$

Denote by w_h the unique minimizer of the above function, namely, the vector characterized by the optimality conditions

$$0 \in \partial f(w_h) + \lambda(w_h - h). \quad (38)$$

Then, for any $\lambda > 0$ and $h \in \mathbb{R}^d$, we have that

$$\partial_\epsilon \mathcal{L}(h) = \bigcup_{0 \le \alpha \le \epsilon} \partial_{\epsilon - \alpha} f(w_h) \cap \mathcal{B}\left(-\lambda(w_h - h), \sqrt{2\lambda\alpha}\right), \quad (39)$$

where, for any center $c \in \mathbb{R}^d$ and any radius $r \ge 0$, we recall the notation

$$\mathcal{B}(c, r) = \left\{ u \in \mathbb{R}^d : \|u - c\| \le r \right\}. \quad (40)$$

For $\epsilon = 0$ we retrieve the well-known result according to which $\mathcal{L}$ is differentiable, with λ-Lipschitz gradient given by

$$\nabla \mathcal{L}(h) = -\lambda(w_h - h). \quad (41)$$

Finally, from Eq. (39), we can deduce that, if $u \in \partial_\epsilon \mathcal{L}(h)$, then

$$\left\| \nabla \mathcal{L}(h) - u \right\| \le \sqrt{2\lambda\epsilon}. \quad (42)$$

B. Primal-Dual Formulation of the Within-Task Problem

In this section, we give the primal-dual formulation of the biased regularized empirical risk minimization problem outlined in Eq, (3) for each single task. Specifically, rewriting for any $w \in \mathbb{R}^d$ and $u \in \mathbb{R}^n$, the empirical risk

$$\mathcal{R}_{Z_n}(w) = (g \circ X_n)(w) \qquad g(u) = \frac{1}{n}\sum_{k=1}^{n} \ell_k(u_k), \quad (43)$$

for any $h \in \mathbb{R}^d$, we can express our meta-objective function in Eq. (15) as

$$\mathcal{L}_{Z_n}(h) = \min_{w \in \mathbb{R}^d} (g \circ X_n)(w) + \frac{\lambda}{2}\|w - h\|^2. \quad (44)$$

We remark that, in the optimization community, this function coincides with the Moreau envelope of the empirical error at the point h, see also Ex. 2. In this section, in order to simplify the presentation, we omit the dependence on the dataset Z_n in the notation. The unique minimizer of the above function

$$w_h = \underset{w \in \mathbb{R}^d}{\mathrm{argmin}} \ \big(g \circ X_n\big)(w) + \frac{\lambda}{2} \, \|w - h\|^2 \tag{45}$$

is known as the proximity operator of the empirical error at the point h and it coincides with the ERM algorithm introduced in Eq. (3) in the paper. We interpret the vector w_h in Eq. (3)–(45) as the solution of the primal problem

$$w_h = \underset{w \in \mathbb{R}^d}{\mathrm{argmin}} \ \Phi_h(w) \qquad \Phi_h(w) = \big(g \circ X_n\big)(w) + \frac{\lambda}{2} \, \|w - h\|^2. \tag{46}$$

The next proposition is a standard result stating that, in this setting, strong duality holds and the optimality conditions, also known as Karush–Kuhn–Tucker (KKT) conditions provide a unique way to determine the primal variables from the dual ones.

Proposition 13 (Strong Duality, (Borwein & Zhu, 2005, Thm. 4.4.2), (Bauschke & Combettes, 2011, Prop. 15.18))**.** *Consider the primal problem in Eq.* (131). *Then, its dual problem admits a solution*

$$u_h \in \underset{u \in \mathbb{R}^n}{\mathrm{argmin}} \ \Psi_h(u) \qquad \Psi_h(u) = g^*(u) + \frac{1}{2\lambda} \big\|X_n^\top u\big\|^2 - \big\langle X_n h, u \big\rangle, \tag{47}$$

where, thanks to the separability of g, for any $u \in \mathbb{R}^n$, we have that

$$g^*(u) = \frac{1}{n} \sum_{k=1}^{n} \ell_k^*(n u_k). \tag{48}$$

Moreover, strong duality holds, namely,

$$\mathcal{L}(h) = \Phi_h(w_h) = \min_{w \in \mathbb{R}^d} \Phi_h(w) = - \min_{u \in \mathbb{R}^n} \Psi_h(u) = -\Psi_h(u_h) \tag{49}$$

and the optimality (KKT) conditions read as follows

$$\begin{aligned}
w_h &= -\frac{1}{\lambda} X_n^\top u_h + h \iff \lambda(w_h - h) = -X_n^\top u_h \\
u_h &\in \partial g(X_n w_h) \iff X_n w_h \in \partial g^*(u_h).
\end{aligned} \tag{50}$$

C. Properties of the Meta-Objective

In this section we recall some properties of the meta-objective function $\mathcal{L}_{Z_n}$ already outlined in the text in Prop. 4.

Proposition 4 (Properties of $\mathcal{L}_{Z_n}$)**.** *The function $\mathcal{L}_{Z_n}$ in Eq.* (15) *is convex and λ-smooth over $\mathbb{R}^d$. Moreover, for any $h \in \mathbb{R}^d$, its gradient is given by the formula*

$$\nabla \mathcal{L}_{Z_n}(h) = -\lambda\big(w_h(Z_n) - h\big), \tag{17}$$

where $w_h(Z_n)$ is the ERM algorithm in Eq. (3). *Finally, when Asm. 1 and Asm. 2 hold, $\mathcal{L}_{Z_n}$ is LR-Lipschitz.*

Proof. The first part of the statement is a well-known fact, see (Bauschke & Combettes, 2011, Prop. 12.29) and also Ex. 2. In order to prove the second part of the statement, we exploit Asm. 1 and Asm. 2 and we proceed as follows. According to the change of variables $v = w - h$, exploiting the fact that, for any two convex functions f_1 and f_2, we have

$$\Big| \min_{v \in \mathbb{R}^d} f_1(v) - \min_{v \in \mathbb{R}^d} f_2(v) \Big| \leq \sup_{v \in \mathbb{R}^d} \big| f_1(v) - f_2(v) \big|, \tag{51}$$

for any $h_1, h_2 \in \mathbb{R}^d$, we can write the following

$$
\begin{aligned}
\Big| \mathcal{L}_{Z_n}(h_1) &- \mathcal{L}_{Z_n}(h_2) \Big| \\
&= \Big| \min_{w \in \mathbb{R}^d} \Big(\frac{1}{n} \sum_{k=1}^{n} \ell_k(\langle x_k, w \rangle) + \frac{\lambda}{2} \|w - h_1\|^2 \Big) - \min_{w \in \mathbb{R}^d} \Big(\frac{1}{n} \sum_{k=1}^{n} \ell_k(\langle x_k, w \rangle) + \frac{\lambda}{2} \|w - h_2\|^2 \Big) \Big| \\
&= \Big| \min_{v \in \mathbb{R}^d} \Big(\frac{1}{n} \sum_{k=1}^{n} \ell_k(\langle x_k, v + h_1 \rangle) + \frac{\lambda}{2} \|v\|^2 \Big) - \min_{v \in \mathbb{R}^d} \Big(\frac{1}{n} \sum_{k=1}^{n} \ell_k(\langle x_k, v + h_2 \rangle) + \frac{\lambda}{2} \|v\|^2 \Big) \Big| \\
&\leq \sup_{v \in \mathbb{R}^d} \Big| \frac{1}{n} \sum_{k=1}^{n} \ell_k(\langle x_k, v + h_1 \rangle) + \frac{\lambda}{2} \|v\|^2 - \frac{1}{n} \sum_{k=1}^{n} \ell_k(\langle x_k, v + h_2 \rangle) - \frac{\lambda}{2} \|v\|^2 \Big| \\
&= \sup_{v \in \mathbb{R}^d} \Big| \frac{1}{n} \sum_{k=1}^{n} \Big(\ell_k(\langle x_k, v + h_1 \rangle) - \ell_k(\langle x_k, v + h_2 \rangle) \Big) \Big| \\
&\leq \sup_{v \in \mathbb{R}^d} \frac{1}{n} \sum_{k=1}^{n} \Big| \ell_k(\langle x_k, v + h_1 \rangle) - \ell_k(\langle x_k, v + h_2 \rangle) \Big| \\
&\leq \frac{L}{n} \sup_{v \in \mathbb{R}^d} \sum_{k=1}^{n} \Big| \langle x_k, v + h_1 \rangle - \langle x_k, v + h_2 \rangle \Big| \\
&= \frac{L}{n} \sum_{k=1}^{n} \Big| \langle x_k, h_1 - h_2 \rangle \Big| \\
&\leq \frac{L}{n} \sum_{k=1}^{n} \|x_k\| \|h_1 - h_2\| \\
&\leq LR \|h_1 - h_2\|,
\end{aligned}
\tag{52}
$$

where, in the third inequality we have used Asm. 2, in the fourth inequality we have applied Cauchy-Schwartz inequality and in the last step we have used Asm. 1. Consequently, we can state that $\mathcal{L}_{Z_n}$ is LR-Lipschitz. ∎

To conclude this section, in the next proposition, we recall the closed form of the conjugate of the function $\mathcal{L}_{Z_n}$.

Lemma 14 (Fenchel Conjugate of $\mathcal{L}_{Z_n}$). *For any $\alpha \in \mathbb{R}^d$, the Fenchel conjugate function of $\mathcal{L}_{Z_n}$ is*

$$
\mathcal{L}_{Z_n}^*(\alpha) = \big(g \circ X_n \big)^*(\alpha) + \frac{1}{2\lambda} \|\alpha\|^2.
\tag{53}
$$

Proof. We recall that the infimal convolution of two proper closed convex functions f_1 and f_2 is defined as $\big(f_1 \,\square\, f_2 \big)(\cdot) = \inf_w f_1(w) + f_2(\cdot - w)$ and its Fenchel conjugate is give by $\big(f_1 \,\square\, f_2 \big)^* = f_1^* + f_2^*$, see (Bauschke & Combettes, 2011, Chap. XII). Hence, the statement follows from observing that, for any $h \in \mathbb{R}^d$ and any $\alpha \in \mathbb{R}^d$, $\mathcal{L}_{Z_n}(h) = \big(g \circ X_n \big) \,\square\, \frac{\lambda}{2} \| \cdot \|^2 (h)$ and $\Big(\frac{\lambda}{2} \| \cdot \|^2 \Big)^*(\alpha) = \frac{1}{2\lambda} \|\alpha\|^2$. ∎

D. From the Dual an ϵ-Subgradient for the Meta-Objective

In this section, we show how to exploit an ϵ-minimizer $\hat{u}_h$ of the dual problem in Eq. (47) in order to get an ϵ-subgradient of the function $\mathcal{L}_{Z_n}$ in Eq. (15)–(44) at the point h. This is described in the following proposition, which will play a fundamental role in our analysis.

Proposition 15 (ϵ-Subgradient for the Meta-Objective $\mathcal{L}_{Z_n}$). *In the setting described above, for a fixed value $h \in \mathbb{R}^d$ and a fixed parameter $\lambda > 0$, consider an ϵ-minimizer $\hat{u}_h \in \mathbb{R}^n$ of the dual objective Ψ_h in Eq. (47), for some value $\epsilon \geq 0$. Then, the vector $X_n^\top \hat{u}_h \in \mathbb{R}^d$ is an ϵ-subgradient of $\mathcal{L}_{Z_n}$ at the point h.*

Proof. By Lemma 8, the assumption that $\hat{u}_h$ is an ϵ-minimizer of Ψ_h is equivalent to the condition $0 \in \partial_\epsilon \Psi_h(\hat{u}_h)$. Now recall that, for any $u \in \mathbb{R}^n$, the expression of the dual objective is given by

$$\Psi_h(u) = g^*(u) + \frac{1}{2\lambda}\|X_n^\top u\|^2 - \langle X_n h, u \rangle. \tag{54}$$

Consequently, thanks to Lemma 10, for any $u \in \mathrm{Dom}(\Psi_h) = \mathrm{Dom}(g^*)$, we have that

$$\partial_\epsilon \Psi_h(u) = \bigcup_{0 \le \epsilon_1 + \epsilon_2 \le \epsilon} \partial_{\epsilon_1} g^*(u) + \partial_{\epsilon_2}\Big\{\frac{1}{2\lambda}\|X_n^\top \cdot\|^2 - \langle X_n h, \cdot\rangle\Big\}(u). \tag{55}$$

Thanks to Ex. 1, for any $u \in \mathbb{R}^n$, we can write

$$\partial_{\epsilon_2}\Big\{\frac{1}{2\lambda}\|X_n^\top \cdot\|^2 - \langle X_n h, \cdot\rangle\Big\}(u) = \Big\{ X_n\Big(\frac{X_n^\top u}{\lambda} - h + \frac{X_n^\top s}{\lambda}\Big) \ : \ \frac{1}{2}\Big\langle \frac{X_n X_n^\top s}{\lambda}, s \Big\rangle \le \epsilon_2 \Big\}. \tag{56}$$

Hence, we know that $0 \in \partial_\epsilon \Psi_h(\hat{u}_h)$ iff

$$\exists\, \epsilon_1, \epsilon_2, s \in \mathbb{R}^n \ : \ 0 \le \epsilon_1 + \epsilon_2 \le \epsilon, \ \frac{1}{2}\Big\langle \frac{X_n X_n^\top s}{\lambda}, s \Big\rangle \le \epsilon_2 \tag{57}$$

such that the following relations hold true

$$
\begin{aligned}
0 \in \partial_{\epsilon_1} g^*(\hat{u}_h) + X_n\Big(\frac{X_n^\top \hat{u}_h}{\lambda} - h + \frac{X_n^\top s}{\lambda}\Big) &\iff X_n\Big(h - \frac{X_n^\top(\hat{u}_h + s)}{\lambda}\Big) \in \partial_{\epsilon_1} g^*(\hat{u}_h) \\
\text{Lemma 9} \iff & \hat{u}_h \in \partial_{\epsilon_1} g\Big(X_n\Big(h - \frac{X_n^\top(\hat{u}_h + s)}{\lambda}\Big)\Big) \\
\implies & X_n^\top \hat{u}_h \in X_n^\top \partial_{\epsilon_1} g\Big(X_n\Big(h - \frac{X_n^\top(\hat{u}_h + s)}{\lambda}\Big)\Big) \\
\text{Lemma 12} \implies & X_n^\top \hat{u}_h \in \partial_{\epsilon_1}\big(g \circ X_n\big)\Big(h - \frac{X_n^\top(\hat{u}_h + s)}{\lambda}\Big) \\
\text{Lemma 9} \iff & h - \frac{X_n^\top(\hat{u}_h + s)}{\lambda} \in \partial_{\epsilon_1}\big(g \circ X_n\big)^*\big(X_n^\top \hat{u}_h\big) \\
\iff & h \in \partial_{\epsilon_1}\big(g \circ X_n\big)^*\big(X_n^\top \hat{u}_h\big) + \frac{X_n^\top(\hat{u}_h + s)}{\lambda}.
\end{aligned}
\tag{58}
$$

Now, thanks to Lemma 14, we have that, for any $\alpha \in \mathbb{R}^d$, the Fenchel conjugate function of $\mathcal{L}_{Z_n}$ is

$$\mathcal{L}_{Z_n}^*(\alpha) = \big(g \circ X_n\big)^*(\alpha) + \frac{1}{2\lambda}\|\alpha\|^2. \tag{59}$$

Hence, thanks to Lemma 10, for any $\alpha \in \mathrm{Dom}(\mathcal{L}_{Z_n}^*) = \mathrm{Dom}\big((g \circ X_n)^*\big) \supset X_n^\top \mathrm{Dom}(g^*)$, we have that

$$\partial_\epsilon \mathcal{L}_{Z_n}^*(\alpha) = \bigcup_{0 \le \epsilon_1 + \epsilon_2 \le \epsilon} \partial_{\epsilon_1}\big(g \circ X_n\big)^*(\alpha) + \partial_{\epsilon_2}\Big\{\frac{1}{2\lambda}\|\cdot\|^2\Big\}(\alpha). \tag{60}$$

Moreover, thanks to Ex. 1, we observe that

$$\partial_{\epsilon_2}\Big\{\frac{1}{2\lambda}\|\cdot\|^2\Big\}(\alpha) = \Big\{ \frac{\alpha + \tilde{s}}{\lambda} \ : \ \frac{1}{2\lambda}\|\tilde{s}\|^2 \le \epsilon_2 \Big\}. \tag{61}$$

Therefore, making the identification $\tilde{s} = X_n^\top s$, the last relation in Eq. (58) tells us

$$
\begin{aligned}
0 \in \partial_\epsilon \Psi_h(\hat{u}_h) \implies h \in \partial_{\epsilon_1}\big(g \circ X_n\big)^*\big(X_n^\top \hat{u}_h\big) + \frac{X_n^\top(\hat{u}_h + s)}{\lambda} &\implies h \in \partial_\epsilon \mathcal{L}_{Z_n}^*(X_n^\top \hat{u}_h) \\
\iff & X_n^\top \hat{u}_h \in \partial_\epsilon \mathcal{L}_{Z_n}(h),
\end{aligned}
\tag{62}
$$

where, in the last equivalence, we have used again Lemma 9. This proves the desired statement. ∎

E. SGD on the Primal: Coordinate Descent on the Dual

In this section, we focus on the within-task algorithm we adopt in the paper, namely Alg. 1. More precisely, we start from describing how the iterations generated by Alg. 1 can be considered as the primal iterations of a primal-dual algorithm in which the dual scheme consists of a coordinate descent algorithm on the dual problem. After this, we report in App. E.1 a key inequality for the dual decrease of this approach. From this result, a regret bound for Alg. 1 and the proof of Prop. 5, the key result describing the ϵ-subgradients of our meta-algorithm, can be deduced as corollaries. This is done in App. E.2 and App. E.3, respectively.

What follows is an adaptation of the theory developed in (Shalev-Shwartz & Kakade, 2009), where the authors do not emphasize the presence of the linear operator X_n and consider a slightly different dual problem. Specifically, proceeding as in (Shalev-Shwartz & Kakade, 2009), the primal-dual setting we need to consider is the following. At each iteration $k \in [n]$, we define the instantaneous primal problem

$$w_{h,k+1} = \underset{w \in \mathbb{R}^d}{\operatorname{argmin}} \; \Phi_{h,k+1}(w) \qquad \Phi_{h,k+1}(w) = \sum_{i=1}^{k} \ell_i\big(\langle x_i, w\rangle\big) + \frac{k\lambda}{2}\, \|w - h\|^2, \tag{63}$$

where, $X_k \in \mathbb{R}^{k \times d}$ is the matrix with rows only the first k input vectors. The associated dual problem reads as follows

$$\tilde{u}_{h,k+1} = \underset{\tilde{u} \in \mathbb{R}^k}{\operatorname{argmin}} \; \Psi_{h,k+1}(\tilde{u}) \qquad \Psi_{h,k+1}(\tilde{u}) = \sum_{i=1}^{k} \ell_i^*(\tilde{u}_i) - \big\langle h, X_k^\top \tilde{u}\big\rangle + \frac{1}{2k\lambda}\big\|X_k^\top \tilde{u}\big\|^2. \tag{64}$$

In the following we will adopt the convention $\Phi_{h,1} \equiv \Psi_{h,1} \equiv 0$.

Remark 1 (Strong Duality). *Similarly to what observed in Prop. 15, also in this case, strong duality holds for each instantaneous couple of primal-dual problems above, namely, for any $k \in [n]$*

$$\Phi_{h,k+1}\big(w_{h,k+1}\big) = \min_{w \in \mathbb{R}^d} \Phi_{h,k+1}(w) = - \min_{\tilde{u} \in \mathbb{R}^k} \Psi_{h,k+1}(\tilde{u}) = -\Psi_{h,k+1}\big(\tilde{u}_{h,k+1}\big). \tag{65}$$

Moreover, by the KKT conditions, we can express the primal solution by the dual one as follows

$$w_{h,k+1} = -\frac{1}{k\lambda} X_k^\top \tilde{u}_{h,k+1} + h. \tag{66}$$

Remark 2 (Link Between the Instantaneous Problems and the Original Ones). *We observe that the original primal objective Φ_h in Eq. (131) and the corresponding dual objective Ψ_h in Eq. (47) are respectively linked with the above instantaneous primal and dual objective functions in the following way*

$$\frac{1}{n}\Phi_{h,n+1}(w) = \Phi_h(w), \qquad \frac{1}{n}\Psi_{h,n+1}(\tilde{u}) = \Psi_h\Big(\frac{\tilde{u}}{n}\Big), \tag{67}$$

for any $w \in \mathbb{R}^d$ and any $\tilde{u} \in \mathbb{R}^n$.

Algorithm 3 Within-Task Algorithm, Primal-Dual Version

   Input $\lambda > 0$ regularization parameter, $h \in \mathbb{R}^d$ bias

   Initialization $\tilde{u}_h^{(1)} = 0 \in \mathbb{R}$, $w_h^{(1)} = h \in \mathbb{R}^d$

   For $k = 1$ to n

        Receive $\ell_{k,h}(\cdot) = \ell_k\big(\langle x_k, \cdot\rangle\big) + \frac{\lambda}{2}\| \cdot -h\|^2$

        Pay $\ell_{k,h}\big(w_h^{(k)}\big)$

        Update $\tilde{u}_h^{(k+1)}$ according to Eq. (68)

        Define $w_h^{(k+1)} = -\frac{1}{k\lambda} X_k^\top \tilde{u}_h^{(k+1)} + h$

   Return $\big(\tilde{u}_h^{(k)}\big)_{k=1}^{n+1}, \big(w_h^{(k)}\big)_{k=1}^{n+1}, \bar{w}_h = \frac{1}{n}\sum_{k=1}^{n} w_h^{(k)}$

As described in (Shalev-Shwartz & Kakade, 2009), we apply the coordinate descent algorithm on the instantaneous dual problem outlined in Alg. 3. More specifically, at the iteration k, the algorithm adds a coordinate at the last k-th position of the dual variable $\tilde{u}_h^{(k)}$ in the following way

$$
\tilde{u}_{h,i}^{(k+1)} = \begin{cases} u_k' & \text{if } i = k \\ \tilde{u}_{h,i}^{(k)} & \text{if } i \in [k-1], \end{cases} \tag{68}
$$

where, $u_k' \in \partial \ell_k(\langle x_k, w_h^{(k)} \rangle)$. We stress again that $\tilde{u}_h^{(k+1)} \in \mathbb{R}^k$ and $\tilde{u}_h^{(k)} \in \mathbb{R}^{k-1}$. The primal variable is then updated by the KKT conditions outlined in Eq. (66) in Rem. 1. In the next lemma, we show that, in this way, we exactly retrieve the iterations $(w_h^{(k)})_k$ generated by Alg. 1, and, consequently, the notation does not conflict with the one used in the main body.

Lemma 16. *Let $w_h^{(k+1)}$ be the update of the primal variable in Alg. 3. Then, introducing the subgradient*

$$
s_k = x_k u_k' + \lambda(w_h^{(k)} - h) \in \partial \ell_{k,h}(w_h^{(k)}), \tag{69}
$$

we can rewrite

$$
w_h^{(k+1)} = w_h^{(k)} - \frac{1}{k\lambda} s_k. \tag{70}
$$

Consequently, the primal iterations generated by Alg. 3 coincides with the iterations generated by Alg. 1 in the paper.

Proof. We start from observing that, for any $k \in [n]$, by definition, we have

$$
w_h^{(k+1)} = -\frac{1}{k\lambda} X_k^\top \tilde{u}_h^{(k+1)} + h. \tag{71}
$$

For $k = 1$ the statement holds, as a matter of fact, introducing the subgradient $s_1 = x_1 \tilde{u}_1' + \lambda(w_h^{(1)} - h) \in \partial \ell_{1,h}(w_h^{(1)})$, we can write

$$
w_h^{(2)} = -\frac{1}{\lambda} x_1 \tilde{u}_1' + h = -\frac{1}{\lambda}\big(s_1 - \lambda(w_h^{(1)} - h)\big) + h = w_h^{(1)} - \frac{1}{\lambda} s_1. \tag{72}
$$

Now, we show that the statement holds also for $k = 2, \dots, n$. Since $X_k^\top \tilde{u}_h^{(k+1)} = X_{k-1}^\top \tilde{u}_h^{(k)} + x_k u_k'$, recalling again the subgradient $s_k = x_k u_k' + \lambda(w_h^{(k)} - h) \in \partial \ell_{k,h}(w_h^{(k)})$ of the regularized loss, we can write the following

$$
\begin{aligned}
w_h^{(k+1)} &= -\frac{1}{k\lambda} X_k^\top \tilde{u}_h^{(k+1)} + h = -\frac{1}{k\lambda}\Big(X_{k-1}^\top \tilde{u}_h^{(k)} + x_k u_k'\Big) + h \\
&= \frac{(k-1)\lambda}{k\lambda}\Big(-\frac{1}{(k-1)\lambda} X_{k-1}^\top \tilde{u}_h^{(k)}\Big) - \frac{x_k u_k'}{k\lambda} + h \\
&= \frac{(k-1)\lambda(w_h^{(k)} - h) - s_k + \lambda(w_h^{(k)} - h)}{k\lambda} + h \\
&= \frac{k\lambda w_h^{(k)} - s_k}{k\lambda} = w_h^{(k)} - \frac{1}{k\lambda} s_k.
\end{aligned} \tag{73}
$$

where, in the fourth equality, we have exploited the definition of the primal iterates in Alg. 3. ∎

E.1. Main Inequality on the Dual Decrease

The next proposition is a key tool in our analysis. It coincides with a combination of slightly different versions of Lemma 2 and Thm. 1 in (Shalev-Shwartz & Kakade, 2009).

Proposition 17 (Dual Decrease of Alg. 3, (Shalev-Shwartz & Kakade, 2009, Lemma 2 and Thm. 1)). *Let $(\tilde{u}_h^{(k)})_k$, $(w_h^{(k)})_k$ be generated according to Alg. 3 for a fixed bias of $h \in \mathbb{R}^d$ and a regularization parameter $\lambda > 0$. Then, under Asm. 1 and Asm. 2, we have that*

$$
\Psi_{h,n+1}(\tilde{u}_h^{(n+1)}) - \Psi_{h,n+1}(\tilde{u}_{h,n+1}) \leq -\Big(\sum_{k=1}^n \ell_{k,h}(w_h^{(k)}) - \Phi_{h,n+1}(w_{h,n+1})\Big) + \frac{2R^2 L^2(\log(n) + 1)}{\lambda}.
$$

Proof. For any $k \in [n]$, using the convention $\Psi_{h,1} \equiv 0$, define the dual decrease

$$\Delta_k = \Psi_{h,k+1}\big(\tilde{u}_h^{(k+1)}\big) - \Psi_{h,k}\big(\tilde{u}_h^{(k)}\big). \tag{74}$$

Hence, thanks to the telescopic sum and again the assumption $\Psi_{h,1} \equiv 0$, we can write

$$\Psi_{h,n+1}\big(\tilde{u}_h^{(n+1)}\big) = \sum_{k=1}^{n} \Delta_k + \Psi_{h,1}\big(\tilde{u}_h^{(1)}\big) = \sum_{k=1}^{n} \Delta_k. \tag{75}$$

We now show that, for any $k \in [n]$, the following relation holds

$$\Delta_k = -\ell_{k,h}\big(w_h^{(k)}\big) + \frac{1}{2k\lambda}\big\|x_k u_k' + \lambda\big(w_h^{(k)} - h\big)\big\|^2. \tag{76}$$

We start from considering the case $k = 2, \dots, n$. In this case, thanks to the updating rule in Eq. (68), the fact $X_k^\top \tilde{u}_h^{(k+1)} = X_{k-1}^\top \tilde{u}_h^{(k)} + x_k u_k'$ and the closed form of the dual objective, we have that

$$
\begin{aligned}
\Delta_k &= \Psi_{h,k+1}\big(\tilde{u}_h^{(k+1)}\big) - \Psi_{h,k}\big(\tilde{u}_h^{(k)}\big) \\
&= \sum_{i=1}^{k-1} \ell_i^*\big(\tilde{u}_{h,i}^{(k)}\big) + \ell_k^*\big(u_k'\big) + \frac{1}{2k\lambda}\big\|X_{k-1}^\top \tilde{u}_h^{(k)} + x_k u_k'\big\|^2 - \big\langle h, X_{k-1}^\top \tilde{u}_h^{(k)} + x_k u_k'\big\rangle \\
&\qquad\qquad\qquad - \sum_{i=1}^{k-1} \ell_i^*\big(\tilde{u}_{h,i}^{(k)}\big) - \frac{1}{2(k-1)\lambda}\big\|X_{k-1}^\top \tilde{u}_h^{(k)}\big\|^2 + \big\langle h, X_{k-1}^\top \tilde{u}_h^{(k)}\big\rangle \\
&= \ell_k^*\big(u_k'\big) + \frac{1}{2k\lambda}\big\|X_{k-1}^\top \tilde{u}_h^{(k)} + x_k u_k'\big\|^2 - \big\langle h, x_k u_k'\big\rangle - \frac{1}{2(k-1)\lambda}\big\|X_{k-1}^\top \tilde{u}_h^{(k)}\big\|^2 \\
&= \ell_k^*\big(u_k'\big) + \frac{1}{2\lambda}\Big(\frac{1}{k} - \frac{1}{k-1}\Big)\big\|X_{k-1}^\top \tilde{u}_h^{(k)}\big\|^2 + \frac{1}{2k\lambda}\big\|x_k u_k'\big\|^2 + \Big\langle \frac{X_{k-1}^\top \tilde{u}_h^{(k)}}{k\lambda} - h, x_k u_k'\Big\rangle \\
&= \ell_k^*\big(u_k'\big) + \frac{1}{2\lambda}\Big(\frac{1}{k} - \frac{1}{k-1}\Big)\lambda^2 (k-1)^2\big\|w_h^{(k)} - h\big\|^2 + \frac{1}{2k\lambda}\big\|x_k u_k'\big\|^2 - \Big\langle \frac{(k-1)\lambda}{k\lambda}\big(w_h^{(k)} - h\big) + h, x_k u_k'\Big\rangle \\
&= \ell_k^*\big(u_k'\big) + \frac{\lambda}{2}\Big(\frac{1}{k} - 1\Big)\big\|w_h^{(k)} - h\big\|^2 + \frac{1}{2k\lambda}\big\|x_k u_k'\big\|^2 - \Big\langle \Big(1 - \frac{1}{k}\Big)\big(w_h^{(k)} - h\big) + h, x_k u_k'\Big\rangle \\
&= \Big(\ell_k^*\big(u_k'\big) - \big\langle w_h^{(k)}, x_k u_k'\big\rangle - \frac{\lambda}{2}\big\|w_h^{(k)} - h\big\|^2\Big) + \frac{1}{2k\lambda}\Big(\lambda^2\big\|w_h^{(k)} - h\big\|^2 + \big\|x_k u_k'\big\|^2 + 2\big\langle \lambda\big(w_h^{(k)} - h\big), x_k u_k'\big\rangle\Big) \\
&= -\Big(\ell_k\big(\langle x_k, w_h^{(k)}\rangle\big) + \frac{\lambda}{2}\big\|w_h^{(k)} - h\big\|^2\Big) + \frac{1}{2k\lambda}\big\|x_k u_k' + \lambda\big(w_h^{(k)} - h\big)\big\|^2 \\
&= -\ell_{k,h}\big(w_h^{(k)}\big) + \frac{1}{2k\lambda}\big\|x_k u_k' + \lambda\big(w_h^{(k)} - h\big)\big\|^2,
\end{aligned}
$$

where, in the fifth equality we have used the definition of the primal variable $w_h^{(k)} = -\frac{1}{(k-1)\lambda} X_{k-1}^\top \tilde{u}_h^{(k)} + h$, in the sixth equality we have used the relation

$$\frac{1}{\lambda}\Big(\frac{1}{k} - \frac{1}{k-1}\Big)\lambda^2 (k-1)^2 = \lambda\Big(\frac{1}{k} - 1\Big), \tag{77}$$

and, finally, in the eighth equality we have exploited the assumption $u_k' \in \partial\ell_k\big(\langle x_k, w_h^{(k)}\rangle\big)$, implying by Fenchel–Young equality

$$\ell_k^*\big(u_k'\big) - \big\langle w_h^{(k)}, x_k u_k'\big\rangle = -\ell_k\big(\langle x_k, w_h^{(k)}\rangle\big). \tag{78}$$

We now observe that the above relation in Eq. (76) holds also in the case $k = 1$, as a matter of fact, by definition, since

$\Psi_{h,1} \equiv 0$, we have

$$
\begin{aligned}
\Delta_1 &= \Psi_{h,2}\big(\tilde{u}_h^{(2)}\big) - \Psi_{h,1}\big(\tilde{u}_h^{(1)}\big) = \Psi_{h,2}\big(\tilde{u}_h^{(2)}\big) \\
&= \ell_1^*(u_1') - \langle h, x_1 u_1' \rangle + \frac{1}{2\lambda}\big\|x_1 u_1'\big\|^2 \\
&= \left(\ell_1^*(u_1') - \big\langle w_h^{(1)}, x_1 u_1' \big\rangle - \frac{\lambda\big\|w_h^{(1)} - h\big\|^2}{2}\right) + \frac{1}{2\lambda}\big\|x_1 u_1' + \lambda\big(w_h^{(1)} - h\big)\big\|^2 \\
&= -\left(\ell_1\big(\langle x_1, w_h^{(1)} \rangle\big) + \frac{\lambda}{2}\big\|w_h^{(1)} - h\big\|^2\right) + \frac{1}{2\lambda}\big\|x_1 u_1' + \lambda\big(w_h^{(1)} - h\big)\big\|^2 \\
&= -\ell_{1,h}\big(w_h^{(1)}\big) + \frac{1}{2\lambda}\big\|x_1 u_1' + \lambda\big(w_h^{(1)} - h\big)\big\|^2,
\end{aligned}
\tag{79}
$$

where, in the fourth equality we have rewritten

$$
\frac{1}{2\lambda}\big\|x_1 u_1'\big\|^2 = \frac{1}{2\lambda}\big\|x_1 u_1' + \lambda\big(w_h^{(1)} - h\big)\big\|^2 - \big\langle w_h^{(1)} - h, x_1 u_1' \big\rangle - \frac{\lambda\big\|w_h^{(1)} - h\big\|^2}{2},
\tag{80}
$$

and, in the fifth equality, we have used again the assumption $u_1' \in \partial\ell_1\big(\langle x_1, w_h^{(1)} \rangle\big)$, implying by Fenchel–Young equality

$$
\ell_1^*(u_1') - \big\langle w_h^{(1)}, x_1 u_1' \big\rangle = -\ell_1\big(\langle x_1, w_h^{(1)} \rangle\big).
\tag{81}
$$

Therefore, using Eq. (75) and summing over $k \in [n]$, we get the following

$$
\begin{aligned}
\Psi_{h,n+1}\big(\tilde{u}_h^{(n+1)}\big) &= \sum_{k=1}^n \Delta_k = -\sum_{k=1}^n \ell_{k,h}\big(w_h^{(k)}\big) + \sum_{k=1}^n \frac{1}{2k\lambda}\big\|x_k u_k' + \lambda\big(w_h^{(k)} - h\big)\big\|^2 \\
&= -\sum_{k=1}^n \ell_{k,h}\big(w_h^{(k)}\big) + \frac{1}{2\lambda}\sum_{k=1}^n \frac{1}{k}\big\|x_k u_k' + \lambda\big(w_h^{(k)} - h\big)\big\|^2.
\end{aligned}
\tag{82}
$$

Now, for $k = 2, \dots, n$, thanks to the definition of $w_h^{(k)}$, we can write

$$
\lambda\big(w_h^{(k)} - h\big) = -\frac{1}{k-1}X_{k-1}^\top \tilde{u}_h^{(k)} = -\frac{1}{k-1}\sum_{i=1}^{k-1} x_i u_i'.
\tag{83}
$$

Hence, under Asm. 1 and Asm. 2, since $|u_i'| \le L$, for any i, for $k = 2, \dots, n$, we get

$$
\big\|\lambda\big(w_h^{(k)} - h\big)\big\| \le LR.
\tag{84}
$$

Moreover, we observe that the above majorization holds also for the case $k = 1$, as a matter of fact, thanks to the definition $w_h^{(1)} = h$, we have that

$$
\big\|\lambda\big(w_h^{(1)} - h\big)\big\| = 0.
\tag{85}
$$

Hence, using the inequality $\|a + b\|^2 \le 2\|a\|^2 + 2\|b\|^2$ for any $a, b \in \mathbb{R}^d$, for $k = 1, \dots, n$, we get

$$
\big\|x_k u_k' + \lambda\big(w_h^{(k)} - h\big)\big\|^2 \le 2\big\|x_k u_k'\big\|^2 + 2\big\|\lambda\big(w_h^{(k)} - h\big)\big\|^2 \le 4R^2 L^2.
\tag{86}
$$

Finally, coming back to Eq. (82), using the inequality $\sum_{k=1}^n 1/k \le \log(n) + 1$, we get

$$
\begin{aligned}
\Psi_{h,n+1}\big(\tilde{u}_h^{(n+1)}\big) &\le -\sum_{k=1}^n \ell_{k,h}\big(w_h^{(k)}\big) + \frac{1}{2\lambda}\sum_{k=1}^n \frac{1}{k}\big\|x_k u_k' + \lambda\big(w_h^{(k)} - h\big)\big\|^2 \\
&\le -\sum_{k=1}^n \ell_{k,h}\big(w_h^{(k)}\big) + \frac{2R^2 L^2}{\lambda}\sum_{k=1}^n \frac{1}{k} \\
&\le -\sum_{k=1}^n \ell_{k,h}\big(w_h^{(k)}\big) + \frac{2R^2 L^2\big(\log(n) + 1\big)}{\lambda}.
\end{aligned}
\tag{87}
$$

The desired statement follows by adding to both sides $-\Psi_{h,n+1}\big(\tilde{u}_{h,n+1}\big)$ and observing that, by strong duality, as already observed in Eq. (65) in Rem. 1, we have that

$$\Phi_{h,k+1}\big(w_{h,k+1}\big) = \min_{w \in \mathbb{R}^d} \Phi_{h,k+1}(w) = -\min_{\tilde{u} \in \mathbb{R}^k} \Psi_{h,k+1}(\tilde{u}) = -\Psi_{h,k+1}\big(\tilde{u}_{h,k+1}\big). \tag{88}$$

∎

E.2. Regret Bound for Alg. 3

The following result is a direct corollary of Prop. 17. It is a well-known fact and it coincides with a regret bound for the iterations in Alg. 1. This result will be then used in the following App. F in order to prove Prop. 1.

Corollary 18. *Let* $\big(w_h^{(k)}\big)_k$ *be the iterations generated by Alg. 1. Then, under the same assumptions of Prop. 17, for any* $w \in \mathbb{R}^d$, *the following regret bound holds*

$$\frac{1}{n}\sum_{k=1}^{n} \ell_{k,h}\big(w_h^{(k)}\big) - \Phi_h(w) \le \frac{1}{n}\sum_{k=1}^{n} \ell_{k,h}\big(w_h^{(k)}\big) - \Phi_h(w_h) \le \frac{2R^2L^2\big(\log(n)+1\big)}{\lambda n}. \tag{89}$$

Proof. We start from observing that, as already pointed out in Rem. 2, for any $w \in \mathbb{R}^d$, we have $\Phi_{h,n+1}(w)/n = \Phi_h(w)$. Consequently, $w_h = \operatorname{argmin}_{w \in \mathbb{R}^d} \Phi(w) = \operatorname{argmin}_{w \in \mathbb{R}^d} \Phi_{h,n+1}(w) = w_{h,n+1}$. This implies $\Phi_{h,n+1}\big(w_{h,n+1}\big)/n = \Phi_h(w_h)$. Hence, thanks to this last observation, the definition of $\tilde{u}_{h,n+1}$ and Prop. 17, we can write

$$0 \le \Psi_{h,n+1}\big(\tilde{u}_h^{(n+1)}\big) - \Psi_{h,n+1}\big(\tilde{u}_{h,n+1}\big) \le -\bigg(\sum_{k=1}^{n} \ell_{k,h}\big(w_h^{(k)}\big) - \Phi_{h,n+1}\big(w_{h,n+1}\big)\bigg) + \frac{2R^2L^2\big(\log(n)+1\big)}{\lambda}.$$

The statement derives from dividing by n. The first inequality simply derives from the definition of w_h. ∎

E.3. Proof of Prop. 5

The second corollary deriving from Prop. 17 is the main tool used to prove Prop. 5. It essentially states that the last dual iteration of Alg. 3 is an ϵ-minimizer of our original dual objective Ψ_h in Eq. (47), for an appropriate value of ϵ. This observation, combined with an expectation argument and Prop. 15, allows us to build an ϵ-subgradient for the meta-objective function, as described in Prop. 5.

Corollary 19. *Let* $\tilde{u}_h^{(n+1)}$ *be the last dual iteration of Alg. 3. Then, under the same assumptions of Prop. 17, for any* $w \in \mathbb{R}^d$, *the vector* $\hat{u}_h = \tilde{u}_h^{(n+1)}/n$ *is an* ϵ-*minimizer of the dual objective* Ψ_h *in Eq. (47), with*

$$\epsilon = -\bigg(\frac{1}{n}\sum_{k=1}^{n} \ell_{k,h}\big(w_h^{(k)}\big) - \Phi_h\big(w\big)\bigg) + \frac{2R^2L^2\big(\log(n)+1\big)}{\lambda n}, \tag{90}$$

where $\big(w_h^{(k)}\big)_k$ *is the iteration generated by Alg. 1.*

Proof. We start from recalling that, as already observed in Prop. 17, the primal iterations generated by Alg. 3 coincide with the iterations generated by Alg. 1. Now, thanks to Prop. 17, dividing by n, we have that

$$\frac{1}{n}\Psi_{h,n+1}\big(\tilde{u}_h^{(n+1)}\big) - \frac{1}{n}\Psi_{h,n+1}\big(\tilde{u}_{h,n+1}\big) \le \tilde{\epsilon},$$

with

$$\tilde{\epsilon} = -\bigg(\frac{1}{n}\sum_{k=1}^{n} \ell_{k,h}\big(w_h^{(k)}\big) - \frac{1}{n}\Phi_{h,n+1}\big(w_{h,n+1}\big)\bigg) + \frac{2R^2L^2\big(\log(n)+1\big)}{\lambda n}. \tag{91}$$

As already pointed out, we now observe that, $\Phi_{h,n+1}\big(w_{h,n+1}\big)/n = \Phi_h(w_h)$, hence, for any $w \in \mathbb{R}^d$, we can rewrite

$$\begin{aligned}
\tilde{\epsilon} &= -\bigg(\frac{1}{n}\sum_{k=1}^{n} \ell_{k,h}\big(w_h^{(k)}\big) - \Phi_h(w_h)\bigg) + \frac{2R^2L^2\big(\log(n)+1\big)}{\lambda n} \\
&\le -\bigg(\frac{1}{n}\sum_{k=1}^{n} \ell_{k,h}\big(w_h^{(k)}\big) - \Phi_h(w)\bigg) + \frac{2R^2L^2\big(\log(n)+1\big)}{\lambda n} = \epsilon.
\end{aligned} \tag{92}$$

Summarizing, we have obtained that

$$0 \in \partial_\epsilon \Big(\frac{1}{n}\Psi_{h,n+1}\Big)(\tilde{u}_h^{(n+1)}), \tag{93}$$

where the value of ϵ is the one in Eq. (92). Now, we observe that, thanks to the relation $\frac{1}{n}\Psi_{h,n+1}(\tilde{u}) = \Psi_h\Big(\frac{\tilde{u}}{n}\Big) = \Big(\Psi_h \circ \frac{1}{n}\Big)(\tilde{u})$ (see Rem. 2), exploiting Lemma 11, for any $\tilde{u} \in \mathbb{R}^n$, we have that,

$$\partial_\epsilon \Big(\frac{1}{n}\Psi_{h,n+1}\Big)(\tilde{u}) = \partial_\epsilon \Big(\Psi_h \circ \frac{1}{n}\Big)(\tilde{u}) = \frac{1}{n}\partial_\epsilon \Psi_h\Big(\frac{\tilde{u}}{n}\Big). \tag{94}$$

Consequently, Eq. (93), implies $0 \in \partial_\epsilon \Psi_h\Big(\frac{\tilde{u}_h^{(n+1)}}{n}\Big)$, which is equivalent, as already observed in Lemma 8, to the desired statement. ∎

The last ingredient we need to prove Prop. 5 is the following expectation argument.

Corollary 20. *Let $\big(w_h^{(k)}\big)_k$ be the iterations generated by Alg. 1, Let ϵ be the value in Cor. 19 with $w = w_{\mu,h}$, where $w_{\mu,h} = \underset{w \in \mathbb{R}^d}{\mathrm{argmin}}\, \mathcal{R}_\mu(w) + \frac{\lambda}{2}\|w - h\|^2$. Then, under the same assumptions of Prop. 17, we have that*

$$\mathbb{E}_{Z_n \sim \mu^n}[\epsilon] \leq \frac{2R^2 L^2 (\log(n) + 1)}{\lambda n}. \tag{95}$$

Proof. We recall that the value of ϵ in Cor. 19 with $w = w_{\mu,h}$ is explicitly given by

$$\epsilon = -\Big(\frac{1}{n}\sum_{k=1}^n \ell_k(\langle x_k, w_h^{(k)}\rangle) + \frac{\lambda}{2}\|w_h^{(k)} - h\|^2 - \Phi_h(w_{\mu,h})\Big) + \frac{2R^2 L^2 (\log(n) + 1)}{\lambda n}. \tag{96}$$

Hence, to prove the statement we just need to show that

$$0 \leq \mathbb{E}_{Z_n \sim \mu^n}\Big[\frac{1}{n}\sum_{k=1}^n \ell_k(\langle x_k, w_h^{(k)}\rangle) + \frac{\lambda}{2}\|w_h^{(k)} - h\|^2 - \Phi_h(w_{\mu,h})\Big]. \tag{97}$$

In order to do this, we recall that $\bar{w}_h$ denotes the average of the first n iterations $\big(w_h^{(k)}\big)_k$ and we observe the following

$$
\begin{aligned}
0 &\leq \mathbb{E}_{Z_n \sim \mu^n}\Big[\mathcal{R}_\mu(\bar{w}_h(Z_n)) + \frac{\lambda}{2}\|\bar{w}_h(Z_n) - h\|^2\Big] - \mathbb{E}_{Z_n \sim \mu^n}\Big[\mathcal{R}_\mu(w_{\mu,h}) + \frac{\lambda}{2}\|w_{\mu,h} - h\|^2\Big] \\
&\leq \mathbb{E}_{Z_n \sim \mu^n}\Big[\frac{1}{n}\sum_{i=1}^n \mathcal{R}_\mu(w_h^{(i)}) + \frac{\lambda}{2}\|w_h^{(i)} - h\|^2\Big] - \mathbb{E}_{Z_n \sim \mu^n}\Big[\mathcal{R}_\mu(w_{\mu,h}) + \frac{\lambda}{2}\|w_{\mu,h} - h\|^2\Big] \\
&= \mathbb{E}_{Z_n \sim \mu^n}\Big[\frac{1}{n}\sum_{i=1}^n \mathcal{R}_\mu(w_h^{(i)}) + \frac{\lambda}{2}\|w_h^{(i)} - h\|^2\Big] - \mathbb{E}_{Z_n \sim \mu^n}\Big[\frac{1}{n}\sum_{i=1}^n \ell_i(\langle x_i, w_{\mu,h}\rangle) + \frac{\lambda}{2}\|w_{\mu,h} - h\|^2\Big] \\
&= \mathbb{E}_{Z_n \sim \mu^n}\Big[\frac{1}{n}\sum_{i=1}^n \ell_i(\langle x_i, w_h^{(i)}\rangle) + \frac{\lambda}{2}\|w_h^{(i)} - h\|^2\Big] - \mathbb{E}_{Z_n \sim \mu^n}\Big[\frac{1}{n}\sum_{i=1}^n \ell_i(\langle x_i, w_{\mu,h}\rangle) + \frac{\lambda}{2}\|w_{\mu,h} - h\|^2\Big] \\
&= \mathbb{E}_{Z_n \sim \mu^n}\Big[\frac{1}{n}\sum_{k=1}^n \ell_k(\langle x_k, w_h^{(k)}\rangle) + \frac{\lambda}{2}\|w_h^{(k)} - h\|^2 - \Phi_h(w_{\mu,h})\Big],
\end{aligned} \tag{98}
$$

where, the first inequality is a consequence of the definition of $w_{\mu,h}$, the second inequality derives from Jensen's inequality, the first equality holds since $w_{\mu,h}$ does not depend on the data and, finally, the second equality holds by standard online-to-batch arguments, more precisely, since $w_h^{(i)}$ does not depend on the point z_i, we have that, $\mathbb{E}_{Z_n \sim \mu^n}\ell_i(\langle x_i, w_h^{(i)}\rangle) = \mathbb{E}_{Z_n \sim \mu^n}\mathcal{R}_\mu(w_h^{(i)})$. ∎

We now are ready to prove Prop. 5.

Proposition 5 (An ϵ-Subgradient for $\mathcal{L}_{Z_n}$). *Let* $w_h{}^{(n+1)}(Z_n)$ *be the last iterate of Alg. 1. Then, under Asm. 1 and Asm. 2, the vector*

$$\hat{\nabla}\mathcal{L}_{Z_n}(h) = -\lambda\big(w_h{}^{(n+1)}(Z_n) - h\big) \tag{18}$$

is an ϵ-subgradient of $\mathcal{L}_{Z_n}$ at point h, where ϵ is such that

$$\mathbb{E}_{Z_n\sim\mu^n}\big[\epsilon\big] \leq \frac{2R^2 L^2\big(\log(n)+1\big)}{\lambda n}. \tag{19}$$

Moreover, introducing $\Delta_{Z_n}(h) = \nabla\mathcal{L}_{Z_n}(h) - \hat{\nabla}\mathcal{L}_{Z_n}(h)$,

$$\mathbb{E}_{Z_n\sim\mu^n}\big\|\Delta_{Z_n}(h)\big\|^2 \leq \frac{4R^2 L^2\big(\log(n)+1\big)}{n}. \tag{20}$$

Proof. We start from observing that, thanks to Lemma 16, we have that

$$-\lambda\big(w_h{}^{(n+1)} - h\big) = X_n^\top \frac{\tilde{u}_h^{(n+1)}}{n}. \tag{99}$$

Hence, thanks to Prop. 15 and Cor. 19 applied to the vector $\hat{u}_h = \tilde{u}_h^{(n+1)}/n$, we can state that $-\lambda\big(w_h{}^{(n+1)}-h\big) \in \partial_\epsilon \mathcal{L}_{Z_n}(h)$, with ϵ given as in Eq. (90), choosing $w = w_{\mu,h}$. Hence, the statement in Eq. (19) is a consequence of these observations and Cor. 20. Finally, we observe that, thanks to the fact $-\lambda\big(w_h{}^{(n+1)} - h\big) \in \partial_\epsilon \mathcal{L}_{Z_n}(h)$ and Eq. (42) in Ex. 2, we know that

$$\big\|\nabla\mathcal{L}_{Z_n}(h) - \hat{\nabla}\mathcal{L}_{Z_n}(h)\big\|^2 \leq 2\lambda\epsilon, \tag{100}$$

where ϵ is the same value as before. The statement in Eq. (20) derives from taking the expectation with respect to the dataset Z_n and applying again the result in Cor. 20. ∎

F. Proof of Prop. 1

In this section, we report the proof of Prop. 1 which is often used in the main body of this work. The proof exploits the regret bound for Alg. 1 given in Cor. 18 in App. E.2 and it essentially relies on online-to-batch conversion arguments.

Proposition 1. *Let Asm. 1 and Asm. 2 hold and let $\bar{w}_h$ be the output of Alg. 1. Then, we have that*

$$\mathbb{E}_{Z_n\sim\mu^n}\big[\mathcal{R}_\mu\big(\bar{w}_h(Z_n)\big) - \mathcal{R}_{Z_n,h}(w_h(Z_n))\big] \leq c_{n,\lambda}$$

$$\tag{7}$$

$$c_{n,\lambda} = \frac{2R^2 L^2\big(\log(n)+1\big)}{\lambda n}.$$

Proof. The proof is similar to the one of Cor. 20. More precisely, we can write

$$\mathbb{E}_{Z_n\sim\mu^n}\big[\mathcal{R}_\mu\big(\bar{w}_h(Z_n)\big)\big] - \mathbb{E}_{Z_n\sim\mu^n}\Big[\mathcal{R}_{Z_n}\big(w_h(Z_n)\big) + \frac{\lambda}{2}\big\|w_h(Z_n) - h\big\|^2\Big]$$

$$\leq \mathbb{E}_{Z_n\sim\mu^n}\Big[\mathcal{R}_\mu\big(\bar{w}_h(Z_n)\big) + \frac{\lambda}{2}\big\|\bar{w}_h(Z_n) - h\big\|^2\Big] - \mathbb{E}_{Z_n\sim\mu^n}\Big[\mathcal{R}_{Z_n}\big(w_h(Z_n)\big) + \frac{\lambda}{2}\big\|w_h(Z_n) - h\big\|^2\Big]$$

$$\leq \mathbb{E}_{Z_n\sim\mu^n}\Big[\frac{1}{n}\sum_{i=1}^n \mathcal{R}_\mu\big(w_h{}^{(i)}\big) + \frac{\lambda}{2}\big\|w_h{}^{(i)} - h\big\|^2\Big] - \mathbb{E}_{Z_n\sim\mu^n}\Big[\mathcal{R}_{Z_n}\big(w_h(Z_n)\big) + \frac{\lambda}{2}\big\|w_h(Z_n) - h\big\|^2\Big]$$

$$= \mathbb{E}_{Z_n\sim\mu^n}\Big[\frac{1}{n}\sum_{i=1}^n \ell_i\big(\langle x_i, w_h{}^{(i)}\rangle\big) + \frac{\lambda}{2}\big\|w_h{}^{(i)} - h\big\|^2\Big] - \mathbb{E}_{Z_n\sim\mu^n}\Big[\mathcal{R}_{Z_n}\big(w_h(Z_n)\big) + \frac{\lambda}{2}\big\|w_h(Z_n) - h\big\|^2\Big]$$

$$= \mathbb{E}_{Z_n\sim\mu^n}\Big[\frac{1}{n}\sum_{i=1}^n \ell_i\big(\langle x_i, w_h{}^{(i)}\rangle\big) + \frac{\lambda}{2}\big\|w_h{}^{(i)} - h\big\|^2 - \Phi_h(w_h(Z_n))\Big]$$

$$\leq \frac{2R^2 L^2\big(\log(n)+1\big)}{\lambda n},$$

where, in the first inequality we have exploited the non-negativity of the regularizer and in the second inequality we have applied Jensen's inequality. The first equality above holds by standard online-to-batch arguments, more precisely, since $w_h^{(i)}$ does not depend on the point z_i, we have that, $\mathbb{E}_{Z_n \sim \mu^n} \ell_i(\langle x_i, w_h^{(i)} \rangle) = \mathbb{E}_{Z_n \sim \mu^n} \mathcal{R}_\mu(w_h^{(i)})$. Finally, the last inequality is due to the application of the regret bound given in Cor. 18. ∎

G. Convergence Rate of Alg. 2

In this section, we give the convergence rate bound of Alg. 2 which is used in the paper for the proof of the excess transfer risk bound given in Thm. 6.

We recall that the meta-algorithm we adopt to estimate the bias h is SGD applied to the function $\hat{\mathcal{E}}_n(\cdot) = \mathbb{E}_{\mu \sim \rho} \mathbb{E}_{Z_n \sim \mu^n} \mathcal{L}_{Z_n}(\cdot)$. We recall also that, at each iteration t, the meta-algorithm approximate the gradient of the function $\mathcal{L}_t$ at the point $h^{(t)}$ by the vector $\hat{\nabla}_t$ which is computed as described in Eq. (21). In the subsequent analysis we use the notation

$$\mathbb{E}_{Z_n^{(t)}}[\cdot] = \mathbb{E}\left[\cdot \,\middle|\, Z_n^{(1)}, \ldots, Z_n^{(t-1)}\right], \tag{101}$$

where, the expectation must be intended with respect to the sampling of the dataset from the distribution induced by the sampling of the task $\mu \sim \rho$ and then the sampling of the dataset from that task. We observe that, thanks to Prop. 5 and the independence of $h^{(t)}$ on $Z_n^{(t)}$, we can state that this vector $\hat{\nabla}_t$ is an ϵ_t-subgradient of $\mathcal{L}_t$ at the point $h^{(t)}$, where, ϵ_t is such that

$$\mathbb{E}_{Z_n^{(t)}}[\epsilon_t] \leq \frac{2R^2 L^2 (\log(n) + 1)}{\lambda n} \tag{102}$$

$$\mathbb{E}_{Z_n^{(t)}} \|\nabla^{(t)} - \hat{\nabla}^{(t)}\|^2 \leq \frac{4R^2 L^2 (\log(n) + 1)}{n}. \tag{103}$$

Before proceeding with the proof of the convergence rate of Alg. 2, we need to introduce the following result contained in (Shalev-Shwartz & Ben-David, 2014).

Lemma 21 (Shalev-Shwartz & Ben-David (2014, Lemma 14.1)). *Let $h^{(t)}$ be update of Alg. 2. Then, for any $\hat{h} \in \mathbb{R}^d$, we have*

$$\sum_{t=1}^{T} \langle h^{(t)} - \hat{h}, \hat{\nabla}^{(t)} \rangle \leq \frac{1}{2} \left(\frac{1}{\gamma} \|h^{(1)} - \hat{h}\|^2 + \gamma \sum_{t=1}^{T} \|\hat{\nabla}^{(t)}\|^2 \right).$$

Proof. Thanks to the definition of the update, for any $\hat{h} \in \mathbb{R}^d$, we have that

$$\|h^{(t+1)} - \hat{h}\|^2 \leq \|h^{(t)} - \gamma \hat{\nabla}^{(t)} - \hat{h}\|^2 = \|h^{(t)} - \hat{h}\|^2 - 2\gamma \langle h^{(t)} - \hat{h}, \hat{\nabla}^{(t)} \rangle + \gamma^2 \|\hat{\nabla}^{(t)}\|^2.$$

Hence, rearranging the terms, we get the following

$$\langle h^{(t)} - \hat{h}, \hat{\nabla}^{(t)} \rangle = \frac{1}{2\gamma} \left(\|h^{(t)} - \hat{h}\|^2 - \|h^{(t+1)} - \hat{h}\|^2 \right) + \frac{\gamma}{2} \|\hat{\nabla}^{(t)}\|^2.$$

Summing over $t \in [T]$, exploiting the telescopic sum and the fact $-\|h^{(T+1)} - \hat{h}\|^2 \leq 0$, the statement follows. ∎

We now are ready to study the convergence rate of Alg. 2.

Proposition 22 (Convergence Rate of Alg. 2). *Let Asm. 1 and Asm. 2 hold and let $\bar{h}_T$ be the output of Alg. 2 run with step size*

$$\gamma = \frac{\sqrt{2} \|\hat{h}\|}{LR} \sqrt{\left(T \left(1 + \frac{4(\log(n) + 1)}{n} \right) \right)^{-1}}. \tag{104}$$

Then, for any $\hat{h} \in \mathbb{R}^d$, we have that

$$\mathbb{E}\, \hat{\mathcal{E}}_n(\bar{h}_T) - \hat{\mathcal{E}}_n(\hat{h}) \leq \|\hat{h}\|\, LR \sqrt{2 \left(1 + \frac{4(\log(n) + 1)}{n} \right) \frac{1}{T}} + \frac{2R^2 L^2 (\log(n) + 1)}{\lambda n},$$

where, the expectation above is with respect to the sampling of the datasets $Z_n^{(1)}, \ldots, Z_n^{(T)}$ from the environment ρ.

Proof. We start from observing that, by convexity of $\mathcal{L}_{Z_n^{(t)}}$, thanks to the fact that $\hat{\nabla}_t$ is an ϵ_t-subgradient of $\mathcal{L}_{Z_n^{(t)}}$ at the point $h^{(t)}$, for any $\hat{h} \in \mathbb{R}^d$, we can write

$$\mathcal{L}_{Z_n^{(t)}}\big(h^{(t)}\big) - \mathcal{L}_{Z_n^{(t)}}\big(\hat{h}\big) \leq \big\langle \hat{\nabla}^{(t)}, h^{(t)} - \hat{h} \big\rangle + \epsilon_t. \tag{105}$$

Now, taking the expectation with respect to the sampling of $Z_n^{(t)}$, thanks to what observed in Eq. (102), we have

$$\mathbb{E}_{Z_n^{(t)}}\Big[\mathcal{L}_{Z_n^{(t)}}\big(h^{(t)}\big) - \mathcal{L}_{Z_n^{(t)}}\big(\hat{h}\big)\Big] \leq \mathbb{E}_{Z_n^{(t)}}\big\langle \hat{\nabla}^{(t)}, h^{(t)} - \hat{h} \big\rangle + \underbrace{\mathbb{E}_{Z_n^{(t)}}\big[\epsilon_t\big]}$$

$$\leq \mathbb{E}_{Z_n^{(t)}}\big\langle \hat{\nabla}^{(t)}, h^{(t)} - \hat{h} \big\rangle + \underbrace{\frac{2R^2 L^2 \big(\log(n)+1\big)}{\lambda n}}_{\epsilon_{\lambda,n}}. \tag{106}$$

Hence, taking the global expectation, we get

$$\mathbb{E}\Big[\mathcal{L}_{Z_n^{(t)}}\big(h^{(t)}\big) - \mathcal{L}_{Z_n^{(t)}}\big(\hat{h}\big)\Big] \leq \mathbb{E}\big\langle \hat{\nabla}^{(t)}, h^{(t)} - \hat{h} \big\rangle + \epsilon_{\lambda,n}. \tag{107}$$

Summing over $t \in [T]$ and dividing by T, we get

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\Big[\mathcal{L}_{Z_n^{(t)}}\big(h^{(t)}\big) - \mathcal{L}_{Z_n^{(t)}}\big(\hat{h}\big)\Big] \leq \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\big\langle \hat{\nabla}^{(t)}, h^{(t)} - \hat{h} \big\rangle + \epsilon_{\lambda,n}. \tag{108}$$

Now, applying Lemma 21, as regards the first term of the RHS in the bound above, we can write

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\big\langle \hat{\nabla}^{(t)}, h^{(t)} - \hat{h} \big\rangle \leq \frac{1}{2}\Big(\frac{1}{\gamma T}\big\|h^{(1)} - \hat{h}\big\|^2 + \frac{\gamma}{T}\sum_{t=1}^{T}\mathbb{E}\big\|\hat{\nabla}^{(t)}\big\|^2\Big). \tag{109}$$

Now we observe that, thanks to Asm. 1, Asm. 2 and Prop. 4, $\big\|\nabla^{(t)}\big\| \leq RL$ for any $t \in [T]$. Consequently, using the inequality $\|a + b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$ for any two vectors $a, b \in \mathbb{R}^d$ and applying Eq. (103), we can write the following

$$\mathbb{E}_{Z_n^{(t)}}\big\|\hat{\nabla}^{(t)}\big\|^2 = \mathbb{E}_{Z_n^{(t)}}\big\|\hat{\nabla}^{(t)} \pm \nabla^{(t)}\big\|^2 \leq 2\,\mathbb{E}_{Z_n^{(t)}}\big\|\nabla^{(t)}\big\|^2 + 2\,\mathbb{E}_{Z_n^{(t)}}\big\|\nabla^{(t)} - \hat{\nabla}^{(t)}\big\|^2$$

$$\leq 2L^2 R^2 + \frac{8R^2 L^2 \big(\log(n)+1\big)}{n} = 2L^2 R^2 \Big(1 + \frac{4\big(\log(n)+1\big)}{n}\Big). \tag{110}$$

Hence, taking the global expectation of the above relation and combining with Eq. (109), we get

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\big\langle \hat{\nabla}^{(t)}, h^{(t)} - \hat{h} \big\rangle \leq \frac{1}{2}\Big(\frac{1}{\gamma T}\big\|h^{(1)} - \hat{h}\big\|^2 + 2L^2 R^2 \Big(1 + \frac{4\big(\log(n)+1\big)}{n}\Big)\gamma\Big). \tag{111}$$

We now observe that, as regards the LHS member in Eq. (108), by Jensen's inequality and the independence of $h^{(t)}$ on $Z_n^{(t)}$, we have that

$$\mathbb{E}\,\hat{\mathcal{E}}_n(\bar{h}_T) - \hat{\mathcal{E}}_n(\hat{h}) = \mathbb{E}\Big[\mathcal{L}_{Z_n}(\bar{h}_T) - \mathcal{L}_{Z_n}(\hat{h})\Big] \leq \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\Big[\mathcal{L}_{Z_n^{(t)}}\big(h^{(t)}\big) - \mathcal{L}_{Z_n^{(t)}}\big(\hat{h}\big)\Big]. \tag{112}$$

Hence, substituting Eq. (111) and Eq. (112) into Eq. (108), since $h^{(1)} = 0$, we get

$$\mathbb{E}\,\hat{\mathcal{E}}_n(\bar{h}_T) - \hat{\mathcal{E}}_n(\hat{h}) \leq \frac{1}{2}\Big(\frac{1}{\gamma T}\big\|\hat{h}\big\|^2 + 2L^2 R^2 \Big(1 + \frac{4\big(\log(n)+1\big)}{n}\Big)\gamma\Big) + \frac{2R^2 L^2 \big(\log(n)+1\big)}{\lambda n}.$$

The desired statement follows from optimizing the above bound with respect to γ. ∎

H. Analysis for ERM Algorithm

In this section, we repeat the statistical study described in the paper for the family of ERM algorithms introduced in Eq. (3). We obtain excess transfer risk bounds which are equivalent, up to constants and logarithmic factors, to those given in the paper for the SGD family.

We start from recalling the definition of the biased ERM algorithm in Eq. (3)

$$w_h(Z_n) = \underset{w \in \mathbb{R}^d}{\operatorname{argmin}} \ \mathcal{R}_{Z_n,h}(w), \tag{113}$$

where, for any $w, h \in \mathbb{R}^d$, the empirical error and its biased regularized version are defined respectively as

$$
\mathcal{R}_{Z_n}(w) = \frac{1}{n} \sum_{k=1}^{n} \ell_k\big(\langle x_k, w\rangle\big)
$$
$$
\mathcal{R}_{Z_n,h}(w) = \mathcal{R}_{Z_n}(w) + \frac{\lambda}{2}\|w - h\|^2. \tag{114}
$$

We assume to have an oracle providing us with this exact estimator. The study proceeds as in the paper, for the SGD family. The main difference relies on using in the decompositions, instead of Prop. 1, the following standard result on the generalization error of ERM algorithm.

Proposition 23. *Let Asm. 1 and Asm. 2 hold. Let w_h the ERM algorithm in Eq. (113). Then, for any $h \in \mathbb{R}^d$, we have that*

$$\mathbb{E}_{Z_n \sim \mu^n}\big[\mathcal{R}_\mu\big(w_h(Z_n)\big) - \mathcal{R}_{Z_n}\big(w_h(Z_n)\big)\big] \le \frac{L^2 R^2}{\lambda n}. \tag{115}$$

In order to prove Prop. 23, we recall the following standard result linking the generalization error of the algorithm with its stability. We refer to (Bousquet & Elisseeff, 2002) for more details.

Lemma 24 (Bousquet & Elisseeff (2002, Lemma 7))**.** *Let $A(Z_n)$ be a (replace-one) uniformly stable algorithm with parameter β_n. Then, we have that*

$$\mathbb{E}_{Z_n \sim \mu}\big[\mathcal{R}_\mu(A(Z_n)) - \mathcal{R}_{Z_n}(A(Z_n))\big] \le \beta_n. \tag{116}$$

We now are ready to present the proof of Prop. 23.

Proof. of Prop. 23 The proof of the statement proceeds by stability arguments. Specifically, we show that, for any h, $w_h(Z_n)$ is (replace-one) uniformly stable with parameter β_n satisfying $\beta_n \le L^2 R^2/(\lambda n)$. Denote by Z_n^i the dataset Z_n in which we change the point z_i with another independent point sample from the same task μ. Thanks to Asm. 1 and Asm. 2, we have that

$$\sup_i \left| \ell_i(\langle x_i, w_h(Z_n)\rangle) - \ell_i(\langle x_i, w_h(Z_n^i)\rangle) \right| \le LR\|w_h(Z_n) - w_h(Z_n^i)\|. \tag{117}$$

Now thanks to the λ-strong convexity of $\mathcal{R}_{Z_n,h}$, and the definition of the algorithm, we have that

$$
\frac{\lambda}{2}\big\|w_h(Z_n^i) - w_h(Z_n)\big\|^2 \le \mathcal{R}_{Z_n,h}(w_h(Z_n^i)) - \mathcal{R}_{Z_n,h}(w_h(Z_n))
$$
$$
\frac{\lambda}{2}\big\|w_h(Z_n) - w_h(Z_n^i)\big\|^2 \le \mathcal{R}_{Z_n^i,h}(w_h(Z_n)) - \mathcal{R}_{Z_n^i,h}(w_h(Z_n^i)). \tag{118}
$$

Hence, summing these two inequalities, observing that

$$
\mathcal{R}_{Z_n,h}(w_h(Z_n^i)) - \mathcal{R}_{Z_n,h}(w_h(Z_n)) + \mathcal{R}_{Z_n^i,h}(w_h(Z_n)) - \mathcal{R}_{Z_n^i,h}(w_h(Z_n^i))
$$
$$
\le \frac{1}{n}\sup_i \left| \ell_i(\langle x_i, w_h(Z_n)\rangle) - \ell_i(\langle x_i, w_h(Z_n^i)\rangle) \right|
$$

and using again Asm. 1 and Asm. 2, we can write

$$\lambda\big\|w_h(Z_n^i) - w_h(Z_n)\big\|^2 \le \frac{1}{n}\sup_i \left| \ell_i(\langle x_i, w_h(Z_n)\rangle) - \ell_i(\langle x_i, w_h(Z_n^i)\rangle) \right| \le \frac{LR}{n}\big\|w_h(Z_n) - w_h(Z_n^i)\big\|. \tag{119}$$

Hence, we get

$$\left\| w_h(Z_n^i) - w_h(Z_n) \right\| \leq \frac{LR}{\lambda n}. \tag{120}$$

Therefore, continuing with Eq. (117), we get

$$\sup_i \left| \ell_i(\langle x_i, w_h(Z_n)\rangle) - \ell_i(\langle x_i, w_h(Z_n^i)\rangle) \right| \leq \frac{L^2 R^2}{\lambda n}. \tag{121}$$

The statement follows by applying Lemma 24. ∎

We now are ready to proceed with the statistical analysis of the biased ERM algorithm. In the following App. H.1 we report the analysis for a fixed bias, while in App. H.2 we focus on the bias returned by running Alg. 2.

H.1. Analysis for a Fixed Bias

Here we study the performance of a fixed bias h. The following theorem should be compared with Thm. 2 in the paper.

Theorem 25 (Excess Transfer Risk Bound for a Fixed Bias h, ERM). *Let Asm. 1 and Asm. 2 hold. Let w_h the biased ERM algorithm in Eq. (3) with regularization parameter*

$$\lambda = \frac{RL}{\mathrm{Var}_h} \sqrt{\frac{1}{n}}. \tag{122}$$

Then, the following bound holds

$$\mathcal{E}_n(w_h) - \mathcal{E}_\rho \leq \mathrm{Var}_h \, 2RL \sqrt{\frac{1}{n}}. \tag{123}$$

Proof. For $\mu \sim \rho$, consider the following decomposition

$$\mathbb{E}_{Z_n \sim \mu^n} \mathcal{R}_\mu(w_h(Z_n)) - \mathcal{R}_\mu(w_\mu) \leq A + B, \tag{124}$$

where, A and B are respectively defined by

$$\begin{aligned} A &= \mathbb{E}_{Z_n \sim \mu^n} \left[\mathcal{R}_\mu(w_h(Z_n)) - \mathcal{R}_{Z_n}(w_h(Z_n)) \right] \\ B &= \mathbb{E}_{Z_n \sim \mu^n} \left[\mathcal{R}_{Z_n,h}(w_h(Z_n)) - \mathcal{R}_\mu(w_\mu) \right]. \end{aligned} \tag{125}$$

In order to bound the term A, we use Prop. 23. As regards the term B, we apply Eq. (14) in the paper. The desired statement derives from combining the bounds on the two terms, taking the average over $\mu \sim \rho$ and optimizing with respect to λ. ∎

H.2. Analysis for the Bias $\bar{h}_T$ Returned by Alg. 2

We now study the performance of the bias $\bar{h}_T$ returned by an exact version of Alg. 2. In this case, differently from the case analyzed in the paper for the SGD family, thanks to the assumption on the availability of the ERM algorithm in exact form and the closed form of the gradient of the meta-objective $\mathcal{L}_{Z_n}$ (see Prop. 4), Alg. 2 is assumed to run with exact meta-gradients. The following theorem should be compared with Thm. 6 in the paper.

Theorem 26 (Excess Transfer Risk Bound for the Bias $\bar{h}_T$ Estimated by Alg. 2, ERM). *Let Asm. 1 and Asm. 2 hold. Let $\bar{h}_T$ be the output of Alg. 2 with exact meta-gradients and*

$$\gamma = \frac{\|m\|}{LR} \sqrt{\frac{1}{T}}. \tag{126}$$

Consider $w_{\bar{h}_T}$ the biased ERM algorithm in Eq. (3) with bias $h = \bar{h}_T$ and regularization parameter

$$\lambda = \frac{RL}{\mathrm{Var}_m} \sqrt{\frac{1}{n}}. \tag{127}$$

Then, the following bound holds

$$\mathbb{E} \, \mathcal{E}_n(w_{\bar{h}_T}) - \mathcal{E}_\rho \leq \mathrm{Var}_m \, 2RL \sqrt{\frac{1}{n}} + \|m\| \, LR \sqrt{\frac{1}{T}}, \tag{128}$$

where the expectation above is with respect to the sampling of the datasets $Z_n^{(1)}, \dots, Z_n^{(T)}$ from the environment ρ.

Proof. We consider the following decomposition

$$\mathbb{E} \, \mathcal{E}_n(w_{\bar{h}_T}) - \mathcal{E}_\rho \leq A + B + C, \tag{129}$$

where we have defined the following terms

$$
\begin{aligned}
A &= \mathcal{E}_n(w_{\bar{h}_T}) - \hat{\mathcal{E}}_n(\bar{h}_T) \\
B &= \mathbb{E} \, \hat{\mathcal{E}}_n(\bar{h}_T) - \hat{\mathcal{E}}_n(\mathsf{m}) \\
C &= \hat{\mathcal{E}}_n(\mathsf{m}) - \mathcal{E}_\rho.
\end{aligned}
\tag{130}
$$

To bound the term A, we use Prop. 23 with $h = \bar{h}_T$ and we average with respect to $\mu \sim \rho$. As regards the term C, we apply the inequality given in Eq. (14) with $h = \mathsf{m}$ and we take again the average on $\mu \sim \rho$. Finally, the term B is the convergence rate of Alg. 2, but this time, with exact meta-gradients. Now, repeating exactly the same steps described in the proof Prop. 22 with $\hat{h} = \mathsf{m}$ and $\epsilon_{n,\lambda} = 0$, it is immediate to show that for the choice of γ given in the statement we have that

$$B = \mathbb{E} \, \hat{\mathcal{E}}_n(\bar{h}_T) - \hat{\mathcal{E}}_n(\mathsf{m}) \leq \|\mathsf{m}\| \, LR \, \sqrt{\frac{1}{T}} + \frac{2R^2 L^2}{\lambda n}.$$

The desired statement follows from combining the bounds on the three terms above and optimizing with respect to λ. ∎

Looking at the results above, we immediately see that, up to constants and logarithmic factors, the LTL bounds we have stated in the paper for the SGD family are equivalent to the ones we have reported in this appendix for the biased ERM family.

I. Hyper-parameters Tuning in the LTL Setting

Denote by $\bar{h}_{T,\lambda,\gamma}$ the output of Alg. 2 computed with T iterations (hence T tasks) with values λ and γ. In all experiments, we obtain this estimator $\bar{h}_{T_{\mathrm{tr}},\lambda,\gamma}$ by learning it on a dataset $\mathbf{Z}_{\mathrm{tr}}$ of T_{tr} *training* tasks, each comprising a dataset Z_n of n input-output pairs $(x,y) \in \mathcal{X} \times \mathcal{Y}$. We perform this meta-training for different values of $\lambda \in \{\lambda_1, \dots, \lambda_p\}$ and $\gamma \in \{\gamma_1, \dots, \gamma_r\}$ and we select the best estimator based on the prediction error measured on a separate set $\mathbf{Z}_{\mathrm{va}}$ of T_{va} *validation* tasks. Once such optimal λ and γ values have been selected, we report the average risk of the corresponding estimator on a set $\mathbf{Z}_{\mathrm{te}}$ of T_{te} *test* tasks.

In particular, for the synthetic data we considered 10 (30 for the real data) candidates values for both λ and γ in the range $[10^{-6}, 10^3]$ ($[10^{-3}, 10^3]$ for the real data) with logarithmic spacing.

Note that the tasks in the test and validation sets $\mathbf{Z}_{\mathrm{te}}$ and $\mathbf{Z}_{\mathrm{va}}$ are all provided with both a training and test dataset both sampled from the same distribution. Since we are interested in measuring the performance of the algorithm trained with n points, the training datasets have all the same sample size n as those in the meta-training datasets in $\mathbf{Z}_{\mathrm{tr}}$, while the test datasets contain n' points each, for some positive integer n'. Indeed, in order to evaluate the performance of a bias h, we need to first train the corresponding algorithm $\bar{w}_h$ on the training dataset Z_n, and then test its performance on the test set $Z'_{n'}$, by computing the empirical risk $\mathcal{R}_{Z_{n'}}(\bar{w}_h(Z_n))$.

In addition to this, since we are considering the online setting, the training datasets arrive one at the time, therefore model selection is performed *online*: the system keeps track of all candidate values $\bar{h}_{T_{\mathrm{tr}},\lambda_j,\gamma_k}$, $j \in [p]$, $k \in [r]$, and, whenever a new training task is presented, these vectors are all updated by incorporating the corresponding new observations. The best bias h is then returned at each iteration, based on its performance on the validation set $\mathbf{Z}_{\mathrm{va}}$. The previous procedure describes how to tune simultaneously both λ and γ. When the bias h we use is fixed a priori (e.g. in ITL), we just need to tune the parameter λ; in such a case the procedure is analogous to that described above.

J. Additional Experiments

Our method uses SGD (Alg. 1) in two ways (i) to estimate the meta-gradient during meta-training and (ii) to evaluate the bias during the meta-validation or testing phase. In this section, we report additional experiments, in which we compared the proposed approach with exact meta-gradient approaches based on ERM. In the following experiments we approximate the ERM algorithm by running FISTA algorithm (see e.g. Beck & Teboulle, 2009) up to convergence on the within-taks dual problem introduced in App. B, see App. J.1 below for more details.
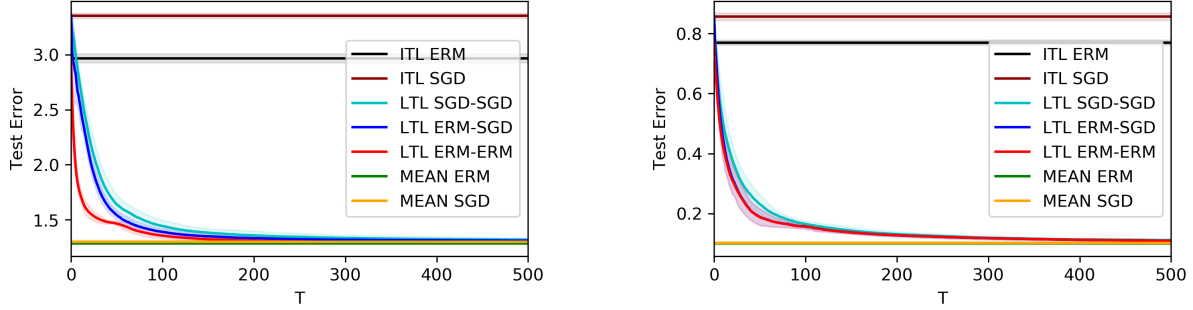
Figure 3. **Synthetic Data.** Test performance of different bias with respect to an increasing number of tasks. (Left) Regression with absolute loss. (Right) Classification with hinge loss. The results are averaged over 10 independent runs (datasets generations).
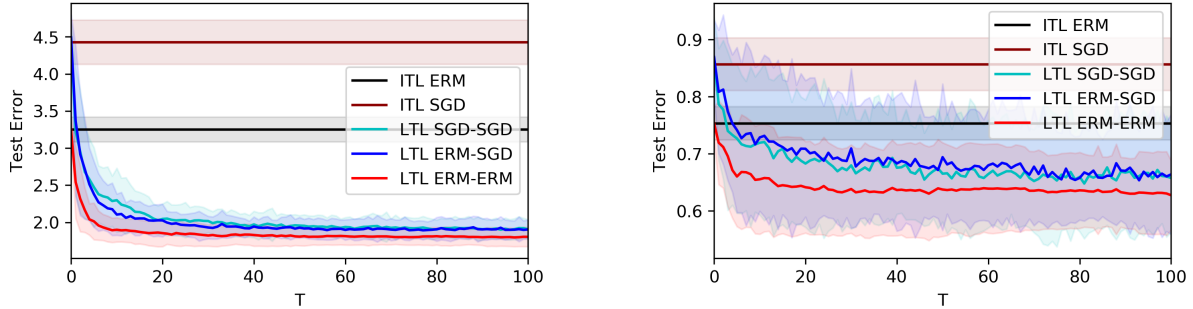

Figure 4. **Real Data.** Test performance of different bias with respect to an increasing number of tasks. (Left) Lenk Dataset Regression. (Right) Lenk Dataset Classification. The results are averaged over 30 independent runs (datasets generations).

In particular, we evaluated the following three settings:

- LTL SGD-SGD (our LTL method described in the paper): we use SGD both during meta-training and meta-validation / testing phases;

- LTL ERM-SGD: we use exact meta-gradients (computed by the ERM, as described in Prop. 4 in the text) during the meta-training phase, but we apply SGD during the meta-validation/testing;

- LTL ERM-ERM: we use ERM both for meta-training process (to compute the exact meta-gradients) and during meta-validation/testing. This is the approach we theoretically analyzed in App. H.

We also compare the above method with four ITL settings:

- ITL ERM: we perform independent task learning using the ERM algorithm with bias $h = 0$;

- ITL SGD: we perform independent task learning using the SGD algorithm with bias $h = 0$;

- MEAN ERM: we perform independent task learning using the ERM algorithm with bias $h = m$ (only in synthetic experiments, in which this quantity is available);

- MEAN SGD: we perform independent task learning using the SGD algorithm with bias $h = m$ (only in synthetic experiments in which this quantity is available).

We evaluated the performance of all the settings described above in the synthetic and real datasets used in the paper in Sec. 6. The results are reported in Fig. 3 and Fig. 4, respectively. Looking at the plots, we can observe that, in all the experiments,

SGD-SGD and ERM-SGD perform similarly. This confirms our theoretical finding: approximating the meta-gradients by SGD introduces an error which does not significanlty affect the resulting generalization performance, and, at the same time, it allows us to obtain an overall method with a very low computational cost.

We also point out that ERM-ERM achieves lower loss values than the other two LTL methods but, especially on the synthetic experiments, the difference is almost negligible and this is coherent with the results obtained in App. H. Finally, as already observed in the paper, all the LTL methods perform better than the ITL approaches (ITL ERM and ITL SGD) by a large margin, and, as expected, in the synthetic experiments, they almost match the performance of both MEAN ERM and MEAN SGD when the number of training tasks T is sufficiently large.

J.1. Approximating ERM by FISTA

In this section we describe how we apply FISTA algorithm (Beck & Teboulle, 2009) on the dual within-task problem in order to compute an approximation of the ERM algorithm in Eq. (3).

We start from recalling the the primal within-task problem

$$w_h = \underset{w \in \mathbb{R}^d}{\mathrm{argmin}}\, \Phi_h(w) \qquad \Phi_h(w) = \frac{1}{n}\sum_{i=1}^{n}\ell_i\big(\langle x_i, w\rangle\big) + \frac{\lambda}{2}\|w - h\|^2 \qquad (131)$$

and we rewrite its dual as follows

$$u_h \in \underset{u \in \mathbb{R}^n}{\mathrm{argmin}}\, \Psi_h(u) \qquad \Psi_h(u) = G(u) + F_h(u) \qquad (132)$$

$$G(u) = \frac{1}{n}\sum_{i=1}^{n}\ell_i^*(nu_i) \qquad F_h(u) = \frac{1}{2\lambda}\big\|X_n^\top u\big\|^2 - \langle X_n h, u\rangle. \qquad (133)$$

We apply FISTA to this function Ψ_h, treating F_h as the smooth part and G as the non-smooth proximable part. The primal variable is then defined as before from the dual one by the KKT conditions. The algorithm is reported in Alg. 4 below. In the experiments reported above, we run Alg. 4 for $K = 2000$ iterations or until the duality gap

$$\Phi_h\big(w_h^{(k)}\big) + \Psi_h\big(u_h^{(k)}\big) \qquad (134)$$

is lower than 10^{-6}.

Algorithm 4 Approximation of ERM by FISTA Algorithm

Input K number of iterations, $\gamma = \lambda/(nR^2)$ step size, $\lambda > 0$, $h \in \mathbb{R}^d$, $t_1 = 1$

Initialization $u_h^{(0)} = p_h^{(1)} \in \mathbb{R}^n$

For $k = 1$ to K

    Update $u_h^{(k)} = \mathrm{prox}_{\gamma G}\Big(p_h^{(k)} - \gamma \nabla F_h\big(p_h^{(k)}\big)\Big)$

    Define $w_h^{(k)} = -\frac{1}{\lambda}X_n^\top u_h^{(k)} + h$ KKT conditions

    Update $t_{k+1} = \dfrac{1 + \sqrt{1 + 4t_k^2}}{2}$

    Update $p_h^{(k+1)} = u_h^{(k)} + \dfrac{t_k - 1}{t_{k+1}}\big(u_h^{(k)} - u_h^{(k-1)}\big)$

Return $w_h^{(K)} \approx w_h$

More precisely, we observe that, thanks to Asm. 1, for any $h \in \mathbb{R}^d$, F_h is (nR^2/λ)-smooth. As a matter of fact, for any $u \in \mathbb{R}^n$, its gradient is given by

$$\nabla F_h(u) = \frac{1}{\lambda}X_n X_n^\top u - X_n h \qquad (135)$$

and $\left\| X_n X_n^\top \right\|_\infty \le nR^2$. The term G play the role of the non-smooth part and, thanks to its separability, for any step-size $\gamma > 0$, any $i \in [n]$ and any $u \in \mathbb{R}^n$, we have

$$\left(\mathrm{prox}_{\gamma G}(u) \right)_i = \frac{1}{n} \, \mathrm{prox}_{n\gamma \ell_i^*}(nu_i). \tag{136}$$

Note that, by Moreau's Identity (Bauschke & Combettes, 2011, Thm. 14.3), for any $\eta > 0$ and any $a \in \mathbb{R}$, we have $\mathrm{prox}_{\eta \ell_i^*}(a) = a - \eta \mathrm{prox}_{\frac{1}{\eta}\ell_i}(a/\eta)$. At last, we report the conjugate, the subdifferential and the closed form of the proximity operator for the absolute and the hinge loss used in our experiments.

Example 3 (Absolute Loss for Regression and Binary Classification). *Let $\mathcal{Y} \subseteq \mathbb{R}$ or $\mathcal{Y} = \{\pm 1\}$. For any $\hat{y}, y \in \mathcal{Y}$, let $\ell(\hat{y}, y) = |\hat{y} - y|$ and denote $\ell_y(\cdot) = \ell(\cdot, y)$. Then, we have*

$$u \in \partial \ell_y(\hat{y}) \iff u \in \begin{cases} \{1\} & \text{if } \hat{y} - y > 0 \\ \{-1\} & \text{if } \hat{y} - y < 0 \\ [-1, 1] & \text{if } \hat{y} - y = 0. \end{cases} \tag{137}$$

Moreover, for any $y \in \mathcal{Y}$, $\ell_y(\cdot)$ is 1-Lipschitz, and, for any $u \in \mathbb{R}$, $\eta > 0$, $a \in \mathbb{R}$, we have that

$$\ell_y^*(u) = \iota_{[-1,1]}(u) + uy \tag{138}$$

$$\mathrm{prox}_{\frac{1}{\eta}\ell_y}(a) = \begin{cases} a - 1/\eta & \text{if } a - y > 1/\eta \\ y & \text{if } a - y \in \left[-1/\eta, 1/\eta \right] \\ a + 1/\eta & \text{if } a - y < -1/\eta. \end{cases} \tag{139}$$

Example 4 (Hinge Loss for Binary Classification). *Let $\mathcal{Y} = \{\pm 1\}$. For any $\hat{y}, y \in \mathcal{Y}$, let $\ell(\hat{y}, y) = \max\{0, 1 - y\hat{y}\}$ and denote $\ell_y(\cdot) = \ell(\cdot, y)$. Then, we have*

$$u \in \partial \ell_y(\hat{y}) \iff u \in \begin{cases} \{-y\} & \text{if } 1 - y\hat{y} > 0 \\ \{0\} & \text{if } 1 - y\hat{y} < 0 \\ [-1, 1]\{-y\} & \text{if } 1 - y\hat{y} = 0. \end{cases} \tag{140}$$

Moreover, for any $y \in \mathcal{Y}$, $\ell_y(\cdot)$ is 1-Lipschitz, and, for any $u \in \mathbb{R}$, $\eta > 0$, $a \in \mathbb{R}$, we have that

$$\ell_i^*(u) = \frac{u}{y} + \iota_{[-1,0]}\left(\frac{u}{y}\right) \tag{141}$$

$$\mathrm{prox}_{\frac{1}{\eta}\ell_y}(a) = \begin{cases} a + y/\eta & \text{if } ya < 1 - y^2/\eta \\ 1/y & \text{if } ya \in \left[1 - y^2/\eta, 1\right] \\ a & \text{if } ya > 1. \end{cases} \tag{142}$$