Vers une meilleure compréhension des méthodes de méta-apprentissage à travers la théorie de l'apprentissage de représentations multi-tâches

Quentin Bouniot[†‡], Ievgen Redko[‡], Romaric Audigier[†], Angélique Loesch[†]

[†]Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
{firstname.lastname}@cea.fr
[‡]Université de Lyon, UJM-Saint-Etienne, CNRS,
Institut d'Optique Graduate School, Laboratoire Hubert Curien UMR 5516,
F-42023, Saint-Etienne, France
{firstname.lastname}@univ-st-etienne.fr

1er juin 2021

Résumé

Dans ce papier, nous considérons le cadre de l'apprentissage de représentations multi-tâches où l'objectif est d'utiliser des tâches sources pour apprendre une représentation qui réduit la complexité en données nécessaires pour la résolution d'une tâche cible. Nous commençons par passer en revue les avancées récentes de la théorie en apprentissage multi-tâches et nous montrons qu'elles peuvent fournir de nouveaux éclaircissements pour les algorithmes populaires de méta-apprentissage lorsque ceux-ci sont analysés dans ce cadre. En particulier, nous mettons en évidence une différence fondamentale entre les algorithmes basés sur les gradients et ceux basés sur un calcul de distance et nous proposons une analyse théorique pour l'expliquer. Enfin, nous utilisons les résultats obtenus pour améliorer la capacité de généralisation des méthodes de méta-apprentissage par le biais d'un nouveau terme de régularisation spectral et nous confirmons son efficacité par des études expérimentales sur des bases de données classiques de classification avec peu d'images. À notre connaissance, il s'agit de la première contribution qui met en pratique les plus récentes bornes issues de la théorie de l'apprentissage de représentations multi-tâches.

1 Introduction

Even though many machine learning methods now enjoy a solid theoretical justification, some more recent advances in the field are still in their preliminary state which requires the hypotheses put forward by the theoretical studies to be implemented and verified in practice. One such notable example is the success of *meta-learning*, also called *learning to learn* (LTL), methods where the goal is to produce a model on data coming from a set of (meta-train) source tasks to use it as a starting point for learning successfully a new previously unseen (meta-test) target task. The success of many meta-learning approaches is directly related to their capacity of learning a good representation [RRBV20] from a set of tasks making it closely related to multi-task representation learning (MTR). For this latter, several theoretical studies [Bax00, PL14, MPR16, AM18, YTZ$^+$20] provided probabilistic learning bounds that require the amount of data in the meta-train source task *and* the number of meta-train tasks to tend to infinity for it to be efficient. While capturing the underlying general intuition, these bounds do not suggest that all the source data is useful in such learning setup due to the additive relationship between the two terms mentioned above and thus, for instance, cannot explain the empirical success of MTR in few-shot classification (FSC) task. To tackle this drawback, two very recent studies [DHK$^+$20, TJJ20] aimed at finding deterministic assumptions that lead to faster learning rates allowing

MTR algorithms to benefit from all the source data. Contrary to probabilistic bounds that have been used to derive novel learning strategies for meta-learning algorithms [AM18, YTZ$^+$20], there has been no attempt to verify the validity of the assumptions leading to the fastest known learning rates in practice or to enforce them through an appropriate optimization procedure.

In this paper, we aim to use the recent advances in MTR theory [TJJ20, DHK$^+$20] to explore the inner workings of popular meta-learning methods. In particular, we take a closer look at two popular families of meta-learning algorithms, notably : gradient-based algorithms [RL17, NAS18, LMRS19, BHTV19, PO19] including MAML [FAL17] and metric-based algorithms [KZS15, VBL$^+$16, SSZ17, ASST19] with its most prominent example given by PROTONET [SSZ17].

Our main contributions are then two-fold :

1. We empirically show that tracking the validity of assumptions on optimal predictors used in [TJJ20, DHK$^+$20] reveals a striking difference between the behavior of gradient-based and metric-based methods in how they learn their optimal feature representations. We provide elements of theoretical analysis that explain this behavior and explain the implications of it in practice.

2. We show that theoretical assumptions mentioned above can be forced during the training of meta-learning algorithms for both families of considered methods and that enforcing them leads to better generalization of the considered algorithms for FSC baselines.

The rest of the paper is organized as follows. We present the existing theoretical results for the MTR problem with their corresponding assumptions, and introduce considered meta-learning algorithms in Section 2. In Section 3, we investigate how metric-based and gradient-based algorithms behave in practice with respect to the identified assumptions and provide theoretical explanation to the observed behavior. We further show that one can force meta-learning algorithms to satisfy such assumptions through adding an appropriate spectral regularization term to their objective function. In Section 4, we provide an experimental evaluation of several state-of-the-art meta-learning methods and highlight the different advantages brought by the proposed regularization technique in practice for the FSC. Finally, we conclude and outline the future research perspectives in Section 5.

2 Preliminary Knowledge

2.1 Multi-Task Representation Learning Setup

Given a set of T source tasks observed through finite size samples of size n_1 grouped into matrices $\mathbf{X}_t = (\mathbf{x}_{t,1}, \dots, \mathbf{x}_{t,n_1}) \in \mathbb{R}^{n_1 \times d}$ and vectors of outputs $y_t = (y_{t,1}, \dots, y_{t,n_1}) \in \mathbb{R}^{n_1},\ \forall t \in [[T]] := \{1, \dots, T\}$ generated by their respective distributions μ_t, the goal of MTR is to learn a shared representation ϕ belonging to a certain class of functions $\Phi := \{\phi \mid \phi : \mathbb{X} \to \mathbb{V},\ \mathbb{X} \subseteq \mathbb{R}^d,\ \mathbb{V} \subseteq \mathbb{R}^k\}$ and linear predictors $\mathbf{w}_t \in \mathbb{R}^k,\ \forall t \in [[T]]$ grouped in a matrix $\mathbf{W} \in \mathbb{R}^{T \times k}$. More formally, this is done by solving the following optimization problem :

$$\hat{\phi}, \widehat{\mathbf{W}} = \underset{\phi \in \Phi, \mathbf{W} \in \mathbb{R}^{T \times k}}{\arg\min} \frac{1}{T n_1} \sum_{t=1}^{T} \sum_{i=1}^{n_1} \ell(y_{t,i}, \langle \mathbf{w}_t, \phi(\mathbf{x}_{t,i}) \rangle),$$

where $\ell : \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}_+$, with $\mathbb{Y} \subseteq \mathbb{R}$, is a loss function. Once such a representation is learned, we want to apply it to a new previously unseen target task observed through a pair $(\mathbf{X}_{T+1} \in \mathbb{R}^{n_2 \times d}, y_{T+1} \in \mathbb{R}^{n_2})$ containing n_2 samples generated by the distribution μ_{T+1}. We expect that a linear classifier $\mathbf{w}$ learned on top of the obtained representation leads to a low true risk over the whole distribution μ_{T+1}. For this, we first use $\hat{\phi}$ to solve the following problem :

$$\hat{\mathbf{w}}_{T+1} = \underset{\mathbf{w} \in \mathbb{R}^k}{\arg\min} \frac{1}{n_2} \sum_{i=1}^{n_2} \ell(y_{T+1,i}, \langle \mathbf{w}, \hat{\phi}(\mathbf{x}_{T+1,i}) \rangle).$$

Then, we define the true target risk of the learned linear classifier $\hat{\mathbf{w}}_{T+1}$ as :

$$\mathcal{L}(\hat{\phi}, \hat{\mathbf{w}}_{T+1}) = \underset{(\mathbf{x},y) \sim \mu_{T+1}}{\mathbb{E}} [\ell(y, \langle \hat{\mathbf{w}}_{T+1}, \hat{\phi}(\mathbf{x}) \rangle)]$$

and want it to be as close as possible to the ideal true risk $\mathcal{L}(\phi^*, \mathbf{w}^*_{T+1})$ where $\mathbf{w}^*_{T+1}$ and ϕ^* satisfy :

$$\begin{aligned} &\forall t \in [[T+1]] \text{ and } (\mathbf{x}, y) \sim \mu_t, \\ &\quad y = \langle \mathbf{w}^*_t, \phi^*(\mathbf{x}) \rangle + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2). \end{aligned} \tag{1}$$

Equivalently, most of the works found in the literature seek to upper-bound the *excess risk* defined as $\mathrm{ER}(\hat{\phi}, \hat{\mathbf{w}}_{T+1}) := \mathcal{L}(\hat{\phi}, \hat{\mathbf{w}}_{T+1}) - \mathcal{L}(\phi^*, \mathbf{w}^*_{T+1})$.

2.2 Learning Bounds and Assumptions

First studies in the context of MTR relied on probabilistic assumption [Bax00, PL14, MPR16, AM18, YTZ$^+$20] stating that meta-train and meta-test tasks

distributions are all sampled i.i.d. from the same random distribution. This assumption, however, is considered unrealistic as in many learning settings, such as FSC, source and target tasks' data are often given by different draws (without replacement) from the same dataset. In this setup, the above-mentioned works obtained the bounds having the following form :

$$\mathrm{ER}(\hat{\phi}, \hat{\mathbf{w}}_{T+1}) \leq O\left(\frac{1}{\sqrt{n_1}} + \frac{1}{\sqrt{T}}\right).$$

Such a guarantee implies that even with the increasing number of source data, one would still have to increase the number of tasks as well, in order to draw the second term to 0. A natural improvement to this bound was then proposed by [DHK$^+$20] and [TJJ20] that obtained the bounds on the excess risk behaving as

$$\mathrm{ER}(\hat{\phi}, \hat{\mathbf{w}}_{T+1}) \leq O\left(\frac{1}{n_1 T} + \frac{1}{n_2}\right).$$

Both these results show that all the source and target samples are useful in minimizing the excess risk. Thus, in the FSC regime where target data is scarce, all source data helps to learn well. From a set of assumptions made by the authors in both of these works, we note the following two :

Assumption 1 : Diversity of the source tasks The matrix of optimal predictors $\mathbf{W}^*$ should cover all the directions in $\mathbb{R}^k$ evenly. More formally, this can be stated as

$$\kappa(\mathbf{W}^*) = \frac{\sigma_1(\mathbf{W}^*)}{\sigma_k(\mathbf{W}^*)} = O(1),$$

where $\sigma_i(\cdot)$ denotes the i^{th} singular value of $\mathbf{W}^*$. As pointed out by the authors, such an assumption can be seen as a measure of diversity between the source tasks that are expected to be complementary to each other to provide a useful representation for a previously unseen target task.

Assumption 2 : Consistency of the classification margin The norm of the optimal predictors $\mathbf{w}^*$ should not increase with the number of tasks seen during meta-training [1]. This assumption says that the classification margin of linear predictors should remain constant thus avoiding over- or under-specialization to the seen tasks.

While being highly insightful, the authors did not provide any experimental evidence suggesting that verifying these assumptions in practice helps to learn more efficiently in the considered learning setting. Furthermore, from the proof given by [DHK$^+$20], and with the same assumptions, we can easily derive a more explicit bound :

If $\forall t, \|\mathbf{w}_t^*\| = O(1)$ then,

$$\mathrm{ER}(\hat{\phi}, \hat{\mathbf{w}}_{T+1}) \leq O\left(\frac{1}{n_1 T} \cdot \kappa(\mathbf{W}^*) + \frac{1}{n_2}\right).$$

suggesting that the terms $\|\mathbf{w}_t^*\|$ and $\kappa(\mathbf{W}^*)$ underlying the assumptions directly impact the tightness of the established bound on the excess risk.

2.3 Meta-Learning Algorithms

Meta-learning algorithms considered below learn an optimal representation sequentially via the so-called episodic training strategy introduced by [VBL$^+$16], instead of jointly minimizing the training error on a set of source tasks as done in MTR. Episodic training mimics the training process at the task scale with each task data being decomposed into a training set *(support set S)* and a testing set *(query set Q)*. Recently, [CWL$^+$20] showed that the episodic training setup used in meta-learning leads to a generalization bounds of $O(\frac{1}{\sqrt{T}})$. This bound is independent of the task sample size n_1, which could explain the success of this training strategy for FSC in the asymptotic limit. However, unlike the results obtained by [DHK$^+$20] studied in this paper, the lack of dependence on n_1 makes such a result uninsightful in practice as we are in a finite-sample size setting. This bound does not give information on other parameters to leverage when the task number cannot increase. We now present two major families of meta-learning approaches below.

Metric-based methods These methods learn an embedding space in which feature vectors can be compared using a similarity function (usually a L_2 distance or cosine similarity) [KZS15, VBL$^+$16, SSZ17, ASST19]. They typically use a form of contrastive loss as their objective function, similarly to Neighborhood Component Analysis (NCA) [GHRS05]. In this paper, we focus our analysis on the popular Prototypical Networks [SSZ17] (ProtoNet) that computes prototypes as the mean vector of support points belonging to the same class : $\mathbf{c}_i = \frac{1}{|S_i|}\sum_{\mathbf{s}\in S_i} \phi(\mathbf{s})$, with S_i the subset of support points belonging to class i.

ProtoNet minimizes the negative log-probability of the true class i computed as the softmax over distances to prototypes $\mathbf{c}_i$:

$$\mathcal{L}_{proto}(S, Q, \phi) :=$$

1. While not stated separately, this assumption is used in [DHK$^+$20] to derive the final result on p.5 after the discussion of Assumption 4.3.

$$\mathbb{E}_{\mathbf{q}\sim Q}\left[-\log\frac{\exp(-d(\phi(\mathbf{q}),\mathbf{c}_i))}{\sum_j \exp\left(-d(\phi(\mathbf{q}),\mathbf{c}_j)\right)}\right]$$

with d being a distance function used to measure similarity between points in the embedding space. In what follows, we establish our theoretical analysis for ProtoNet and add its recent improved variation called Infinite Mixture Prototypes [ASST19] (IMP) in the experiments to confirm that the deduced findings apply to other metric-based methods as well.

Gradient-based methods These methods learn through end-to-end or two-step optimization [RL17, FAL17, NAS18, LMRS19, BHTV19, PO19] where given a new task, the goal is to learn a model from the task's training data specifically adapted for this task. Maml [FAL17] updates its parameters θ using an end-to-end optimization process to find the best initialization such that a new task can be learned quickly, *i.e.* with few examples. More formally, given the loss ℓ_t for each task $t \in [[T]]$, Maml minimizes the expected task loss after an *inner loop* or *adaptation* phase, computed by a few steps of gradient descent initialized at the model's current parameters :

$$\mathcal{L}_{\text{Maml}}(\theta) := \mathbb{E}_{t\sim\eta}[\ell_t(\theta - \alpha\nabla\ell_t(\theta))],$$

with η the distribution of the meta-training tasks and α the learning rate for the adaptation phase. For simplicity, we take a single step of gradient update in this equation.

Again, we concentrate our theoretical analysis on the most popular method (Maml) and add its recent improvement Meta-Curvature [PO19] (MC) to validate our findings for gradient-based methods experimentally.

3 Understanding Meta-learning Algorithms through MTR Theory

In this section, we study the behavior of gradient- and metric-based meta-learning algorithms with respect to the theoretical insights from MTR theory. We start by empirically verifying that, despite a mismatch between the multi-task setup considered in theoretical works and the actual episodic training used by meta-learning methods, the behavior of such methods reveals very distinct features when looked at through the prism of the considered theoretical assumptions. We then set on a quest of explaining the differences in their behavior leading to novel insights into meta-learning algorithms and interesting open problems for future research.

3.1 What happens in practice ?

To verify whether theoretical results from MTR setting are also insightful for episodic training used by popular meta-learning algorithms, we first investigate the natural behavior of Maml and ProtoNet when solving the few-shot image classification problem on the popular *miniImageNet* [RL17] and *tieredImageNet* [RTR$^+$18] datasets. The full experimental setup is detailed in Section 4.1 and additional experiments for *Omniglot* [LST15] benchmark dataset portraying the same behavior are postponed to the Appendix.

To verify Assumption 1 from MTR theory, we seek to compute singular values of $\mathbf{W}$ during the meta-training stage and to follow their evolution. In practice, as T is typically quite large, we propose a more computationally efficient solution that is to calculate the condition number only for the last batch of N predictors (with $N \ll T$) grouped in the matrix $\mathbf{W}_N \in \mathbb{R}^{N\times k}$ that capture the latest dynamics in the learning process. We further note that $\sigma_i(\mathbf{W}_N\mathbf{W}_N^\top) = \sigma_i^2(\mathbf{W}_N),\ \forall i \in [[N]]$ implying that we can calculate the SVD of $\mathbf{W}_N\mathbf{W}_N^\top$ (or $\mathbf{W}_N^\top\mathbf{W}_N$ for $k \leq N$) and retrieve the singular values from it afterwards. We now want to verify whether $\mathbf{w}_t$ cover all directions in the embedding space and track the evolution of the ratio of singular values $\kappa(\mathbf{W}_N)$ during training. For the sake of conciseness, we use κ instead of $\kappa(\mathbf{W}_N)$ thereafter.

For the first assumption to be satisfied, we expect κ to decrease gradually during the training thus improving the generalization capacity of the learned predictors and preparing them for the target task. To verify the second assumption, the norm of the linear predictors should not increase with the number of tasks seen during training, *i.e.*, $\|\mathbf{w}\|_2 = O(1)$ or, equivalently, $\|\mathbf{W}\|_F^2 = O(T)$ and $\|\mathbf{W}_N\|_F = O(1)$.

From Fig. 1, we can see that for Maml (left), both $\|\mathbf{W}_N\|_F$ and κ increase with the number of tasks seen during training, whereas ProtoNet (right) naturally learns the prototypes with a good coverage of the embedding space, and minimizes their norm. This behavior is rather peculiar as neither of the two methods explicitly controls the theoretical quantities of interest, and still, ProtoNet manages to do it implicitly. Before confirming this claim through extensive empirical evaluations involving more baseline methods and benchmark datasets, we first prove several results that provide explanation to the difference of behavior of these two families of methods.

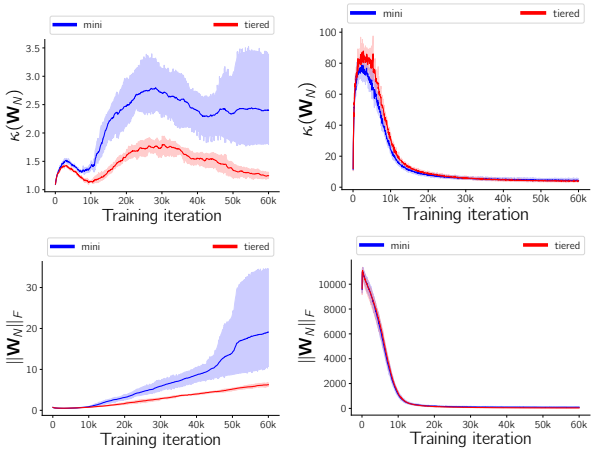

FIGURE 1 – Evolution of $\kappa(\mathbf{W}_N)$ (*top*), $\|\mathbf{W}_N\|_F$ (*middle*) and accuracy (*bottom*) during the training of MAML (*left*) and PROTONET (*right*) on miniImageNet (*mini*) and tieredImageNet (*tiered*) with 5-way 1-shot episodes. All training curves are averaged over 4 different random seeds. The light blue/red areas shows 95% confidence intervals.

3.2 Metric- vs Gradient-based Methods

The differences observed above for the two methods call for a deeper analysis of their behavior. To this end, we provide a full explanation of why PROTONET naturally leads to small condition number of the obtained predictors and a consistent behavior of their norm, while for MAML we consider a common simplified learning model leaving the general result as an open problem for future works.

PROTONET We start by first explaining why PROTONET learns prototypes that cover the embedding space efficiently. This result is given by the following theorem [2].

Theorem 1. (Normalized PROTONET) *If* $\forall i \ \|\mathbf{c}_i\| = 1$, *then an encoder* $\phi \in \arg\min \mathcal{L}_{proto}$ *has* $\kappa(\mathbf{W}^*) = 1$.

This theorem explains the empirical behavior of PROTONET in FSC task : the minimization of its objective function naturally minimizes the condition number when the norm of the prototypes is low.

MAML Unfortunately, the analysis of MAML in the most general case is notoriously harder, as even expressing its loss function and gradients in the case of an overparametrized linear regression model with only 2 parameters requires using a symbolic toolbox for derivations [AIS21].

To this end, we resort to the linear regression model considered in this latter paper and defined as follows. We assume for all $t \in [[T]]$ that the task parameters $\boldsymbol{\theta}_t$ are normally distributed with $\boldsymbol{\theta}_t \sim \mathcal{N}(\mathbf{0}_d, \boldsymbol{I}_d)$, the inputs $\mathbf{x}_t \sim \mathcal{N}(\mathbf{0}_d, \boldsymbol{I}_d)$ and the output $y_t \sim \mathcal{N}(\langle \boldsymbol{\theta}_t, \mathbf{x}_t \rangle, 1)$. For each t, we consider the following learning model and its associated square loss :

$$\hat{y}_t = \langle \mathbf{w}_t, \mathbf{x}_t \rangle, \quad \ell_t = \mathbb{E}_{p(\mathbf{x}_t, y_t | \boldsymbol{\theta}_t)}(y_t - \langle \mathbf{w}_t, \mathbf{x}_t \rangle)^2. \quad (2)$$

We can now state the following result.

Proposition 1. *Let* $\forall t \in [[T]]$, $\boldsymbol{\theta}_t \sim \mathcal{N}(\mathbf{0}_d, \boldsymbol{I}_d)$, $\mathbf{x}_t \sim \mathcal{N}(\mathbf{0}_d, \boldsymbol{I}_d)$ *and* $y_t \sim \mathcal{N}(\langle \boldsymbol{\theta}_t, \mathbf{x}_t \rangle, 1)$. *Consider the learning model from Eq. 2, let* $\boldsymbol{\Theta}_i := [\boldsymbol{\theta}_i, \boldsymbol{\theta}_{i+1}]^T$, *and denote by* $\widehat{\mathbf{W}}_2^i$ *the matrix of last two predictors learned by* MAML *at iteration* i *starting from* $\widehat{\mathbf{w}}_0 = \mathbf{0}_d$. *Then, we have that :*

$$\forall i, \quad \kappa(\widehat{\mathbf{W}}_2^{i+1}) \geq \kappa(\widehat{\mathbf{W}}_2^i), \quad \textit{if } \sigma_{min}(\boldsymbol{\Theta}_i) = 0.$$

This proposition provides an explanation of why MAML may tend to increase the ratio during the iterations. Indeed, the condition when this happens indicates that the optimal predictors forming matrix $\boldsymbol{\Theta}_i$ are linearly dependent implying that its smallest singular values becomes equal to 0. While this is not expected to be the case for all iterations, we note, however, that in FSC task the draws from the dataset are in general not i.i.d. and thus may correspond to co-linear optimal predictors. In every such case, the ratio is expected to remain non-decreasing, as illustrated in Figure 1 where MAML, contrary to PROTONET, exhibits plateaus and the intervals where this latter is increasing. This highlights a major difference between the two approaches : MAML does not specifically seek to diversify the learned predictors, while PROTONET does.

3.3 Can we force the assumptions ?

So far, we have gathered evidence for the fact that MTR theory seems to be insightful for meta-learning algorithms as well. As satisfying the assumptions from MTR theory is expected to come in hand with better generalization performance, we now study what impact forcing these assumptions may have on the learning process when the optimal predictors involved in the data generating process do not naturally satisfy them. To this end, we aim to answer the following question :

Given $\mathbf{W}^*$ *such that* $\kappa(\mathbf{W}^*) \gg 1$, *can we learn* $\widehat{\mathbf{W}}$ *with* $\kappa(\widehat{\mathbf{W}}) \approx 1$ *while solving the underlying classification problems equally well ?*

2. We refer the reader to the Appendix for the full proofs.

While obtaining such a result for any distribution seems to be very hard in the considered learning setup, we provide a constructive proof for the existence of a distribution for which the answer to the above-mentioned question is positive in the case of two tasks. The latter restriction comes out of the necessity to analytically calculate the singular values of $\mathbf{W}$ but we expect our example to generalize to more general setups and a larger number of tasks as well.

Proposition 2. *Let $T = 2$, $\mathbb{X} \subseteq \mathbb{R}^d$ be the input space and $\mathbb{Y} = \{-1, 1\}$ be the output space. Then, there exist distributions μ_1 and μ_2 over $\mathbb{X} \times \mathbb{Y}$, representations $\widehat{\phi} \neq \phi^*$ and matrices of predictors $\widehat{\mathbf{W}} \neq \mathbf{W}^*$ that satisfy Eq. 1 with $\kappa(\widehat{\mathbf{W}}) \approx 1$ and $\kappa(\mathbf{W}^*) \gg 1$.*

The established results show that in some cases even when $\mathbf{W}^*$ does not satisfy Assumptions 1-2 in the ϕ^* space, it may still be possible to learn a new representation $\widehat{\phi}$ such that the optimal predictors in this space do satisfy them. This can be done by using a common strategy that consists in adding $\kappa(\mathbf{W})$ and $\|\mathbf{W}\|_F^2$ directly as regularization terms :

$$\widehat{\phi}, \widehat{\mathbf{W}} = \underset{\phi \in \Phi, \mathbf{W} \in \mathbb{R}^{T \times k}}{\arg\min} \frac{1}{Tn_1} \sum_{t=1}^{T} \sum_{i=1}^{n_1} \ell(y_{t,i}, \langle \mathbf{w}_t, \phi(\mathbf{x}_{t,i}) \rangle) + \lambda_1 \kappa(\mathbf{W}) + \lambda_2 \|\mathbf{W}\|_F^2. \quad (3)$$

Below, we explain how to implement this idea in practice for popular meta-learning algorithms.

3.4 Related Work

Understanding meta-learning [RRBV20] investigate whether MAML algorithm works well due to rapid learning with significant changes in the representations when deployed on target task, or due to feature reuse where the learned representation remains almost intact and establish that the latter factor is dominant. In [GRF+20], the authors explain the success of meta-learning approaches by their capability to either cluster classes more tightly in feature space (task-specific adaptation approach), or to search for meta-parameters that lie close in weight space to many task-specific minima (full fine-tuning approach). Our paper is complementary to all other works mentioned above as it investigates a new aspect of meta-learning that has never been studied before and provides a more complete experimental evaluation with the two different approaches of meta-learning, separately presented in [RRBV20], and [GRF+20].

Common regularization strategies Even though our work does not aim at proposing a new regularization strategy for meta-learning, regularizing the condition number of the matrix of linear predictors and its norm as suggested by MTR theory appears to be novel and drastically different from existing regularization strategies. In general, we note that regularization in meta-learning (i) is applied to either the weights of the whole neural network [BSC18, YTZ+20], or (ii) the predictions [JQ19, GRF+20] or (iii) is introduced via a prior hypothesis biased regularized empirical risk minimization [PL14, KO17, DCSP18a, DCSP18b, DCGP19]. Contrary to the first group of methods and the famous weight decay approach [KH92], we do not regularize the whole weight matrix learned by the neural network but the linear predictors of its last layer. Similarly, spectral normalization proposed by [MKKY18] does not affect $\kappa(\mathbf{W})$ and serves a completely different purpose. Second, we regularize the singular values of the matrix of linear predictors obtained in the last batch of tasks instead of the predictions used by the methods of the second group (*e.g.*, using the theoretic-information quantities in [JQ19]). Finally, the works of the last group are related to the online setting with convex loss functions only, and, similarly to the algorithms from the second group, do not specifically target the spectral properties of the learned predictors.

4 Experiments

In this section, we investigate the impact of enforcing the aforementioned theoretical assumptions for meta-learning algorithms in practice.

4.1 Experimental Setup

We consider the few-shot image classification problem on three benchmark datasets, namely : 1) **Omniglot** [LST15] consisting of 1,623 classes with 20 images/class of size 28×28 ; 2) **miniImageNet** [RL17] consisting of 100 classes with 600 images of size 84×84 per class and 3) **tieredImageNet** [RTR+18] consisting of 779,165 images divided into 608 classes.

For each dataset, we follow a common experimental protocol used in [FAL17, CWL+19] and use a four-layer convolution backbone (Conv-4) with 64 filters as done by [CWL+19]. We perform 20-way classification with 1 shot and 5 shots on *Omniglot*, while on *miniImageNet* and *tieredImageNet* we perform 5-way classification with 1 shot and 5 shots. We measure the performance using the top-1 accuracy with 95% confidence intervals, reproduce the experiments with 4 different random seeds using a single NVIDIA V100 GPU, and average the results over 2400 test tasks.

TABLE 1 – Accuracy gap (in p.p.) when adding the normalization of prototypes (PROTONET and IMP), and both spectral and norm regularization (MAML and MC) enforcing the theoretical assumptions. Statistically significant results (out of confidence intervals) are reported with *. (Cf. Appendix for absolute performances)

Dataset	Episodes	PROTONET	IMP	MAML	MC
Omniglot	1-shot	$+0.33^*$	$+0.08$	$+3.95^*$	-0.61^*
	5-shot	$+0.01$	$+0.07^*$	$+1.17^*$	-0.10^*
miniImageNet	1-shot	$+0.76^*$	$+1.84^*$	$+1.23^*$	$+0.36$
	5-shot	$+2.03^*$	$+0.86^*$	$+1.96^*$	$+1.93^*$
tieredImageNet	1-shot	$+2.10^*$	$+1.30^*$	$+1.42^*$	$+0.70$
	5-shot	$+0.23$	$+0.59$	$+2.66^*$	$+1.39^s$

4.2 Metric-based Methods

Theorem 1 tells us that with normalized class prototypes that act as linear predictors, PROTONET naturally decreases the condition number of their matrix. To this end, we choose to ensure the theoretical assumptions for metric-based methods (PROTONET and IMP) only with the prototype normalization similarly to the constrained problem given in Eq. 6. From Table 1, we note that normalizing the prototypes from the very beginning of the training process has an overall positive effect on the obtained performance.

4.3 Gradient-based Methods

Gradient-based methods learn a batch of linear predictors for each task and we can directly take them as $\mathbf{W}_N$ to compute its SVD. In the following experiments, we consider the regularized problem of Eq. 3 for MAML as well as Meta-Curvature (MC) and set $\lambda_1 = \lambda_2 = 1$ to avoid hyper-parameter tuning. As expected, the dynamics of $\|\mathbf{W}_N\|_F$ and κ during the training of the regularized methods remain bounded (cf. Appendix).

The impact of our regularization on the results is quantified in Table 1 where a statistically significant accuracy gain is achieved in most cases. The obtained improvement is on average more substantial when compared to metric-based methods.

5 Conclusion

In this paper, we studied the validity of the theoretical assumptions made in recent papers on Multi-Task Representation Learning theory when applied to popular metric- and gradient-based meta-learning algorithms. We found a striking difference in their behavior and provided both theoretical and experimental arguments explaining that metric-based methods satisfy the considered assumptions, while gradient-based don't. We further used this as a starting point to implement a regularization strategy ensuring these assumptions and observed that it leads to faster learning and better generalization.

While this paper proposes an initial approach to bridging the gap between theory and practice for Meta-Learning, some questions remain open on the inner workings of these algorithms. In particular, being able to take better advantage of the particularities of the training tasks during meta-training could help improve the effectiveness of these approaches.

Références

[AIS21] S.M.R Arnold, S. Iqbal, and F. Sha. When MAML can adapt fast and how to assist when it cannot. In *AISTATS*, 2021.

[AM18] Ron Amit and Ron Meir. Meta-learning by adjusting priors based on extended PAC-Bayes theory. In *ICML*, 2018.

[ASST19] Kelsey Allen, Evan Shelhamer, Hanul Shin, and Joshua Tenenbaum. Infinite mixture prototypes for few-shot learning. In *ICML*, 2019.

[Bax00] Jonathan Baxter. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12, 2000.

[BHTV19] Luca Bertinetto, João F. Henriques, Philip H. S. Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *arXiv :1805.08136 [cs, stat]*, 2019.

[BSC18] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. MetaReg : Towards domain generalization using meta-regularization. In *NeurIPS*, 2018.

[CLF20] Tianshi Cao, Marc T. Law, and Sanja Fidler. A theoretical analysis of the number of shots in few-shot learning. In *ICLR*, 2020.

[CWL+19] Wei-Yu Chen, Yu-Chiang Frank Wang, Yen-Cheng Liu, Zsolt Kira, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019.

[CWL+20] Jiaxin Chen, Xiao-Ming Wu, Yanke Li, Qimai Li, Li-Ming Zhan, and Fu-lai Chung. A closer look at the training strategy for modern meta-learning. *NeurIPS*, 2020.

[DCGP19] Giulia Denevi, Carlo Ciliberto, Riccardo Grazzi, and Massimiliano Pontil. Learning-

to-learn stochastic gradient descent with biased regularization. In *ICML*, 2019.

[DCSP18a] Giulia Denevi, Carlo Ciliberto, Dimitris Stamos, and Massimiliano Pontil. Incremental learning-to-learn with statistical guarantees. In *Conference on Uncertainty in Artificial Intelligence*, 2018.

[DCSP18b] Giulia Denevi, Carlo Ciliberto, Dimitris Stamos, and Massimiliano Pontil. Learning to learn around a common mean. In *NeurIPS*, 2018.

[DHK+20] Simon S. Du, Wei Hu, Sham M. Kakade, Jason D. Lee, and Qi Lei. Few-shot learning via learning the representation, provably. In *arXiv :2002.09434*, 2020.

[FAL17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.

[GHRS05] Jacob Goldberger, Geoffrey E Hinton, Sam Roweis, and Russ R Salakhutdinov. Neighbourhood components analysis. In *NeurIPS*, 2005.

[GRF+20] Micah Goldblum, Steven Reich, Liam Fowl, Renkun Ni, Valeriia Cherepanova, and Tom Goldstein. Unraveling meta-learning : Understanding feature representations for few-shot tasks. In *ICML*, 2020.

[JQ19] Muhammad Abdullah Jamal and Guo-Jun Qi. Task agnostic meta-learning for few-shot learning. In *CVPR*, 2019.

[KH92] Anders Krogh and John A. Hertz. A simple weight decay can improve generalization. In *NeurIPS*, 1992.

[KO17] Ilja Kuzborskij and Francesco Orabona. Fast rates by transferring from auxiliary hypotheses. *Machine Learning*, 106(2), 2017.

[KZS15] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, 2015.

[LMRS19] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *CVPR*, 2019.

[LST15] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266), 2015.

[MKKY18] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018.

[MPR16] Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17, 2016.

[NAS18] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv :1803.02999 [cs]*, 2018.

[PL14] Anastasia Pentina and Christoph H. Lampert. A PAC-Bayesian bound for lifelong learning. In *ICML*, 2014.

[PO19] Eunbyung Park and Junier B. Oliva. Meta-curvature. In *NeurIPS*, 2019.

[RL17] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017.

[RRBV20] Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid learning or feature reuse ? Towards understanding the effectiveness of MAML. In *ICLR*, 2020.

[RTR+18] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. Meta-learning for semi-supervised few-shot classification. In *ICLR*, 2018.

[SSZ17] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017.

[TJJ20] Nilesh Tripuraneni, Chi Jin, and Michael I. Jordan. Provable meta-learning of linear representations. In *arXiv :2002.11684*, 2020.

[VBL+16] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NeurIPS*, 2016.

[WI20] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, Proceedings of machine learning research, 2020.

[YTZ+20] Mingzhang Yin, George Tucker, Mingyuan Zhou, Sergey Levine, and Chelsea Finn. Meta-learning without memorization. In *ICLR*, 2020.

A Intuition behind the assumptions

An intuition behind the assumptions of Section 2.2 and the regularization terms introduced in Section 3.3 can be seen in Fig. 2. When the assumptions are not verified, the linear predictors can be biased towards a single part of the space and over-specialized. The representation learned will not generalize well to unseen tasks. If the assumptions are respected, the linear predictors are complementary and will not under- or over-specialize to the tasks seen. The representation learned can adapt to the target tasks and better generalize.

B Behavior on Omniglot

The behavior of the norm and condition number of the predictor for MAML and PROTONET on the benchmark dataset *Omniglot* [LST15] is shown in Figure 3. We observe similar trends as on *miniImageNet* and *tieredImageNet* detailed in Section 3.1.

C Proofs of Section 3

Prototypical Loss We start by recalling the prototypical loss $\mathcal{L}_{proto}$ used during training of Prototypical Networks for a single episode with support set S and query set Q :

$$\begin{aligned}\mathcal{L}_{proto}(S,Q,\phi) &= \mathbb{E}_{(\mathbf{q},i)\sim Q}\left[-\log\frac{\exp(-d(\phi(\mathbf{q}),\mathbf{c}_i))}{\sum_j \exp\left(-d(\phi(\mathbf{q}),\mathbf{c}_j)\right)}\right] \\ &= \underbrace{\mathbb{E}_{(\mathbf{q},i)\sim Q}\left[d(\phi(\mathbf{q}),\mathbf{c}_i)\right]}_{(1)} \\ &\quad + \underbrace{\mathbb{E}_{\mathbf{q}\sim Q}\log\sum_{j=1}^{n}\exp\left(-d(\phi(\mathbf{q}),\mathbf{c}_j)\right)}_{(2)}\end{aligned}$$

with $\mathbf{c}_i = \frac{1}{k}\sum_{\mathbf{s}\in S_i}\phi(\mathbf{s})$ the prototype for class i, $S_i \subseteq S$ being the subset containing instances of S labeled with class i.

Distance For PROTONET, we consider the Euclidean distance between the representation of a query example $\phi(\mathbf{q})$ and the prototype of a class i $\mathbf{c}_i$:

$$\begin{aligned}-d(\phi(\mathbf{q}),\mathbf{c}_i) &= -\|\phi(\mathbf{q})-\mathbf{c}_i\|_2^2 \\ &= -\phi(\mathbf{q})^\top\phi(\mathbf{q}) + 2\mathbf{c}_i^\top\phi(\mathbf{q}) - \mathbf{c}_i^\top\mathbf{c}_i.\end{aligned}$$

Then, with respect to class i, the first term is constant and do not affect the softmax probabilities. The remaining terms are :

$$\begin{aligned}-d(\phi(\mathbf{q}),\mathbf{c}_i) &= 2\mathbf{c}_i^\top\phi(\mathbf{q}) - \|\mathbf{c}_i\|_2^2 \\ &= \frac{2}{|S_i|}\sum_{\mathbf{s}\in S_i}\phi(\mathbf{s})^\top\phi(\mathbf{q}) - \|\mathbf{c}_i\|_2^2.\end{aligned}$$

C.1 Proof of Theorem 1

Démonstration. We can rewrite the first term in $\mathcal{L}_{proto}$ as

$$\begin{aligned}&\mathbb{E}_{(\mathbf{q},i)\sim Q}\left[d(\phi(\mathbf{q}),\mathbf{c}_i)\right] \\ &= -\mathbb{E}_{(\mathbf{q},i)\sim Q}\left[\frac{2}{|S_i|}\sum_{\mathbf{s}\in S_i}\phi(\mathbf{s})^\top\phi(\mathbf{q}) - \|\mathbf{c}_i\|_2^2\right] \\ &= -\mathbb{E}_{(\mathbf{q},i)\sim Q}\left[\frac{2}{|S_i|}\sum_{\mathbf{s}\in S_i}\phi(\mathbf{s})^\top\phi(\mathbf{q})\right] \\ &\quad + \mathbb{E}_{(\mathbf{q},i)\sim Q}\left[\|\mathbf{c}_i\|_2^2\right],\end{aligned}$$

and the second term as

$$\begin{aligned}&\mathbb{E}_{\mathbf{q}\sim Q}\left[\log\sum_{j=1}^{n}\exp\left(-d(\phi(\mathbf{q}),\mathbf{c}_j)\right)\right] \\ &= \mathbb{E}_{\mathbf{q}\sim Q}\left[\log\sum_{j=1}^{n}\exp\left(\frac{2}{|S_j|}\sum_{\mathbf{s}\in S_j}\phi(\mathbf{s})^\top\phi(\mathbf{q}) - \|\mathbf{c}_j\|_2^2\right)\right] \\ &= \mathbb{E}_{\mathbf{q}\sim Q}\left[\log\sum_{j=1}^{n}\exp\left(2\mathbf{c}_j^\top\phi(\mathbf{q}) - \|\mathbf{c}_j\|_2^2\right)\right] \\ &= \mathbb{E}_{\mathbf{q}\sim Q}\left[\log\left(n\sum_{j=1}^{n}\frac{1}{n}\left[\exp\left(2\mathbf{c}_j^\top\phi(\mathbf{q}) - \|\mathbf{c}_j\|_2^2\right)\right]\right)\right] \\ &= \mathbb{E}_{\mathbf{q}\sim Q}\left[\log\sum_{j=1}^{n}\frac{1}{n}\left[\exp\left(2\mathbf{c}_j^\top\phi(\mathbf{q}) - \|\mathbf{c}_j\|_2^2\right)\right] + \log n\right].\end{aligned}$$

By dropping the constant part in the loss, we obtain :

$$\begin{aligned}\mathcal{L}_{proto}(S,Q,\phi) &= -\mathbb{E}_{(\mathbf{q},i)\sim Q}\left[\frac{2}{|S_i|}\sum_{\mathbf{s}\in S_i}\phi(\mathbf{s})^\top\phi(\mathbf{q})\right] \\ &\quad + \mathbb{E}_{\mathbf{q}\sim Q}\left[\log\sum_{j=1}^{n}\frac{1}{n}\left[\exp\left(2\mathbf{c}_j^\top\phi(\mathbf{q})\right)\right]\right].\end{aligned}$$

Let us note $\mathcal{S}^d$ the hypersphere of dimension d, and $\mathcal{M}(\mathcal{S}^d)$ the set of all possible Borel probability measures

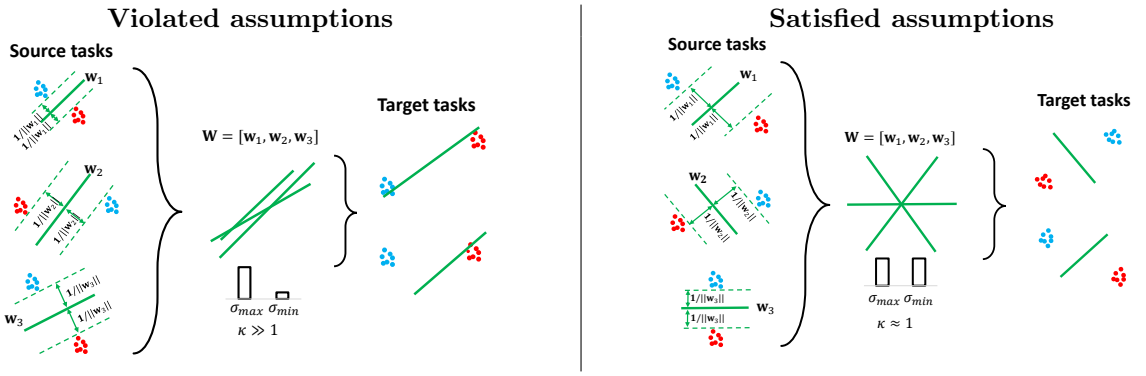

FIGURE 2 – Illustration of the intuition behind the assumptions derived from the MTR learning theory. **(left)** Lack of diversity and increasing norm of the linear predictors restrict them from being useful on the target task. **(right)** When the assumptions are satisfied, the linear predictors cover the embedding space evenly and their norm remains roughly constant on source tasks making them useful for a previously unseen task.

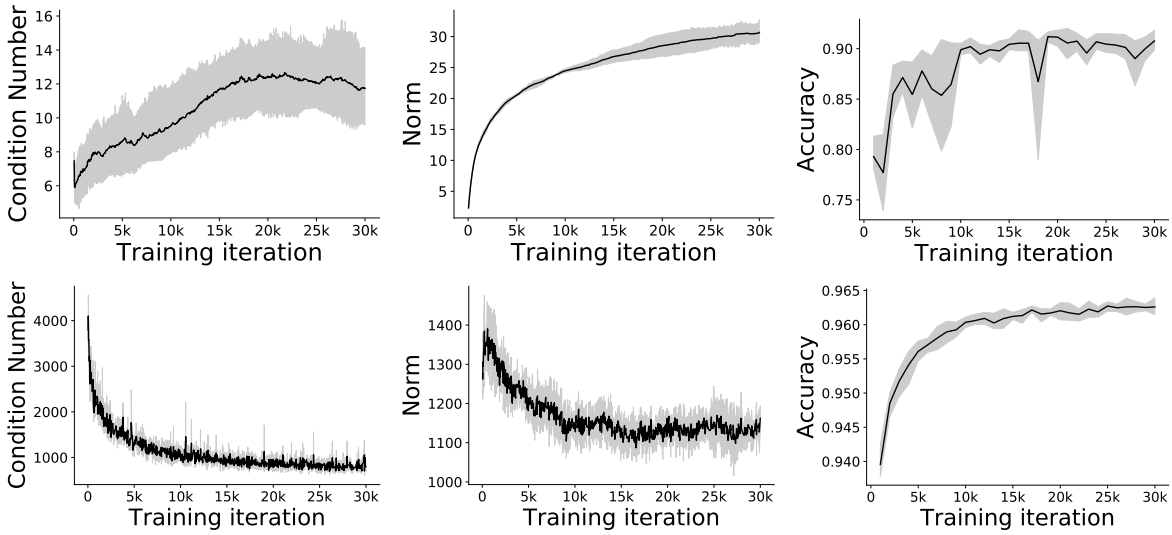

FIGURE 3 – Evolution of κ (*left*), $\|\mathbf{W}_N\|_F$ (*middle*), and validation accuracy (*right*) during training of MAML (*top*) and PROTONET (*bottom*) on Omniglot with 20-way 1-shot episodes. All training curves are averaged over 4 different random seeds. The light gray area shows 95% confidence intervals.

on $\mathcal{S}^d$. $\forall \mu \in \mathcal{M}(\mathcal{S}^d), u \in \mathcal{S}^d$, we further define the continuous and Borel measurable function :

$$U_\mu(u) := \int_{\mathcal{S}^d} \exp(2u^\top v) d\mu(v).$$

Then, we can write the second term as

$$\begin{aligned} &\mathbb{E}_{\mathbf{q}\sim Q}\left[\log \mathbb{E}_{\mathbf{c}\sim C\circ\phi^{-1}}\left[\exp\left(2\phi(\mathbf{c})^\top \phi(\mathbf{q})\right)\right]\right] \\ &= \mathbb{E}_{\mathbf{q}\sim Q}\left[\log U_{C\circ\phi^{-1}}(\phi(\mathbf{q}))\right], \end{aligned}$$

where C is the distribution of prototypes of S, *i.e.* each data point in C is the mean of all the points in S that share the same label, and $C \circ \phi^{-1}$ is the probability measure of prototypes, *i.e.* the pushforward measure of C via ϕ.

We now consider the following problem :

$$\min_{\mu\in\mathcal{M}(\mathcal{S}^d)} \int_{\mathcal{S}^d} \log U_\mu(u) d\mu(u). \tag{4}$$

The unique minimizer of Eq. 4 is the *uniform distribution on* $\mathcal{S}^d$, as shown in [WI20]. This means that learning with $\mathcal{L}_{proto}$ leads to prototypes uniformly distributed in the embedding space and thus with $\kappa = 1$. □

C.2 Proof of Proposition 1

Démonstration. We follow [AIS21] and note that in the considered setup the gradient of the loss for each task is given by

$$\frac{\partial \ell_t(\widehat{\mathbf{w}} - \alpha \nabla \ell_t(\boldsymbol{\theta}))}{\partial \widehat{\mathbf{w}}} \propto (1-\alpha)^2(\widehat{\mathbf{w}}_t - \boldsymbol{\theta}_t)$$

so that the meta-training update for a single gradient step becomes :

$$\widehat{\mathbf{w}}_t \leftarrow \widehat{\mathbf{w}}_{t-1} - \beta(1-\alpha)^2(\widehat{\mathbf{w}}_{t-1} - \boldsymbol{\theta}_t),$$

where β is the meta-training update learning rate. Starting at $\widehat{\mathbf{w}}_0 = \mathbf{0}_d$, we have that

$$\begin{aligned}
\widehat{\mathbf{w}}_1 &= c\boldsymbol{\theta}_1, \\
\widehat{\mathbf{w}}_2 &= c((c-1)\boldsymbol{\theta}_1 + \boldsymbol{\theta}_2), \\
&\dots \\
\widehat{\mathbf{w}}_n &= c\sum_{i=1}^{n} \boldsymbol{\theta}_i (c-1)^{n-i},
\end{aligned}$$

where $c := \beta(1-\alpha)^2$. We can now define matrices $\widehat{\mathbf{W}}_2^i$ as follows :

$$\begin{aligned}
\widehat{\mathbf{W}}_2^1 &= \begin{pmatrix} c\boldsymbol{\theta}_1, \\ c((c-1)\boldsymbol{\theta}_1 + \boldsymbol{\theta}_2) \end{pmatrix}, \\
\widehat{\mathbf{W}}_2^2 &= \begin{pmatrix} c((c-1)\boldsymbol{\theta}_1 + \boldsymbol{\theta}_2), \\ c((c-1)^2\boldsymbol{\theta}_1 + (c-1)\boldsymbol{\theta}_2 + \boldsymbol{\theta}_3) \end{pmatrix}, \\
&\dots \\
\widehat{\mathbf{W}}_2^n &= \begin{pmatrix} c\sum_{i=1}^{n} \boldsymbol{\theta}_i (c-1)^{n-i}, \\ c\sum_{i=1}^{n+1} \boldsymbol{\theta}_i (c-1)^{n-i} \end{pmatrix}.
\end{aligned}$$

We can note that for all $i > 1$:

$$\widehat{\mathbf{W}}_2^{i+1} = (c-1)\widehat{\mathbf{W}}_2^i + c\boldsymbol{\Theta}_i.$$

Now, we can write :

$$\begin{aligned}
\kappa(\widehat{\mathbf{W}}_2^{i+1}) = \frac{\sigma_1(\widehat{\mathbf{W}}_2^{i+1})}{\sigma_2(\widehat{\mathbf{W}}_2^{i+1})} &= \frac{\sigma_1((c-1)\widehat{\mathbf{W}}_2^i + c\boldsymbol{\Theta}_i)}{\sigma_2((c-1)\widehat{\mathbf{W}}_2^i + c\boldsymbol{\Theta}_i)} \\
&\geq \frac{\sigma_1((c-1)\widehat{\mathbf{W}}_2^i) - \sigma_2(c\boldsymbol{\Theta}_i)}{\sigma_2((c-1)\widehat{\mathbf{W}}_2^i + c\boldsymbol{\Theta}_i)} \\
&\geq \frac{\sigma_1((c-1)\widehat{\mathbf{W}}_2^i) - \sigma_2(c\boldsymbol{\Theta}_i)}{\sigma_2((c-1)\widehat{\mathbf{W}}_2^i) + \sigma_2(c\boldsymbol{\Theta}_i)} \\
&\geq \kappa(\widehat{\mathbf{W}}_2^i).
\end{aligned}$$

where the second and third lines follow from the inequalities for singular values $\sigma_1(A+B) \leq \sigma_1(A) + \sigma_2(B)$ and $\sigma_i(A+B) \geq \sigma_i(A) - \sigma_{\min}(B)$ and the desired result is obtained by setting $\sigma_{\min}(\boldsymbol{\theta}_i) = 0$. □

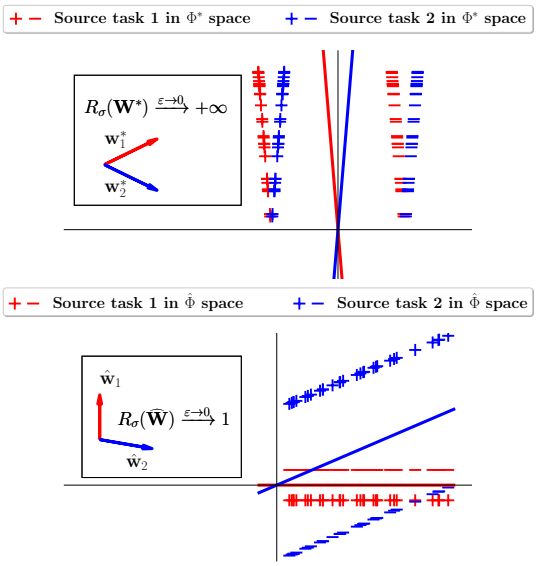

FIGURE 4 – Illustration of Proposition 2 and the construction used in its proof.

C.3 Proof of Proposition 2

Démonstration. Let us define two uniform distributions μ_1 and μ_2 parametrized by a scalar $\varepsilon > 0$ satisfying the data generating process from Eq. 1 :

1. μ_1 is uniform over $\{1 - k\varepsilon, k, 1, \underbrace{\dots}_{d-3}\} \times \{1\} \cup \{1 + k\varepsilon, k, -1, \underbrace{\dots}_{d-3}\} \times \{-1\}$;
2. μ_2 is uniform over $\{1 + k\varepsilon, k, \frac{k-1}{\varepsilon}, \underbrace{\dots}_{d-3}\} \times \{1\} \cup \{-1 + k\varepsilon, k, \frac{1+k}{\varepsilon}, \underbrace{\dots}_{d-3}\} \times \{-1\}$.

where last $d-3$ coordinates of the generated instances are arbitrary numbers. We now define the optimal representation and two optimal predictors for each distribution as the solution to the MTR problem over the two data generating distributions and $\Phi = \{\phi |\ \phi(\mathbf{x}) = \boldsymbol{\Phi}^T\mathbf{x},\ \boldsymbol{\Phi} \in \mathbb{R}^{d\times 2}\}$:

$$\phi^*, \mathbf{W}^* = \underset{\phi\in\Phi, \mathbf{W}\in\mathbb{R}^{2\times 2}}{\arg\min} \sum_{i=1}^{2} \underset{(\mathbf{x},y)\sim\mu_i}{\mathbb{E}} \ell(y, \langle \mathbf{w}_i, \phi(\mathbf{x})\rangle), \quad (5)$$

One solution to this problem can be given as follows :

$$\boldsymbol{\Phi}^* = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \end{pmatrix}^T, \quad \mathbf{W}^* = \begin{pmatrix} 1 & \varepsilon \\ 1 & -\varepsilon \end{pmatrix},$$

where $\boldsymbol{\Phi}^*$ projects the data generated by μ_i to a two-dimensional space by discarding its $d-2$ last dimensions

and the linear predictors satisfy the data generating process from Eq. 1 with $\varepsilon = 0$. One can verify that in this case $\mathbf{W}^*$ have singular values equal to $\sqrt{2}$ and $\sqrt{2}\varepsilon$, and $\kappa(\mathbf{W}^*) = \frac{1}{\varepsilon}$. When $\varepsilon \to 0$, the optimal predictors make the ratio arbitrary large thus violating Assumption 1.

Let us now consider a different problem where we want to solve Eq. 5 with constraints that force linear predictors to satisfy both assumptions :

$$\widehat{\phi}, \widehat{\mathbf{W}} = \underset{\phi \in \Phi, \mathbf{W} \in \mathbb{R}^{2\times 2}}{\arg\min} \sum_{i=1}^{2} \underset{(\mathbf{x},y)\sim\mu_i}{\mathbb{E}} \ell(y, \langle \mathbf{w}_i, \phi(\mathbf{x})\rangle), \quad \text{s.t. } \kappa(\mathbf{W}) \approx 1 \quad \text{and} \quad \forall i, \quad \|\mathbf{w}_i\| \approx 1. \tag{6}$$

Its solution is different and is given by

$$\widehat{\mathbf{\Phi}} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \end{pmatrix}^T, \quad \widehat{\mathbf{W}} = \begin{pmatrix} 0 & 1 \\ 1 & -\varepsilon \end{pmatrix}.$$

Similarly to $\mathbf{\Phi}^*$, $\widehat{\mathbf{\Phi}}$ projects to a two-dimensional space by discarding the first and last $d-3$ dimensions of the data generated by μ_i. The learned predictors in this case also satisfy Eq. 1 with $\varepsilon = 0$, but contrary to $\mathbf{W}^*$, $\kappa(\widehat{\mathbf{W}}) = \sqrt{\frac{2+\varepsilon^2+\varepsilon\sqrt{\varepsilon^2+4}}{2+\varepsilon^2-\varepsilon\sqrt{\varepsilon^2+4}}}$ tends to 1 when $\varepsilon \to 0$. □

D Related work

Understanding meta-learning [RRBV20] investigate whether MAML algorithm works well due to rapid learning with significant changes in the representations when deployed on target task, or due to feature reuse where the learned representation remains almost intact. They establish that the latter factor is dominant and propose a new variation of MAML that freezes all but task-specific layers of the neural network when learning new tasks. In [GRF+20], the authors explain the success of meta-learning approaches by their capability to either cluster classes more tightly in feature space (task-specific adaptation approach), or to search for meta-parameters that lie close in weight space to many task-specific minima (full fine-tuning approach). Finally, the effect of the number of shots on the classification accuracy was studied in [CLF20] for PROTONET algorithm. Our paper is complementary to all other works mentioned above as it investigates a new aspect of meta-learning that has never been studied before and provides a more complete experimental evaluation with the two different approaches of meta-learning, separately presented in [RRBV20], [CLF20] and [GRF+20].

Common regularization strategies Even though our work does not aim at proposing a new regularization strategy for meta-learning, regularizing the condition number of the matrix of linear predictors and its norm as suggested by MTR theory appears to be novel and drastically different from existing regularization strategies. In general, we note that regularization in meta-learning (i) is applied to either the weights of the whole neural network [BSC18, YTZ+20], or (ii) the predictions [JQ19, GRF+20] or (iii) is introduced via a prior hypothesis biased regularized empirical risk minimization [PL14, KO17, DCSP18a, DCSP18b, DCGP19]. Contrary to the first group of methods and the famous weight decay approach [KH92], we do not regularize the whole weight matrix learned by the neural network but the linear predictors of its last layer. The purpose of the regularization in our case is also completely different : weight decay is used to avoid overfitting by penalizing large magnitudes of weights, while our goal is to keep the classification margin unchanged during the training to avoid over-/under-specialization to some source tasks. Similarly, spectral normalization proposed by [MKKY18] to satisfy the Lipschitz constraint in GANs through dividing $\mathbf{W}$ values by $\sigma_{\max}(\mathbf{W})$ does not affect the ratio between $\sigma_{\max}(\mathbf{W})$ and $\sigma_{\min}(\mathbf{W})$ and serves a completely different purpose. Second, we regularize the singular values of the matrix of linear predictors obtained in the last batch of tasks instead of the predictions used by the methods of the second group (*e.g.*, using the theoretic-information quantities in [JQ19]). Finally, the works of the last group are related to the online setting with convex loss functions only, and, similarly to the algorithms from the second group, do not specifically target the spectral properties of the learned predictors.

E Detailed performance comparisons

Table 2 provides the detailed performance of our reproduced methods with and without our regularization or normalization and Figure 5 shows the performance gap throughout training for all methods on miniImageNet. Theses results are summarized in Table 1 of our paper and discussions about them can be found in Section 4.2 and 4.3. We can see on both Table 2 and Figure 5 that the gap is globally positive throughout the training on both validation and test sets, which shows the increased generalization capabilities of enforcing the assumptions. There is also generally a high gap at the beginning of training suggesting faster learning.

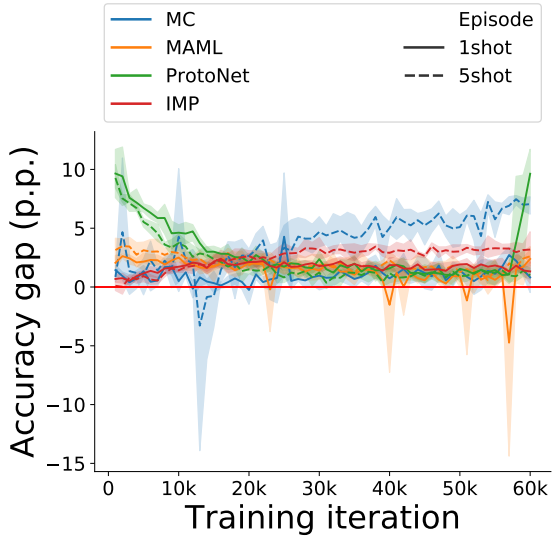

FIGURE 5 – Performance gap (in p.p.) when applying regularization for gradient-based and normalization for metric-based methods throughout the training process on 5-way 1-shot and 5-shot episodes on miniImageNet (*better viewed in color*). Each data point is averaged over 2400 validation episodes and 4 different seeds and reported with 95% confidence interval. We can see that the gap is globally positive throughout training and generally higher at the beginning of training. The increase in the gap at the end of training is linked to a lower overfitting.

F Intended effect of the regularization/normalization

Figure 6 provides the detailed evolution of κ and $\|\mathbf{W}_N\|_F$ during training for all methods on miniImageNet with 5-way 1-shot episodes. Adding our regularization for gradient-based or normalization for metric-based to enforce the assumptions has the intended effect of both terms as explained in Section 4. We can see that κ and $\|\mathbf{W}_N\|_F$ remain constant and bounded throughout the training.

G Further enforcing a low condition number on Metric-based methods

A first idea for further enforcing a low condition number for metric-based methods would be to regularize the norm and condition number of the prototypes in the same way as Gradient-based methods. Unfortunately, this latter strategy hinders the convergence of the network and leads to numerical instabilities. Most likely this is explained by prototypes being computed from image features which suffer from rapid changes across batches making the smallest singular value $\sigma_N(\mathbf{W}_N)$ close to 0. Consequently, we propose to replace the ratio of the vector of singular values by its entropy as follows :

$$H_\sigma(\mathbf{W}_N) := \\ -\sum_{i=1}^{N} \text{softmax}(\sigma(\mathbf{W}_N))_i \cdot \log \text{softmax}(\sigma(\mathbf{W}_N))_i,$$

where $\text{softmax}(\cdot)_i$ is the i^{th} output of the softmax function. As with κ, we write H_σ instead of $H_\sigma(\mathbf{W}_N)$ from now on. Since uniform distribution has the highest entropy, regularizing with κ or $-H_\sigma$ leads to a better coverage of $\mathbb{R}^k$ by ensuring a nearly identical importance regardless of the direction. Then, to ensure Assumption 2 and following Theorem 1, we also normalize the prototypes. We obtain the following regularized optimization problem :

$$\widehat{\phi}, \widehat{\mathbf{W}} = \underset{\phi \in \Phi, \mathbf{W} \in \mathbb{R}^{T \times k}}{\arg\min} \frac{1}{Tn_1} \sum_{t=1}^{T} \sum_{i=1}^{n_1} \ell(y_{t,i}, \langle \tilde{\mathbf{w}}_t, \phi(\mathbf{x}_{t,i}) \rangle) \\ - \lambda_1 H_\sigma(\mathbf{W}), \tag{7}$$

where $\tilde{\mathbf{w}} = \frac{\mathbf{w}}{\|\mathbf{w}\|}$ are the normalized prototypes.

In Table 3, we report the performance of our reproduced PROTONET without normalization, with normalization and with both normalization and regularization on the entropy. As mentioned in Section 4.2 of our paper, we can see that further enforcing a regularization on the singular values (through the entropy) does not help the training since PROTONET naturally learns to minimize the singular values of the prototypes. In Table 4, we show that reducing the *strength* of the regularization with the entropy can help retrieve good performance.

H Ablative studies

In the following, we include ablative studies on the effect of each term in our regularization scheme for gradient-based methods to complete results given in Section 4.3 of our paper. In Table 5, we compared the performance of our reproduced MAML without regularization ($\lambda_1 = \lambda_2 = 0$), with a regularization on the condition number κ ($\lambda_1 = 1$ and $\lambda_2 = 0$), on the norm of the linear predictors ($\lambda_1 = 0$ and $\lambda_2 = 1$), and with both regularization terms ($\lambda_1 = \lambda_2 = 1$) on Omniglot and miniImageNet. We can see that both regularization

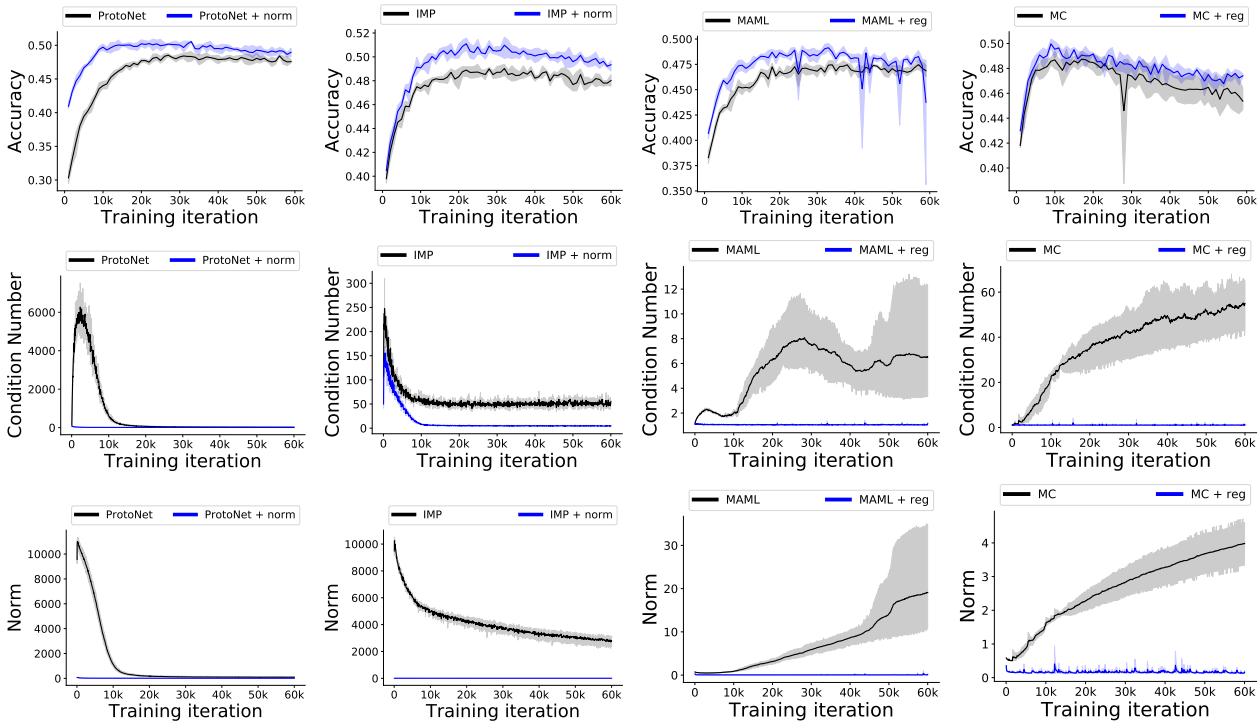

FIGURE 6 – Evolution of accuracy of 5-way 1-shot meta-test episodes from *miniImageNet* during meta-training on 5-way 1-shot episodes from *miniImageNet*, for PROTONET, IMP(*top*) and MAML, MC (*bottom*) (in black) and their regularized or normalized counterparts (in blue).

terms are important in the training and that using only a single term can be detrimental to the training results.

TABLE 2 – Performance of several meta-learning algorithms without and with our regularization (or normalization in the case of PROTONET and IMP) to enforce the theoretical assumptions. All accuracy results (in %) are averaged over 2400 test episodes and 4 different seeds and are reported with 95% confidence interval. Episodes are 20-way classification for Omniglot and 5-way classification for miniImageNet and tieredImageNet.

Method	Dataset	Episodes	without Reg./Norm.	with Reg./Norm.
PROTONET	Omniglot	1-shot	$95.56 \pm 0.10\%$	$\mathbf{95.89 \pm 0.10}\%$
		5-shot	$\mathbf{98.80 \pm 0.04}\%$	$\mathbf{98.80 \pm 0.04}\%$
	miniImageNet	1-shot	$49.53 \pm 0.41\%$	$\mathbf{50.29 \pm 0.41}\%$
		5-shot	$65.10 \pm 0.35\%$	$\mathbf{67.13 \pm 0.34}\%$
	tieredImageNet	1-shot	$51.95 \pm 0.45\%$	$\mathbf{54.05 \pm 0.45}\%$
		5-shot	$\mathbf{71.61 \pm 0.38}\%$	$\mathbf{71.84 \pm 0.38}\%$
IMP	Omniglot	1-shot	$\mathbf{95.77 \pm 0.20}\%$	$\mathbf{95.85 \pm 0.20}\%$
		5-shot	$\mathbf{98.77 \pm 0.08}\%$	$\mathbf{98.83 \pm 0.07}\%$
	miniImageNet	1-shot	$48.85 \pm 0.81\%$	$\mathbf{50.69 \pm 0.80}\%$
		5-shot	$66.43 \pm 0.71\%$	$\mathbf{67.29 \pm 0.68}\%$
	tieredImageNet	1-shot	$52.16 \pm 0.89\%$	$\mathbf{53.46 \pm 0.89}\%$
		5-shot	$\mathbf{71.79 \pm 0.75}\%$	$\mathbf{72.38 \pm 0.75}\%$
MAML	Omniglot	1-shot	$91.72 \pm 0.29\%$	$\mathbf{95.67 \pm 0.20}\%$
		5-shot	$97.07 \pm 0.14\%$	$\mathbf{98.24 \pm 0.10}\%$
	miniImageNet	1-shot	$47.93 \pm 0.83\%$	$\mathbf{49.16 \pm 0.85}\%$
		5-shot	$64.47 \pm 0.69\%$	$\mathbf{66.43 \pm 0.69}\%$
	tieredImageNet	1-shot	$50.08 \pm 0.91\%$	$\mathbf{51.5 \pm 0.90}\%$
		5-shot	$67.5 \pm 0.79\%$	$\mathbf{70.16 \pm 0.76}\%$
MC	Omniglot	1-shot	$\mathbf{96.56 \pm 0.18}\%$	$95.95 \pm 0.20\%$
		5-shot	$\mathbf{98.88 \pm 0.08}\%$	$98.78 \pm 0.08\%$
	miniImageNet	1-shot	$\mathbf{49.28 \pm 0.83}\%$	$\mathbf{49.64 \pm 0.83}\%$
		5-shot	$63.74 \pm 0.69\%$	$\mathbf{65.67 \pm 0.70}\%$
	tieredImageNet	1-shot	$55.16 \pm 0.94\%$	$\mathbf{55.85 \pm 0.94}\%$
		5-shot	$71.95 \pm 0.77\%$	$\mathbf{73.34 \pm 0.76}\%$

TABLE 3 – Performance of PROTONET with and without our regularization on the entropy and/or normalization. All accuracy results (in %) are averaged over 2400 test episodes and 4 different random seeds and are reported with 95% confidence interval. Further enforcing regularization on the singular values can be detrimental to performance.

Dataset	Episodes	without Norm., $\lambda_1 = 0$	with Norm., $\lambda_1 = 0$	with Norm., $\lambda_1 = 1$
Omniglot	20-way 1-shot	$95.56 \pm 0.10\%$	$\mathbf{95.89 \pm 0.10}\%$	$91.90 \pm 0.14\%$
	20-way 5-shot	$\mathbf{98.80 \pm 0.04}\%$	$\mathbf{98.80 \pm 0.04}\%$	$96.40 \pm 0.07\%$
miniImageNet	5-way 1-shot	$49.53 \pm 0.41\%$	$\mathbf{50.29 \pm 0.41}\%$	$49.43 \pm 0.40\%$
	5-way 5-shot	$65.10 \pm 0.35\%$	$\mathbf{67.13 \pm 0.34}\%$	$65.71 \pm 0.35\%$
tieredImageNet	5-way 1-shot	$51.95 \pm 0.45\%$	$\mathbf{54.05 \pm 0.45}\%$	$53.54 \pm 0.44\%$
	5-way 5-shot	$\mathbf{71.61 \pm 0.38}\%$	$\mathbf{71.84 \pm 0.38}\%$	$70.30 \pm 0.40\%$

TABLE 4 – Ablative study on the strength of the regularization with normalized PROTONET. All accuracy results (in %) are averaged over 2400 test episodes and 4 random seeds and are reported with 95% confidence interval.

Dataset	Episodes	Reproduced	$\lambda_1 = 0$	$\lambda_1 = 1$	$\lambda_1 = 0.1$	$\lambda_1 = 0.01$	$\lambda_1 = 0.001$	$\lambda_1 = 0.0001$
miniImageNet	5-way 1-shot	$49.53 \pm 0.41\%$	$\mathbf{50.29 \pm 0.41}\%$	$49.43 \pm 0.40\%$	$\mathbf{50.19 \pm 0.41}\%$	$\mathbf{50.44 \pm 0.42}\%$	$\mathbf{50.46 \pm 0.42}\%$	$\mathbf{50.45 \pm 0.42}\%$
	5-way 5-shot	$65.10 \pm 0.35\%$	$\mathbf{67.13 \pm 0.34}\%$	$65.71 \pm 0.35\%$	$66.69 \pm 0.36\%$	$66.69 \pm 0.34\%$	$\mathbf{67.2 \pm 0.35}\%$	$\mathbf{67.12 \pm 0.35}\%$
Omniglot	20-way 1-shot	$95.56 \pm 0.10\%$	$\mathbf{95.89 \pm 0.10}\%$	$91.90 \pm 0.14\%$	$94.38 \pm 0.12\%$	$95.60 \pm 0.10\%$	$95.7 \pm 0.10\%$	$95.77 \pm 0.10\%$
	20-way 5-shot	$98.80 \pm 0.04\%$	$98.80 \pm 0.04\%$	$96.40 \pm 0.07\%$	$97.93 \pm 0.05\%$	$98.62 \pm 0.04\%$	$98.76 \pm 0.04\%$	$\mathbf{98.91 \pm 0.03}\%$

TABLE 5 – Ablative study of the regularization parameter for MAML on Omniglot and miniImageNet. All accuracy results (in %) are averaged over 2400 test episodes and 4 different random seeds and are reported with 95% confidence interval. Using both regularization terms is important.

Dataset	Episodes	$\lambda_1 = \lambda_2 = 0$	$\lambda_1 = 1, \lambda_2 = 0$	$\lambda_1 = 0, \lambda_2 = 1$	$\lambda_1 = \lambda_2 = 1$
Omniglot	20-way 1-shot	$91.72 \pm 0.29\%$	$89.86 \pm 0.31\%$	$92.80 \pm 0.26\%$	$\mathbf{95.67 \pm 0.20}\%$
	20-way 5-shot	$97.07 \pm 0.14\%$	$72.47 \pm 0.17\%$	$96.99 \pm 0.14\%$	$\mathbf{98.24 \pm 0.10}\%$
miniImageNet	5-way 1-shot	$47.93 \pm 0.83\%$	$47.76 \pm 0.84\%$	$48.27 \pm 0.81\%$	$\mathbf{49.16 \pm 0.85}\%$
	5-way 5-shot	$64.47 \pm 0.69\%$	$64.44 \pm 0.68\%$	$64.16 \pm 0.72\%$	$\mathbf{66.43 \pm 0.69}\%$