Adaptive Gradient-Based Meta-Learning Methods

Anonymous Authors[1]

Abstract

We build a theoretical framework for understanding practical meta-learning methods that enables the integration of sophisticated formalizations of task-similarity with the extensive literature on online convex optimization and sequential prediction algorithms in order to provide within-task performance guarantees. Our approach improves upon recent analyses of parameter-transfer by enabling the task-similarity to be learned adaptively and by improving transfer-risk bounds in the setting of statistical learning-to-learn. It also leads to straightforward derivations of average-case regret bounds for efficient algorithms in settings where the task-environment changes dynamically or the tasks share a certain geometric structure.

1. Introduction

Meta-learning, or *learning-to-learn* (LTL) (Thrun & Pratt, 1998), has recently re-emerged as an important direction for developing algorithms capable of performing well in multitask learning, changing environments, and federated settings. By using the data of numerous training tasks, meta-learning algorithms seek to perform well on new, potentially related test tasks without using many samples from them. Successful modern approaches have also focused on exploiting the capacity of deep neural networks, whether by learning multi-task data representations passed to simple classifiers (Snell et al., 2017) or by neural control of the optimization algorithms themselves (Ravi & Larochelle, 2017).

Because of its simplicity and flexibility, a common approach is that of *parameter-transfer*, in which all tasks use the same class of Θ-parameterized functions $f_\theta : \mathcal{X} \mapsto \mathcal{Y}$; usually a shared global model $\phi \in \Theta$ is learned that can then be used to train task-specific parameters. In *gradient-based meta-learning* (GBML) (Finn et al., 2017), ϕ is a meta-initialization such that a few stochastic gradient steps on a few samples from a new task suffice to learn a good task-specific model. GBML is now used in a variety of LTL domains such as vision (Li et al., 2017; Nichol et al., 2018; Kim et al., 2018), federated learning (Chen et al., 2018), and robotics (Al-Shedivat et al., 2018). However, its simplicity also raises many practical and theoretical questions concerning what task-relationships it is able to exploit and in which settings it may be expected to succeed.

While theoretical LTL has a long history (Baxter, 2000; Maurer, 2005; Pentina & Lampert, 2014), there has recently been an effort to understand GBML in particular. This has naturally lead to online convex optimization (OCO) (Zinkevich, 2003), either directly (Finn et al., 2019; Khodak et al., 2019) or via online-to-batch conversion to statistical LTL (Khodak et al., 2019; Denevi et al., 2019). These efforts all consider learning a shared initialization of a descent method; Finn et al. (2019) then prove learnability of a meta-learning algorithm while Khodak et al. (2019) and Denevi et al. (2019) give meta-test-time performance guarantees.

However, this line of work has so far considered at most a very restricted, if natural, notion of task-similarity – closeness to a single fixed point in the parameter space. We introduce a new theoretical framework, **Averaged-Regret Upper-Bound Analysis (ARUBA)**, that enables the derivation of meta-learning algorithms that can provably take advantage of much more sophisticated task-structure. Expanding significantly upon the work of Khodak et al. (2019), ARUBA treats meta-learning as the online learning of a sequence of losses that each upper bound the regret on a single task. These bounds frequently have convenient functional forms that are (a) nice enough for us to easily draw on the existing OCO literature and (b) strongly dependent on both the task-data and the meta-initialization, thus encoding task-similarity in a mathematically accessible way. Using ARUBA we provide new or dramatically improved meta-learning algorithms in the following settings:

- **Adaptive Meta-Learning:** A major drawback of previous work is the reliance on knowing the task-similarity beforehand to set the learning rate (Finn et al., 2019) or regularization (Denevi et al., 2019), or the use of a sub-optimal guess-and-tune approach based on the doubling trick (Khodak et al., 2019). ARUBA yields a simple and efficient gradient-based algorithm that eliminates the need to guess the task-similarity by learning it on-the-fly.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

- **Statistical LTL:** ARUBA allows us to leverage powerful results in online-to-batch conversion (Zhang, 2005; Kakade & Tewari, 2008) to derive new upper-bounds on the transfer risk when using GBML for statistical LTL (Baxter, 2000), including fast rates in the number of tasks when the task-similarity is known and fully high-probability guarantees for a class of losses that includes linear regression. These results improve directly upon the guarantees of Khodak et al. (2019) and Denevi et al. (2019) for similar or identical GBML algorithms.
- **LTL in Dynamic Environments:** Many practical applications of GBML include settings where the optimal initialization may change over time due to a changing task-environment (Al-Shedivat et al., 2018). However, current theoretical work on GBML has only considered learning a fixed initialization (Finn et al., 2019; Denevi et al., 2019). ARUBA reduces the problem of meta-learning in changing environments to a dynamic regret-minimization problem, for which there exists a vast array of online algorithms with provable guarantees.
- **Meta-Learning the Task Geometry:** A recurring theme in parameter-transfer LTL is the idea that certain model weights, such as those encoding a shared representation, are common to all tasks, whereas others, such as those performing a task-specific classification, need to be updated on each one. However, by simply using a fixed initialization we are forced to re-learn this structure on every task. Using ARUBA we provide an algorithm that can learn and take advantage of such structure by adaptively determining which directions in parameter-space need to be updated. We further provide a fully adaptive, per-coordinate variant that may be viewed as an analog for Reptile (Nichol et al., 2018) of the Meta-SGD modification of MAML (Finn et al., 2017; Li et al., 2017), which learns a per-coordinate learning rate; in addition to its provable guarantees, our version is more efficient and can be applied to a variety of GBML methods.

In the current paper we provide in Section 2 an introduction to ARUBA and use it to show guarantees for adaptive and statistical LTL. We defer our theory for meta-learning in dynamic environments and of different task-geometries, as well as our empirical results, to the full version of the paper.

1.1. Related Work

Theoretical Learning-to-Learn: The statistical analysis of LTL as learning over a task-distribution was formalized by Baxter (2000) and expanded upon by Maurer (2005). Recently, several works have built upon this theory to understand modern LTL, either from a PAC-Bayesian perspective (Amit & Meir, 2018) or in the ridge regression setting with a learned kernel (Denevi et al., 2018). However, due to the nature of the data, tasks, and algorithms involved, much effort has been devoted to the online setting, often through the framework of lifelong learning (Pentina & Lampert, 2014; Balcan et al., 2015; Alquier et al., 2017). The latter work considers a many-task notion of regret similar to our own in order to learn a shared data representations, although our algorithms are significantly more practical. Very recently, Bullins et al. (2019) also developed a more efficient online approach to learning a linear embedding of the data. However, such work is related to popular shared-representation methods such as ProtoNets (Snell et al., 2017), whereas we consider the parameter-transfer setting of GBML.

Gradient-Based Meta-Learning: GBML developed from the model-agnostic meta-learning (MAML) algorithm of Finn et al. (2017) and has been widely used in practice (Li et al., 2017; Al-Shedivat et al., 2018; Nichol et al., 2018; Jerfel et al., 2018). An expressivity result was shown for MAML by Finn & Levine (2018), proving that the meta-learner could approximate any permutation-invariant learning algorithm given enough data and a specific neural network architecture. Under strong-convexity and smoothness assumptions and using a fixed learning rate, Finn et al. (2019) show that the MAML meta-initialization is learnable, albeit via a somewhat impractical Follow-the-Leader (FTL) method. In contrast to these efforts, Khodak et al. (2019) and Denevi et al. (2019) focus on providing finite-sample meta-test-time performance guarantees in the convex setting, the former for the SGD-based Reptile algorithm of Nichol et al. (2018) and the latter for a more strongly-regularized variant. Our work improves upon these analyses by considering the case when the learning rate, a proxy for the task-similarity, is not known beforehand as in Finn et al. (2019) and Denevi et al. (2019) but must be learned online; Khodak et al. (2019) do consider an unknown task-similarity but use a rough doubling-trick-based approach that considers the absolute deviation of the task-parameters from the meta-initialization and is thus average-case suboptimal and sensitive to outliers. Furthermore, ARUBA can handle more sophisticated and dynamic notions of task-similarity and in certain settings can provide better statistical guarantees than those of Khodak et al. (2019) and Denevi et al. (2019).

2. Averaged-Regret Upper-Bound Analysis

Following the setup of Alquier et al. (2017), we consider a sequence of tasks $t = 1, \dots, T$; each task has rounds $i = 1, \dots, m$, on each of which we see a loss function $\ell_{t,i} : \Theta \mapsto \mathbb{R}$ for $\Theta \subset \mathbb{R}^d$. In the online setting, our goal will be to design algorithms taking actions $\theta_{t,i} \in \Theta$ that result in small **task-averaged regret** (TAR) (Khodak et al., 2019), which averages the within-task regret over $t \in [T]$:

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

This quantity measures within-task performance by dynamically comparing to the best action on individual tasks.

A common approach in this setting is to run an online algorithm, such as online gradient descent (OGD) with learning rate $\eta_t > 0$ and initialization $\phi_t \in \Theta$, on each task t:

$$\theta_{t,i+1} = \underset{\theta\in\Theta}{\arg\min}\ \frac{1}{2}\|\theta - \phi_t\|_2^2 + \eta_t \sum_{t=1}^{T} \langle \nabla \ell_{t,i}(\theta_{t,i}), \theta\rangle$$

The meta-learning problem is then reduced to determining which learning rate and initialization to use on each task t. Specific cases of this setup include the Reptile method of Nichol et al. (2018) and the algorithms in several recent theoretical analyses (Alquier et al., 2017; Khodak et al., 2019; Denevi et al., 2019). The observation that enables the results in the current paper is the fact that the online algorithms of interest in few-shot learning and meta-learning often have existing regret guarantees that depend strongly on both the parameters and the data; for example, the within-task regret of OGD for G-Lipschitz convex losses is

$$\mathbf{R}_t = \sum_{i=1}^{m} \ell_{t,i} - \ell_{t,i}(\theta_t^*) \le \frac{1}{2\eta_t}\|\theta_t^* - \phi_t\|_2^2 + \eta_t G m$$

for θ_t^* the optimal parameter in hindsight. Whereas more sophisticated adaptive methods for online learning attempt to reduce this dependence on initialization, in our setting each task does not have enough data to do so. Instead we can observe that if the upper bound $\hat{\mathbf{R}}_t(\phi_t, \eta_t) \ge \mathbf{R}_t$ on the task-t regret is low on average over $t \in [T]$ then the TAR of the actions $\theta_{t,i}$ due to running OGD initialized at ϕ_t with learning rate η_t at each task t will also be low, i.e.

$$\hat{\bar{\mathbf{R}}} = \frac{1}{T}\sum_{t=1}^{T} \hat{\mathbf{R}}_t(\phi_t, \eta_t) \ge \sum_{t=1}^{T}\sum_{i=1}^{m} \ell_{t,i}(\theta_{t,i}) - \ell_{t,i}(\theta_t^*) = \bar{\mathbf{R}}$$

Often this upper-bound $\hat{\mathbf{R}}_t$ will have a nice functional form; for example, the OGD bound above is jointly convex in the learning rate η_t and the initialization ϕ_t. Then standard OCO results can be applied directly.

While this approach was taken implicitly by Khodak et al. (2019), and indeed is related to earlier work on adaptive bound optimization for online learning (McMahan & Streeter, 2010), in this work we make explicit this framework, which we call **Averaged-Regret Upper-Bound Analysis (ARUBA)**, and showcase its usefulness in deriving a variety of new results in both online and batch LTL. Specifically, our approach will reduce LTL to the online learning of a sequence of regret upper-bounds $\hat{\mathbf{R}}_1(x), \dots, \hat{\mathbf{R}}_T(x)$, where x parameterizes the within-task algorithms. The resulting guarantees will then have the generic form

$$\bar{\mathbf{R}} \le \hat{\bar{\mathbf{R}}} \le o_T(1) + \min_x \frac{1}{T}\sum_{t=1}^{T} \hat{\mathbf{R}}_t(x)$$

Thus as $T \to \infty$ the algorithm competes with the best parameterization x, which encodes the task-relatedness through the task-data-dependence of $\hat{\mathbf{R}}_t$.

Algorithm 1: General form of meta-learning algorithm we study. $\text{TASK}_{\eta,\phi}$ corresponds to online mirror descent (OMD) or follow-the-regularized-leader (FTRL) with initialization $\phi \in \Theta$, learning rate $\eta > 0$, and regularization $R : \Theta \mapsto \mathbb{R}$. $\text{META}^{(1)}$ is follow-the-leader (FTL). $\text{META}^{(2)}$ is some OCO algorithm.

Set meta-initialization $\phi_1 \in \Theta$ and learning rate $\eta_1 > 0$.
for task $t \in [T]$ **do**
  for round $i \in [m_t]$ **do**
    $\theta_{t,i} \leftarrow \text{TASK}_{\eta_t,\phi_t}(\ell_{t,1}, \dots, \ell_{t,i-1})$
    Suffer loss $\ell_{t,i}(\theta_{t,i})$
  $\phi_{t+1} \leftarrow \text{META}^{(1)}(\mathcal{B}_R(\theta_1^*||\cdot), \dots, \mathcal{B}_R(\theta_t^*||\cdot))$
  $D_{t+1} \leftarrow \text{META}^{(2)}(f_1, \dots, f_t)$
    for $f_t(x) = \frac{\mathcal{B}_R(\theta_t^*||\phi_t)}{x} + x$
  $\eta_{t+1} = \frac{D_{t+1}}{G\sqrt{m}}$

2.1. Adaptive Task-Similarity Learning

Our first result is an adaptive algorithm for a simple notion of task-similarity that serves also to demonstrate how our framework may be applied. We consider tasks $t = 1, \dots, T$ whose optimal actions θ_t^* are close to some unknown global $\phi^* \in \Theta$ according to some metric. For ℓ_2-distance this assumption was made, explicitly or implicitly, by Finn et al. (2019) and Denevi et al. (2019); Khodak et al. (2019) also consider the case of a Bregman divergence $\mathcal{B}_R(\theta_t^*||\phi^*)$ for 1-strongly-convex $R : \Theta \mapsto \mathbb{R}$ (Bregman, 1967), with $R(\cdot) = \frac{1}{2}\|\cdot\|_2^2$ recovering $\mathcal{B}_R(\theta||\phi) = \frac{1}{2}\|\theta - \phi\|_2^2$. However, their methods were not adaptive to the strength of the task similarity, i.e. the average deviation $V = \sqrt{\frac{1}{T}\sum_{t=1}^{T} \mathcal{B}_R(\theta_t^*||\phi^*)}$ of the task-parameters; for OCO methods V is proportional to the learning rate or the inverse of the regularization coefficient, which were fixed by Finn et al. (2019) and Denevi et al. (2019). Khodak et al. (2019) instead used the doubling trick to learn the maximum deviation $\max_t \sqrt{\mathcal{B}_R(\theta_t^*||\phi^*)} \ge V$, which is suboptimal and extremely sensitive to outliers.

We first formalize the setting we consider, extensions of which will also be used for later results:

Setting 2.1. *Each task $t \in [T]$ has m convex loss functions $\ell_{t,i}\Theta \mapsto \mathbb{R}$ that are G-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.*

We will consider variants of Algorithm 1, in which a parameterized OCO method $\text{TASK}_{\eta,\phi}$ is run within-task and two OCO methods, $\text{META}^{(1)}$ and $\text{META}^{(2)}$, are run in the outer loop to determine the learning rate $\eta > 0$ and initialization $\phi \in \Theta$. We provide the following guarantee:

Theorem 2.1. *In Setting 2.1 Algorithm 1 achieves TAR*

$$\bar{\mathbf{R}} \leq \frac{\mathbf{R}_T}{T} + \min_{\substack{0<V\leq D\\ \phi\in\Theta}} \frac{G\sqrt{m}}{VT}\left(\mathcal{O}(\log T) + \sum_{t=1}^T \mathcal{B}_R(\theta_t^*||\phi) + V^2\right)$$

where $D^2 = \max_t \mathcal{B}_R(\theta_t^*||\phi_t)$ *and* $\mathbf{R}_T$ *is the regret of* $\text{META}^{(2)}$ *on a sequence* $f_1, \dots, f_T$ *of functions of form* $f_t(x) = \frac{B_t^2}{x} + x$ *for* $B_t \leq D\ \forall\, t$.

Proof Sketch. The proof follows from the well-known regret $\hat{\mathbf{R}}_t = \frac{1}{\eta_t}\mathcal{B}_R(\theta_t^*||\phi_t) + \eta_t G^2 m$ of FTRL and OMD (Shalev-Shwartz, 2011). Summing this upper-bound over tasks $t \in [T]$, apply the regret of $\text{META}^{(2)}$ to replace η_t by $\frac{V}{G\sqrt{m}}$ at the cost of the $\frac{\mathbf{R}_T}{T}$ term. Conclude by applying the regret of $\text{META}^{(1)}$, i.e. FTL, over $\mathcal{B}_R(\theta_t^*||\cdot)$; this is well-known to be logarithmic for the case $R(\cdot) = \frac{1}{2}\|\cdot\|_2^2$ (Shalev-Shwartz, 2011) and can be shown to be so in general by a novel strongly-convex coupling argument. □

It remains to show a low-regret algorithm $\text{META}^{(2)}$ for the sequence $f_t(x) = \frac{B_t^2}{x} + x$. This is nontrivial, as while the functions are convex they are non-Lipschitz near 0. However, using strongly-convex coupling once more one can show that using the actions of FTL on the modified loss functions $\tilde{f}_t(x) = \frac{B_t^2+\varepsilon^2}{x} + x$ will achieve regret $\tilde{\mathcal{O}}(\min\{T^{\frac{3}{5}}/V, T^{\frac{4}{5}}\})$ for $\varepsilon = T^{-\frac{1}{5}}$ for the original sequence. This can be improved to $\tilde{\mathcal{O}}(\min\{\sqrt{T}/V, T^{\frac{3}{4}}\})$ for $\varepsilon = T^{-\frac{1}{4}}$ by proving the exp-concavity of $\tilde{f}_t$ and using the Exponentially-Weighted Online Optimization (EWOO) algorithm of Hazan et al. (2007), which can be implemented efficiently in this single-dimensional case, instead of FTL. We thus have the following corollary:

Corollary 2.1. *Algorithm 1 with* $\text{META}^{(2)} = \text{FTL}$ *achieves TAR*

$$\bar{\mathbf{R}} \leq \min_{\substack{0<V\leq D\\ \phi\in\Theta}} \frac{G\sqrt{m}}{VT}\left(o(T) + \sum_{t=1}^T \mathcal{B}_R(\theta_t^*||\phi) + V^2\right)$$

For $\phi = \frac{1}{T}\sum_{t=1}^T \theta_t^*$ and $V^2 = \frac{1}{T}\sum_{t=1}^T \mathcal{B}_R(\theta_t^*||\phi)$, i.e. the mean and squared average deviation of the optimal task parameters, we have an asymptotic per-task regret of $VG\sqrt{m}$, which is much better than the minimax-optimal single-task guarantee $DG\sqrt{m}$ when $V \ll D$, i.e. when the tasks are on-average close in parameter space. As in Khodak et al. (2019) and assuming a quadratic growth condition on each task, in the full version we extend this result to the case when θ_t^* is not known and either the last or average within-task iterate is used to perform the meta-updates.

2.2. Improved Rates for Statistical Learning-to-Learn

An important motivation for studying LTL via online learning has been to provide batch-setting bounds on the transfer risk (Alquier et al., 2017; Denevi et al., 2019). While Khodak et al. (2019) provide an in-expectation bound on the expected transfer risk of any low-TAR algorithm, their result cannot exploit the many stronger results in the online-to-batch conversion literature. Following the classical distribution over task-distributions setup of Baxter (2000), ARUBA yields strong bounds on the expected transfer risk in the general case of convex $\hat{\mathbf{R}}$, as well as fast rates in the strongly-convex case using Kakade & Tewari (2008) and high probability bounds for linear regression using Zhang (2005).

Theorem 2.2. *Let convex losses* $\ell_{t,i} : \Theta \mapsto [0,1]$ *be sampled i.i.d.* $\mathcal{P}_t \sim \mathcal{Q}, \{\ell_{t,i}\}_i \sim \mathcal{P}_t^m$ *for some distribution* $\mathcal{Q}$ *over task distributions* $\mathcal{P}_t$. *If the losses are given to an algorithm with averaged regret upper-bound* $\hat{\bar{\mathbf{R}}}_T$ *that on each task runs an algorithm with regret upper-bound* $\hat{\mathbf{R}}_t(s_t)$ *a convex, nonnegative, and* $B\sqrt{m}$*-bounded function of the state* s_t *of the algorithm at the beginning of time* t *then we have the following bound on the expected transfer risk:*

$$\mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\mathcal{P}^m}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}} \ell(\bar{\theta}) \leq \mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}} \ell(\theta^*) + \mathcal{L}_T$$

w.p. $1-\delta$, *where* $\bar{\theta} = \frac{1}{m}\sum_{i=1}^m \theta_i$ *is generated by running the task-algorithm with state* $\bar{s} = \frac{1}{T}\sum_{t=1}^T s_{1:T}$ *and averaging the task-actions* $\{\theta_t\}_{i\in[m]}$ *and where for the general case we have* $\mathcal{L}_T = \frac{\hat{\bar{\mathbf{R}}}}{m} + B\sqrt{\frac{8}{mT}\log\frac{1}{\delta}}$. *If the regret upper-bounds are* α*-strongly-convex then this term is instead*

$$\mathcal{L}_T = \frac{\hat{\bar{\mathbf{R}}} + \min_s \mathbb{E}_{\mathcal{P}\sim\mathcal{Q}}\hat{\mathbf{R}}(s)}{m} + \frac{4G}{T}\sqrt{\frac{\hat{\bar{\mathbf{R}}}}{\alpha m}\log\frac{8\log T}{\delta}} + \frac{\max\{16G^2, 6\alpha B\sqrt{m}\}}{\alpha m T}\log\frac{8\log T}{\delta}$$

If the losses ℓ *satisfy a certain self-bounding property then we have a high probability bound on the transfer risk itself:*

$$\mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}} \ell(\bar{\theta}) \leq \mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}} \ell(\theta^*) + \mathcal{L}_T + \sqrt{\frac{2\rho\mathcal{L}_T}{m}\log\frac{2}{\delta}} + \frac{3\rho+2}{m}\log\frac{2}{\delta}$$

w.p. $1-\delta$ *for some* $\rho > 0$.

In the case of a known task-similarity, when we know the expected task-parameter deviation V and can fix the learning rate in Algorithm 1 accordingly, the above result yields

$$\mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\mathcal{P}^m}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}} \ell(\bar{\theta}) \leq \mathop{\mathbb{E}}_{\mathcal{P}\sim\mathcal{Q}}\mathop{\mathbb{E}}_{\ell\sim\mathcal{P}} \ell(\theta^*) + \mathcal{O}\left(\frac{V}{\sqrt{m}} + \frac{\log\frac{T}{\delta}}{T\sqrt{m}}\right)$$

This can be compared to results of Denevi et al. (2019), where the last term only decreases as $T^{-\frac{1}{2}}$. Note that their results are fully in-expectation only and the V corresponds to the expected deviation of true-risk minimizers.

References

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.

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

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.

Bullins, B., Hazan, E., Kalai, A., and Livni, R. Generalize across tasks: Efficient algorithms for linear representation learning. In *Proceedings of the 30th International Conference on Algorithmic Learning Theory*, 2019.

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

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*, 2018.

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

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. In *Proceedings of the 36th International Conference on Machine Learning*, 2019. To Appear.

Hazan, E., Agarwal, A., and Kale, S. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69:169–192, 2007.

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 Tewari, A. On the generalization ability of online strongly convex programming algorithm. In *Advances in Neural Information Processing Systems*, 2008.

Khodak, M., Balcan, M.-F., and Talwalkar, A. Provable guarantees for gradient-based meta-learning. In *Proceedings of the 36th International Conference on Machine Learning*, 2019. To Appear.

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.

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.

McMahan, H. B. and Streeter, M. Adaptive bound optimization for online convex optimization. In *Proceedings of the Conference on Learning Theory*, 2010.

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

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

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

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.

Zhang, T. Data dependent concentration bounds for sequential prediction algorithms. In *Proceedings of the International Conference on Learning Theory*, 2005.

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