entropy

MDPI

Article

Meta-Strategy for Learning Tuning Parameters with Guarantees

Dimitri Meunier [1] and Pierre Alquier [2,*]

[1] Istituto Italiano di Tecnologia, 16163 Genoa, Italy; dimitri.meunier.21@ucl.ac.uk
[2] RIKEN Center for Advanced Intelligence Project, Tokyo 103-0027, Japan
* Correspondence: pierrealain.alquier@riken.jp

Abstract: Online learning methods, similar to the online gradient algorithm (OGA) and exponentially weighted aggregation (EWA), often depend on tuning parameters that are difficult to set in practice. We consider an online meta-learning scenario, and we propose a meta-strategy to learn these parameters from past tasks. Our strategy is based on the minimization of a regret bound. It allows us to learn the initialization and the step size in OGA with guarantees. It also allows us to learn the prior or the learning rate in EWA. We provide a regret analysis of the strategy. It allows to identify settings where meta-learning indeed improves on learning each task in isolation.

Keywords: meta-learning; hyperparameters; priors; online learning; Bayesian inference; online optimization; gradient descent

Citation: Meunier, D.; Alquier, P. Meta-Strategy for Learning Tuning Parameters with Guarantees. *Entropy* **2021**, *23*, 1257. https://doi.org/10.3390/e23101257

Academic Editor: Gholamreza Anbarjafari

Received: 9 August 2021
Accepted: 23 September 2021
Published: 27 September 2021

1. Introduction

In many applications of modern supervised learning, such as medical imaging or robotics, a large number of tasks is available but many of them are associated with a small amount of data. With few datapoints per task, learning them in isolation would give poor results. In this paper, we consider the problem of learning from a (large) sequence of regression or classification tasks with small sample size. By exploiting their similarities we seek to design algorithms that can utilize previous experience to rapidly learn new skills or adapt to new environments.

Inspired by human ingenuity in solving new problems by leveraging prior experience, *meta-learning* is a subfield of machine learning whose goal is to automatically adapt a learning mechanism from past experiences to rapidly learn new tasks with little available data. Since it "learns the learning mechanism" it is also referred to as *learning-to-learn* [1]. It is seen as a critical problem for the future of machine learning [2]. Numerous formulations exist for meta-learning and we focus on the problem of *online meta-learning* where the tasks arrive one at a time and the goal is to efficiently transfer information from the previous tasks to the new ones such that we learn the new tasks as efficiently as possible (this has also been refered to as *lifelong learning*). Each task is in turn processed *online*. To sum up, we have a stream of tasks and for each task a stream of observations.

In order to solve online tasks, diverse well-established strategies exist: perceptron, online gradient algorithm (OGA), online mirror descent, follow-the-regularized-leader, exponentially weighted aggregation (EWA, also refered to as *generalized Bayes* etc.) We refer the reader to [3–6] for introductions to these algorithms and to so-called regret bounds, that control their generalization errors. We refer to these algorithms as the *within-task* strategies. The big challenge is to design a meta-strategy that uses past experiences to adapt a within-task strategy to perform better on the next tasks.

In this paper, we propose a new meta-learning strategy. The main idea to learn the tuning parameters is to minimize its regret bound. We provide a meta-regret analysis for our strategy. We illustrate our results in the case where the within-task strategy is the online gradient algorithm, and exponentially weighted aggregation. In the case of OGA, the tuning parameters considered are the initialization and the gradient steps. For EWA,

we consider either the learning rate, or the prior. In each case, we compare the regret incurred when learning the tasks in isolation to our meta-regret bound. This allows us to identify settings where meta-learning indeed improves on learning in isolation.

1.1. Related Works

Meta-learning is similar to multitask learning [7–9] in the sense that the learner faces many tasks to solve. However, in multitask learning, the learner is given a fixed number of tasks, and can learn the connections between these tasks. In meta-learning, the learner must prepare to face future tasks that are not given yet.

Meta-learning is often referred to as learning-to-learn or lifelong learning. The authors of [10] proposed the following distinction: "learning-to-learn" for situations where the tasks are presented simultaneously, and "lifelong learning" for situations where they are presented sequentially. Following this terminology, learning-to-learn algorithms were proposed very early in the literature, with generalization guarantees [11–16].

On the other hand, in the lifelong learning scenario, until recently, algorithms were proposed without generalization guarantees [17,18]. A theoretical study was proposed by [10], but the strategies in that paper are not feasible in practice. This problem was recently improved [19–26]. In a similar context, in [27], the authors propose an efficient strategy to learn the starting point of OGA. However, an application of this strategy to learning the step size do not show any improvement over learning in isolation [28]. The closest work to this paper is [29] in which they also suggest a regret bound minimization strategy. This paper indeed provides a meta-regret bound for learning both the initialization and the gradient step. Note, however, that this paper remains specific to OGA, while our work can be potentially applied to any online learning algorithm. Indeed, we provide another example: the generalized Bayesian algorithm EWA, for which we learn the prior, or the learning rate. To learn the prior is new in the online setting, to our knowledge. It can be related to works in the batch setting [11,13,15,16], but the improvement with respect to learning in isolation is not quantified in these works.

Finally, it is important to note that we focus on the case where the number of tasks T is large, while the sample size n and algorithmic complexity of each task is moderately small. When each task is extremely complex, for example training a deep neural network on a huge dataset, our procedure (as well as those discussed above) will become too expansive. Alternative approaches were proposed, based on optimization via multi-armed bandits [30,31].

1.2. Organization of the Paper

In Section 2, we introduce the formalism of meta-learning and the notations that will be used throughout the paper. In Section 3, we introduce our meta-learning strategy, and its theoretical analysis. In Section 4, we provide the details of our method in the case of meta-learning the initialization and the step size in the online gradient algorithm. Based on our theoretical results, there are also explicit situations where meta-learning indeed improves on learning the tasks independently. This is confirmed by experiments reported in this section. In Section 5, we provide the details of our methodology when the algorithm used within tasks is a generalized Bayesian algorithm: EWA. We show how our meta-strategy can be used to tune the learning rate; we also discuss how it can be used to learn priors. The proofs of the main results are given in Section 6.

2. Notations and Preliminaries

By convention, vectors $v \in \mathbb{R}^d$ are seen as $d \times 1$ matrices (columns). Let $\|v\|$ denote the Euclidean norm of v. Let A^T denote the transposition of any $d \times k$ matrix A, and I_d the $d \times d$ identity matrix. For two real numbers a and b, let $a \vee b = \max(a, b)$ and $a \wedge b = \min(a, b)$. For $z \in \mathbb{R}$, z_+ is its positive part $z_+ = z \vee 0$. Given a finite set S, we let $\mathrm{card}(S)$ denote the cardinality of S.

The learner has to solve tasks $t = 1, \dots, T$ sequentially. Each task t consists in n rounds $i = 1, \dots, n$. At each round i of task t, the learner has to take a decision $\theta_{t,i}$ in a decision space $\Theta \subseteq \mathbb{R}^d$ for some $d > 0$. Then, a convex loss function $\ell_{t,i} : \Theta \to \mathbb{R}$ is revealed to the learner, who incurs the loss $\ell_{t,i}(\theta_{t,i})$. Classical examples with $\Theta \subset \mathbb{R}^d$ include regression tasks, where $\ell_{t,i}(\theta) = (y_{t,i} - x_{t,i}^T\theta)^2$ for some $x_{t,i} \in \mathbb{R}^d$ and $y_{t,i} \in \mathbb{R}$. For classification tasks, $\ell_{t,i}(\theta) = (1 - y_{t,i}x_{t,i}^T\theta)_+$ for some $x_{t,i} \in \mathbb{R}^d$, $y_{t,i} \in \{-1, +1\}$.

Throughout the paper, we will assume that the learner uses, for each task, an online decision strategy called *within-task strategy*, parametrized by a tuning parameter $\lambda \in \Lambda$ where Λ is a closed, convex subset of $\mathbb{R}^p$ for some $p > 0$. Example of such strategies include the online gradient algorithm, given by $\theta_{t,i} = \theta_{t,i-1} - \gamma\nabla\ell_{t,i}(\theta_{t,i-1})$. In this case, the tuning parameters are the initialization, or starting point, $\theta_{t,1} = \vartheta$ and the learning rate, or step size, γ. That is, $\lambda = (\vartheta, \gamma)$, so $p = d + 1$. The parameter λ is kept fixed during the whole task. It is of course possible to use the same parameter λ in *all* the tasks. However, we will be interested here in defining *meta-strategies* that will allow us to improve λ task after task, based on the information available so far. In Section 3, we will define such strategies. For now, let λ_t denote the tuning parameter used by the learner all along task t. Figure 1 provides a recap of all the notations.

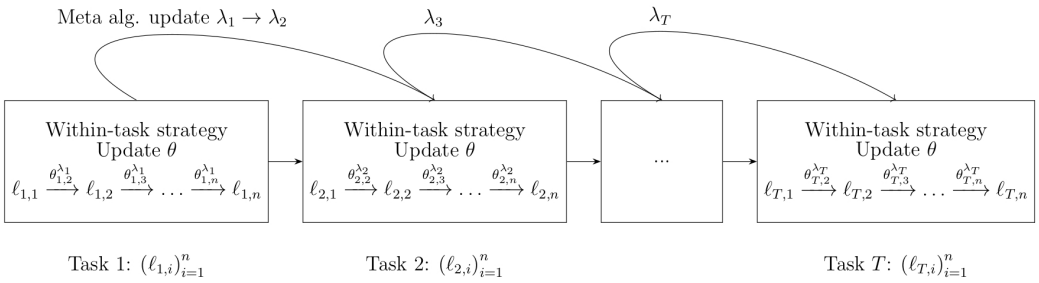

Figure 1. The dynamics of meta-learning.

Let $\theta_{t,i}^\lambda$ denote the decision at round i of task t when the online strategy is used with parameter λ. We will assume that a regret bound is available for the within-task strategy. By this, we mean that there is a set $\Theta_0 \subset \Theta$ of parameters of interest, and that the learner knows a function $\mathcal{B}_n : \Theta \times \Lambda \to \mathbb{R}$ such that, for any task t, for any $\lambda \in \Lambda$,

$$\sum_{i=1}^n \ell_{t,i}(\theta_{t,i}^\lambda) \leq \underbrace{\inf_{\theta\in\Theta_0} \left\{ \sum_{i=1}^n \ell_{t,i}(\theta) + \mathcal{B}_n(\theta, \lambda) \right\}}_{=:\mathcal{L}_t(\lambda)}. \tag{1}$$

For OGA, regret bounds can be found, for example, in [4,6] (in this case, $\Theta_0 = \Theta$). Other examples include exponentially weighted aggregation (bounds in [3], here Θ_0 is a finite set of predictors while decisions Θ are probability distributions on Θ_0). More examples will be discussed in the paper. For a fixed parameter θ, the quantity $\sum_{i=1}^n \ell_{t,i}(\theta_{t,i}^\lambda) - \sum_{i=1}^n \ell_{t,i}(\theta)$ measures the difference between the total loss suffered during task t, and the loss what one would have suffered using the parameter θ. It is thus called "the regret with respect to parameter θ", and $\mathcal{B}_n(\theta, \lambda)$ is usually referred to as a "regret bound". We will call $\mathcal{L}_t(\lambda)$ the "meta-loss". In [29], the authors study a meta-strategy that minimizes the meta-loss of OGA. Indeed, if (1) is tight, to minimize the right-hand side is a good way to ensure that the left-hand side, that is, the cumulated loss, is small. In this work, we will focus on meta-strategies minimizing the meta-loss in a more general context.

The simplest meta-strategy is learning in isolation. That is, we keep $\lambda_t = \lambda_0 \in \Lambda$ for all tasks. The total loss after task T is then given by:

$$\sum_{t=1}^T \sum_{i=1}^n \ell_{t,i}(\theta_{t,i}^{\lambda_0}) \leq \sum_{t=1}^T \mathcal{L}_t(\lambda_0). \tag{2}$$

However, when the learner uses a meta-strategy to improve the tuning parameter at the end of each task, the total loss is given by $\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\lambda_t})$. We will, in this paper, investigate strategies with meta-regret bounds; that is, bounds of the form

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\lambda_t}) \leq \inf_{\lambda\in\Lambda}\left\{\sum_{t=1}^{T}\mathcal{L}_t(\lambda)+\mathcal{C}_T(\lambda)\right\}. \tag{3}$$

Of course, such bounds will be relevant only if the right-hand side of (3) is not larger than the right-hand side of (2), and is significantly smaller in some favourable settings. We show when this is the case in Section 4.

3. Meta-Learning Algorithms

In this section, we provide two meta-strategies to update λ at the end of each task. The first one is a direct application of OGA to meta-learning. It is computationally simpler, but feasible only in the special case where we have an explicit formula for the (sub-)gradient of each $\mathcal{L}_t(\lambda)$. The second one is an application of implicit online learning to our setting. In Section 4, we provide an example where this is the case. The second meta-strategy can be used without this assumption. In both cases, we provide a regret bound as (3), under the following condition.

Assumption 1. *For any $t\in\{1,\dots,T\}$, the function $\lambda\mapsto\mathcal{L}_t(\lambda)$ is L-Lipschitz and convex.*

3.1. Special Case: The Gradient of the Meta-Loss Is Available in Closed Form

As each $\mathcal{L}_t$ is convex, its subdifferential at each point of Λ is non-empty. For the sake of simplicity, we will use the notation $\lambda\mapsto\nabla\mathcal{L}_t(\lambda)$ in the following formulas to denote *any* element of its subdifferential at λ. We define the online gradient meta-strategy (OGMS) with step $\alpha>0$ and starting point $\lambda_1\in\Lambda$: for any $t>1$,

$$\lambda_t=\Pi_\Lambda[\lambda_{t-1}-\alpha\nabla\mathcal{L}_{t-1}(\lambda_{t-1})] \tag{4}$$

where Π_Λ denotes the orthogonal projection on Λ.

3.2. The General Case

We now cover the general case, where a formula for the gradient of $\mathcal{L}_t(\lambda)$ might not be available. We propose to apply a strategy that was first defined in [32] for online learning, and studied under the name "implicit online learning" (we refer the reader to [33] and the references therein). In the meta-learning context, this gives the online proximal meta-strategy (OPMS) with step $\alpha>0$ and starting point $\lambda_1\in\Lambda$, defined by:

$$\lambda_t=\operatorname*{argmin}_{\lambda\in\Lambda}\left\{\mathcal{L}_{t-1}(\lambda)+\frac{\|\lambda-\lambda_{t-1}\|^2}{2\alpha}\right\}. \tag{5}$$

Using classical notations, e.g., [34], we can rewrite this definition with the proximal operator (hence the name of the method). Indeed $\lambda_t=\operatorname{prox}_{\alpha\mathcal{L}_{t-1}}(\lambda_{t-1})$ where prox is the proximal operator given for any $x\in\Lambda$ and any convex function $f:\Lambda\to\mathbb{R}$,

$$\operatorname{prox}_f(x)=\operatorname*{argmin}_{\lambda\in\Lambda}\left\{f(\lambda)+\frac{\|x-\lambda\|^2}{2}\right\}. \tag{6}$$

This strategy is feasible in practice in the regime we are interested in; that is, when n is small or moderately large, and $T\to\infty$. The learner has to store all the losses of the current

task $\ell_{t-1,1}, \ldots, \ell_{t-1,n}$. At the end of the task, the learner can use any convex optimization algorithm to minimize, with respect to $(\theta, \lambda) \in \Theta \times \Lambda$, the function

$$F_t(\theta, \lambda) = \sum_{i=1}^{n} \ell_{t,i}(\theta) + \mathcal{B}_n(\theta, \lambda) + \frac{\|\lambda - \lambda_{t-1}\|^2}{2\alpha}. \tag{7}$$

We can use a (projected) gradient descent on F_t or its accelerated variants [35].

3.3. Regret Analysis

A direct application of known results to the setting of this paper leads to the following proposition. For the sake of completeness, we still provide the proofs in Section 6.

Proposition 1. *Under Assumption 1, using either OGMS or OPMS with step $\alpha > 0$ and starting point $\lambda_1 \in \Lambda$ leads to*

$$\sum_{t=1}^{T} \sum_{i=1}^{n} \ell_{t,i}(\theta_{t,i}^{\lambda_t}) \leq \inf_{\lambda \in \Lambda} \left\{ \sum_{t=1}^{T} \mathcal{L}_t(\lambda) + \frac{\alpha T L^2}{2} + \frac{\|\lambda - \lambda_1\|^2}{2\alpha} \right\}. \tag{8}$$

The proof can be found in Section 6.

4. Example: Learning the Tuning Parameters of Online Gradient Descent

In all this section, we work under the following condition.

Assumption 2. *For any $(t, i) \in \{1, \ldots, T\} \times \{1, \ldots, n\}$, the function $\ell_{t,i}$ is Γ-Lipschitz and convex.*

4.1. Explicit Meta-Regret Bound

We study the situation where the learner uses (projected) OGA as a within-task strategy; that is, $\Theta = \{\theta \in \mathbb{R}^d : \|\theta\| \leq C\}$ and, for any $i > 1$,

$$\theta_{t,i} = \Pi_\Theta[\theta_{t,i-1} - \gamma \nabla \ell_{t,i}(\theta_{t,i-1})]. \tag{9}$$

With such a strategy, we already mentioned that $\lambda = (\vartheta, \gamma) \in \Lambda \subset \Theta \times \mathbb{R}_+$ contains an initialization and a step size. An application of the results in Chapter 11 in [3] gives $\mathcal{B}_n(\theta, \lambda) = \mathcal{B}_n(\theta, (\vartheta, \gamma)) = \gamma \Gamma^2 n/2 + \|\theta - \vartheta\|^2/(2\gamma)$. So

$$\mathcal{L}_t((\vartheta, \gamma)) = \inf_{\|\theta\| \leq C} \left\{ \sum_{i=1}^{n} \ell_{t,i}(\theta) + \frac{\gamma \Gamma^2 n}{2} + \frac{\|\theta - \vartheta\|^2}{2\gamma} \right\}. \tag{10}$$

It is quite direct to check Assumption 1. We summarize this in the following proposition.

Proposition 2. *Under Assumption 2, assume that the learner uses OGA as an inner algorithm. Assume $\Lambda = \{\vartheta \in \mathbb{R}^d : \|\vartheta\| \leq C\} \times [\underline{\gamma}, \bar{\gamma}]$ for some $C > 0$ and $0 < \underline{\gamma} < \bar{\gamma} < \infty$. Then Assumption 1 is satisfied with*

$$L := \sqrt{\frac{n^2 \Gamma^4}{4} + \frac{4C^2}{\underline{\gamma}^2} + \frac{4C^4}{\underline{\gamma}^4}}. \tag{11}$$

So, when the learner uses one of the meta-strategies OGMS or OPMS, we can apply Proposition 1 respectively. This leads to the following theorem.

Theorem 1. *Under the assumptions of Proposition 2, with $\underline{\gamma} = 1/n^\beta$ for some $\beta > 0$ and $\bar{\gamma} = C^2$, when the learner uses either OGMS or OPMS with*

$$\alpha = \frac{C}{L}\sqrt{\frac{4+C^2}{T}} \tag{12}$$

(where L is given by (11)), we have:

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\lambda_t}) \leq \inf_{\theta_1,\dots,\theta_T\in\Theta}\left\{\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_t) + \mathcal{C}(\Gamma,C)\left[n^{1\vee 2\beta}\sqrt{T} + \left(n^{1-\beta}+\sigma(\theta_1^T)\sqrt{n}\right)T\right]\right\} \tag{13}$$

where $\mathcal{C}(\Gamma,C) > 0$ depends only on (Γ,C) and where:

$$\sigma(\theta_1^T) = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left\|\theta_t - \frac{1}{T}\sum_{s=1}^{T}\theta_s\right\|^2}. \tag{14}$$

Let us compare this result with learning in isolation, as defined in (2); that is, solving the sequence of tasks with a constant hyperparameter $\lambda = (\vartheta,\gamma)$. For the usual choice $\vartheta = 0$ and $\gamma = c/\sqrt{n}$ where c is a constant that does not depend on n nor T, OGA leads to a regret in $\mathcal{O}(\sqrt{n})$. After T tasks, learning in isolation thus leads to a regret in $T\sqrt{n}$. Our strategies with $\beta = 1$ lead to a regret in

$$n^2\sqrt{T} + \left(1+\sigma(\theta_1^T)\sqrt{n}\right)T. \tag{15}$$

The term $n^2\sqrt{T}$ is the price to pay for meta-learning. In the regime we are interested in (small n, large T), which is smaller than $T\sqrt{n}$. Consider the leading term. In the worst case scenario, this is also $T\sqrt{n}$. However, there are good predictors $\theta_1,\dots,\theta_T$ for tasks $1,\dots,T$, respectively, such that $\sigma(\theta_1^T)$ is small, and in this case we see the improvement with respect to learning in isolation. The extreme case is when there is a good predictor θ^* that predicts well for all tasks. In this case, regret with respect to $\theta_1 = \cdots = \theta_T = \theta^*$ is in $n^2\sqrt{T}+T$, which improves significantly on learning in isolation. Note however that, using a different meta-strategy, specifically designed for OGA, Ref. [29] obtain a better dependence on T when $\sigma(\theta_1^T) = 0$.

Let us now discuss the implementation of our meta-stategy. We first remark that under the quadratic loss, it is possible to derive a formula for $\mathcal{L}_t$, which allows to use OGMS. We then discuss OPMS for the general case.

4.2. Special Case: Quadratic Loss

First, consider $\ell_{t,i} = (y_{t,i} - x_{t,i}^T\theta)^2$ for some $y_{t,i}\in\mathbb{R}$ and $x_{t,i}\in\mathbb{R}^d$. Assumption 2 is satisfied if we assume, moreover that all $|y_{t,i}|\leq c$ and $\|x_{t,i}\|\leq b$, with $\Gamma = 2bc + 2b^2C$. In this case,

$$\mathcal{L}_t((\vartheta,\gamma)) = \inf_{\|\theta\|\leq C}\left\{\sum_{i=1}^{n}(y_{t,i}-x_{t,i}^T\theta)^2 + \frac{\gamma\Gamma^2 n}{2} + \frac{\|\theta-\vartheta\|^2}{2\gamma}\right\}. \tag{16}$$

Define $Y_t = (y_{t,1},\dots,y_{t,n})^T$ and $X_t = (x_{t,1}|\dots|x_{t,n})^T$. The minimizer of $\sum_{i=1}^{n}(y_{t,i} - x_{t,i}^T\theta)^2 + \|\theta-\vartheta\|^2/(2\gamma)$ with respect to θ is known as the ridge regression estimator:

$$\hat{\theta}_t = \left(X_t^TX_t + \frac{I_d}{2\gamma}\right)^{-1}\left(X_t^TY_t + \frac{\vartheta}{2\gamma}\right). \tag{17}$$

This also coincides with the minimizer in the right-hand side of (16) on the condition that $\|\hat{\theta}_t\|\leq C$. In this case, by plugging $\hat{\theta}_t$ in (16), we have a close form formula for $\mathcal{L}_t((\vartheta,\gamma))$, and an explicit (but cumbersome) formula for its gradient. It is thus possible to use the OGMS strategy to update $\lambda = (\vartheta,\gamma)$.

4.3. The General Case

In the general case, denote $\lambda_{t-1} = (\vartheta_{t-1}, \gamma_{t-1})$, then $\lambda_t = (\vartheta_t, \gamma_t)$ is obtained by minimizing

$$F_t(\theta, (\vartheta, \gamma)) = \sum_{i=1}^{n} \ell_{t,i}(\theta) + \frac{\gamma \Gamma^2 n}{2} + \frac{\|\theta - \vartheta\|^2}{2\gamma} + \frac{\|\vartheta - \vartheta_{t-1}\|^2 + (\gamma - \gamma_{t-1})^2}{2\alpha} \tag{18}$$

with respect to $\theta, \vartheta, \gamma$. Any efficient minimization procedure can be used. In our experiments, we used a projected gradient descent, the gradient being given by:

$$\frac{\partial F_t}{\partial \theta} = \sum_{i=1}^{n} \nabla \ell_{t,i}(\theta) + \frac{\theta - \vartheta}{\gamma}, \tag{19}$$

$$\frac{\partial F_t}{\partial \vartheta} = \frac{\vartheta - \theta}{\gamma} + \frac{\vartheta - \vartheta_{t-1}}{\alpha}, \tag{20}$$

$$\frac{\partial F_t}{\partial \gamma} = \frac{\Gamma^2 n}{2} - \frac{\|\theta - \vartheta\|^2}{2\gamma^2} + \frac{\gamma - \gamma_{t-1}}{\alpha}. \tag{21}$$

Note that even though we do not *stricto sensu* obtain the minimizer of F_t, we can get arbitrarily close to it by taking a large enough number of steps. The main difference between this algorithm and the strategy suggested in [29] is that it is obtained by applying the general proximal update introduced in Equation (7), while they decoupled the update for the initialization step and the learning rate.

4.4. Experimental Study

In this section we compare simulated data for the numerical performance of OPMS w.r.t learning the task in isolation with online gradient descent (I-OGA). To measure the impact of learning the gradient step γ, we also introduce mean-OPMS that uses the same strategy as OPMS but only learns the starting point ϑ (it is thus close to [27]). We present the results for regression tasks with the mean-squared-error loss, and then for classification with the hinge loss. The notebooks of the experiments can be found online: https://dimitri-meunier.github.io/ (accessed on 26 September 2021).

4.4.1. Synthetic Regression

At each round $t = 1, \dots, T$, the meta learner sequentially receives a regression task that corresponds to a dataset $(x_{t,i}, y_{t,i})_{i=1,\dots,n}$ generated as $y_{t,i} = x_{t,i}^T \theta_t + \epsilon_{t,i}$, $x_{t,i} \in \mathbb{R}^d$. The noise is $\epsilon_{t,i} \sim \mathcal{U}([-\sigma^2, \sigma^2])$ and the $\epsilon_{t,i}$ are all independent, the inputs are uniformly sampled on the $(d-1)$-unit sphere $\mathcal{S}^{d-1}$ and $\theta_t = ru + \theta_0$, $u \sim \mathcal{U}\left(\mathcal{S}^{d-1}\right)$, $\theta_0 \in \mathbb{R}^d$, $r \in \mathbb{R}_+$. We take $d = 20$, $n = 30$, $T = 200$, $\sigma^2 = 0.5$ and θ_0 with all components equal to 5. In this setting, θ_0 is a common bias between the tasks, σ^2 is the inter-task variance and r characterizes the tasks similarity. We experiment with different values of $r \in \{0, 5, 10, 30\}$ to observe the impact of task similarity on the meta-learning process. The smaller r, the closer are the tasks and for the extreme case of $r = 0$ the tasks are identical, in the sense that the parameters θ_t of the tasks are all the same. We draw attention to the fact that a cross-validation procedure to select α (the parameter of OGMS or OPMS, see Equation (5)) or γ is not valid in the online settings, as it would require having knowledge of several tasks in advance for the former and several datapoints in advance for each task for the latter. Moreover, the theoretical values are based on worst-case analysis and lead in practice to slow learning. In practice, to set these values to the correct order of magnitude without adjusting the constants led to better results. So, for mean-OPMS and OPMS we set $\alpha = 1/\sqrt{T}$, for OPMS and I-OGA we set $\gamma = 1/\sqrt{n}$. Instead of cross-validation, one can launch several online learners in parallel with different parameter values to pick the best one (or aggregate them). That is the strategy we use to select Γ for OPMS. Note that the exact value of Γ is usually unkown in practice; its automatic calibration is an important open question. To solve (18), after each task we use the exact solution for mean-OPMS and projected Newton descent with 10

steps for OPMS. We observed that not reaching the exact solution of (18) does not harm the performance of the algorithm and 10 steps are sufficient to reach convergence. The results are displayed in Table 1 and Figure 2. On Figure 2, for each task $t = 1, \dots, T$, we report the average end-of-task loss $MSE_t = \sum_{i=1}^{n} \ell_{t,i}(\theta_{t,n})/n$ averaged over 50 independent runs (with their confidence intervals). Table 1 reports MSE_t averaged over the 100 most recent tasks. The results confirm our theoretical findings: learning γ can bring a substantial benefit over just learning the starting point, which in turn brings a considerable benefit with respect to learning the tasks in isolation. Learning the gradient step makes the meta-learner more robust to task dissimilarities (i.e. when r increases) as shown in Figure 2. In the regime where r is low, learning the gradient step does not help the meta-learner as it takes more steps to reach convergence. Overall both meta learners are consistently better than learning the task in isolation since the number of observation per task is low.

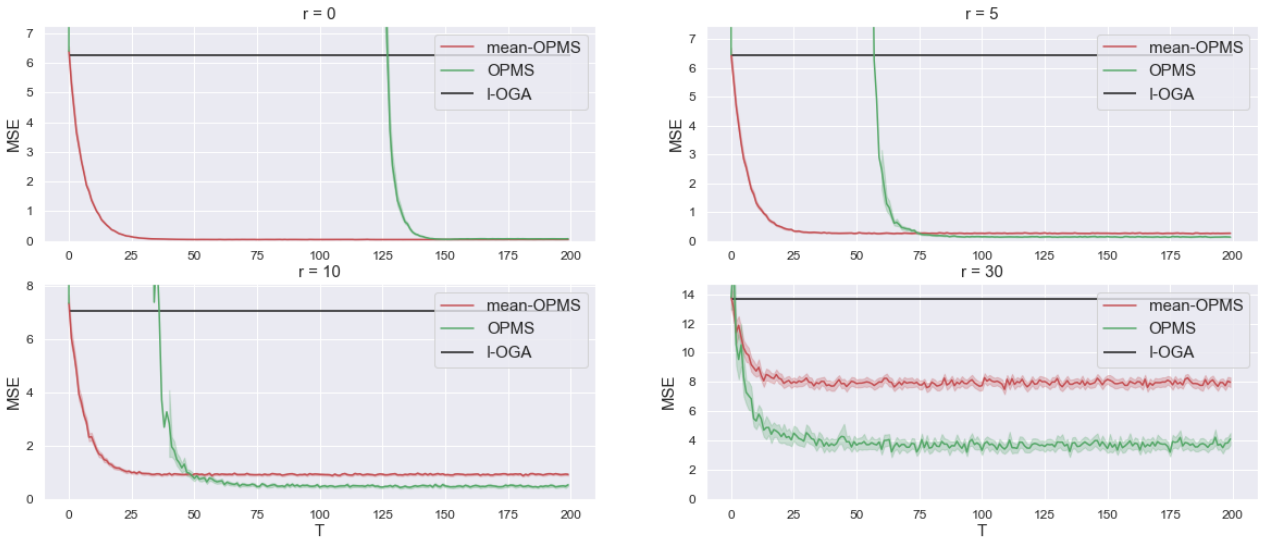

Figure 2. Performance of learning in isolation with OGA (**I-OGA**), OPMS to learn initialization (**mean-OPMS**) and OPMS to learn initialization and step size (**OPMS**). We report the average end-of-task MSE losses at the end of each task, for different values of the task-similarity index $r \in \{0, 5, 10, 30\}$. The results are averaged over 50 independent runs to get confidence intervals.

Table 1. Average end-of-task MSE of the 100 last tasks (averaged over 50 independent runs).

	r = 0	**r = 5**	**r = 10**	**r = 30**
I-OGA	6.24	6.44	7.06	13.60
mean OPMS	0.05	0.27	0.93	7.93
OPMS	0.07	0.15	0.49	3.72

4.4.2. Synthetic Classification

At each round $t = 1, \dots, T$, the meta learner sequentially receives a binary classification task with the Hinge loss that corresponds to a dataset $(x_{t,i}, y_{t,i})_{i=1,\dots,n}$. The binary labels $\{-1, 1\}$ are generated as a logistic model $\mathbb{P}(y = 1) = (1 + \exp(-x^t \theta_t))^{-1}$. The task parameters θ_t and the inputs are generated as in the regression setting. To add some noise, we shuffle 10% of the labels. We take $d = 10$, $n = 100$, $T = 500$, $r = 2$. For mean-OPMS and OPMS we set $\alpha = 1/\sqrt{T}$, for OPMS and I-OGA we set $\gamma = 1/\sqrt{n}$. For the optimisation of F_t (18) with both OPMS and mean-OPMS we use a projected gradient descent with 50 steps.

On Figure 3, for each task $t = 1, \dots, T$, we report the regret on the end-of-task losses: $R(t) = \frac{1}{nt} \sum_{k=1}^{t} \sum_{i=1}^{n} \ell_{k,i}(\theta_{k,n})$, averaged over 10 independent runs (with their confidence intervals). As the for regression setting, the results confirm our theoretical findings: by learning γ (OPMS), we reach a better overall performance than just learning the initialization (mean-OPMS) and a substantially stronger than independent task learning (I-OGA). Note that, in the classification regime, there is no known closed formed expression for the meta-gradient; therefore, OGMS cannot be used.

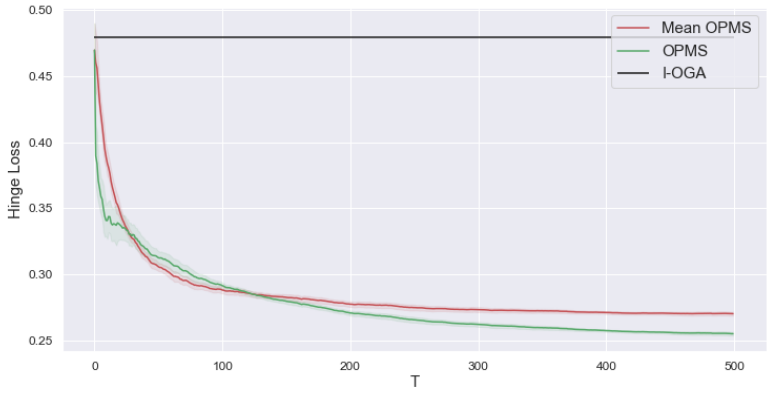

Figure 3. Performance of learning in isolation with OGA (**I-OGA**), OPMS to learn the initialization (**mean-OPMS**) and OPMS to learn the initialization and step size (**OPMS**) on a sequence of classification tasks with the Hinge loss. We report the meta-regret of the Hinge loss. The results are averaged over 10 independent runs (dataset generation) to get confidence intervals.

5. Second Example: Learning the Prior or the Learning Rate in Exponentially Weighted Aggregation

In this section, we will study a generalized Bayesian method, exponentially weighted aggregation. Consider a *finite* set $\Theta_0 = \{\theta_1, \dots, \theta_M\} \subset \mathbb{R}^d$. EWA depends on a prior distribution π on Θ_0, and on a learning rate $\eta > 0$, and returns a decision in $\Theta = \mathrm{conv}(\theta_1, \dots, \theta_M)$ the convex envelope of Θ_0. In this section, we work under the following condition.

Assumption 3. *There is a $B \in \mathbb{R}_+^*$, such that for any $(t,i) \in \{1,\dots,T\} \times \{1,\dots,n\}$, the function $\ell_{t,i}$ is $\Theta \to [0,B]$ and convex.*

We will sometimes use a stronger assumption.

Assumption 4. *There is a $C \in \mathbb{R}_+^*$, such that for any $(t,i) \in \{1,\dots,T\} \times \{1,\dots,n\}$, the function $\theta \mapsto \exp(-\ell_{t,i}(\theta)/C)$ is concave.*

Examples of a situation in which Assumption 4 is satisfied are provided in [3]. Note that Assumption 4 implies Assumption 3.

5.1. Reminder on EWA

The update in EWA is given by:

$$\theta_{t,i} = \sum_{\theta \in \Theta_0} p_{t,i}(\theta)\theta \tag{22}$$

where $p_{t,i}$ are weights defined by

$$p_{t,i}(\theta) = \frac{\exp\left[-\eta \sum_{j=1}^{i-1} \ell_{t,j}(\theta)\right] \pi(\theta)}{\sum_{\vartheta \in \Theta_0} \exp\left[-\eta \sum_{j=1}^{i-1} \ell_{t,j}(\vartheta)\right] \pi(\vartheta)}. \tag{23}$$

The strategy is studied in detail in [3]. We refer the reader to [36] and the references therein for connections to Bayesian inference. We recall the following regret bounds from [3]. First, under Assumption 3,

$$\sum_{i=1}^{n} \ell_{t,i}(\theta_{t,i}) \leq \min_{\theta \in \Theta_0} \left[\sum_{i=1}^{n} \ell_{t,i}(\theta) + \frac{\eta n B^2}{8} + \frac{\log \frac{1}{\pi(\theta)}}{\eta} \right]. \tag{24}$$

Moreover, under the stronger Assumption 4,

$$\sum_{i=1}^{n} \ell_{t,i}(\theta_{t,i}) \leq \min_{\theta \in \Theta_0} \left[\sum_{i=1}^{n} \ell_{t,i}(\theta) + C \log \frac{1}{\pi(\theta)} \right]. \tag{25}$$

In Section 5.2, we work in the general setting (Assumption 3), and we use our meta-strategy OPMS or OGMS to learn η. In Section 5.3, we use OPMS or OGMS to learn π under Assumption 4.

5.2. Learning the Rate η

Consider the uniform prior $\pi(\theta) = 1/M$ for any $\theta \in \Theta_0$. Then, the regret bound (24) becomes:

$$\sum_{i=1}^{n} \ell_{t,i}(\theta_{t,i}) \leq \min_{\theta \in \Theta_0} \sum_{i=1}^{n} \ell_{t,i}(\theta) + \frac{\eta n B^2}{8} + \frac{\log M}{\eta} \tag{26}$$

and it is then possible to optimize it explicitly with respect to η. The value minimizing the bound is $\eta = (2/B)\sqrt{2\log(M)/n}$ and the regret bound becomes:

$$\sum_{i=1}^{n} \ell_{t,i}(\theta_{t,i}) \leq \min_{\theta \in \Theta_0} \sum_{i=1}^{n} \ell_{t,i}(\theta) + B\sqrt{\frac{n \log M}{2}}. \tag{27}$$

In practice, however, while it is often reasonable to assume that the loss function is bounded (as in Assumption 3), very often one does not know a tight upper bound. Thus, one may use a constant B that satisfies Assumption 3, but that is far too large. Even though one does not know a better upper bound than B, one would like a regret bound that depends on the tightest possible upper bound.

In the meta-learning framework, define:

$$\mathcal{L}_t(\eta) = \min_{\theta \in \Theta_0} \sum_{i=1}^{n} \ell_{t,i}(\theta) + \frac{\eta n \left[\max_{\vartheta \in \Theta_0, 1 \leq i \leq n} \ell_{t,i}(\vartheta)\right]^2}{8} + \frac{\log M}{\eta} \tag{28}$$

for $\eta \in \Lambda = [1/n, 1]$. It is immediately necessary to prove that this function is convex and L-Lipschitz with $L = n^2 \log(M) + nB^2/8$. So, Assumption 1 is satisfied, allowing for the use of the OPMS or OGMS strategy without needed a tight upper bound on the losses. Note that, in this context, the OGMS strategy is given by:

$$\eta_t = \frac{1}{n} \vee \left[\eta_{t-1} - \alpha \left(\frac{n \left[\max_{\theta \in \Theta_0, 1 \leq i \leq n} \ell_{t,i}(\theta)\right]^2}{8} - \frac{\log M}{\eta_{t-1}^2} \right) \right] \wedge 1.$$

Theorem 2. *Under Assumption 3, using OGMS or OPMS on $\mathcal{L}_t(\eta)$, as in (28) with $\eta_1 = 1$, $L = n^2 \log(M) + nB^2/8$ and*

$$\alpha = \frac{1}{L}\sqrt{\frac{2}{T}} \tag{29}$$

we have

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\eta_t}) \leq \sum_{t=1}^{T}\min_{\theta\in\Theta_0}\sum_{i=1}^{n}\ell_{t,i}(\theta) + bT\sqrt{\frac{n\log(M)}{2}} + T\log(M) + \frac{b^2T}{8} + \left(n^2\log M + \frac{nB^2}{8}\right)\sqrt{2T} \quad (30)$$

where

$$b = \max_{\theta\in\Theta_0, 1\leq t\leq T, 1\leq i\leq n}|\ell_{t,i}(\theta)|. \quad (31)$$

Let us compare learning in isolation with meta-learning in this context. When learning in isolation, the hyperparameter η is fixed (as in (2)). If we fix it to the value $\eta_0 = (2/B)\sqrt{2\log(M)/n}$ as in (27), the meta-regret is in $BT\sqrt{n\log(M)/2}$. On the other hand, meta-learning leads to a meta-regret in $bT\sqrt{n\log(M)/2} + n^2\log M\sqrt{2T} + \mathcal{O}(nB^2\sqrt{T}+T)$. In other words, we replace the potentially loose upper bound B by the tightest possible bound b, at the cost of an additional $n^2\log M\sqrt{2T} + \mathcal{O}(nB^2\sqrt{T}+T)$ term. Here again, when T is large enough with respect to n, this term is negligible.

5.3. Learning the Prior π

Under Assumption 4, we have the regret bound in (25). Without any information on Θ_0, it seems natural to use the uniform prior π on $\Theta_0 = \{\theta_1, \ldots, \theta_M\}$, which leads to

$$\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}) \leq \min_{\theta\in\Theta_0}\sum_{i=1}^{n}\ell_{t,i}(\theta) + C\log M. \quad (32)$$

If some additional information was available, such as, for example: "the best θ is always either θ_1 or θ_2", one would rather chose the uniform prior on $\{\theta_1, \theta_2\}$, and obtain the bound:

$$\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}) \leq \min_{\theta\in\Theta_0}\sum_{i=1}^{n}\ell_{t,i}(\theta) + C\log 2. \quad (33)$$

Unfortunately, such information is generally not available. However, in the context of meta-learning, we can take advantage of the previous tasks to learn such information.

Thus, let us define, for any task t,

$$\theta_t^* = \operatorname*{argmin}_{\theta\in\Theta_0}\sum_{i=1}^{n}\ell_{t,i}(\theta) \quad (34)$$

and

$$\mathcal{L}_t(\pi) = \sum_{i=1}^{n}\ell_{t,i}(\theta_t^*) + C\log\frac{1}{\pi(\theta_t^*)} \quad (35)$$

for $\pi = (\pi(\theta_1), \ldots, \pi(\theta_M)) \in \Lambda$ with

$$\Lambda = \left\{x \in (\mathbb{R}_+)^M \colon \sum_{h=1}^{M} x_h = 1 \text{ and } x_h \geq \frac{1}{2M}\right\}. \quad (36)$$

It is important to check that $\mathcal{L}_t$ is convex and L-Lipschitz with $L = 2CM$ on Λ; this allows us to use OPMS (or OGMS).

Theorem 3. *Under Assumption 4, using OPMS on $\mathcal{L}_t(\pi)$ as in (35) with $\pi_1 = (1/M, \ldots, 1/M)$, $L = 2CM$ and*

$$\alpha = \frac{1}{2CM\sqrt{T}}, \quad (37)$$

define $I^* = \{\theta_1^*, \dots, \theta_T^*\}$ *where each* θ_t^* *is as in* (34) *and* $m^* = \mathrm{card}(I^*)$. *We have*

$$\sum_{t=1}^{T}\sum_{i=1}^{n} \ell_{t,i}(\theta_{t,i}^{\pi_t}) \leq \sum_{t=1}^{T}\sum_{i=1}^{n} \ell_{t,i}(\theta_t^*) + CT\log(2m^*) + 2CM\sqrt{T}. \tag{38}$$

When learning in isolation with a uniform prior, the meta-regret is $TC\log(M)$. On the other hand, if m^* is small (that is, many of the θ_i^*s are similar), meta-learning leads to a meta-regret in $CT\log(2m^*) + 2CM\sqrt{T}$. For a T that is large enough, this is an important improvement.

5.4. Discussion on the Continuous Case

Let us now discuss the possibility of meta-learning for generalized Bayesian methods when Θ_0 is no longer a finite set. There is a general formula for EWA, given by

$$\rho_{t,i}(\mathrm{d}\theta) = \underset{\rho}{\mathrm{argmin}}\left\{ \mathbb{E}_{\theta\sim\rho}\left[\sum_{j=1}^{i-1}\ell_{t,j}(\theta)\right] + \frac{\mathcal{K}(\rho,\pi)}{\eta}\right\} \tag{39}$$

where the minimum is taken over for all probability distributions that are absolutely continuous with π, and where π is a prior distribution, $\eta > 0$ a learning rate and $\mathcal{K}$ is the Kullback–Leibler divergence (KL). Meta-learning for such an update rule is proven in [10,37] but usually does not lead to feasible strategies. Online variational inference [38,39] consists in replacing the minimization on the set of all probability distributions by minimization in a smaller set in order to define a feasible approximation of $\rho_{t,i}$. For example, let $(q_\mu)_{\mu\in M}$ be a parametric family of probability distributions, Thus, we define:

$$\mu_{t,i} = \underset{\mu\in M}{\mathrm{argmin}}\left\{ \mathbb{E}_{\theta\sim q_\mu}\left[\sum_{j=1}^{i-1}\ell_{t,j}(\theta)\right] + \frac{\mathcal{K}(q_\mu,\pi)}{\eta}\right\}. \tag{40}$$

It is discussed in [40] that, generally, when μ is a location-scale parameter and $\ell_{t,j}$ is Γ-Lipschitz and convex, then $\bar{\ell}_{t,i}(\mu) := \mathbb{E}_{\theta\sim q_\mu}[\ell_{t,j}(\theta)]$ is 2Γ-Lipschitz and convex. In this case, under the assumption that $\mathcal{K}(q_\mu,\pi)$ is α-strongly convex in μ, a regret bound for such strategies was derived in [39]:

$$\sum_{i=1}^{n}\mathbb{E}_{\theta\sim q_{\mu_{t,i}}}[\ell_{t,i}(\theta)] \leq \inf_{\mu\in\mathcal{M}}\left\{ \mathbb{E}_{\theta\sim q_\mu}\left[\sum_{i=1}^{n}\ell_{t,i}(\theta)\right] + \frac{\eta 4\Gamma^2 n}{\alpha} + \frac{\mathcal{K}(q_\mu,\pi)}{\eta}\right\}. \tag{41}$$

A complete study of meta-learning of the rate $\eta > 0$ and of the prior π in this context is an important objective (possibly, with a restriction that $\pi \in \{q_\mu, \mu \in M\}$). However, this raises many problems. For example, the KL divergence $\mathcal{K}(q_\mu, q_{\mu'})$ is not always convex with respect to the parameter μ'. In this case, it might help to replace it by a convex relaxation that would allow for the use of OGMS or OPMS. This relates to [41,42], who advocate going beyond the KL divergence in (39); see also [36] and the references therein. This will be the object of future works.

6. Proofs

We start with a preliminary lemma that will be used in the proof of Proposition 1.

Lemma 1. *Let* a, b, c *be three vectors in* $\mathbb{R}^p$. *Then:*

$$(a-b)^T(b-c) = \frac{\|a-c\|^2 - \|a-b\|^2 - \|b-c\|^2}{2}. \tag{42}$$

Proof. expand $\|a-c\|^2 = \|a\|^2 + \|c\|^2 - 2a^Tc$ in the r.h.s, as well as $\|a-b\|^2$ and $\|b-c\|^2$. Then simplify. □

We now prove Proposition 1 separately for the general OGMS strategy, and then for OGMS.

Proof of Proposition 1 for OPMS. As mentioned earlier, this strategy is an application to the meta-learning setting of implicit online learning [32,33]. We follow here a proof from Chapter 11 in [3]. We refer the reader to [43] and the references therein for tighter bounds under stronger assumptions.

First, λ_t is defined as the minimizer of a convex function in (5). So, the subdifferential of this function at λ_t contains 0. In other words, there is a $z_t \in \partial\mathcal{L}_{t-1}(\lambda_t)$, such that

$$z_t = \frac{\lambda_{t-1} - \lambda_t}{\alpha}. \tag{43}$$

By convexity, for any λ, for any $z \in \partial\mathcal{L}_{t-1}(\lambda_t)$,

$$\mathcal{L}_{t-1}(\lambda) \geq \mathcal{L}_{t-1}(\lambda_t) + (\lambda - \lambda_t)^T z. \tag{44}$$

The choice $z = z_t$ gives:

$$\mathcal{L}_{t-1}(\lambda) \geq \mathcal{L}_{t-1}(\lambda_t) + \frac{(\lambda - \lambda_t)^T(\lambda_{t-1} - \lambda_t)}{\alpha}, \tag{45}$$

that is,

$$\begin{aligned}
\mathcal{L}_{t-1}(\lambda_t) &\leq \mathcal{L}_{t-1}(\lambda) + \frac{(\lambda - \lambda_t)^T(\lambda_t - \lambda_{t-1})}{\alpha} \\
&= \mathcal{L}_{t-1}(\lambda) + \frac{\|\lambda - \lambda_{t-1}\|^2 - \|\lambda - \lambda_t\|^2}{2\alpha} - \frac{\|\lambda_t - \lambda_{t-1}\|^2}{2\alpha} \\
&= \mathcal{L}_{t-1}(\lambda) + \frac{\|\lambda - \lambda_{t-1}\|^2 - \|\lambda - \lambda_t\|^2}{2\alpha} - \alpha\frac{\|z_t\|^2}{2}
\end{aligned} \tag{46}$$

where we used Lemma 1. Then, note that

$$\begin{aligned}
\mathcal{L}_{t-1}(\lambda_{t-1}) &= \mathcal{L}_{t-1}(\lambda_t) + [\mathcal{L}_{t-1}(\lambda_{t-1}) - \mathcal{L}_{t-1}(\lambda_t)] \\
&\leq \mathcal{L}_{t-1}(\lambda_t) + \|\lambda_{t-1} - \lambda_t\|L \\
&\leq \mathcal{L}_{t-1}(\lambda_t) + \alpha\|z_t\|L.
\end{aligned} \tag{47}$$

Combining this inequality with (46) gives

$$\mathcal{L}_{t-1}(\lambda_{t-1}) \leq \mathcal{L}_{t-1}(\lambda) + \frac{\|\lambda - \lambda_{t-1}\|^2 - \|\lambda - \lambda_t\|^2}{2\alpha} + \alpha\left(\|z_t\|L - \frac{\|z_t\|^2}{2}\right). \tag{48}$$

Now, for any $x \in \mathbb{R}$, $-x^2/2 + xL - L^2/2 \leq 0$. In particular, $\|z_t\|L - \|z_t\|^2/2 \leq L^2/2$ and so the above can be rewritten:

$$\mathcal{L}_{t-1}(\lambda_{t-1}) \leq \mathcal{L}_{t-1}(\lambda) + \frac{\|\lambda - \lambda_{t-1}\|^2 - \|\lambda - \lambda_t\|^2}{2\alpha} + \frac{\alpha L^2}{2}. \tag{49}$$

Summing the inequality for $t = 2$ to $T + 1$ leads to:

$$\sum_{t=1}^{T} \mathcal{L}_t(\lambda_t) \leq \sum_{t=1}^{T} \mathcal{L}_t(\lambda) + \frac{\|\lambda - \lambda_1\|^2 - \|\lambda - \lambda_{T+1}\|^2}{2\alpha} + \frac{\alpha T L^2}{2}. \tag{50}$$

This ends the proof. □

Proof of Proposition 1 for OGMS. The beginning of the proof follows the proof of Theorem 11.1 in [3].

Note that we can rewrite (4) as

$$\begin{cases} \tilde{\lambda}_t = \lambda_{t-1} - \alpha \nabla \mathcal{L}_{t-1}(\lambda_{t-1}) \\ \lambda_t = \Pi_\Lambda(\tilde{\lambda}_t) \end{cases}$$

rearranging the first line, we obtain:

$$\nabla \mathcal{L}_{t-1}(\lambda_{t-1}) = \frac{\lambda_{t-1} - \tilde{\lambda}_t}{\alpha}. \tag{51}$$

By convexity, for any λ,

$$\mathcal{L}_{t-1}(\lambda) \geq \mathcal{L}_{t-1}(\lambda_{t-1}) + (\lambda - \lambda_{t-1})^T \nabla \mathcal{L}_{t-1}(\lambda_{t-1}) \tag{52}$$

$$= \mathcal{L}_{t-1}(\lambda_{t-1}) + \frac{(\lambda - \lambda_{t-1})^T (\lambda_{t-1} - \tilde{\lambda}_t)}{\alpha}, \tag{53}$$

that is,

$$\mathcal{L}_{t-1}(\lambda_{t-1}) \leq \mathcal{L}_{t-1}(\lambda) - \frac{(\lambda - \lambda_{t-1})^T (\lambda_{t-1} - \tilde{\lambda}_t)}{\alpha}. \tag{54}$$

Lemma 1 gives:

$$(\lambda - \lambda_{t-1})^T (\lambda_{t-1} - \tilde{\lambda}_t) = \frac{\|\lambda - \tilde{\lambda}_t\|^2 - \|\lambda - \lambda_{t-1}\|^2 - \|\lambda_{t-1} - \tilde{\lambda}_t\|^2}{2}$$

$$= \frac{\|\lambda - \tilde{\lambda}_t\|^2 - \|\lambda - \lambda_{t-1}\|^2 - \alpha^2 \|\nabla \mathcal{L}_{t-1}(\lambda_{t-1})\|^2}{2} \tag{55}$$

$$\geq \frac{\|\lambda - \lambda_t\|^2 - \|\lambda - \lambda_{t-1}\|^2 - \alpha^2 \|\nabla \mathcal{L}_{t-1}(\lambda_{t-1})\|^2}{2}, \tag{56}$$

the last step being justified by:

$$\|\lambda - \tilde{\lambda}_t\|^2 \geq \|\lambda - \Pi_\Lambda(\tilde{\lambda}_t)\|^2 = \|\lambda - \lambda_t\|^2 \tag{57}$$

for any $\lambda \in \Lambda$. Plug (56) in (54) to get:

$$\mathcal{L}_{t-1}(\lambda_{t-1}) \leq \mathcal{L}_{t-1}(\lambda) + \frac{\|\lambda - \lambda_{t-1}\|^2 - \|\lambda - \lambda_t\|^2}{2\alpha} + \frac{\alpha \|\nabla \mathcal{L}_{t-1}(\lambda_{t-1})\|^2}{2} \tag{58}$$

and the Lipschitz assumption gives:

$$\mathcal{L}_{t-1}(\lambda_{t-1}) \leq \mathcal{L}_{t-1}(\lambda) + \frac{\|\lambda - \lambda_{t-1}\|^2 - \|\lambda - \lambda_t\|^2}{2\alpha} + \frac{\alpha L^2}{2} \tag{59}$$

sum the inequality for $t = 2$ to $T + 1$ to get:

$$\sum_{t=1}^{T} \mathcal{L}_t(\lambda_t) \leq \sum_{t=1}^{T} \mathcal{L}_t(\lambda) + \frac{\|\lambda - \lambda_1\|^2 - \|\lambda - \lambda_{T+1}\|^2}{2\alpha} + \frac{\alpha T L^2}{2}. \tag{60}$$

This ends the proof of the statement for OGMS. □

We now provide a lemma that will be useful for the proof of Proposition 2.

Lemma 2. *Let $G(u, v)$ be a convex function of $(u, v) \in U \times V$. Define $g(u) = \inf_{v \in V} G(u, v)$. Then g is convex.*

Proof. indeed, let $\lambda \in [0,1]$ and $(x,y) \in U^2$,

$$g(\lambda x + (1-\lambda)y) = \inf_{v \in V} G(\lambda x + (1-\lambda)y, v) \tag{61}$$
$$\leq G(\lambda x + (1-\lambda)y, \lambda x' + (1-\lambda)y') \tag{62}$$
$$\leq \lambda G(x,x') + (1-\lambda)G(y,y') \tag{63}$$

where the last two inequalities hold for any $(x',y') \in V^2$. Let us now take the infimum with respect to $(x',y') \in V^2$ in both sides, this gives:

$$g(\lambda x + (1-\lambda)y) \leq \inf_{x' \in V} \lambda G(x,x') + \inf_{y' \in V} (1-\lambda)G(y,y') \tag{64}$$
$$= \lambda g(x) + (1-\lambda)g(y), \tag{65}$$

that is, g is convex. □

Proof of Proposition 2. Apply Lemma 2 to $u = (\vartheta, \gamma)$, $v = \theta$, $U = \Lambda$, $V = \Theta$ and

$$G(u,v) = \sum_{i=1}^{n} \ell_{i,t}(\theta) + \frac{\gamma \Gamma^2 n}{2} + \frac{\|\vartheta - \theta\|^2}{2\gamma}. \tag{66}$$

This shows $g(u) = \mathcal{L}_t((\vartheta,\gamma))$ is convex with respect (ϑ,γ). Additionally, G is differentiable w.r.t $u = (\vartheta,\gamma)$, so

$$\frac{\partial G}{\partial \vartheta} = \frac{\vartheta - \theta}{\gamma}, \text{ and } \frac{\partial G}{\partial \gamma} = \frac{n\Gamma^2}{2} - \frac{\|\vartheta - \theta\|^2}{2\gamma^2}. \tag{67}$$

As a consequence, for $(\theta,\vartheta) \in \Theta^2$ and $\underline{\gamma} \leq \gamma \leq \overline{\gamma}$,

$$\left\| \frac{\partial G}{\partial \vartheta} \right\|^2 \leq \frac{4C^2}{\underline{\gamma}^2}, \text{ and } \left| \frac{\partial G}{\partial \gamma} \right|^2 \leq \frac{n^2\Gamma^4}{4} + \frac{4C^4}{\underline{\gamma}^4}. \tag{68}$$

This leads to

$$\|\nabla_u G(u,v)\| = \sqrt{\left\| \frac{\partial G}{\partial \vartheta} \right\|^2 + \left| \frac{\partial G}{\partial \gamma} \right|^2} \tag{69}$$
$$\leq \sqrt{\frac{n^2\Gamma^4}{4} + \frac{4C^2}{\underline{\gamma}^2} + \frac{4C^4}{\underline{\gamma}^4}} =: L, \tag{70}$$

that is, for each v, $G(u,v)$ is L-Lipschitz in u. So, $g(u) = \inf_{v \in V} G(u,v)$ is L-Lipschitz in u. □

Proof of Theorem 1. Thanks to the Assumption 2, we can apply Proposition 2. That is, Assumption (1) is satisfied, and we can apply Proposition 1. This gives:

$$\sum_{t=1}^{T} \sum_{i=1}^{n} \ell_{t,i}(\theta_{t,i}^{\lambda_t}) \leq \inf_{\theta_1,\ldots,\theta_T \in \Theta} \inf_{(\vartheta,\gamma) \in \Lambda} \left\{ \sum_{t=1}^{T} \left[\sum_{i=1}^{n} \ell_{t,i}(\theta_t) + \frac{\gamma\Gamma^2 n}{2} + \frac{\|\theta_t - \vartheta\|^2}{2\gamma} \right] + \frac{\alpha T L^2}{2} + \frac{\|\vartheta - \vartheta_1\|^2 + |\gamma - \gamma_1|^2}{2\alpha} \right\}. \tag{71}$$

We use direct bounds for the last two terms: $\|\vartheta - \vartheta_1\|^2 \leq 4C^2$ and $|\gamma - \gamma_1|^2 \leq |\overline{\gamma} - \underline{\gamma}|^2 \leq \overline{\gamma}^2 = C^4$. Then note that

$$\sum_{t=1}^{T} \|\theta_t - \vartheta\|^2 = T\left\|\vartheta - \frac{1}{T}\sum_{s=1}^{T}\theta_s\right\|^2 + \sum_{t=1}^{T}\left\|\theta_t - \frac{1}{T}\sum_{s=1}^{T}\theta_s\right\|^2 \tag{72}$$

$$= T\left\|\vartheta - \frac{1}{T}\sum_{s=1}^{T}\theta_s\right\|^2 + T\sigma^2(\theta_1^T). \tag{73}$$

Upper bounding the infimum on ϑ in (71) by $\vartheta = \frac{1}{T}\sum_{s=1}^{T}\theta_s$ leads to

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\lambda_t}) \leq \inf_{\theta_1,\ldots,\theta_T\in\Theta}\inf_{\gamma\in[\underline{\gamma},\overline{\gamma}]}\left\{\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_t) + \frac{\gamma\Gamma^2 nT}{2} + \frac{T\sigma^2(\theta_1^T)}{2\gamma} + \frac{\alpha TL^2}{2} + \frac{C^2(4+C^2)}{2\alpha}\right\}. \tag{74}$$

The right-hand side of (74) is minimized with respect to α if $\alpha = \frac{C}{L}\sqrt{\frac{4+C^2}{T}}$, which is the value proposed in the theorem, and we obtain:

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\lambda_t}) \leq \inf_{\theta_1,\ldots,\theta_T\in\Theta}\inf_{\gamma\in[\underline{\gamma},\overline{\gamma}]}\left\{\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_t) + \frac{\gamma\Gamma^2 nT}{2} + \frac{T\sigma^2(\theta_1^T)}{2\gamma} + CL\sqrt{(4+C^2)T}\right\}. \tag{75}$$

The infimum with respect to γ in the r.h.s is reached for

$$\gamma^* = \left(\underline{\gamma} \vee \frac{\sigma(\theta_1^T)}{\Gamma\sqrt{n}}\right) \wedge \overline{\gamma}. \tag{76}$$

First, note that

$$\frac{\gamma^*\Gamma^2 nT}{2} \leq \left(\underline{\gamma} \vee \frac{\sigma(\theta_1^T)}{\Gamma\sqrt{n}}\right)\frac{\Gamma^2 nT}{2} \tag{77}$$

$$\leq \left(\underline{\gamma} + \frac{\sigma(\theta_1^T)}{\Gamma\sqrt{n}}\right)\frac{\Gamma^2 nT}{2} \tag{78}$$

$$= \frac{\Gamma^2 Tn^{1-\beta}}{2} + \frac{\sigma(\theta_1^T)\Gamma T\sqrt{n}}{2}, \tag{79}$$

using $\underline{\gamma} = n^{-\beta}$. Then,

$$\frac{T\sigma^2(\theta_1^T)}{2\gamma^*} \leq \frac{T\sigma^2(\theta_1^T)}{2}\left(\frac{1}{\overline{\gamma}} \vee \frac{\Gamma\sqrt{n}}{\sigma(\theta_1^T)}\right) \tag{80}$$

$$\leq \frac{T\sigma^2(\theta_1^T)}{2}\left(\frac{1}{\overline{\gamma}} + \frac{\Gamma\sqrt{n}}{\sigma(\theta_1^T)}\right) \tag{81}$$

$$= \frac{T\sigma^2(\theta_1^T)}{2C^2} + \frac{\sigma(\theta_1^T)\Gamma T\sqrt{n}}{2} \tag{82}$$

$$\leq \frac{T\sigma(\theta_1^T)}{C} + \frac{\sigma(\theta_1^T)\Gamma T\sqrt{n}}{2}, \tag{83}$$

using $\overline{\gamma} = C^2$ and $\sigma(\theta_1^T) \leq 2C$. Plugging (77), (80) and the definition of L into (75) gives

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\lambda_t}) \leq \inf_{\theta_1,\dots,\theta_T\in\Theta}\left\{\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_t) + C\sqrt{\left(\frac{n^2\Gamma^4}{4}+4C^2n^{2\beta}+4C^4n^{4\beta}\right)(4+C^2)T}\right. \tag{84}$$

$$\left. + \frac{\Gamma^2 T n^{1-\beta}}{2} + \sigma(\theta_1^T)T\left(\Gamma\sqrt{n}+\frac{1}{C}\right)\right\} \tag{85}$$

$$= \inf_{\theta_1,\dots,\theta_T\in\Theta}\left\{\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_t) + C\sqrt{(4+C^2)\left(\frac{n^2\Gamma^4}{4n^{2\vee 4\beta}}+\frac{4C^2n^{2\beta}}{n^{2\vee 4\beta}}+\frac{4C^4n^{4\beta}}{n^{2\vee 4\beta}}\right)}n^{1\vee 2\beta}\sqrt{T}\right. \tag{86}$$

$$\left. + \left[\frac{\Gamma^2}{2}n^{1-\beta}+\left(\Gamma+\frac{1}{nC}\right)\sigma(\theta_1^T)\sqrt{n}\right]T\right\} \tag{87}$$

$$\leq \inf_{\theta_1,\dots,\theta_T\in\Theta}\left\{\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_t) + C\sqrt{(4+C^2)\left(\frac{\Gamma^2}{4}+4C^2+4C^4\right)}n^{1\vee 2\beta}\sqrt{T}\right. \tag{88}$$

$$\left. + \left[\frac{\Gamma^2}{2}n^{1-\beta}+\left(\Gamma+\frac{1}{C}\right)\sigma(\theta_1^T)\sqrt{n}\right]T\right\} \tag{89}$$

$$\leq \inf_{\theta_1,\dots,\theta_T\in\Theta}\left\{\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_t) + \mathcal{C}(\Gamma,C)\left[n^{1\vee 2\beta}\sqrt{T}+\left(n^{1-\beta}+\sigma(\theta_1^T)\sqrt{n}\right)T\right]\right. \tag{90}$$

where we took

$$\mathcal{C}(\Gamma,C) = \max\left(C\sqrt{(4+C^2)\left(\frac{\Gamma^2}{4}+4C^2+4C^4\right)}, \frac{\Gamma^2}{2}, \Gamma+\frac{1}{C}\right). \tag{91}$$

This ends the proof. □

Proof of Theorem 2. A direct application of Proposition 1 gives

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\eta_t}) \leq \inf_{\eta\geq\frac{1}{n}}\left\{\sum_{t=1}^{T}\min_{\theta\in\Theta_0}\left[\sum_{i=1}^{n}\ell_{t,i}(\theta)\right.\right.$$

$$\left.\left. + \frac{\eta n\left[\max_{\vartheta\in\Theta_0,1\leq i\leq n}\ell_{t,i}(\vartheta)\right]^2}{8}+\frac{\log M}{\eta}\right] + \frac{\alpha T L^2}{2}+\frac{(\eta-1)^2}{2\alpha}\right\}. \tag{92}$$

Thus, we have

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\eta_t}) \leq \inf_{\eta\geq\frac{1}{n}}\left\{\sum_{t=1}^{T}\min_{\theta\in\Theta_0}\left[\sum_{i=1}^{n}\ell_{t,i}(\theta)+\frac{\eta n b^2}{8}+\frac{\log M}{\eta}\right] + \frac{\alpha T L^2}{2}+\frac{(\eta-1)^2}{2\alpha}\right\}. \tag{93}$$

Now, plugging in the right-hand side

$$\eta = \frac{1}{n}\vee\left(\frac{2}{b}\sqrt{\frac{2\log M}{n}}\right)\wedge 1, \tag{94}$$

we obtain:

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\eta_t}) \leq \sum_{t=1}^{T}\min_{\theta\in\Theta_0}\left[\sum_{i=1}^{n}\ell_{t,i}(\theta)+\frac{b^2}{8}+b\sqrt{\frac{n\log(M)}{2}}+\log(M)\right] + \frac{\alpha T L^2}{2}+\frac{1}{2\alpha}. \tag{95}$$

Now, we see that the value $\alpha = \sqrt{2/(TL^2)}$ leads to:

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\eta_t}) \leq \sum_{t=1}^{T}\min_{\theta\in\Theta_0}\left[\sum_{i=1}^{n}\ell_{t,i}(\theta) + \frac{b^2}{8} + b\sqrt{\frac{n\log(M)}{2}} + \log(M)\right] + L\sqrt{2T}. \quad (96)$$

Rearranging terms, and replacing L by its value,

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\eta_t}) \leq \sum_{t=1}^{T}\min_{\theta\in\Theta_0}\sum_{i=1}^{n}\ell_{t,i}(\theta) + bT\sqrt{\frac{n\log(M)}{2}} + \frac{b^2T}{8} + T\log(M) + \left(n^2\log M + \frac{nB^2}{8}\right)\sqrt{2T}, \quad (97)$$

that is the statement of the theorem. □

Proof of Theorem 3. An application of Proposition 1 leads to

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\pi_t}) \leq \min_{\pi\in\Lambda}\left\{\sum_{t=1}^{T}\left[\sum_{i=1}^{n}\ell_{t,i}(\theta_t^*) + C\log\frac{1}{\pi(\theta_t^*)}\right] + \frac{\alpha TL^2}{2} + \frac{\|\pi-\pi_1\|^2}{2\alpha}\right\} \quad (98)$$

and so

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\pi_t}) \leq \min_{\pi\in\Lambda}\left\{\sum_{t=1}^{T}\left[\sum_{i=1}^{n}\ell_{t,i}(\theta_t^*) + C\log\frac{1}{\pi(\theta_t^*)}\right] + \frac{\alpha TL^2}{2} + \frac{1}{2\alpha}\right\} \quad (99)$$

define π_{I^*} such that $\pi_{I^*}(\theta_j) = 1/(2m^*)$ if $j \in I^*$ and $\pi_{I^*}(\theta_j) = 1/(2(M-m^*))$ otherwise. We have $\pi_I^* \in \Lambda$ and thus

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\ell_{t,i}(\theta_{t,i}^{\pi_t}) \leq \sum_{t=1}^{T}\left[\sum_{i=1}^{n}\ell_{t,i}(\theta_t^*) + C\log(2m^*)\right] + \frac{\alpha TL^2}{2} + \frac{1}{2\alpha}. \quad (100)$$

Replace L and α by their values to get the theorem. □

7. Conclusions

We proposed two simple meta-learning strategies together with their theoretical analysis. Our results clearly show an improvement on learning in isolation if the tasks are similar enough. These theoretical findings are confirmed by our numerical experiments. Important questions remain open. In [27], a purely online method is proposed, in the sense that it does not require storing all of the information of the current task. In the case of OGA, this method allows us to learn the starting point. However, its application to learn the step size is not direct [28]. An important question is, then: is there a purely online method that would provably improve on learning in isolation in this case? Another important question is the automatic calibration of Γ. However, as mentioned in Section 5, we believe that a very general and efficient meta-learning method for learning priors in Bayesian statistics (or in generalized Bayesian inference) would be extremely valuable in practice.

Author Contributions: Investigation, D.M. and P.A.; Software, D.M.; Writing—original draft, D.M. and P.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: This project was initiated as Dimitri Meunier's internship project at RIKEN AIP, in the Approximate Bayesian Inference team. The internship was cancelled because of the pandemic. We would like to thank Arnak Dalalyan (ENSAE Paris), who provided fundings so that the internship could take place at ENSAE Paris instead. We would like to thank Emtiyaz Khan (RIKEN AIP), Sébastien Gerchinovitz (IRT Saint-Exupéry, Toulouse), Vianney Perchet (ENSAE Paris) and all the members of the Approximate Bayesian Inference team for valuable feedback.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Thrun, S.; Pratt, L. *Learning to Learn*; Kluwer Academic Publishers: New York, NY, UK, 1998.
2. Chollet, F. On the measure of intelligence. *arXiv* **2019**, arXiv:1911.01547.
3. Cesa-Bianchi, N.; Lugosi, G. *Prediction, Learning, and Games*; Cambridge University Press: Cambridge, UK, 2006.
4. Hazan, E. Introduction to online convex optimization. *arXiv* **2019**, arXiv:1909.05207.
5. Orabona, F. A modern introduction to online learning. *arXiv* **2019**, arXiv:1912.13213.
6. Shalev-Shwartz, S. Online learning and online convex optimization. *Found. Trends Mach. Le.* **2012**, *4*, 107–194. [CrossRef]
7. Maurer, A. Bounds for linear multi-task learning. *J. Mach. Learn. Res.* **2006**, *7*, 117–139.
8. Romera-Paredes, B.; Aung, H.; Bianchi-Berthouze, N.; Pontil, M. Multilinear multitask learning. In Proceedings of the International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; pp. 1444–1452.
9. Yamada, M.; Koh, T.; Iwata, T.; Shawe-Taylor, J.; Kaski, S. Localized lasso for high-dimensional regression. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, PMLR, Fort Lauderdale, FL, USA, 20–22 April 2017; Volume 54, pp. 325–333.
10. Alquier, P.; Mai, T.T.; Pontil, M. Regret Bounds for Lifelong Learning. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017; Volume 54, pp. 261–269.
11. Amit, R.; Meir, R. Meta-learning by adjusting priors based on extended PAC-Bayes theory. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 205–214.
12. Baxter, J. Theoretical models of learning to learn. In *Learning to Learn*; Springer: Berlin, Germany, 1998; pp. 71–94.
13. Jose, S.T.; Simeone, O.; Durisi, G. Transfer meta-learning: Information-theoretic bounds and information meta-risk minimization. *arXiv* **2020**, arXiv:2011.02872.
14. Maurer, A.; Pontil, M.; Romera-Paredes, B. The benefit of multitask representation learning. *J. Mach. Learn. Res.* **2016**, *17*, 1–32.
15. Pentina, A.; Lampert, C. A PAC-Bayesian bound for lifelong learning. In Proceedings of the 31st International Conference on Machine Learning, Bejing, China, 22–24 June 2014; pp. 991–999.
16. Rothfuss, J.; Fortuin, V.; Krause, A. Pacoh: Bayes-optimal meta-learning with pac-guarantees. *arXiv* **2020**, arXiv:2002.05551.
17. Andrychowicz, M.; Denil, M.; Gomez, S.; Hoffman, M.W.; Pfau, D.; Schaul, T.; Shillingford, B.; De Freitas, N. Learning to learn by gradient descent by gradient descent. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 3981–3989.
18. Ruvolo, P.; Eaton, E. Ella: An efficient lifelong learning algorithm. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 17–19 June 2013; pp. 507–515.
19. Balcan, M.-F.; Khodak, M.; Talwalkar, A. Provable guarantees for gradient-based meta-learning. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 424–433.
20. Denevi, G.; Ciliberto, C.; Stamos, D.; Pontil, M. Learning to learn around a common mean. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; pp. 10169–10179.
21. Denevi, G.; Ciliberto, C.; Grazzi, R.; Pontil, M. Learning-to-learn stochastic gradient descent with biased regularization. *arXiv* **2019**, arXiv:1903.10399.
22. Denevi, G.; Pontil, M.; Ciliberto, C. The advantage of conditional meta-learning for biased regularization and fine tuning. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–12 December 2020; Volume 33.
23. Fallah, A.; Mokhtari, A.; Ozdaglar, A. On the convergence theory of gradient-based model-agnostic meta-learning algorithms. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Online, 26–28 August 2020; pp. 1082–1092.
24. Finn, C.; Rajeswaran, A.; Kakade, S.; Levine, S. Online meta-learning. *arXiv* **2019**, arXiv:1902.08438.
25. Konobeev, M.; Kuzborskij, I.; Szepesvári, C. On optimality of meta-learning in fixed-design regression with weighted biased regularization. *arXiv* **2020**, arXiv:2011.00344.
26. Zhou, P.; Yuan, X.; Xu, H.; Yan, S.; Feng, J. Efficient meta learning via minibatch proximal update. In Proceedings of the 2019 Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 1534–1544.
27. Denevi, G.; Stamos, D.; Ciliberto, C.; Pontil, M. Online-within-online meta-learning. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 13110–13120.
28. Meunier, D. Meta-Learning Meets Variational Inference: Learning Priors with Guarantees. Master's Thesis, Université Paris Saclay, Paris, France, 2020. Available online: https://dimitri-meunier.github.io/files/RikenReport.pdf (accessed on 26 September 2021).
29. Khodak, M.; Balcan, M.-F.; Talwalkar, A. Adaptive Gradient-Based Meta-Learning Methods. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 5917–5928.

30. Li, L.; Jamieson, K.; DeSalvo, G.; Rostamizadeh, A.; Talwalkar, A. Hyperband: A novel bandit-based approach to hyperparameter optimization. *J. Mach. Learn. Res.* **2017**, *18*, 6765–6816.
31. Shang, X.; Kaufmann, E.; Valko, M. A simple dynamic bandit algorithm for hyper-parameter tuning. In Proceedings of the 6th ICML Workshop on Automated Machine Learning, Long Beach, CA, USA, 14–15 June 2019.
32. Kivinen, J.; Warmuth, M.K. Exponentiated gradient versus gradient descent for linear predictors. *Inf. Comput.* **1997**, *132*, 1–63. [CrossRef]
33. Kulis, B.; Bartlett, P.L. Implicit online learning. In Proceedings of the 27th International Conference on Machine Learning (ICML-10), Haifa, Israel, 21–24 June 2010; pp. 575–582.
34. Parikh, N.; Boyd, S. Proximal algorithms. *Found. Trends Optim.* **2014**, *1*, 127–239. [CrossRef]
35. Nesterov, Y. *Introductory Lectures on Convex Optimization: A Basic Course*; Springer Science & Business Media: Berlin, Germany, 2004; Volume 87.
36. Alquier, P. Approximate Bayesian Inference. *Entropy* **2020**, *22*, 1272. [CrossRef] [PubMed]
37. Mai, T.T. On continual single index learning. *arXiv* **2021**, arXiv:2102.12961.
38. Lin, W.; Khan, M.E.; Schmidt, M. Fast and simple natural-gradient variational inference with mixture of exponential-family approximations. In Proceedings of the 36th International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; Volume 97, pp. 3992–4002.
39. Chérief-Abdellatif, B.-E.; Alquier, P.; Khan, M.E. A generalization bound for online variational inference. In Proceedings of the Eleventh Asian Conference on Machine Learning, PMLR, Nagoya, Japan, 17–19 November 2019; Volume 101, pp. 662–677.
40. Domke, J. Provable smoothness guarantees for black-box variational inference. In Proceedings of the 37th International Conference on Machine Learning, Online, 12–18 July 2021; Volume 119, pp. 2587–2596.
41. Alquier, P. Non-exponentially weighted aggregation: Regret bounds for unbounded loss functions. In Proceedings of the 38th International Conference on Machine Learning, Online, 18–24 July 2021; Volume 139, pp. 207–218.
42. Knoblauch, J.; Jewson, J.; Damoulas, T. Generalized variational inference: Three arguments for deriving new posteriors. *arXiv* **2019**, arXiv:1904.02063.
43. Campolongo, N.; Orabona, F. Temporal variability in implicit online learning. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 6–12 December 2020; Volume 33.