Provable Guarantees for Gradient-Based Meta-Learning

Mikhail Khodak [1] **Maria-Florina Balcan** [1] **Ameet Talwalkar** [1 2]

Abstract

We study the problem of meta-learning through the lens of online convex optimization, developing a meta-algorithm bridging the gap between popular gradient-based meta-learning and classical regularization-based multi-task transfer methods. Our method is the first to simultaneously satisfy good sample efficiency guarantees in the convex setting, with generalization bounds that improve with task-similarity, while also being computationally scalable to modern deep learning architectures and the many-task setting. Despite its simplicity, the algorithm matches, up to a constant factor, a lower bound on the performance of any such parameter-transfer method under natural task similarity assumptions. We use experiments in both convex and deep learning settings to verify and demonstrate the applicability of our theory.

1. Introduction

The goal of *meta-learning* can be broadly defined as using the data of existing tasks to learn algorithms or representations that enable better or faster performance on unseen tasks. As the modern iteration of learning-to-learn (LTL) (Thrun & Pratt, 1998), research on meta-learning has been largely focused on developing new tools that can exploit the power of the latest neural architectures. Examples include the control of stochastic gradient descent (SGD) itself using a recurrent neural network (Ravi & Larochelle, 2017) and learning deep embeddings that allow simple classification methods to work well (Snell et al., 2017). A particularly simple but successful approach has been *parameter-transfer* via *gradient-based meta-learning*, which learns a *meta-initialization* ϕ for a class of parametrized functions $f_\theta : \mathcal{X} \mapsto \mathcal{Y}$ such that one or a few stochastic gradient steps on a few samples from a new task suffice to learn good task-specific model parameters $\hat{\theta}$. For example, when presented with examples $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ for an unseen task, the popular MAML algorithm (Finn et al., 2017) outputs

$$\hat{\theta} = \phi - \eta \sum_i \nabla L(f_\phi(x_i), y_i) \tag{1}$$

for loss function $L : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}_+$ and learning rate $\eta > 0$; $\hat{\theta}$ is then used for inference on the task. Despite its simplicity, gradient-based meta-learning is a leading approach for LTL in numerous domains including vision (Li et al., 2017; Nichol et al., 2018; Kim et al., 2018), robotics (Al-Shedivat et al., 2018), and federated learning (Chen et al., 2018).

While meta-initialization is a more recent approach, methods for parameter-transfer have long been studied in the multi-task, transfer, and lifelong learning communities (Evgeniou & Pontil, 2004; Kuzborskij & Orabona, 2013; Pentina & Lampert, 2014). A common classical alternative to (1), which in modern parlance may be called *meta-regularization*, is to learn a good bias ϕ for the following regularized empirical risk minimization (ERM) problem:

$$\hat{\theta} = \arg\min_\theta \frac{\|\theta - \phi\|_2^2}{2\eta} + \sum_i L(f_\theta(x_i), y_i) \tag{2}$$

Although there exist statistical guarantees and poly-time algorithms for learning a meta-regularization for simple models (Pentina & Lampert, 2014; Denevi et al., 2018b), such methods are impractical and do not scale to modern settings with deep neural architectures and many tasks. On the other hand, while the theoretically less-studied meta-initialization approach is often compared to meta-regularization (Finn et al., 2017), their connection is not rigorously understood.

In this work, we formalize this connection using the theory of online convex optimization (OCO) (Zinkevich, 2003), in which an intimate connection between initialization and regularization is well-understood due to the equivalence of online gradient descent (OGD) and follow-the-regularized-leader (FTRL) (Shalev-Shwartz, 2011; Hazan, 2015). In the lifelong setting of an agent solving a sequence of OCO tasks, we use this connection to analyze an algorithm that learns a ϕ, which can be a meta-initialization for OGD or a meta-regularization for FTRL, such that the within-task regret of these algorithms improves with the similarity of the online tasks; here the similarity is measured by the distance between the optimal actions θ^* of each task and is not known

[1]Carnegie Mellon University [2]Determined AI.
Correspondence to: Mikhail Khodak <khodak@cmu.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

beforehand. This algorithm, which we call Follow-the-Meta-Regularized-Leader (FMRL or **Ephemeral**), scales well in both computation and memory requirements, and in fact generalizes the gradient-based meta-learning algorithm Reptile (Nichol et al., 2018), thus providing a convex-case theoretical justification for a leading method in practice.

More specifically, we make the following contributions:

- Our first result assumes a sequence of OCO tasks t whose optimal actions θ_t^* are inside a small subset Θ^* of the action space. We show how Ephemeral can use these θ_t^* to make the average regret decrease in the diameter of Θ^* and do no worse on dissimilar tasks. Furthermore, we extend a lower bound of Abernethy et al. (2008) to the multi-task setting to show that one can do no more than a small constant-factor better sans stronger assumptions.
- Under a realistic assumption on the loss functions, we show that Ephemeral also has low-regret guarantees in the practical setting where the optimal actions θ_t^* are difficult or impossible to compute and the algorithm only has access to a statistical or numerical approximation. In particular, we show high probability regret bounds in the case when the approximation uses the gradients observed during within-task training, as is done in practice by Reptile (Nichol et al., 2018).
- We prove an online-to-batch conversion showing that the task parameters learned by a meta-algorithm with low task-averaged regret have low risk, connecting our guarantees to statistical LTL (Baxter, 2000; Maurer, 2005).
- We verify several assumptions and implications of our theory using a new meta-learning dataset we introduce consisting of text-classification tasks solvable using convex methods. We further study the empirical suggestions of our theory in the deep learning setting.

1.1. Related Work

Gradient-Based Meta-Learning: The model-agnostic meta-learning (MAML) algorithm of Finn et al. (2017) pioneered this recent approach to LTL. A great deal of empirical work has studied and extended this approach (Li et al., 2017; Grant et al., 2018; Nichol et al., 2018; Jerfel et al., 2018); in particular, Nichol et al. (2018) develop Reptile, a simple yet equally effective first-order simplification of MAML for which our analysis shows provable guarantees as a subcase. Theoretically, Franceschi et al. (2018) provide computational convergence guarantees for gradient-based meta-learning for strongly-convex functions, while Finn & Levine (2018) show that with infinite data MAML can approximate any function of task samples assuming a specific neural architecture as the model. In contrast to both results, we show finite-sample learning-theoretic guarantees for convex functions under a natural task-similarity assumption.

Online LTL: Learning-to-learn and multi-task learning (MTL) have both been extensively studied in the online setting, although our setting differs significantly from the one usually studied in online MTL (Abernethy et al., 2007; Dekel et al., 2007; Cavallanti et al., 2010). There, in each round an agent is told which of a fixed set of tasks the current loss belongs to, whereas our analysis is in the lifelong setting, in which tasks arrive one at a time. Here there are many theoretical results for learning useful data representations (Ruvolo & Eaton, 2013; Pentina & Lampert, 2014; Balcan et al., 2015; Alquier et al., 2017); the PAC-Bayesian result of Pentina & Lampert (2014) can also be used for regularization-based parameter transfer, which we also consider. Such methods are provable variants of practical shared-representation approaches, e.g. ProtoNets (Snell et al., 2017), but unlike our algorithms they do not scale to deep neural networks. Our work is especially related to Alquier et al. (2017), who also consider a many-task regret. We achieve similar bounds with a significantly more practical algorithm, although within-task their results hold for any low-regret method whereas ours only hold for OCO. Lastly, we note two concurrent works, by Denevi et al. (2019) and Finn et al. (2019), that address LTL via online learning, either directly or through online-to-batch conversion.

Statistical LTL: While we focus on the online setting, our online-to-batch results also imply risk bounds for distributional meta-learning. This setting was formalized by Baxter (2000); Maurer (2005) further extended the hypothesis-space-learning framework to algorithm-learning. Recently, Amit & Meir (2018) showed PAC-Bayesian generalization bounds for this setting, although without implying an efficient algorithm. Also closely related are the regularization-based approaches of Denevi et al. (2018a;b), which provide statistical learning guarantees for Ridge regression with a meta-learned kernel or bias. Denevi et al. (2018b) in particular focuses on usefulness relative to single-task learning, showing that their method is better than the ℓ_2-regularized ERM, but neither addresses the connection between loss-regularization and gradient-descent-initialization.

2. Meta-Initialization & Meta-Regularization

We study simple methods of the form of Algorithm 1, where we run a *within-task* online algorithm on each task and then update the initialization or regularization of this algorithm using a *meta-update* online algorithm. Alquier et al. (2017) study such a method where the meta-update is conducted using exponentially-weighted averaging. Our use of OCO for the meta-update makes this class of algorithms much more practical; for example, in the case of OGD for both the inner and outer loop we recover the Reptile algorithm of Nichol et al. (2018). To analyze Algorithm 1, we first discuss the OCO methods that make up both its inner and outer loop and the inherent connection they provide between initialization

Algorithm 1: The generic online-within-online algorithm we study. First-order gradient-based meta-learning uses OGD in both the inner and outer loop.

Pick a first meta-initialization ϕ_1.
for task $t \in [T]$ **do**
  Run a within-task online algorithm (e.g. OGD) on the losses of task t using initialization ϕ_t.
  Compute (exactly or approximately) the best fixed action in hindsight θ_t^* for task t.
  Update ϕ_t using a meta-update online algorithm (e.g. OGD) on the meta-loss $\ell_t(\phi) = \|\theta_t^* - \phi_t\|^2$.

and regularization. We then make this connection explicit by formalizing the notion of learning a meta-initialization or meta-regularization as learning a parameterized Bregman regularizer. We conclude this section by proving convex-case upper and lower bounds on the task-averaged regret.

2.1. Online Convex Optimization

In the online learning setting, at each time $t = 1, \dots, T$ an agent chooses action $\theta_t \in \Theta \subset \mathbb{R}^d$ and suffers loss $\ell_t(\theta_t)$ for some adversarially chosen function $\ell_t : \Theta \mapsto \mathbb{R}$ that subsumes the loss, model, and data in $L(f_\theta(x), y)$ into one function of θ. The goal is to minimize *regret* – the difference between the total loss and that of the optimal fixed action:

$$\mathbf{R}_T = \sum_{t=1}^{T} \ell_t(\theta_t) - \min_{\theta \in \Theta} \sum_{t=1}^{T} \ell_t(\theta)$$

When $\mathbf{R}_T = o(T)$ then as $T \to \infty$ the average loss of the agent will approach that of an optimal fixed action.

For OCO, ℓ_t is assumed convex and Lipschitz for all t. This setting provides many practically useful algorithms such as online gradient descent (OGD). Parameterized by a starting point $\phi \in \Theta$ and learning rate $\eta > 0$, OGD plays

$$\theta_t = \text{Proj}_\Theta \left(\phi - \eta \sum_{s<t} \nabla \ell_s(\theta_s) \right) \tag{3}$$

and achieves sublinear regret $\mathcal{O}(D\sqrt{T})$ when $\eta \propto \frac{D}{\sqrt{T}}$, where D is the diameter of the action space Θ.

Note the similarity between OGD and the meta-initialization update in Equation 1. In fact another fundamental OCO algorithm, follow-the-regularized-leader (FTRL), is a direct analog for the meta-regularization algorithm in Equation 2, with its action at each time being the output of ℓ_2-regularized ERM over the previous data:

$$\theta_t = \arg\min_{\theta \in \Theta} \frac{1}{2\eta} \|\theta - \phi\|_2^2 + \sum_{s<t} \ell_s(\theta) \tag{4}$$

Note that most definitions set $\phi = 0$. A crucial connection here is that on linear functions $\ell_t(\cdot) = \langle \nabla_t, \cdot \rangle$, OGD initialized at $\phi = 0$ plays the same actions $\theta_t \in \Theta \; \forall \; t \in [T]$ as FTRL. Since linear losses are the hardest losses, in that low regret for them implies low regret for convex functions (Zinkevich, 2003), in the online setting this equivalence suggests that meta-initialization is a reasonable surrogate for meta-regularization because it is solving the hardest version of the problem. The OGD-FTRL equivalence can be extended to other geometries by replacing the squared-norm in (4) by a strongly-convex function $R : \Theta \mapsto \mathbb{R}_+$:

$$\theta_t = \arg\min_{\theta \in \Theta} \frac{1}{\eta} R(\theta) + \sum_{s<t} \ell_s(\theta)$$

In the case of linear losses this is the online mirror descent (OMD) generalization of OGD. For G-Lipschitz losses, OMD and FTRL have the following well-known regret guarantee $\forall \; \theta^* \in \Theta$ (Shalev-Shwartz, 2011, Theorem 2.11):

$$\mathbf{R}_T \leq \frac{1}{\eta} R(\theta^*) + \eta G^2 T \tag{5}$$

2.2. Task-Averaged Regret and Task Similarity

We consider the *lifelong* extension of online learning, where $t = 1, \dots, T$ now index a sequence of online learning problems, in each of which the agent must sequentially choose m_t actions $\theta_{t,i} \in \Theta$ and suffer loss $\ell_{t,i} : \Theta \mapsto \mathbb{R}$. Since in meta-learning we are interested in doing well on individual tasks, we will aim to minimize a dynamic notion of regret in which the comparator changes with each task, so that the comparator corresponds to the best within-task parameter:

Definition 2.1. *The* **task-averaged regret** *(TAR) of an online algorithm after T tasks with $\{m_t\}_{t=1}^T$ steps is*

$$\bar{\mathbf{R}} = \frac{1}{T} \sum_{t=1}^{T} \left(\sum_{i=1}^{m_t} \ell_{t,i}(\theta_{t,i}) - \min_{\theta_t \in \Theta} \sum_{i=1}^{m_t} \ell_{t,i}(\theta_t) \right)$$

Note that, unlike in standard regret one cannot achieve TAR decreasing in T, the number of tasks, because the comparator is dynamic and so can force a constant loss at each task t. Furthermore, the average is taken over T and *not* the number of rounds per task m_t, so in our results we expect TAR to grow sub-linearly in m_t. This corresponds to achieving sub-linear single-task regret on-average.

An alternative comparator that is seemingly natural in the study of gradient-based meta-learning is the best fixed initialization in hindsight; however, this quantity overlooks the fact that meta-initialization is simply a tool to achieve what we actually care about, which is within-task performance. If the difference between the task loss when starting from the best meta-initialization and that of the optimal within-task parameter is high, comparing to the best meta-initialization may not be very meaningful. On the other hand, a low TAR ensures that the task loss of an algorithm compared to that of the optimal within-task parameter is low on average.

We now formalize our similarity assumption on the tasks $t \in [T]$: their optimal actions θ_t^* lie within a small subset

Figure 1. Random projection of ERM parameters of 1-shot (left) and 32-shot (right) Mini-Wiki tasks, described in Section 4.

Θ^* of the action space. This is natural for studying gradient-based meta-learning, as the notion that there exists a meta-parameter ϕ from which a good parameter for any individual task is reachable with only a few steps implies that they are all close together. We develop algorithms whose TAR scales with the diameter D^* of Θ^*; notably, this means they will not do much worse if $\Theta^* = \Theta$, i.e. if the tasks are not related in this way, but will do well if $D^* \ll D$. Importantly, our methods will not require knowledge of Θ^*.

Setting 2.1. *Each task $t \in [T]$ has m_t convex loss functions $\ell_{t,i} : \Theta \mapsto \mathbb{R}$ that are G_t-Lipschitz on-average. Let $\theta_t^* \in \arg\min_{\theta\in\Theta} \sum_{i=1}^{m_t} \ell_{t,i}(\theta)$ be the minimum-norm optimal fixed action for task t. Define $\Theta^* \subset \Theta$ to be the minimal subset containing $\theta_t^*\ \forall\ t \in [T]$. Assume that Θ^* has non-empty interior (and thus $T > 1$).*

Note θ_t^* is unique as the minimum of $\|\cdot\|^2$, a strongly convex function, over minima of a convex function. The algorithms in Section 2.4 assume an efficient oracle computing θ_t^*.

2.3. Parameterizing Bregman Regularizers

Following the main idea of gradient-based meta-learning, our goal is to learn a $\phi \in \Theta$ such that an online algorithm such as OGD starting from ϕ will have low regret. We thus treat regret as our objective and observe that in the regret of FTRL (5), the regularizer R effectively encodes a distance from the initialization to θ^*. This is clear in the Euclidean geometry for $R(\theta) = \frac{1}{2}\|\theta - \phi\|_2^2$, but can be extended via the *Bregman divergence* (Bregman, 1967), defined for $f : S \mapsto \mathbb{R}$ everywhere-sub-differentiable and convex as

$$\mathcal{B}_f(x||y) = f(x) - f(y) - \langle \nabla f(y), x - y\rangle$$

The Bregman divergence has many useful properties (Banerjee et al., 2005) that allow us to use it almost directly as a parameterized regularization function. However, in order to use OCO for the meta-update we also require it to be strictly convex in the second argument, a property that holds for the Bregman divergence of both the ℓ_2 regularizer and the entropic regularizer $R(\theta) = \langle \theta, \log\theta\rangle$ used for online learning over the probability simplex, e.g. with expert advice.

Definition 2.2. *Let $R : S \mapsto \mathbb{R}$ be 1-strongly-convex w.r.t. norm $\|\cdot\|$ on convex $S \subset \mathbb{R}^d$. Then we call the Bregman divergence $\mathcal{B}_R(x||y) : S \times S \mapsto \mathbb{R}_+$ a* **Bregman regularizer** *if $\mathcal{B}_R(x||\cdot)$ is strictly convex for any fixed $x \in S$.*

Within each task, the regularizer is parameterized by the second argument and acts on the first. More specifically, for $R = \frac{1}{2}\|\cdot\|_2^2$ we have $\mathcal{B}_R(\theta||\phi) = \frac{1}{2}\|\theta - \phi\|_2^2$, and so in the case of FTRL and OGD, ϕ is a parameterization of the regularization and the initialization, respectively. In the case of the entropic regularizer, the associated Bregman regularizer is the KL-divergence from ϕ to θ and thus meta-learning ϕ can very explicitly be seen as learning a prior.

Finally, we use Bregman regularizers to formally define our parameterized learning algorithms:

Definition 2.3. $\mathrm{FTRL}_{\eta,\phi}$*, for $\eta \in \mathbb{R}_+, \phi \in \Theta$, where Θ is some bounded convex subset $\Theta \subset \mathbb{R}^d$, plays*

$$\theta_t = \arg\min_{\theta\in\Theta} \mathcal{B}_R(\theta||\phi) + \eta \sum_{s<t} \ell_s(\theta)$$

for Bregman regularizer $\mathcal{B}_R$. Similarly, $\mathrm{OMD}_{\eta,\phi}$ *plays*

$$\theta_t = \arg\min_{\theta\in\Theta} \mathcal{B}_R(\theta||\phi) + \eta \sum_{s<t} \langle \nabla_s, \theta\rangle$$

Here FTRL and OMD correspond to the meta-regularization (2) and meta-initialization (1) approaches, respectively. As $\mathcal{B}_R(\cdot||\phi)$ is strongly-convex, both algorithms have the same regret bound (5), allowing us to analyze them jointly.

2.4. Follow-the-Meta-Regularized-Leader

We now specify the first variant of our main algorithm, Follow-the-Meta-Regularized-Leader (Ephemeral). First assume the diameter D^* of Θ^*, as measured by the square root of the maximum Bregman divergence between any two points, is known. Starting with $\phi_1 \in \Theta$, run $\mathrm{FTRL}_{\eta,\phi_t}$ or $\mathrm{OMD}_{\eta,\phi_t}$ with $\eta \propto \frac{D^*}{\sqrt{m}}$ on the losses in each task t. After each task, compute ϕ_{t+1} using an OCO meta-update algorithm operating on the Bregman divergences $\mathcal{B}_R(\theta_t^*||\cdot)$. For D^* unknown, make an underestimate $\varepsilon > 0$ and multiply it by a factor $\gamma > 1$ each time $\mathcal{B}_R(\theta_t^*||\phi_t) > \varepsilon^2$.

The following is a regret bound for this algorithm when the meta-update is either *Follow-the-Leader* (FTL), which plays the minimizer of all past losses, or OGD with adaptive step size. We call this Ephemeral variant *Follow-the-Average-Leader* (FAL) because in the case of FTL the algorithm uses the mean of the previous optimal parameters in hindsight as the initialization. Pseudo-code for this and other variants is given in Algorithm 2. For brevity, we state results for constant $G_t = G, m_t = m\ \forall\ t$; detailed statements are in the supplement together with the full proof.

Theorem 2.1. *In Setting 2.1, the FAL variant of Algorithm 2 with task similarity guess $\varepsilon = D\frac{1+\log T}{T}$, tuning parameter $\gamma = \frac{1+\log T}{\log T}$, and $\mathcal{B}_R$ that is Lipschitz on Θ^* achieves TAR*

$$\bar{\mathbf{R}} \le \mathcal{O}\left(D^* + \frac{D\log T}{D^* T}\right)\sqrt{m}$$

for diameter $D^* = \max_{\theta,\phi\in\Theta^*} \sqrt{\mathcal{B}_R(\theta||\phi)}$ of Θ^*.

Proof Sketch. We give a proof for $R(\cdot) = \frac{1}{2}\|\cdot\|_2^2$ and known task similarity, i.e. $\varepsilon = D^*, \gamma = 1$. Denote the divergence to θ_t^* by $\Delta_t(\phi) = \mathcal{B}_R(\theta_t^* || \phi) = \frac{1}{2}\|\theta_t^* - \phi\|_2^2$ and let $\phi^* = \frac{1}{T}\sum_{t=1}^T \theta_t^*$. Note Δ_t is 1-strongly-convex and ϕ^* is the minimizer of their sum, with the variance $\bar{D}^2 = \frac{1}{T}\sum_{t=1}^T \Delta_t(\phi^*) \le D^{*2}$. Now by Definition 2.1:

$$\begin{aligned}\bar{\mathbf{R}} &= \frac{1}{T}\sum_{t=1}^T \left(\sum_{i=1}^m \ell_{t,i}(\theta_{t,i}) - \min_{\theta_t \in \Theta} \sum_{i=1}^m \ell_{t,i}(\theta_t)\right) \\ &\le \frac{1}{T}\sum_{t=1}^T \frac{\Delta_t(\phi_t)}{\eta} + \eta G^2 m \\ &= \frac{1}{T}\sum_{t=1}^T \frac{\Delta_t(\phi_t) - \Delta_t(\phi^*)}{\eta} + \frac{1}{T}\sum_{t=1}^T \frac{\Delta_t(\phi^*)}{\eta} + \eta G^2 m\end{aligned}$$

The first two lines apply the regret bound (5) of FTRL and OMD. The key step is the last one, with the regret is split into the loss of the meta-update algorithm on the left and the loss if we had always initialized at the mean ϕ^* of the optimal actions θ_t^* on the right. Since $\Delta_1, \dots, \Delta_T$ are 1-strongly-convex with minimizer ϕ^*, and since each ϕ_t is determined by playing FTL or OGD on these same functions, the left term is the regret of these algorithms on strongly-convex functions, which is known to be $\mathcal{O}(\log T)$ (Bartlett et al., 2008; Kakade & Shalev-Shwartz, 2008). Substituting the definition of ϕ^* and $\eta = \frac{D^*}{G\sqrt{m}}$ sets the right term to

$$\frac{1}{T}\sum_{t=1}^T \frac{\Delta_t(\phi^*)}{\eta} + \eta G^2 m = G\bar{D}\sqrt{m} + GD^*\sqrt{m}$$

□

The full proof uses the doubling trick to tune task similarity D^*, requiring an analysis of the location of meta-parameter ϕ_t to ensure that we only increase the guess when needed. The extension to non-Euclidean geometries uses a novel logarithmic regret bound for FTL over Bregman regularizers.

Remark 2.1. *Note that if we know the variance $\bar{D}^2$ of the task parameters from their mean ϕ^*, setting $\eta_t = \frac{\bar{D}}{G_t\sqrt{m_t}}$ in Algorithm 2 and following the analysis above replaces D^* in Theorem 2.1 with $\bar{D}$, which is better since $\bar{D} \le D^*$ and is furthermore less sensitive to possible outlier tasks.*

Theorem 2.1 shows that the TAR of Ephemeral scales with task similarity D^*, and that if tasks are not similar then we only do a constant factor worse than FTRL or OMD. This shows that gradient-based meta-learning is useful in convex settings: under a simple notion of similarity, having more tasks yields better performance than the $\mathcal{O}(D\sqrt{m})$ regret of single-task learning. The algorithm also scales well and in the ℓ_2 setting is similar to Reptile (Nichol et al., 2018).

However, it is easy to see that an even simpler "strawman" algorithm achieves regret only a constant factor worse: at time $t+1$, simply initialize FTRL or OMD using the optimal parameter θ_t^* of task t. Of course, in the few-shot setting of small m, a reduction in the average regret is still practically significant; we observe this empirically in Figure 3. Indeed, in the proof of Theorem 2.1 the regret converges to that obtained by always playing the mean optimal action, which will not occur when playing the strawman algorithm. Furthermore, the following lower-bound on the task-averaged regret, a multi-task extension of Abernethy et al. (2008, Theorem 4.2), shows that such constant factor reductions are the best we can achieve under our task similarity assumption:

Algorithm 2: Follow-the-Meta-Regularized-Leader (Ephemeral) meta-algorithm for meta-learning. For the FAL variant we assume $\arg\min_{\theta\in\Theta} L(\theta)$ returns the minimum-norm θ among all minimizers of L over Θ. For META = OGD we assume $R(\cdot) = \frac{1}{2}\|\cdot\|_2^2$ and adaptive step size $\left(\sum_{s<t}\sqrt{m_s}\right)^{-1}$ at each time t.

Data:
- initialization ϕ_1 in action space Θ
- meta-update algorithm META_ϕ (FTL or OGD)
- within-task algorithm $\text{TASK}_{\eta,\phi}$ (FTRL or OMD) with Bregman regularizer $\mathcal{B}_R$ w.r.t. $\|\cdot\|$
- Lipschitz constant G_t w.r.t. $\|\cdot\|_*$ on each task t
- similarity guess $\varepsilon > 0$ and tuning parameter $\gamma \ge 1$

```
// set first-task similarity guess
   to be the full action space
```
$D_1 \leftarrow \max_{\theta\in\Theta}\sqrt{\mathcal{B}_R(\theta||\phi_1)} + \varepsilon$
$k \leftarrow 0$
for $t \in [T]$ **do**
```
// set learning rate using task
   similarity guess; run
   within-task algorithm
```
$\eta_t \leftarrow \frac{D_t}{G_t\sqrt{m_t}}$
for $i \in [m_t]$ **do**
  $\theta_{t,i} \leftarrow \text{TASK}_{\eta_t,\phi_t}(\ell_{t,1}, \dots, \ell_{t,i-1})$
  suffer loss $\ell_{t,i}(\theta_{t,i})$
```
// compute meta-update vector θ_t
   depending on Ephemeral variant
```
case FAL **do**
  $\theta_t \leftarrow \arg\min_{\theta\in\Theta}\sum_{i=1}^{m_t}\ell_{t,i}(\theta)$
case FLI-Online **do**
  $\theta_t \leftarrow \text{TASK}_{\eta_t,\phi_t}(\ell_{t,1}, \dots, \ell_{t,m_t})$
case FLI-Batch **do**
  $\theta_t \leftarrow \frac{1}{m_t}\sum_{i=1}^{m_t}\theta_{t,i}$
```
// increase task similarity guess
   if violated; run meta-update
```
if $D_t < \sqrt{\mathcal{B}_R(\theta_t||\phi_t)}$ **then**
  $k \leftarrow k + 1$
$D_{t+1} \leftarrow \gamma^k\varepsilon$
$\phi_{t+1} \leftarrow \text{META}_{\theta_1}(\{\mathcal{B}_R(\theta_s||\cdot)G_s\sqrt{m_s}\}_{s=1}^t)$

Theorem 2.2. *Assume $d \geq 3$ and that for each $t \in [T]$ an adversary must play a sequence of m convex G-Lipschitz functions $\ell_{t,i} : \Theta \mapsto \mathbb{R}$ whose optimal actions in hindsight $\arg\min_{\theta \in \Theta} \sum_{i=1}^{m} \ell_{t,i}(\theta)$ are contained in some fixed ℓ_2-ball $\Theta^* \subset \Theta$ with center ϕ^* and diameter D^*. Then the adversary can force the agent to have TAR at least $\frac{GD^*}{4}\sqrt{m}$.*

More broadly, this lower bound shows that the learning-theoretic benefits of gradient-based meta-learning are inherently limited without stronger assumptions on the tasks. Nevertheless, Ephemeral-style algorithms are very attractive from a practical perspective, as their memory and computation requirements per iteration scale linearly in the dimension and not at all in the number of tasks.

3. Provable Guarantees for Practical Gradient-Based Meta-Learning

In the previous section we gave an algorithm with access to the best actions in hindsight θ_t^* of each task that can learn a good meta-initialization or meta-regularization. While θ_t^* is efficiently computable in some cases, often it is more practical to use an approximation. This holds in the deep learning setting, e.g. Nichol et al. (2018) use the average within-task gradient. Furthermore, in the batch setting a more natural similarity notion depends on the true risk minimizers and not the optimal actions for a few samples. In this section we first show how two simple variants of Ephemeral handle these settings, one for the adversarial setting which uses the final action on task t as the meta-update and one for the stochastic setting using the average iterate. We call these methods FLI-Online and FLI-Batch, respectively, where *FLI* stands for *Follow-the-Last-Iterate*. We then provide an online-to-batch conversion result for TAR that implies good generalization guarantees when any of the variants of Ephemeral are run in the distributional LTL setting.

3.1. Simple-to-Compute Meta-Updates

To achieve guarantees using approximate meta-updates we need to make some assumptions on the within-task loss functions. This is unavoidable because we need estimates of the optimal actions of different tasks to be nearby; in general, for some $\theta \in \Theta$ a convex function $f : \Theta \mapsto \mathbb{R}$ can have small $f(\theta) - f(\theta^*)$ but large $\|\theta - \theta^*\|$ if f does not increase quickly away from the minimum. This makes it impossible to use guarantees on the loss of an estimate of θ_t^* to bound its distance from θ_t^*. We therefore make assumptions that some aggregate loss, e.g. the expectation or sum of the within-task losses, satisfies the following growth condition:

Definition 3.1. *A function $f : \Theta \mapsto \mathbb{R}$ has α-**quadratic-growth** (α-QG) w.r.t. $\|\cdot\|$ for $\alpha > 0$ if for any $\theta \in \Theta$ and θ^* its closest minimum of f we have*

$$\frac{\alpha}{2}\|\theta - \theta^*\|^2 \leq f(\theta) - f(\theta^*)$$

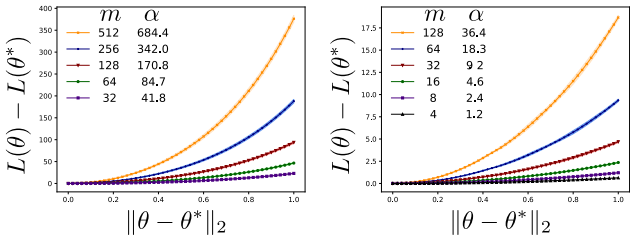

Figure 2. Plot of the smallest $L(\theta) - L(\theta^*)$ as $\|\theta - \theta^*\|_2$ increases for logistic regression over a mixture of four 50-dimensional Gaussians (left) and over a four-class text classification task over 50-dimensional CBOW (right). For both the α factor of the quadratic-growth condition scales linearly with the number of samples m.

QG has recently been used to provide fast rates for GD that hold for practical problems such as LASSO and logistic regression under data-dependent assumptions (Karimi et al., 2016; Garber, 2019). It can be shown when $f(\theta) = g(A\theta)$ for g strongly-convex and some $A \in \mathbb{R}^{m \times d}$; in this case $\alpha \geq \sigma_{\min}(A)$ (Karimi et al., 2016). Note that α-QG is also a weaker condition than α-strong-convexity.

To prove FLI guarantees, we require in Setting 3.1 that some notion of average loss on each task grows quadratically away from the optimum, which is shown to hold in both a real and a synthetic setting in Figure 2.

Setting 3.1. *In Setting 2.1, for each task $t \in [T]$ define average loss L_t according to one of the following two cases:*

(a) $L_t(\theta) = \frac{1}{m_t}\sum_{i=1}^{m_t} \ell_{t,i}(\theta)$

(b) assume losses $\ell_{t,i} : \Theta \mapsto [0,1]$ are i.i.d. from distribution $\mathcal{P}_t$ s.t. $L_t(\theta) = \mathbb{E}_{\mathcal{P}_t}\ell(\theta)$ has a unique minimum

Assume the corresponding L_t in each case is α-QG w.r.t. $\|\cdot\|$ and define $\Theta^* \subset \Theta$ s.t. $\Theta^* \supset \arg\min_{\theta \in \Theta} L(\theta)\ \forall\, t \in [T]$.

Here case (b) is the batch-within-online setting, also studied by Alquier et al. (2017). In this case the distance defining the similarity is between the true-risk minimizers and *not* the optimal parameters in hindsight. Under such data-dependent assumptions we have the following bound on using approximate meta-updates:

Theorem 3.1. *In Setting 3.1(a), the FLI-Online variant of Algorithm 2 with $\varepsilon = \Omega\left(\frac{1}{\sqrt[6]{m}}\right)$, tuning parameter $\gamma \geq 1$, and within-task algorithm FTRL with Bregman regularizer $\mathcal{B}_R$ for R strongly-smooth w.r.t. $\|\cdot\|$ achieves TAR*

$$\bar{\mathbf{R}} \leq \mathcal{O}\left(D^* + \frac{D}{D^*}\left(\frac{\log T}{T} + o_m(1)\right)\right)\sqrt{m}$$

for D^* as in Theorem 2.1 and $o_m(1) = \mathcal{O}(m^{-\frac{1}{6}})$. In Setting 3.1(b) the same bound holds w.p. $1 - \delta$ and $o_m(1) = \mathcal{O}\left(m^{-\frac{1}{6}}\sqrt{\log\frac{Tm}{\delta}}\right)$ for both the FAL and FLI-Batch variants and using either FTRL or OMD within-task.

This bound is very similar to Theorem 2.1 apart from a per-task error term due to the use of an estimate of θ_t^*.

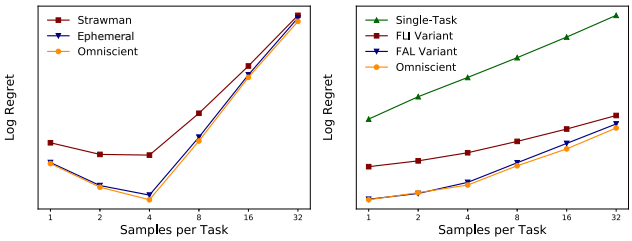

Figure 3. TAR of Ephemeral and the strawman method for FTRL (left) and of variants of Ephemeral for OGD (right). Ephemeral is much better than the strawman at low m, showing the significance of Theorem 2.1 in the few-shot case. As predicted by Theorem 3.1, FLI regret converges to that of FAL as m increases.

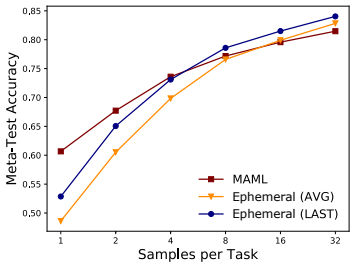

Figure 4. Meta-test accuracy of MAML and Ephemeral in the batch setting. Both using the average iterate, as recommended by online-to-batch conversion, and using the last iterate, as done in practice, provides performance comparable to that of MAML.

3.2. Distributional Learning-to-Learn

While gradient-based LTL methods are largely online, their goals are often statistical. The usual setting due to Baxter (2000) assumes a distribution $\mathcal{Q}$ over task-distributions $\mathcal{P}$ over functions ℓ, which can correspond to a single-sample loss. Given i.i.d. samples from each of T i.i.d. task-samples $\mathcal{P}_t \sim \mathcal{Q}$, we seek to do learn how to do well given m samples from a new distribution $\mathcal{P} \sim \mathcal{Q}$. Here we hope that samples from $\mathcal{Q}$ can reduce the amount needed from $\mathcal{P}$.

Theorem 3.2 gives an online-to-batch conversion for which low TAR implies low expected risk of a new task sampled from $\mathcal{Q}$. For Ephemeral, the procedure draws $t \sim \mathcal{U}[T]$, runs $\mathrm{FTRL}_{\eta_t,\phi_t}$ or $\mathrm{OMD}_{\eta_t,\phi_t}$ on samples from $\mathcal{P} \sim \mathcal{Q}$, and outputs the average iterate $\bar{\theta}$. Such guarantees on random or mean iterates are standard, although in practice the last iterate is used. The proof uses Jensen's inequality to combine two standard conversions (Cesa-Bianchi et al., 2004).

Theorem 3.2. *Suppose convex losses $\ell_{t,i} : \Theta \mapsto [0,1]$ are drawn i.i.d. from $\mathcal{P}_t \sim \mathcal{Q}, \{\ell_{t,i}\}_i \sim \mathcal{P}_t^m$ for some distribution $\mathcal{Q}$ over task distributions $\mathcal{P}_t$. Let $\mathcal{A}_t$ be the state (e.g. the initialization ϕ_t and similarity guess D_t) before task $t \in [T]$ of an algorithm $\mathcal{A}$ with TAR $\bar{\mathbf{R}}$. Then w.p. $1-\delta$ if m loss functions $\{\ell_i\}_i \sim \mathcal{P}^m$ are sampled from task distribution $\mathcal{P} \sim \mathcal{Q}$, running $\mathcal{A}_t$ on these losses will generate $\theta_1, \ldots, \theta_m \in \Theta$ s.t. their mean $\bar{\theta}$ satisfies*

$$\mathop{\mathbb{E}}_{t\sim\mathcal{U}[T]} \mathop{\mathbb{E}}_{\substack{\ell\sim\mathcal{P}\\ \mathcal{P}\sim\mathcal{Q}}} \mathop{\mathbb{E}}_{\mathcal{P}^m} \ell(\bar{\theta}) = \mathop{\mathbb{E}}_{\substack{\ell\sim\mathcal{P}\\ \mathcal{P}\sim\mathcal{Q}}} \ell(\theta^*) + \frac{\bar{\mathbf{R}}}{m} + \sqrt{\frac{8}{T}\log\frac{1}{\delta}}$$

4. Empirical Results

An important aspect of Ephemeral is its practicality. n particular, FLI-Batch is scalable without modification to high-dimensional, non-convex models. This is demonstrated by the success of Reptile (Nichol et al., 2018), a sub-case of our method that competes with MAML on standard meta-learning benchmarks. Given this evidence, empirically our goal is to validate our theory in the convex setting, although we also examine implications for deep meta-learning.

4.1. Convex Setting

We introduce a new dataset of 812 classification tasks, each consisting of sentences from one of four Wikipedia pages which we use as labels. It is derived from the raw super-set of the Wiki3029 corpus collected by Arora et al. (2019). We call the new dataset *Mini-Wiki* and make it available in the supplement. Our use of text classification to examine the convex setting is motivated by the well-known effectiveness of linear models over simple representations (Wang & Manning, 2012; Arora et al., 2018). We use logistic regression over 50-dimensional continuous-bag-of-words (CBOW) using GloVe embeddings (Pennington et al., 2014). The similarity of these tasks is verified by seeing if their optimal parameters are close together. As shown before in Figure 1, we find when Θ is the unit ball that even in the 1-shot setting the tasks have non-vacuous similarity; for 32-shots the parameters are contained in a set of radius 0.32.

We next compare Ephemeral to the "strawman" algorithm from Section 2, which uses the previous optimal action as the initialization. For both algorithms we use similarity guess $\varepsilon = 0.1$ and tune with $\gamma = 1.1$. As expected, in Figure 3 we see that Ephemeral is superior to the strawman algorithm, especially for few-shot learning, demonstrating that our TAR improvement is significant in the low-sample regime. We also see that FLI-Batch, which uses approximate meta-updates, approaches FAL as the number of samples increases and thus its estimate improves.

Finally, we evaluate Ephemeral and (first-order) MAML in the statistical setting. On each task we standardize data using the mean and deviation of the training features. For Ephemeral we use the FAL variant with OGD as the within-task algorithm, with learning rate set using the average deviation of the task parameters from the mean parameter, as suggested in Remark 2.1. For MAML, we use grid search to determine the within-task and meta-update learning rates. As shown in Figure 4, despite using no tuning, Ephemeral performs comparably to MAML – slightly better for $m \geq 8$ and slightly worse for $m < 4$.

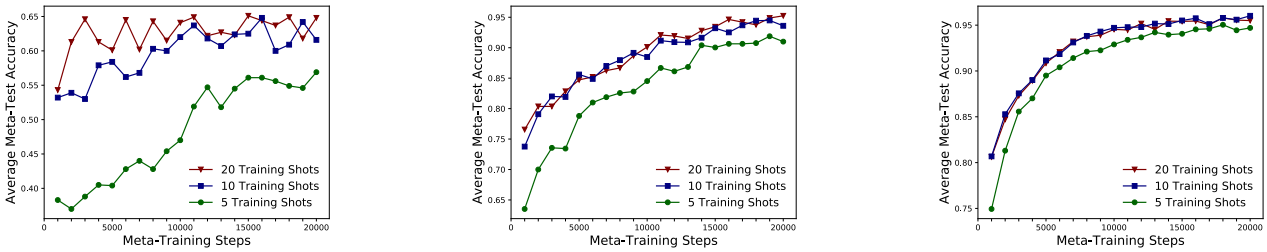

Figure 5. Performance of Reptile (the FLI variant of Ephemeral using OGD within-task) on 5-shot 5-way Mini-ImageNet (left), 1-shot 5-way Omniglot (center), and 5-shot 20-way Omniglot (right) while varying the number of *training* samples. Increasing the number of samples per training task improves performance even when using the same number of samples at meta-test time.

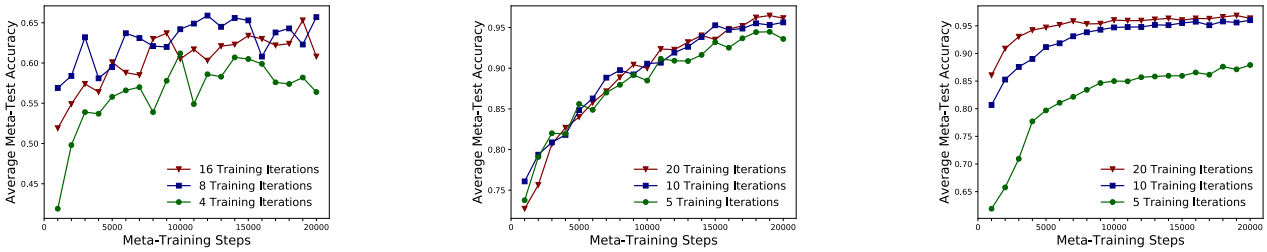

Figure 6. Performance of Reptile (the FLI variant of Ephemeral using OGD within-task) on 5-shot 5-way Mini-ImageNet (left), 1-shot 5-way Omniglot (center), and 5-shot 20-way Omniglot (right) while varying the number of *training* iterations. The benefit of more iterations is not clear for Mini-ImageNet, but an improvement is seen on Omniglot. The number of iterations at meta-test time is 50.

4.2. Deep Learning

While our method generalizes Reptile, an effective meta-learning method (Nichol et al., 2018), we can still examine if our theory can help neural network LTL. We study modifications of Reptile on 5-way and 20-way Omniglot (Lake et al., 2017) and 5-way Mini-ImageNet classification (Ravi & Larochelle, 2017) using the same networks as Nichol et al. (2018). As in these works, we evaluate in the *transductive* setting, where test points are evaluated in batch.

Our theory points to the importance of accurately computing the within-task parameter for the meta-update; Theorem 2.1 assumes access to this parameter, whereas Theorems 3.1 allow computational and stochastic approximations that result in an additional error term decaying with number of task-examples. This becomes relevant in the non-convex setting with many tasks, where it is infeasible to find even a local optimum. Thus we see how a better estimate of the within-task parameter for the meta-update may lead to higher accuracy. We can attain a better estimate by using more samples to reduce stochastic noise or by running more gradient steps on each task to reduce approximation error. It is not obvious that these changes will improve performance – it may be better to learn using the same settings at meta-train and meta-test time. However, for 5-shot evaluation the Reptile authors do indeed use more than 5 task samples – 10 for Omniglot and 15 for Mini-ImageNet. Similarly, they use far fewer within-task gradient steps – 5 for Omniglot and 8 for Mini-ImageNet – at meta-train time than the 50 iterations used for evaluation.

We study how the two settings – the number of task samples and within-task iterations – affect meta-test performance. In Figure 5, we see that more task-samples provide a significant improvement, with fewer meta-iterations needed for good test performance. Reducing this number is equivalent to reducing task-sample complexity, although for a better approximation each task needs more samples. We also see in Figure 6 that taking more gradient steps, which does not use more samples, can also help performance, especially on 20-way Omniglot. However, on Mini-ImageNet using than 8 iterations reduces performance; this may be due to overfitting on specific tasks, with task similarity likely holding for the true rather than empirical risk minimizers, as in Setting 3.1(b). The broad patterns shown above also hold for several other settings, which we discuss in the supplement.

5. Conclusion

In this paper we study a broad class of gradient-based meta-learning methods using the theory of OCO, proving their usefulness compared to single-task learning under a closeness assumption on task parameters. The guarantees of our algorithm, Ephemeral, can be extended to approximate meta-updates, the batch-within-online setting, and statistical LTL. Apart from these results, the algorithm's simplicity makes it extensible to settings of practical interest such as federated learning and differential privacy. Future work can consider more sophisticated notions of task-similarity, such as multi-modal or evolving settings, and theory for practical and scalable shared-representation-learning.

Acknowledgments

This work was supported in part by DARPA FA875017C0141, National Science Foundation grants CCF-1535967, IIS-1618714, IIS-1705121, and IIS-1838017, a Microsoft Research Faculty Fellowship, an Okawa Grant, a Google Faculty Award, an Amazon Research Award, an Amazon Web Services Award, and a Carnegie Bosch Institute Research Award. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA, the National Science Foundation, or any other funding agency.

References

Abernethy, J., Bartlett, P., and Rakhlin, A. Multitask learning with expert advice. In *Proceedings of the International Conference on Computational Learning Theory*, 2007.

Abernethy, J., Bartlett, P. L., Rakhlin, A., and Tewari, A. Optimal strategies and minimax lower bounds for online convex games. Technical report, EECS Department, University of California, Berkeley, 2008.

Al-Shedivat, M., Bansal, T., Burda, Y., Sutskever, I., Mordatch, I., and Abbeel, P. Continuous adaptation via meta-learning in nonstationary and competitive environments. In *Proceedings of the 6th International Conference on Learning Representations*, 2018.

Alquier, P., Mai, T. T., and Pontil, M. Regret bounds for lifelong learning. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 2017.

Amit, R. and Meir, R. Meta-learning by adjusting priors based on extended PAC-Bayes theory. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

Arora, S., Khodak, M., Saunshi, N., and Vodrahalli, K. A compressed sensing view of unsupervised text embeddings, bag-of-n-grams, and LSTMs. In *Proceedings of the 6th International Conference on Learning Representations*, 2018.

Arora, S., Khandeparkar, H., Khodak, M., Plevrakis, O., and Saunshi, N. A theoretical analysis of contrastive unsupervised representation learning. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

Azuma, K. Weighted sums of certain dependent random variables. *Tôhoku Mathematical Journal*, 19:357–367, 1967.

Balcan, M.-F., Blum, A., and Vempala, S. Efficient representations for lifelong learning and autoencoding. In *Proceedings of the Conference on Learning Theory*, 2015.

Banerjee, A., Merugu, S., Dhillon, I. S., and Ghosh, J. Clustering with Bregman divergences. *Journal of Machine Learning Research*, 6:1705–1749, 2005.

Bartlett, P. L., Hazan, E., and Rakhlin, A. Adaptive online gradient descent. In *Advances in Neural Information Processing Systems*, 2008.

Baxter, J. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12:149–198, 2000.

Bregman, L. M. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7: 200–217, 1967.

Cavallanti, G., Cesa-Bianchi, N., and Gentile, C. Linear algorithms for online multitask classification. *Journal of Machine Learning Research*, 11:2901–2934, 2010.

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.

Chen, F., Dong, Z., Li, Z., and He, X. Federated meta-learning for recommendation. arXiv, 2018.

Dekel, O., Long, P. M., and Singer, Y. Online learning of multiple tasks with a shared loss. *Journal of Machine Learning Research*, 8:2233–2264, 2007.

Denevi, G., Ciliberto, C., Stamos, D., and Pontil, M. Incremental learning-to-learn with statistical guarantees. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2018a.

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

Denevi, G., Ciliberto, C., Grazzi, R., and Pontil, M. Learning-to-learn stochastic gradient descent with biased regularization. arXiv, 2019.

Evgeniou, T. and Pontil, M. Regularized multi-task learning. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.

Fellbaum, C. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

Finn, C. and Levine, S. Meta-learning and universality: Deep representations and gradient descent can approximate any learning algorithm. In *Proceedings of the 6th International Conference on Learning Representations*, 2018.

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*, 2017.

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

Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., and Pontil, M. Bilevel programming for hyperparameter optimization and meta-learning. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

Frank, M. and Wolfe, P. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3, 1956.

Freedman, D. A. On tail probabilities for martingales. *The Annals of Probability*, 3(1):100–118, 1975.

Garber, D. Fast rates for online gradient descent without strong convexity via Hoffman's bound. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, 2019.

Grant, E., Finn, C., Levine, S., Darrell, T., and Griffiths, T. Recasting gradient-baed meta-learning as hierarchical Bayes. In *Proceedings of the 6th International Conference on Learning Representations*, 2018.

Hazan, E. Introduction to online convex optimization. In *Foundations and Trends in Optimization*, volume 2, pp. 157–325. now Publishers Inc., 2015.

Jerfel, G., Grant, E., Griffiths, T. L., and Heller, K. Online gradient-based mixtures for transfer modulation in meta-learning. arXiv, 2018.

Kakade, S. and Shalev-Shwartz, S. Mind the duality gap: Logarithmic regret algorithms for online optimization. In *Advances in Neural Information Processing Systems*, 2008.

Karimi, H., Nutini, J., and Schmidt, M. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2016.

Kim, J., Lee, S., Kim, S., Cha, M., Lee, J. K., Choi, Y., Choi, Y., Choi, D.-Y., and Kim, J. Auto-Meta: Automated gradient based meta learner search. arXiv, 2018.

Kuzborskij, I. and Orabona, F. Stability and hypothesis transfer learning. In *Proceedings of the 30th International Conference on Machine Learning*, 2013.

Lake, B. M., Salakhutdinov, R., Gross, J., and Tenenbaum, J. B. One shot learning of simple visual concepts. In *Proceedings of the Conference of the Cognitive Science Society (CogSci)*, 2017.

Li, Z., Zhou, F., Chen, F., and Li, H. Meta-SGD: Learning to learning quickly for few-shot learning. arXiv, 2017.

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

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

Pennington, J., Socher, R., and Manning, C. D. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 2014.

Pentina, A. and Lampert, C. H. A PAC-Bayesian bound for lifelong learning. In *Proceedings of the 31st International Conference on Machine Learning*, 2014.

Polyak, B. T. Gradient methods for minimizing functionals. *USSR Computational Mathematics and Mathematical Physics*, 3(3):864–878, 1963.

Ravi, S. and Larochelle, H. Optimization as a model for few-shot learning. In *Proceedings of the 5th International Conference on Learning Representations*, 2017.

Ruvolo, P. and Eaton, E. ELLA: An efficient lifelong learning algorithm. In *Proceedings of the 30th International Conference on Machine Learning*, 2013.

Shalev-Shwartz, S. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107—-194, 2011.

Snell, J., Swersky, K., and Zemel, R. S. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, 2017.

Thrun, S. and Pratt, L. *Learning to Learn*. Springer Science & Business Media, 1998.

Wang, S. and Manning, C. D. Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, 2012.

Zinkevich, M. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning*, 2003.