Meta Learning in the Continuous Time Limit

Ruitu Xu[1] **Lin Chen**[1,2] **Amin Karbasi**[1]
[1] Yale University [2]Simons Institute for the Theory of Computing

Abstract

In this paper, we establish the ordinary differential equation (ODE) that underlies the training dynamics of Model-Agnostic Meta-Learning (MAML). Our continuous-time limit view of the process eliminates the influence of the manually chosen step size of gradient descent and includes the existing gradient descent training algorithm as a special case that results from a specific discretization. We show that the MAML ODE enjoys a linear convergence rate to an approximate stationary point of the MAML loss function for strongly convex task losses, even when the corresponding MAML loss is non-convex. Moreover, through the analysis of the MAML ODE, we propose a new BI-MAML training algorithm that reduces the computational burden associated with existing MAML training methods, and empirical experiments are performed to showcase the superiority of our proposed methods in the rate of convergence with respect to the vanilla MAML algorithm.

1 Introduction

In machine learning, an ideal learner is able to speed up the learning of new tasks based upon previous experiences. This goal has been shared among different but highly related approaches such as few-shot learning (Snell et al., 2017), domain adaptation (Yu et al., 2018; Li et al., 2018), transfer learning (Pan & Yang, 2009), and meta learning (*a.k.a. learning to learn*) (Thrun & Pratt, 1998; Naik & Mammone, 1992; Schmidhuber, 1987; Hochreiter et al., 2001). In particular, meta-learning addresses a general optimization framework that aims to learn a model based on data from previous tasks, so that the learned model, after fine-tuning, can adapt to and perform well on new tasks. This idea has been successfully applied to different learning scenarios including reinforcement learning (Wang et al., 2016; Duan et al., 2016; Schweighofer & Doya, 2003), deep probabilistic models (Edwards & Storkey, 2016; Lacoste et al., 2017; Grant et al., 2018), language models (Radford et al., 2019), imitation learning (Duan et al., 2017), and unsupervised learning (Hsu et al., 2019), to name a few. Previous works have also investigated meta learning from a variety of perspectives and methods, including memory-based neural networks (Santoro et al., 2016; Munkhdalai & Yu, 2017), black-box optimization (Duan et al., 2016; Wang & Hebert, 2016), learning to design an optimization algorithm (Andrychowicz et al., 2016), attention-based models (Vinyals et al., 2016), and LSTM-based learners (Ravi & Larochelle, 2016).

One of the gradient-based meta-learning algorithms that has been widely used and enjoys great empirical successes is the *model-agnostic meta-learning* proposed in (Finn et al., 2017). Rather than minimizing directly on a combination of task losses, MAML meta-trains by minimizing the loss evaluated at one or multiple gradient descent steps further ahead for each task. The intuition is as follows: First, this meta-trained initialization resides close to the best parameters of all training tasks. Second, for each new task, the model can then be easily fine-tuned for that specific new task with only a small number of gradient descent steps. Following this work, many experimental and theoretical studies of MAML have been carried out (Song et al., 2019; Behl et al., 2019; Grant et al., 2018; Yang et al., 2019; Rajeswaran et al., 2019). In particular, Fallah et al. (2019) and Mendonca et al. (2019) investigated the convergence of MAML to first order stationary points. Moreover, Finn et al. (2019) studied an online variant of MAML and Antoniou et al. (2018) proposed various modifications to MAML to improve its performance. Empirically, Raghu et al. (2019) found that the success of MAML can be primarily associated to feature reuse, given high quality representations provided by the meta-initialization.

MAML assumes a shared parameter space $\mathbb{R}^d$ among all the tasks and learns an initialization $\widehat{w} \in \mathbb{R}^d$ from a batch of tasks at *meta-learning* time, such that

Proceedings of the 24[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

performing a few steps of gradient descent from this initialization at *meta-testing* time minimizes the loss function for a new task on a smaller dataset. More specifically, in a task pool with total M candidate tasks, each task $\mathcal{T}_i$ is sampled from the pool according to a distribution $p(\mathcal{T})$ and has a corresponding risk function $f_i : \mathbb{R}^d \to \mathbb{R}$ that is parameterized by a shared variable $w \in \mathbb{R}^d$. This function measures the performance of the parameter w on task $\mathcal{T}_i$. In practice we estimate this risk function from a set of training data for each task. However, for simplicity, we directly work with the risk function in this paper and assume that we can acquire the knowledge of the task loss f_i from an oracle. At meta-learning time, MAML looks for a warm start $\widehat{w}$ via solving an optimization problem that minimizes the expected loss over f_i after one step of gradient descent, *i.e.*,

$$\widehat{w} = \arg\min_{w} \mathbb{E}_{i\sim p}[f_i(w - \alpha\nabla f_i(w))], \tag{1.1}$$

where $\mathbb{E}_{i\sim p}$ denotes the expectation over sampled tasks and α represents the MAML step size (also referred to as the MAML parameter). To ease the burden on notation, we define the MAML loss function of task $\mathcal{T}_i$ to be $F_i(w) \coloneqq f_i(w - \alpha\nabla f_i(w))$. Further, we define some shorthand for the expected loss $f(w) \coloneqq \mathbb{E}_{i\sim p} f_i(w)$ and the expected MAML loss $F(w) \coloneqq \mathbb{E}_{i\sim p} F_i(w)$. Now we are able to represent the optimization problem (1.1) in a more concise form: $\arg\min_w F(w)$.

Solving (1.1) yields a solution $\widehat{w}$ that in expectation serves as a good initialization point, such that after a step of gradient descent it achieves the best average performance over all possible tasks. However, because the objective function $F(w)$ is non-convex in general, there is no guarantee that one is able to find a global minimum. Hence it is common to consider instead an approximate stationary point $\widehat{w}$ where $\|\nabla F(\widehat{w})\| \le \varepsilon$ for some small ε (Fallah et al., 2019).

In this paper, we study the MAML algorithm proposed in (Finn et al., 2017), in which MAML meta-learns the model by performing gradient descent on the MAML loss. We begin by establishing a smooth approximation of this discrete-time iterative procedure and considering the continuous-time limit by taking an infinitesimally small step size. With any initialization of the parameter, the problem is transformed into an *initial value problem* (IVP) of an ODE, which is the underlying dynamic that governs the training of a MAML model. Specifically, We consider the convergence of the gradient norm $\|\nabla F(w)\|$ and the function value $F(w)$ under the continuous-time limit of MAML.

Our contributions are summarized as follows:

- We propose an ODE that underlies the training dynamics of a MAML model. In this manner, we eliminate the influence of the manually chosen step size. The algorithm in (Finn et al., 2017) can be viewed as an forward Euler integration of this ODE.
- We prove that the aforementioned ODE enjoys a linear convergence rate to an approximate stationary point of F for strongly convex task losses. In particular, it achieves an improved convergence rate of $\mathcal{O}(\log\frac{1}{\varepsilon})$, compared to $\mathcal{O}(\frac{1}{\varepsilon^2})$ in (Fallah et al., 2019). Note that the MAML loss F may not be convex even if the individual task losses f_i are strongly convex.
- We show that for strongly convex task losses with moderate regularization conditions, the MAML loss F has a unique critical point that is also the global minimum.
- We propose a new algorithm named BI-MAML for strongly convex task losses that is computationally efficient and enjoys a similar linear convergence rate to the global minimum compared to the original MAML. It also converges significantly faster than MAML on a variety of learning tasks.

2 Preliminaries

Before turning to the discussion of the continuous-time limit of MAML, we briefly introduce a widely-used approach for taking the continuous-time limit of discrete-time algorithms and the approach we use later for its analysis.

Optimization through lens of ODE. There is an extensive literature on the topic of understanding discrete-time algorithms through the lens of ODEs (Schropp & Singer, 2000; Helmke & Moore, 2012; Lu, 2020), and recent developments in this field offer novel perspectives for looking at discrete-time optimization algorithms (Su et al., 2014; Muehlebach & Jordan, 2019). For example, Shi et al. (2019) developed a first-order optimization algorithm by performing discretization on ODEs that correspond to Nesterov's accelerated gradient descent. Krichene et al. (2015) proposed a family of continuous-time dynamics for convex functions where the corresponding solution converges to the optimum value at an optimal rate. However, there can be multiple ODEs that correspond to the same discrete-time algorithm, and it oftentimes requires strong mathematical intuitions when it comes to taking the continuous limit. In this paper, we take the most intuitive approach by letting the step size of the gradient descent on the MAML loss F go to zero, and the resulting ODE is a gradient flow on F.

Lyapunov's direct method. One of the most commonly used approaches for analyzing the convergence of ODEs is Lyapunov's direct method (Lyapunov, 1992;

Hirsch et al., 2012; Wilson et al., 2016), which is based on constructing a positive definite Lyapunov function $\mathcal{E} : \mathbb{R}^d \to \mathbb{R}$ that decreases along the trajectories of the dynamics $\dot{w}$:

$$\frac{d}{dt}\mathcal{E}(w(t)) = \langle \nabla\mathcal{E}(w(t)), \dot{w}(t)\rangle \leq 0. \tag{2.1}$$

This method is a generalization of the idea that measures the "energy" in a system, and the existence of such a continuously differentiable Lyapunov function guarantees the convergence of the dynamical system.

3 Main Results

In this paper, we analyze the MAML algorithm on strongly convex functions with its continuous-time limit and establish a linear convergence rate. We also propose a new algorithm named BI-MAML based on our theoretical analysis on MAML, which we will elaborate further in Section 4. Moreover, we will present empirical results in Section 5 to show that BI-MAML significantly outperforms vanilla MAML.

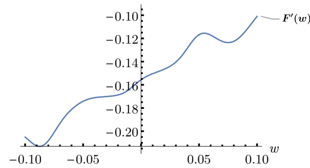
Both our new analyses on MAML and the new BI-MAML algorithm are based on our analysis of the landscape of F. Fig. 1 illustrates an example where F is indeed non-convex. This example has two tasks where we take $f_1(w) = 0.505w^2 - \sin(w)$ and $f_2(w) = 0.505w^2 - 0.0001\sin(100w)$. Both functions are 0.01-strongly convex and 2.01-smooth. We observe that $F'(w)$ is not monotone increasing and that $F''(w)$ is not always positive. These imply that F is non-convex. While F is non-convex in general as illustrated, we prove that F has a unique critical point and is strongly convex on a convex set around the critical point, which implies that it is the global minimum. Of course, the function F can be non-convex outside the convex set.

3.1 MAML Algorithm and ODE

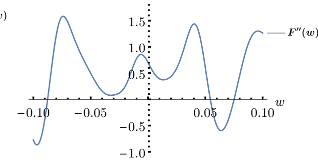
In this section, we present the original MAML. In particular, we investigate MAML under a continuous-time limit. Recall that the update rule of MAML on w follows a gradient descent on $F(w)$, *i.e.*,

$$\begin{aligned} w^+ &= w - \beta\nabla F(w) \\ &= w - \beta\,\mathbb{E}_{i\sim p}\,\nabla F_i(w) \\ &= w - \beta\,\mathbb{E}_{i\sim p}[A_i(w)\nabla f_i(w - \alpha\nabla f_i(w))], \end{aligned} \tag{3.1}$$

where w denotes the iterate input, w^+ denotes the iterate output, β represents the step size, and $A_i(w) := I_d - \alpha\nabla^2 f_i(w)$ is a shorthand for the Hessian correction term. Here we see that computing the gradient of F requires the Hessian of f. In this paper we consider MAML where the step size α of the task-specific gradient descent remains constant while the step size β for the MAML gradient descent goes to zero. Proposition 3.1 presents the continuous-time limit for MAML, which we term as the MAML ODE.



(a) Plot of $F'(w)$.



(b) Plot of $F''(w)$.

Figure 1: An example of non-convex MAML loss $F(w)$ even if its corresponding task losses satisfy all Assumptions 3.2 to 3.5. Here we let $f_1(w) = 0.505w^2 - \sin(w)$ and $f_2(w) = 0.505w^2 - 0.0001\sin(100w)$. Notice that both $f_1''(w) \geq 0.01$ and $f_2''(w) \geq 0.01$ for all $w \in \mathbb{R}^d$. Both functions are 0.01-strongly convex and 2.01-smooth. It is not hard to see also that Assumptions 3.4 and 3.5 are both satisfied for some finite σ and κ. Taking the MAML step size as $\alpha = 0.4$, we have a non-convex MAML loss F with its first- and second-order derivatives as indicated in Figs. 1a and 1b.

Proposition 3.1 (Proof in Appendix B). *If the losses f_i are twice differentiable, the continuous-time limit for MAML is*

$$\dot{w} = -\nabla f(w) + \mathbb{E}_{i\sim p}[B_i(w)\nabla f_i(w)] \tag{3.2}$$

where $B_i(w) := \alpha(\nabla^2 f_i(w) + \nabla^2 f_i(\tilde{w}_i)) - \alpha^2\nabla^2 f_i(w)\nabla^2 f_i(\tilde{w}_i)$ and $\tilde{w}_i$ is a convex combination of w and $w - \alpha\nabla f_i(w)$.

The first term on the right-hand side of (3.2) represents a gradient flow on f, and the second term that follows is the key term that differentiates a MAML gradient descent on the MAML loss function F from a vanilla gradient descent on the expected loss function f. Due to the compositional nature of the MAML loss $F(w)$, the second-order information is required to evaluate its gradient. Recall that by definition, we have $\nabla F(w) = \mathbb{E}_{i\sim p}[(I_d - \alpha\nabla^2 f_i(w))\nabla f_i(w - \alpha\nabla f_i(w))]$. To reduce such cost on computing the Hessian, Finn et al. (2017) proposed the first-order model-agnostic meta-learning (FO-MAML), which is a first-order approximation of MAML that replaces the second-order term $I_d - \alpha\nabla^2 f_i(w)$ with an identity matrix. However, this may cause a failure in convergence, as mentioned in (Fallah et al., 2019). In comparison, our new BI-MAML algorithm achieves a similar computational efficiency by evaluating only first-order information while at the same time still being able to converge to an approximate stationary point, as we will see in Section 4.

3.2 Summary of Assumptions, Results, and Techniques

In this section, we establish our theoretical results for MAML when the loss functions $f_i(w)$ are strongly convex and smooth with bounded gradient variance among the tasks. Formally, we make the following assumptions.

Assumption 3.2. For every $i \in [M]$, the loss $f_i(w)$ is twice differentiable and L_i-smooth, *i.e.*, for every $w, u \in \mathbb{R}^d$, we have $\|\nabla f_i(w) - \nabla f_i(u)\| \le L_i \|w-u\|$.

Assumption 3.3. For every $i \in [M]$, the loss $f_i(w)$ is μ_i-strongly convex, *i.e.*, for every $w, u \in \mathbb{R}^d$, there exists positive μ, such that $\|\nabla f_i(w) - \nabla f_i(u)\| \ge \mu_i \|w-u\|$.

Assumption 3.4. For any $w \in \mathbb{R}^d$, the variance of gradient $\nabla f_i(w)$ is bounded, *i.e.*, there exists non-negative σ, such that $\mathbb{E}_{i\sim p} \|\nabla f(w) - \nabla f_i(w)\|^2 \le \sigma^2$.

Assumption 3.5. For every $i \in [M]$, the Hessian for loss $f_i(w)$ is κ_i-Lipschitz continuous, *i.e.*, for every $w, u \in \mathbb{R}^d$, we have $\|\nabla^2 f_i(w) - \nabla^2 f_i(u)\| \le \kappa_i \|w-u\|$.

To simplify the notation, we denote $L := \max_i L_i$, $\mu := \min_i \mu_i$, and $\kappa := \max_i \kappa_i$ in the rest of the paper. Because f_i is twice differentiable, Assumption 3.2 is equivalent to $-L_i I_d \preceq \nabla^2 f_i(w) \preceq L_i I_d$. We note that Finn et al. (2019) assumed all the assumptions above, except Assumption 3.4 because they considered online meta-learning where functions can be selected in an adversarial manner. On the other hand, they assumed that the functions are Lipschitz (Finn et al., 2019, Assumption 1.1), which may contradict the strong convexity assumption (Finn et al., 2019, Assumption 2) in their paper. Similarly, Fallah et al. (2019) assumed all but Assumption 3.3. We remark that Assumption 3.3 implies the boundedness of the MAML loss F from below, but it does not guarantee the convexity of F. See Fig. 1 for an example of non-convex MAML loss F with corresponding task losses f_i satisfying Assumptions 3.2 to 3.5. In other words, while f_i are strongly convex, F can still be non-convex. Hence, minimizing F is challenging as we are dealing with a non-convex optimization problem.

As we mention above, we make an additional assumption (Assumption 3.3), compared to the set of assumptions in (Fallah et al., 2019). They showed that the MAML algorithm outputs a solution that guarantees $\|\nabla F(w)\| \le \varepsilon$ in $\mathcal{O}(\frac{1}{\varepsilon^2})$ iterations. Under this additional assumption, we significantly improve the result in two aspects. First, we show that our proposed algorithm finds a solution such that $\|\nabla F(w)\| \le \varepsilon$ in only $O(\log \frac{1}{\varepsilon})$ iterations (Theorem 3.6). This is indeed an exponential improvement on (Fallah et al., 2019) in terms of iteration complexity. Second, we characterize the landscape of F. While F is non-convex in general, we prove that its stationary point is also the global minimum (Theorem 3.9). Therefore, the solution returned by our algorithm is close to not only a critical point but also the global minimum.

Our main result in Theorem 3.6 shows that the MAML ODE achieves linear convergence in finding a critical point on the MAML loss F.

Theorem 3.6 (Iteration complexity, proof in Appendix C)**.** *Suppose the loss function $f_i(w)$ satisfies Assumptions 3.2 to 3.5, if*

$$\alpha < \min\left\{\frac{1}{2L}, \frac{7\mu^{3/2}}{288\kappa\sigma + 232\kappa\sqrt{\mu}\sigma}, \sqrt[3]{\frac{2}{15}}\mu^{1/3}L^{-5/3}, \sqrt{\frac{1}{15}}\mu^{1/2}L^{-2}, \sqrt{\frac{1}{15}}\mu L^{-2}\right\}$$

then the MAML ODE finds a solution $\widehat{w}$ such that $\|\nabla F(\widehat{w}(t))\| \le \varepsilon$ after at most running for

$$t = \mathcal{O}\left[\frac{1}{\mu}\log\left(\frac{(5+\frac{9}{\sqrt{\mu}})(\mu^2\sigma\|\nabla f(w(0))\|^2 - \frac{\mu\sigma^3}{2})}{4\iota\sigma^2\varepsilon}\right)\right],$$

if $\|\nabla f(w(0))\|^2 > \frac{\sigma^2}{\mu}$ and

$$t = \mathcal{O}\left[\frac{16}{\mu}\log\left(\frac{(5+\frac{9}{\sqrt{\mu}})\sigma}{4\varepsilon}\right)\right]$$

otherwise, where $\iota > 0$ is a small constant.

Theorem 3.6 says that if the MAML parameter α is small enough, then the MAML ODE finds an approximate stationary point of the MAML loss F in $\mathcal{O}(\log \frac{1}{\varepsilon})$ time. This approximate stationary point is at the same time an approximate global minimum, as implied by Theorem 3.9 later.

We prove Theorem 3.6 with a two-phase analysis where the transition between two phases depends on the norm of $\nabla F(w)$. In the first phase we conduct a Lyapunov function analysis on the Lyapunov candidate function $\|\nabla f(w)\|^2$ and under the dynamics defined by the ODEs. The second phase follows as a landscape characterization of the MAML loss F. Intuitively, $\|\nabla F(w)\|$ can be large at initialization and will become smaller over the course of the gradient flow. We note that $\|\nabla F(w)\|$ is close to $\|\nabla f(w)\|$ if α is small due to the fact that $\nabla F(w) = \mathbb{E}[(I_d - \alpha\nabla^2 f_i(w))\nabla f_i(w - \alpha\nabla f_i(w))]$, and we therefore choose to analyze the gradient norm on f instead of F. When $\|\nabla f(w)\|$ is large and we are far from the stationary point, the analysis on the Lyapunov function $\mathcal{E}(w(t)) = \|\nabla f(w(t))\|^2$ helps establish a linear convergence rate for $\|\nabla f(w)\|$ to be in the order of $\mathcal{O}(\sigma)$. This is due to the proof technique shown in Lemma 3.7.

Proof of Theorem 3.6

Proof. We first define $y(t) = (e^{-\zeta(t+c_0)} + \gamma)/\zeta$, where we denote $y(0) = \|\nabla f(w(0))\|^2$, $\gamma = \sigma^2/2$, and $\zeta = \mu - \frac{5}{4}L^2\alpha(L^3\alpha^2 + 2L^2\alpha + 2)$. Especially, under the assumptions of Theorem C.2, we have $\frac{\mu}{2} < \frac{\mu}{2} + \iota \leq \zeta \leq \mu$ where $\iota > 0$ is a small constant. To achieve $\|\nabla f(w(t))\|^2 \leq \frac{\sigma^2}{\mu}$, it suffices to have

$$y(t) = \frac{e^{-\zeta(t+c_0)} + \gamma}{\zeta} \leq \frac{\sigma^2}{\mu}. \tag{3.3}$$

Since at initialization $e^{-\zeta c_0} = \zeta y(0) - \gamma$, as long as $y(0) = \|\nabla f(w(0))\|^2 > \frac{\sigma^2}{\mu}$, plug into (3.3) to get

$$\begin{aligned} e^{-\zeta t} &\leq \left(\frac{\zeta\sigma^2}{\mu} - \gamma\right) e^{\zeta c_0} \\ t &\geq \frac{1}{\zeta}\log\left(\frac{\zeta y(0) - \frac{\sigma^2}{2}}{\frac{\zeta\sigma^2}{\mu} - \frac{\sigma^2}{2}}\right). \end{aligned}$$

Hence for any initialization $w(0)$ where $y(0) > \frac{\sigma^2}{\mu}$, we have $\|\nabla f(w(t))\|^2 \leq \frac{\sigma^2}{\mu}$ for any

$$\begin{aligned} t &\geq \frac{2}{\mu}\log\left(\frac{\mu y(0) - \frac{\sigma^2}{2}}{(\frac{\mu}{2} + \iota)\frac{\sigma^2}{\mu} - \frac{\sigma^2}{2}}\right) \\ &= \frac{2}{\mu}\log\left(\frac{\mu^2\|\nabla f(w(0))\|^2 - \frac{\mu\sigma^2}{2}}{\iota\sigma^2}\right). \end{aligned}$$

For any initialization $w(0)$ where $y(0) \leq \frac{\sigma^2}{\mu}$, we skip the first phase and go directly into the second phase.

Let us denote $t_1 = \min_t \{t : y(t) \leq \sigma^2/\mu\}$, and especially $t_1 = 0$ if $y(0) \leq \frac{\sigma^2}{\mu}$. Under the assumption $\alpha \leq \frac{1}{2L}$, we have

$$\begin{aligned} \|\nabla F(w(t_1))\| &\leq (1 + 2\alpha L + \alpha^2L^2)\frac{\sigma}{\sqrt{\mu}} + (2\alpha L + \alpha^2L^2)\sigma \\ &\leq \frac{1}{4}\left(\frac{9}{\sqrt{\mu}} + 5\right)\sigma. \end{aligned} \tag{3.4}$$

We prove (3.4) in Lemma C.3. Let us denote $K = \frac{1}{4}\big(\frac{9}{\sqrt{\mu}} + 5\big)\sigma$, and Theorem 3.8 implies that if $\alpha \leq \min\{\frac{1}{2L}, \frac{7\mu}{8\kappa(16K+9\sigma)}\}$ the MAML loss $F(w)$ is $\frac{\mu}{8}$-strongly convex at w, and the MAML ODE (3.2) after time t_1 is a gradient flow on a $\frac{\mu}{8}$-strongly convex loss $F(w)$. This dynamics then converges exponentially fast to an approximate stationary point $\widehat{w}$ where $\|\nabla F(\widehat{w})\| \leq \varepsilon$. More specifically, we have

$$\begin{aligned} \frac{d}{dt}\|\nabla F(w)\|^2 &= \nabla F(w)^\intercal \nabla^2 F(w)\dot{w} \\ &= -\nabla F(w)^\intercal \nabla^2 F(w)\nabla F(w) \\ &\leq -\frac{\mu}{8}\|\nabla F(w)\|^2. \end{aligned}$$

Note that even though the set of w such that $\|\nabla F(w)\| \leq K$ is not necessarily convex, the trajectory of the MAML ODE still converges inside it. Consider a point $w(t_1)$ that has $\|\nabla F(w(t_1))\| \leq K$. If there exist $\tau > 0$, such that $\|\nabla F(w(t_1 + \tau))\| > K$, then we define $\tau_0 := \max_\tau\{\tau : \|\nabla F(w(t_1 + t))\| \leq K, \forall 0 \leq t \leq \tau\}$. Because $\nabla F(w)$ and $\nabla^2 F(w)$ are continuous in w, there exists a neighborhood around $w(t_1 + \tau_0)$ such that for all w in this neighborhood it holds $\|\nabla F(w)\| > K/2$ and $\|\nabla^2 F(w)\| > \mu/16$. Therefore it has $\frac{d}{dt}\|\nabla F(w)\|^2 < -\mu K^2/256$, and integrating it in a short time interval after $t_1 + \tau_0$ yields a contradiction. Consequently, we obtain another upper bound

$$\|\nabla F(w(\tau + t_1))\|^2 \leq \tilde{y}(\tau + t_1) := e^{-\mu\tau/8}\|\nabla F(w(t_1))\|^2$$

for any time $\tau \geq 0$ after t_1. A sufficient condition for the approximate stationary point $\widehat{w}$ writes $e^{-\mu\tau/8}\|\nabla F(w(t_1))\|^2 \leq \varepsilon^2$, which means $w(\tau + t_1)$ is an approximate stationary point if

$$\tau \geq \frac{8}{\mu}\log\left(\frac{\|\nabla F(w(t_1))\|^2}{\varepsilon^2}\right) = \frac{16}{\mu}\log\left(\frac{(5 + \frac{9}{\sqrt{\mu}})\sigma}{4\varepsilon}\right).$$

Combine two parts together to get the major result that the MAML ODE converges to an approximate stationary point $\widehat{w}(t)$ within

$$t = \frac{1}{\mu}\mathcal{O}\left[\log\left(\frac{(5 + \frac{9}{\sqrt{\mu}})(\mu^2\sigma\|\nabla f(w(0))\|^2 - \frac{\mu\sigma^3}{2})}{4\iota\sigma^2\varepsilon}\right)\right].$$

if $\|\nabla f(w(0))\|^2 > \frac{\sigma^2}{\mu}$, and the MAML ODE converges to an approximate stationary point $\widehat{w}(t)$ within $t = \frac{16}{\mu}\mathcal{O}\left[\log\left(\frac{(5+\frac{9}{\sqrt{\mu}})\sigma}{4\varepsilon}\right)\right]$ if $\|\nabla f(w(0))\|^2 \leq \frac{\sigma^2}{\mu}$. □

Lemma 3.7 (Proof in Appendix C)**.** *Suppose the loss functions $f_i(w)$ satisfy Assumptions 3.2 and 3.3, then it holds*

$$\begin{aligned} &\frac{d}{dt}\frac{1}{2}\|\nabla f(w)\|^2 \\ &\leq \frac{\sigma^2}{2} - \Big(\mu - \frac{5}{4}L^2\alpha(L^3\alpha^2 + 2L^2\alpha + 2)\Big)\|\nabla f(w)\|^2. \end{aligned} \tag{3.5}$$

When the gradient norm $\|\nabla f(w)\|$ is small enough, $\|\nabla F(w)\|$ is also controlled. Then we enter the second phase, in which we follow a gradient flow inside a convex set where the MAML loss is strongly convex. This establishes another linear convergence rate from $\mathcal{O}(\sigma)$ down to ε. Combining the above two phases gives us the overall linear rate. In the following subsections, we will explain these two phases in more details.

3.3 Large Gradient Phase: Linear Convergence via Lyapunov Analysis

When $\|\nabla F(w)\|$ is large, its behavior under the MAML ODE (3.2), namely $\frac{d}{dt}\|\nabla F(w(t))\|$, is not easily tractable due to the non-convex nature of the MAML loss F. Hence, we consider instead a Lyapunov candidate function $\mathcal{E}(w(t)) = \|\nabla f(w(t))\|^2$ in the first phase, where

$$\begin{aligned}\frac{d}{dt}\mathcal{E}(w(t)) &= \nabla f(w)^\intercal \nabla^2 f(w)\dot{w} \\ &= -\nabla f(w)^\intercal \nabla^2 f(w)\nabla f(w) \\ &\quad + \nabla f(w)^\intercal \nabla^2 f(w)\,\mathbb{E}_{i\sim p}[B_i(w)\nabla f_i(w)].\end{aligned} \tag{3.6}$$

Even though we are primarily interested in the convergence of $\|\nabla F(w(t))\|$, the convergence analysis of $\|\nabla f(w(t))\|$ is still helpful. We show that when α is small, an upper bound on $\|\nabla f(w)\|$ gives an upper bound on $\|\nabla F(w)\|$, and *vice versa*. Hence we are able to keep track of an upper bound on $\|\nabla F(w(t))\|$ while only having one on $\|\nabla f(w(t))\|$. However, we need to remark that the convergence of $\|\nabla f(w)\|$ to zero does not imply the convergence of $\|\nabla F(w)\|$. In fact, the global minimum of f may not even be a stationary point of F.

Note that the Lyapunov candidate function $\|\nabla f(w)\|^2$ turns into a true Lyapunov function for the dynamic (3.2) if $\frac{d}{dt}\|\nabla f(w)\|^2 \leq 0$ for every $w \in \mathbb{R}^d$. To get a linear rate on $\|\nabla f(w)\|^2$, we need to characterize the right-hand side of (3.6). Notice that the first term is a quadratic form of $\nabla f(w)$, while the second term is less tractable due to the expectation. We build an upper bound in Lemma C.1 to tackle this second term, and it is achieved through "pulling out" the integrand $B_i(w)$ from the expectation and forming a quadratic form that is more friendly to the spectral analysis to follow. Lemma C.1 then leads to a more tractable upper bound for the right-hand side of (3.6), as illustrated in Lemma 3.7. If we replace the inequality in (3.5) with an equality, the resulting ODE on $\|\nabla f(w)\|^2$ is subject to a closed-form solution, which converges to a constant smaller than $\sigma^2/4$ when α is small. This solution serves as an upper bound on $\|\nabla f(w(t))\|^2$ for any $t \geq 0$. By making sure that the upper bound in (3.5) is strictly less than zero, it enables us to provide sufficient conditions on the MAML step size α so that the Lyapunov function is convergent in linear rate to a small constant, as explained in Theorem C.2.

However, the upper bound on the Lyapunov function $\|\nabla f(w)\|^2$ does not converge to zero, we can only guarantee in phase one that $\|\nabla F(w)\|$ goes below a constant. This issue will be resolved in phase two of the analysis.

3.4 Small Gradient Phase: Unique Global Minimum via Landscape Analysis

Due to the aforementioned limitations of the Lyapunov method, we propose a landscape analysis that complements the above argument and guarantees the linear rate of the MAML ODE when the gradient norm $\|\nabla F(w)\|$ on the MAML loss is small enough. Recall that the MAML ODE is a gradient flow on F, thus the landscape of F determines the behavior of the MAML ODE. If a function is strongly convex, then its gradient flow converges linearly to its unique minimizer. Even though the MAML loss F is not convex in general, we are able to show in Theorem 3.8 that for any point $w \in \mathbb{R}^d$ with a bounded gradient norm $\|\nabla F(w)\|$, the MAML loss F is both smooth and strongly convex in its neighborhood. This provides us with a powerful tool that enables us to show the global convergence of the MAML ODE, as indicated in Theorem 3.9.

Theorem 3.8 (Proof in Appendix D)**.** *Suppose $f_i(w)$ satisfies Assumptions 3.2 and 3.3. Then for any $\alpha \leq \min\{\frac{1}{2L}, \frac{7\mu}{8\kappa(16K+9\sigma)}\}$ and $w \in U(K) := \{w \in \mathbb{R}^d : \|\nabla F(w)\| \leq K\}$, we have $\frac{\mu}{8}I_d \preceq \mathrm{Hess}(F(w)) \preceq \frac{9L}{8}I_d$, where $\mathrm{Hess}(F(w))$ is the Hessian matrix of F at point w.*

Theorem 3.9 (Unique global minimum, proof in Appendix E)**.** *If $K \geq \left(\frac{9}{7}\sqrt{\frac{L}{\mu}}+1\right)\sigma$ and $\alpha \leq \min\{\frac{1}{4L}, \frac{7\mu}{8\kappa(16K+9\sigma)}\}$, the function F is strongly convex on the convex set $V\left(\frac{(K-\sigma)^2}{2L}\right) := \{w \in \mathbb{R}^d : f(w) \leq \min_{w'\in\mathbb{R}^d} f(w') + \frac{(K-\sigma)^2}{2L}\}$. Moreover, the set $V\left(\frac{(K-\sigma)^2}{2L}\right)$ contains the unique critical point of F.*

We remark that even though Theorem 3.9 concludes that the MAML loss F is strongly convex within V, F can be non-convex outside V (recall our example in Fig. 1). Being a sublevel set of a strongly convex function f, the set V is convex. Moreover, it is also closed and bounded. Again, by the strong convexity of f, its minimum $\min_{w'\in\mathbb{R}^d} f(w')$ exists and is finite. Theorem 3.9 implies that there is no critical point outside V. Since F is strongly convex within V, the unique critical point inside the convex sublevel set V is consequently the global minimizer of F.

4 Biphasic MAML (BI-MAML)

In this section, we propose a new algorithm named `Biphasic MAML` (BI-MAML) as an alternative to the original MAML. Unlike MAML where we always minimize F, the optimization process of BI-MAML can be divided into two phases. In the first phase, BI-MAML optimizes the expected task loss f until it reaches its approximate global minimum. In the second phase, it runs MAML until it finds an approximate critical point. We show that BI-MAML also enjoys the same $O(\log \frac{1}{\varepsilon})$ iteration complexity on strongly convex functions. While the iteration complexity of MAML and BI-MAML share the same order, BI-MAML has a lower computational complexity, because it performs gradient descent on f rather than F in the first phase and thereby avoids computing the Hessian. In contrast, MAML performs gradient descent on F, which involves computing the Hessian of f.

Inspired by our convergence analysis on MAML ODE, we propose a different dynamic with gradient flow in two stages called BI-MAML ODE. The first stage of BI-MAML ODE is a gradient flow on the expected loss function f that converges to one of its stationary point, *i.e.*, $\dot{w} = -\nabla f(w)$. This is followed by the second stage, which is a MAML ODE starting from an approximate stationary point provided by the first stage. Recall that MAML is a forward Euler integration of its continuous-time limit (3.2): a gradient flow on the MAML loss F. Analogously, the BI-MAML algorithm is a forward Euler integration of BI-MAML ODE: a gradient descent on f followed by a gradient descent of F. More detailed descriptions of BI-MAML and BI-MAML ODE are given in Algorithm 2 and Algorithm 1. As mentioned in Section 3, BI-MAML runs faster because it avoids computing the Hessian of f by performing gradient descent on f in the first phase.

Algorithm 1 BI-MAML ODE

Input: Loss functions $\{f_i(w)\}_{i\in[M]}$, MAML parameter α, step size β, tolerance level $\varepsilon_0, \varepsilon$.

```
initialize w(0) ∈ R^d arbitrarily
for t ∈ [0, ∞) do
    if ‖∇f(w(t))‖ ≥ ε_0 then
        dw/dt ← −β∇f(w(t))
    else
        dw/dt ← −β∇F(w(t))
    end if
    return w(t) if ‖∇F(w(t))‖ ≤ ε
end for
```

Our result in Theorem 4.1 shows that the BI-MAML ODE also achieves linear convergence in finding a critical point on the MAML loss F.

Theorem 4.1 (Proof in Appendix C)**.** *Suppose the loss function $f_i(w)$ satisfies Assumptions 3.2 to 3.5 and ε_0 is the tolerate level set in Algorithm 1, if*

$$\alpha < \min\left\{\frac{1}{2L}, \frac{7\mu}{288\kappa\varepsilon_0 + 232\kappa\sigma}\right\}$$

then the BI-MAML ODE finds a solution $\widehat{w}$ such that $\|\nabla F(\widehat{w}(t))\| \le \varepsilon$ after at most running for

$$t = \frac{1}{\mu}\mathcal{O}\left[\log\left(\frac{(9\varepsilon_0 + 5\sigma)\|\nabla f(w(0))\|}{4\varepsilon_0\varepsilon}\right)\right].$$

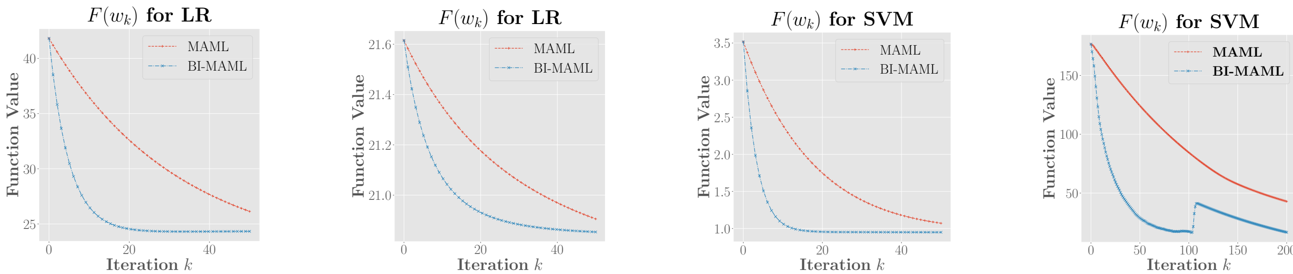
Theorem 4.1 states that whenever the MAML parameter α is small enough so that F is strongly convex for every point w such that $\|\nabla f(w)\| \le \varepsilon_0$, the BI-MAML ODE finds an approximate global minimum of F in $\mathcal{O}(\log \frac{1}{\varepsilon})$ time. The proof for Theorem 4.1 is similar to that of Theorem 3.6, *cf.* Appendix C.

5 Numerical Experiments

Our experiments evaluate the BI-MAML algorithm proposed in Section 3 against the MAML algorithm on a series of learning problems. More specifically, we compare both methods on two different tasks: linear regression and binary classification with a Support Vector Machine (SVM). They correspond to two different types of task loss f_i: strongly convex and convex. For both types of problem we compare the methods on both synthetic and real data.

Linear regression. For the linear regression problem with synthetic data, we meta-learn the model parameter $\gamma \in \mathbb{R}^d$ from a set of generated data that correspond to $M = 10$ individual linear regression tasks, each with dimension $d = 20$. A ground truth vector $\gamma_i \in \mathbb{R}^d$ is generated independently for each individual task. Each γ_i has its coordinates drawn from i.i.d. standard normal distributions. From each task, we generate $n = 100$ training samples $\{x_j, y_j\} \in \mathbb{R}^d \times \mathbb{R}$, where each entry of x_j is subject to an i.i.d. standard normal distribution and $y_j = x_j^\intercal \gamma_i + \sigma Z_j$ where Z_j is a standard normal variable and $\sigma = 1$ in our case. We also initialize the model parameter $\gamma \in \mathbb{R}^d$ randomly in the way we generate γ_i, and the MAML step size $\alpha = 0.3$ is fixed for all tasks. Each linear regression task i has a strongly convex loss function $f_i(\gamma) = \|y - X\gamma\|^2$, where $X = [x_1, \ldots, x_n]^\intercal$ and $y = [y_1, \ldots, y_n]^\intercal$. We take the gradient descent step size $\beta = 0.05$ for both BI-MAML and MAML, and we present the numerical results in Figs. 2a and 2e. For the linear regression problem on the `Diabetes` data set, we divide the original data into $M = 4$ tasks with dimension $d = 8$ according to male/female and whether the person is older than mean age of the data. We initialize the model parameter β randomly with i.i.d. normal entries, and the MAML step size $\alpha = 0.5$ is fixed for all tasks. We choose $\beta = 0.1$ and run the algorithms. Numerical results are presented in Figs. 2b and 2f. Compared to MAML, our BI-MAML algorithm in Fig. 2a converges to a neighborhood of stationary point of F in 25 steps, where MAML is far from convergence. After 50 iterations, BI-MAML also shows its superiority in computational efficiency over MAML by having 37% and 36% decrease in computation time on synthetic and real data, respectively.

Support vector machine. For binary classification with an SVM, we meta-learn from $M = 50$ binary clas-

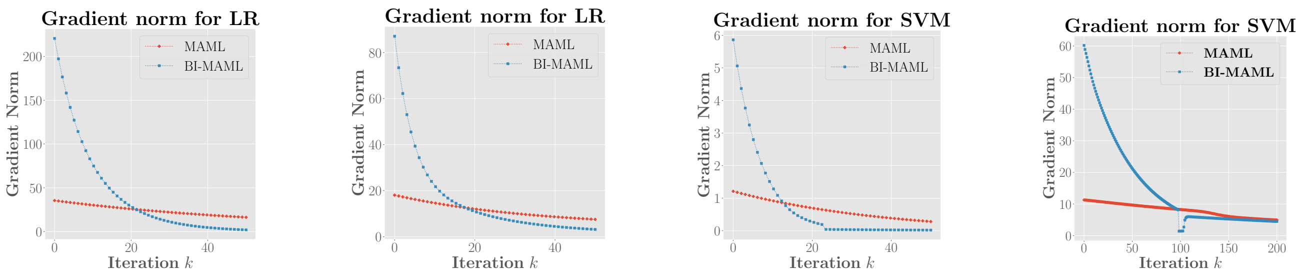


(a) Linear regression on synthetic data

(b) Linear regression on `Diabetes`

(c) Binary class. with SVM on synthetic data

(d) Binary classification with SVM on `MNIST`



(e) Linear regression on synthetic data

(f) Linear regression on `Diabetes`

(g) Binary class. with SVM on synthetic data

(h) Binary classification with SVM on `MNIST`

Figure 2: Comparisons on the performance of BI-MAML versus MAML on linear regression and binary classification with SVM. The figures on the first row compare the BI-MAML and MAML with function values on the MAML loss $F(w_k)$ at each iteration k. The figures in the second row show the trajectory of the corresponding gradient norm values evaluated at every iteration k. The blue lines represent the performance on BI-MAML, and the red lines represent that on MAML.

sification tasks in $d = 20$ dimensional space, where each one of the individual tasks is composed of $n = 300$ samples $\{x_j, y_j\} \in \mathbb{R}^d \times \{\pm 1\}$, evenly split between the positive and negative classes. We generate these training data sets with the default data generating function from `scikit-learn`. For each task i, it is equipped with a convex (but not strongly convex) hinge loss $f_i(w) = \mathrm{ReLu}(y - Xw)$, where $X = [x_1, \ldots, x_n]^\intercal$ and $y = [y_1, \ldots, y_n]^\intercal$. During the experiment, we take the MAML parameter $\alpha = 0.3$ for all tasks, and the gradient descent step size $\beta = 0.1$ is used for both the BI-MAML and the MAML. The corresponding numerical results are presented in Figs. 2c and 2g. For the binary classification on `MNIST`, we design $M = 5$ tasks, each to be a classification on different digits (classify $1, 3, 5, 7, 9$ against $2, 4, 6, 8, 0$), which has dimension $d = 784$. Each task has $n = 200$ samples, and the MAML parameter $\alpha = 0.3$ is fixed for all tasks. We take $\beta = 0.01$ for the experiment, and the result is reported in Figs. 2d and 2h. Compared to MAML, our BI-MAML algorithm in Fig. 2g reaches the region with small gradient norm $\|\nabla F(w)\|$ in less than 30 steps, where MAML still has a much larger gradient. In these experiments, BI-MAML shows its superiority in computational efficiency over MAML by having a 19% and 37% decrease in computation time on the synthetic and real data, respectively.

It is noteworthy that even if our method only provably works for strongly convex loss functions f_i, it empirically achieves good performance on real world data for many strongly convex and even convex loss functions, such as SVM. We remark that the gradient norm for the BI-MAML shown in Figs. 2e to 2h represents different quantities in different stages of the algorithm. Specifically, it denotes $\|\nabla F(w_k)\|$ when the BI-MAML descends on F at the k-th iteration; analogously, it represents $\|\nabla f(w_k)\|$ when the BI-MAML descends on f. The jumps in Figs. 2d, 2g and 2h show clearly the transitions between two phases.

6 Conclusions

In this paper we analyze the MAML ODE, a continuous-time limit of MAML, and establish a linear convergence rate to the global minimum of the MAML loss function for strongly convex task losses. We also propose a computationally efficient algorithm BI-MAML where its continuous-time limit BI-MAML ODE has the same linear convergence guarantee under milder conditions. We experimentally show that the BI-MAML method outperforms MAML in a variety of learning tasks.

Acknowledgements

Amin Karbasi is partially supported by NSF (IIS-1845032), ONR (N00014-19-1-2406), and AFOSR (FA9550-18-1-0160). We would like to thank Marko Mitrovic for his help in preparation of the paper.

References

Andrychowicz, M., Denil, M., Gomez, S., Hoffman, M. W., Pfau, D., Schaul, T., Shillingford, B., & De Freitas, N. (2016). Learning to learn by gradient descent by gradient descent. In *Advances in neural information processing systems* (pp. 3981–3989).

Antoniou, A., Edwards, H., & Storkey, A. (2018). How to train your maml. *arXiv preprint arXiv:1810.09502*.

Behl, H. S., Baydin, A. G., & Torr, P. H. (2019). Alpha maml: Adaptive model-agnostic meta-learning. In *6th ICML Workshop on Automated Machine Learning*.

Duan, Y., Andrychowicz, M., Stadie, B., Ho, O. J., Schneider, J., Sutskever, I., Abbeel, P., & Zaremba, W. (2017). One-shot imitation learning. In *Advances in neural information processing systems* (pp. 1087–1098).

Duan, Y., Schulman, J., Chen, X., Bartlett, P. L., Sutskever, I., & Abbeel, P. (2016). Rl^2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*.

Edwards, H. & Storkey, A. (2016). Towards a neural statistician. *arXiv preprint arXiv:1606.02185*.

Fallah, A., Mokhtari, A., & Ozdaglar, A. (2019). On the convergence theory of gradient-based model-agnostic meta-learning algorithms. *arXiv preprint arXiv:1908.10400*.

Finn, C., Abbeel, P., & Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. *34th International Conference on Machine Learning, ICML 2017*, 3, 1856–1868.

Finn, C., Rajeswaran, A., Kakade, S., & Levine, S. (2019). Online meta-learning. In *ICML*.

Grant, E., Finn, C., Levine, S., Darrell, T., & Griffiths, T. (2018). Recasting gradient-based meta-learning as hierarchical bayes. *arXiv preprint arXiv:1801.08930*.

Helmke, U. & Moore, J. B. (2012). *Optimization and dynamical systems*. Springer Science & Business Media.

Hirsch, M. W., Smale, S., & Devaney, R. L. (2012). *Differential equations, dynamical systems, and an introduction to chaos*. Academic press.

Hochreiter, S., Younger, A. S., & Conwell, P. R. (2001). Learning to learn using gradient descent. In *International Conference on Artificial Neural Networks* (pp. 87–94).: Springer.

Hsu, K., Levine, S., & Finn, C. (2019). Unsupervised learning via meta-learning. In *International Conference on Learning Representations*.

Krichene, W., Bayen, A. M., & Bartlett, P. L. (2015). Accelerated mirror descent in continuous and discrete time. *Advances in Neural Information Processing Systems*, 2015-Janua, 2845–2853.

Lacoste, A., Boquet, T., Rostamzadeh, N., Oreshkin, B., Chung, W., & Krueger, D. (2017). Deep prior. *arXiv preprint arXiv:1712.05016*.

Li, D., Yang, Y., Song, Y.-Z., & Hospedales, T. M. (2018). Learning to generalize: Meta-learning for domain generalization. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Lu, H. (2020). An $O(s^r)$-resolution ODE framework for discrete-time optimization algorithms and applications to convex-concave saddle-point problems. *arXiv preprint arXiv:2001.08826*.

Lyapunov, A. M. (1992). The general problem of the stability of motion. *International journal of control*, 55(3), 531–534.

Mendonca, R., Gupta, A., Kralev, R., Abbeel, P., Levine, S., & Finn, C. (2019). Guided meta-policy search. In *Advances in Neural Information Processing Systems* (pp. 9653–9664).

Muehlebach, M. & Jordan, M. I. (2019). A dynamical systems perspective on nesterov acceleration. *arXiv preprint arXiv:1905.07436*.

Munkhdalai, T. & Yu, H. (2017). Meta networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70* (pp. 2554–2563).: JMLR. org.

Naik, D. K. & Mammone, R. J. (1992). Meta-neural networks that learn by learning. In *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, volume 1 (pp. 437–442).: IEEE.

Pan, S. J. & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10), 1345–1359.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 9.

Raghu, A., Raghu, M., Bengio, S., & Vinyals, O. (2019). Rapid learning or feature reuse? towards understanding the effectiveness of maml. *arXiv preprint arXiv:1909.09157*.

Rajeswaran, A., Finn, C., Kakade, S. M., & Levine, S. (2019). Meta-learning with implicit gradients. In *Advances in Neural Information Processing Systems* (pp. 113–124).

Ravi, S. & Larochelle, H. (2016). Optimization as a model for few-shot learning.

Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., & Lillicrap, T. (2016). Meta-learning with memory-augmented neural networks. In *International conference on machine learning* (pp. 1842–1850).

Schmidhuber, J. (1987). *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, Technische Universität München.

Schropp, J. & Singer, I. (2000). A dynamical systems approach to constrained minimization. *Numerical functional analysis and optimization*, 21(3-4), 537–551.

Schweighofer, N. & Doya, K. (2003). Meta-learning in reinforcement learning. *Neural Networks*, 16(1), 5–9.

Shi, B., Du, S. S., Su, W., & Jordan, M. I. (2019). Acceleration via symplectic discretization of high-resolution differential equations. In *Advances in Neural Information Processing Systems* (pp. 5745–5753).

Snell, J., Swersky, K., & Zemel, R. (2017). Prototypical networks for few-shot learning. In *Advances in neural information processing systems* (pp. 4077–4087).

Song, X., Gao, W., Yang, Y., Choromanski, K., Pacchiano, A., & Tang, Y. (2019). Es-maml: Simple hessian-free meta learning. *arXiv preprint arXiv:1910.01215*.

Su, W., Boyd, S., & Candes, E. (2014). A differential equation for modeling nesterov's accelerated gradient method: Theory and insights. In *Advances in Neural Information Processing Systems* (pp. 2510–2518).

Thrun, S. & Pratt, L. (1998). *Learning to learn*. Springer Science & Business Media.

Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al. (2016). Matching networks for one shot learning. In *Advances in neural information processing systems* (pp. 3630–3638).

Wang, J. X., Kurth-Nelson, Z., Tirumala, D., Soyer, H., Leibo, J. Z., Munos, R., Blundell, C., Kumaran, D., & Botvinick, M. (2016). Learning to reinforcement learn. *arXiv preprint arXiv:1611.05763*.

Wang, Y.-X. & Hebert, M. (2016). Learning to learn: Model regression networks for easy small sample learning. In *European Conference on Computer Vision* (pp. 616–634).: Springer.

Wilson, A. C., Recht, B., & Jordan, M. I. (2016). A lyapunov analysis of momentum methods in optimization. *arXiv preprint arXiv:1611.02635*.

Yang, Y., Caluwaerts, K., Iscen, A., Tan, J., & Finn, C. (2019). Norml: No-reward meta learning. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems* (pp. 323–331).: International Foundation for Autonomous Agents and Multiagent Systems.

Yu, T., Finn, C., Xie, A., Dasari, S., Zhang, T., Abbeel, P., & Levine, S. (2018). One-shot imitation from observing humans via domain-adaptive meta-learning. In *ICLR workshop*.