Appendix
Improving Few-Shot Learning through Multi-task Representation Learning Theory

Quentin Bouniot[1,2][0000−0002−0982−372X], Ievgen Redko[2][0000−0002−3860−5502], Romaric Audigier[1][0000−0002−4757−2052], Angélique Loesch[1][0000−0001−5427−3010], Amaury Habrard[2,3][0000−0003−3038−9347]

[1] Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
{firstname.lastname}@cea.fr
[2] Université de Lyon, UJM-Saint-Etienne, CNRS, IOGS, Laboratoire Hubert Curien UMR 5516, F-42023, Saint-Etienne, France
{firstname.lastname}@univ-st-etienne.fr
[3] Institut Universitaire de France (IUF)

The supplementary material is organized as follows. Section A provides an additional review on Multi-task Representation Learning Theory. Section B provides the full proofs of the theoretical results discussed in the paper. Section C gives more details behind the proposed regularization terms. Section D describes the full experimental setup with all the hyperparameters used. Section E gives the detailed results used to derive Figure 3 in the paper. Section F provides more experiments showing that further enforcing the condition number assumption for ProtoNet is unfavorable. In Section G, we study the effect of each term in the proposed regularization. Section H provides a preliminary analysis in the out-of-domain setting.

A Review of Multi-task Representation Learning Theory

We formulate the main results of the three main theoretical analyses of Multi-task Representation (MTR) Learning Theory provided in [3, 10, 17] in Table 1 to give additional details for Sections 2.2 and 4.4 of our paper.

One may note that all the assumptions presented in this table can be roughly categorized into two groups. First one consists of the assumptions related to the data generating process (A1, A2.1, A2.4-7 and A3.1), technical assumptions required for the manipulated empirical quantities to be well-defined (A2.6) and assumptions specifying the learning setting (A3.3-4). We put them together as they are not directly linked to the quantities that we optimize over in order to solve the meta-learning problem. The second group of assumptions include A2.2 and A3.2: both defined as a measure of diversity between source tasks' predictors that are expected to cover all the directions of $\mathbb{R}^k$ evenly. These assumptions is of primary interest as it involves the matrix of predictors optimized in Eq. 1 as thus one can attempt to force it in order for $\widehat{\mathbf{W}}$ to have the desired properties.

Finally, we note that assumption A2.2 related to the covariance dominance can be seen as being at the intersection between the two groups. On the one

Table 1: Overview of main theoretical contributions related to MTR learning with their assumptions, considered classes of representations and the obtained bounds on the excess risk. Here $\tilde{O}(\cdot)$ hides logarithmic factors.

Paper	Assumptions	Φ	Bound
[10]	**A1**. $\forall t \in [[T+1]],\ \mu_t \sim \eta$	–	$O\left(\frac{1}{\sqrt{n_1}} + \frac{1}{\sqrt{T}}\right)$
[3]	**A2.0**. $\forall t,\ \|\mathbf{w}_t^*\| = \Theta(1)$ **A2.1**. $\forall t,\ \bar{\mathbf{x}}$ is ρ^2-subgaussian **A2.2**. $\forall t \in [[T]], \exists c > 0 : \Sigma_t \succeq c\Sigma_{T+1}$ **A2.3**. $\frac{\sigma_1(\mathbf{W}^*)}{\sigma_k(\mathbf{W}^*)} = O(1)$ **A2.4**. $\mathbf{w}_{T+1}^* \sim \mu_{\mathbf{w}} : \|\mathbb{E}_{\mathbf{w}\sim\mu_{\mathbf{w}}}[\mathbf{w}\mathbf{w}^T]\| \leq O(\frac{1}{k})$ **A2.5**. $\forall t,\ p_t = p, \Sigma_t = \Sigma$ **A2.6**. Point-wise+unif. cov. convergence **A2.7**. Teacher network	**A2.1-2.4**, linear, $k \ll d$ **A2.3-2.5**, general, $k \ll d$ **A2.1,2.5,2.6**, linear + ℓ_2 regul., $k \gg d$ **A2.1,2.5,2.6,2.7**, two-layer NN (ReLUs+ ℓ_2 regul.)	$O\left(\frac{kd}{cn_1T} + \frac{k}{n_2}\right)$ $O\left(\frac{\mathcal{C}(\Phi)}{n_1T} + \frac{k}{n_2}\right)$ $\sigma\bar{R}\tilde{O}\left(\frac{\sqrt{\mathrm{Tr}(\Sigma)}}{\sqrt{n_1T}} + \frac{\sqrt{\|\Sigma\|_2}}{\sqrt{n_2}}\right)$ $\sigma\bar{R}\tilde{O}\left(\frac{\sqrt{\mathrm{Tr}(\Sigma)}}{\sqrt{n_1T}} + \frac{\sqrt{\|\Sigma\|_2}}{\sqrt{n_2}}\right)$
[17]	**A3.1**. $\forall t,\ \mathbf{x} \sim \mu_{\mathbb{X}_t}$ is ρ^2-subgaussian **A3.2**. $\frac{\sigma_1(\mathbf{W}^*)}{\sigma_k(\mathbf{W}^*)} = O(1)$ and $\forall t, \|\mathbf{w}_t\| = \Theta(1)$ **A3.3**. $\widehat{\mathbf{W}}$ learned using the Method of Moments **A3.4**. $\mathbf{w}_{T+1}^*$ is learned using Linear Regression	**A1-4**, linear, $k \ll d$	$\bar{O}\left(\frac{kd}{n_1T} + \frac{k}{n_2}\right)$

hand, this assumption is related to the population covariance and thus is related to the data generating process that is supposed to be fixed. On the other hand, we can think about a pre-processing step that precedes the meta-train step of the algorithm and transforms the source and target tasks' data so that their sample covariance matrices satisfy A2.2. While presenting a potentially interesting research direction, it is not clear how this can be done in practice especially under a constraint of the largest value of c required to minimize the bound. [3] circumvent this problem by adding A2.5, stating that the task data marginal distributions are similar.

An intuition behind the main assumptions studied in this paper (Assumption 1 and 2 in this paper, and A2.0, A2.3, A3.2 in Table 1) can be seen in Figure 1. When the assumptions do not hold, the linear predictors can be biased towards a single part of the space and over-specialized to the tasks. 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 more easily adapt to the target tasks and achieve better generalization.

B Full proofs

B.1 Proof of Theorem 1

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:

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

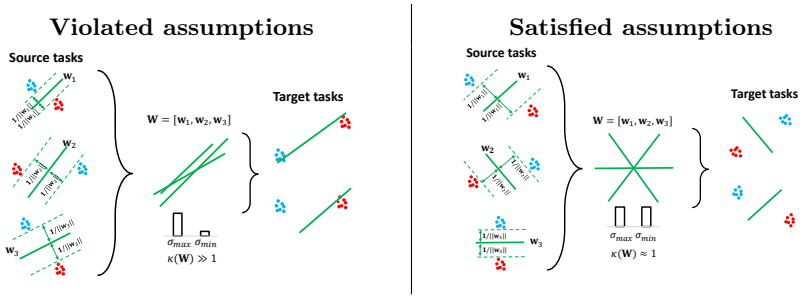

Fig. 1: Illustration of the intuition behind the assumptions derived from the MTR learning theory. **(left)** Lack of diversity and increasing norm of the linear predictors restrict them from being useful on the target task. **(right)** When the assumptions are satisfied, the linear predictors cover the embedding space evenly and their norm remains roughly constant on source tasks making them useful for a previously unseen task.

$$= \underbrace{\mathbb{E}_{(\mathbf{q},i)\sim Q}\left[d(\phi(\mathbf{q}), \mathbf{c}_i)\right]}_{(1)} + \underbrace{\mathbb{E}_{\mathbf{q}\sim Q} \log \sum_{j=1}^{n} \exp\left(-d(\phi(\mathbf{q}), \mathbf{c}_j)\right)}_{(2)}$$

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:

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

Now we can recall Theorem 1:

Theorem 1. (Normalized ProtoNet) *If* $\forall i \ \|\mathbf{c}_i\| = 1$,
then $\forall \hat{\phi} \in \arg\min_\phi \mathcal{L}_{proto}(S, Q, \phi)$, *the matrix of the optimal prototypes* $\mathbf{W}^*$ *is well-conditioned,* i.e. $\kappa(\mathbf{W}^*) = O(1)$.

Proof. 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:

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

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 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:

$$\min_{\mu\in\mathcal{M}(\mathcal{S}^d)}\int_{\mathcal{S}^d}\log U_\mu(u)d\mu(u). \tag{1}$$

The unique minimizer of Eq. 1 is the *uniform distribution on* $\mathcal{S}^d$, as shown in [19]. This means that learning with $\mathcal{L}_{proto}$ leads to prototypes uniformly distributed in the embedding space. By considering $\mathbf{W}^*$ the matrix of the optimal prototypes for each task then $\mathbf{W}^*$ is *well-conditioned*, *i.e.* $\kappa(\mathbf{W}^*)=O(1)$.

B.2 Proof of Proposition 1

Let us recall the learning model of interest and Proposition 1:

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

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:*

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

Proof. We follow [1] 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

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

$$\widehat{\mathbf{w}}_n = c\sum_{i=1}^{n} \boldsymbol{\theta}_i (c-1)^{n-i},$$

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 $\sigma_{\min}(\boldsymbol{\theta}_i) = 0$.

B.3 Proof of Proposition 2

Let us first recall Proposition 2:

Proposition 2. *If* $\forall t \in [[T]], \|\mathbf{w}_t^*\| = O(1)$ *and* $\kappa(\mathbf{W}^*) = O(1)$*, and* $\mathbf{w}_{T+1}$ *follows a distribution* ν *such that* $\|\mathbb{E}_{\mathbf{w}\sim\nu}[\mathbf{w}\mathbf{w}^\top]\| \leq O\left(\frac{1}{k}\right)$*, then*

$$ER(\hat{\phi}, \hat{\mathbf{w}}_{T+1}) \leq O\left(\frac{C(\Phi)}{n_1 T} \cdot \kappa(\mathbf{W}^*) + \frac{k}{n_2}\right). \tag{3}$$

Proof. Du et al. [3] assume that $\sigma_k(\mathbf{W}^*) \gtrsim \frac{T}{k}$ (Assumption 4.3 in their work). However, since we also have $\|\mathbf{w}_t^*\| = O(1)$, it is equivalent to $\frac{\sigma_1(\mathbf{W}^*)}{\sigma_k(\mathbf{W}^*)} = O(1)$. We have $\sigma_1(\mathbf{W}^*) \gtrsim \sigma_k(\mathbf{W}^*) \gtrsim \frac{T}{k}$ and then $\frac{\sigma_1(\mathbf{W}^*)}{T \cdot \sigma_k(\mathbf{W}^*)} = \frac{1}{T} \cdot \kappa(\mathbf{W}^*) \gtrsim \frac{1}{k \cdot \sigma_k(\mathbf{W}^*)}$ which we use in their proof of Theorem 5.1 instead of $\frac{1}{T} \gtrsim \frac{1}{k \cdot \sigma_k(\mathbf{W}^*)}$ to obtain the desired result.

B.4 Proof of Proposition 3

Let us recall the data generating process and Proposition 3:

$$\begin{aligned} &\forall t \in [[T+1]] \text{ and } (\mathbf{x}, y) \sim \mu_t, \\ &\quad y = \langle \mathbf{w}_t^*, \phi^*(\mathbf{x}) \rangle + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2). \end{aligned} \tag{4}$$

Proposition 3. *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. 4 with $\kappa(\widehat{\mathbf{W}}) \approx 1$ and $\kappa(\mathbf{W}^*) \gg 1$.*

Proof. Let us define two uniform distributions μ_1 and μ_2 parametrized by a scalar $\varepsilon > 0$ satisfying the data generating process from Eq. 4:

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 $\Phi = \{\phi |\ \phi(\mathbf{x}) = \boldsymbol{\Phi}^T \mathbf{x},\ \boldsymbol{\Phi} \in \mathbb{R}^{d \times 2}\}$:

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

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. 4 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:

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

Its solution is different and is given by

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

Similarly to $\boldsymbol{\Phi}^*$, $\widehat{\boldsymbol{\Phi}}$ 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. 4 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 $\varepsilon \to 0$. The construction used in this proof is illustrated in Figure 2

C More Details on the regularization terms

By adding $\|\mathbf{W}\|_F^2$ in the loss, we force the model to have a low norm on the weights. Since it cannot be put to 0 or below, the model will keep the norm relatively constant instead of increasing it. The second regularizer term is a softer way to apply the constraint on the norm rather than considering normalized weights as in Eq. 6.

According to Theorem 7.1 from [9], subgradients of singular values function are well-defined for absolutely symmetric functions. In our case, we are computing in practice the squared singular values $\sigma^2(\mathbf{W})$ and we retrieve the singular values by taking the square root, as explained in Section 3.2 of the paper. This means that effectively, we are computing $\kappa(\mathbf{W}) = \max(|\sigma(\mathbf{W})|)/\min(|\sigma(\mathbf{W})|)$, which is an absolutely symmetric function. Consequently, subgradients of the spectral regularization term $\kappa(\mathbf{W})$ are well-defined and can be optimized efficiently when used in the objective function.

D Detailed Experimental Setup

We consider the few-shot image classification problem on three benchmark datasets, namely:

1. **Omniglot** [7] is a dataset of 20 instances of 1623 characters from 50 different alphabets. Each image was hand-drawn by different people. The images are resized to 28×28 pixels and the classes are augmented with rotations by multiples of 90 degrees.

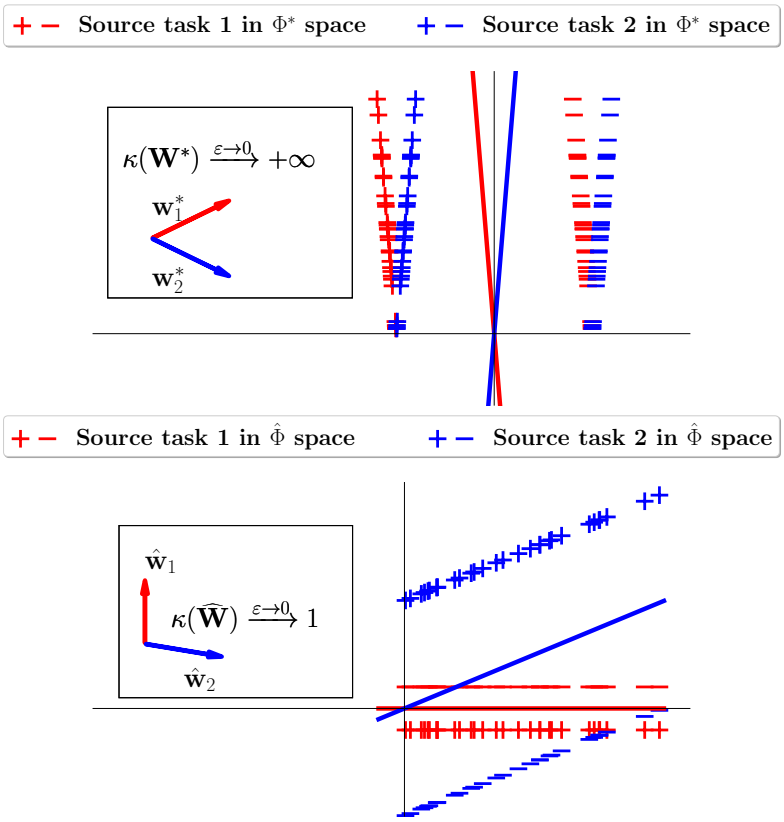

Fig. 2: Visualization of the distributions used in the constructive example for the proof of Proposition 3, with $\epsilon = 0.02$. In this example, $\kappa(\widehat{\mathbf{W}})$ is closer to 1 than $\kappa(\mathbf{W}^*)$. It shows that we can search for a representation $\widehat{\phi}$ such that optimal predictors in this space are fulfilling the assumptions, while solving the underlying problem equally well.

2. **miniImageNet** [13] is a dataset made from randomly chosen classes and images taken from the ILSVRC-12 dataset [15]. The dataset consists of 100 classes and 600 images for each class. The images are resized to 84×84 pixels and normalized.
3. **tieredImageNet** [14] is also a subset of ILSVRC-12 dataset. However, unlike miniImageNet, training classes are semantically unrelated to testing classes. The dataset consists of $779,165$ images divided into 608 classes. Here again, the images are resized to 84×84 pixels and normalized.

For each dataset, we follow a common experimental protocol used in [2, 4] and use a four-layer convolution backbone (Conv-4) with 64 filters as done by [2] optimized with Adam [6] and a learning rate of 0.001. On *miniImageNet* and *tieredImageNet*, models are trained on 60000 5-way 1-shot or 5-shot episodes

and on 30000 20-way 1-shot or 5-shot episodes for *Omniglot*. We use a batch size of 4 and evaluate on the validation set every 1000 episodes. We keep the best performing model on the validation set to evaluate on the test set. 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. The seeds used for all experiments are 1, 10, 100 and 1000. For MAML and MC, we use an inner learning rate of 0.01 for *miniImageNet* and *tieredImageNet*, and 0.1 for *Omniglot*. During training, we perform 5 inner gradient step and 10 step during testing. For all FSC experiments, unless explicitly stated, we use the regularization parameters $\lambda_1 = \lambda_2 = 1$.

We also provide experiments with the ResNet-12 architecture [8]. In this case, we follow the recent practice and initialize the models with the weights pretrained on the entire meta-training set [12, 16, 21]. Like in their protocol, this initialization is updated by meta-training with PROTONET or MAML on at most 20000 episodes, grouping every 100 episodes into an *epoch*. Then, the best performing model on the validation set, evaluated every epoch, is kept and the performance on 10000 test tasks is measured. For all experiments with the ResNet-12 architecture, the SGD optimizer with a weight decay of 0.0005 and momentum of 0.9 and a batch of episodes of size 1 are used. For PROTONET, following the protocol of Ye et al. [21], an initial learning rate of 0.0002, decayed by a factor 0.5 every 40 epochs, is used. For MAML, following Ye et Chao [20], the initial learning rate is set to 0.001, decayed by a factor 0.1 every 20 epochs. The number of inner loop updates are respectively set to 15 and 20 with a step size of 0.05 and 0.1 for 1-shot and 5-shot episodes on the miniImageNet dataset, and respectively 20 and 15 with a step size of 0.001 and 0.05 on the tieredImageNet dataset.

E Detailed performance comparisons

The plots showing the behavior of PROTONET and MAML on Omniglot are shown in Figure 4. The detailed training curves of the regularized and normalized versions of PROTONET, IMP, MAML and MC can be found in Figure 3. The performance gap (difference of accuracy in p.p.) throughout training for all methods is shown in Figure 5. 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. From them, we note that the gap in performance due to our regularization is globally positive throughout the whole training, which shows the increased generalization capabilities from enforcing the assumptions. There is also generally a high gap at the beginning of training suggesting faster learning. The best performance with the proposed regularization is achieved after training on a significantly reduced amount of training data. These results are also summarized in Table 1 of our paper and discussions about them can be found in Section 4.2 and 4.3.

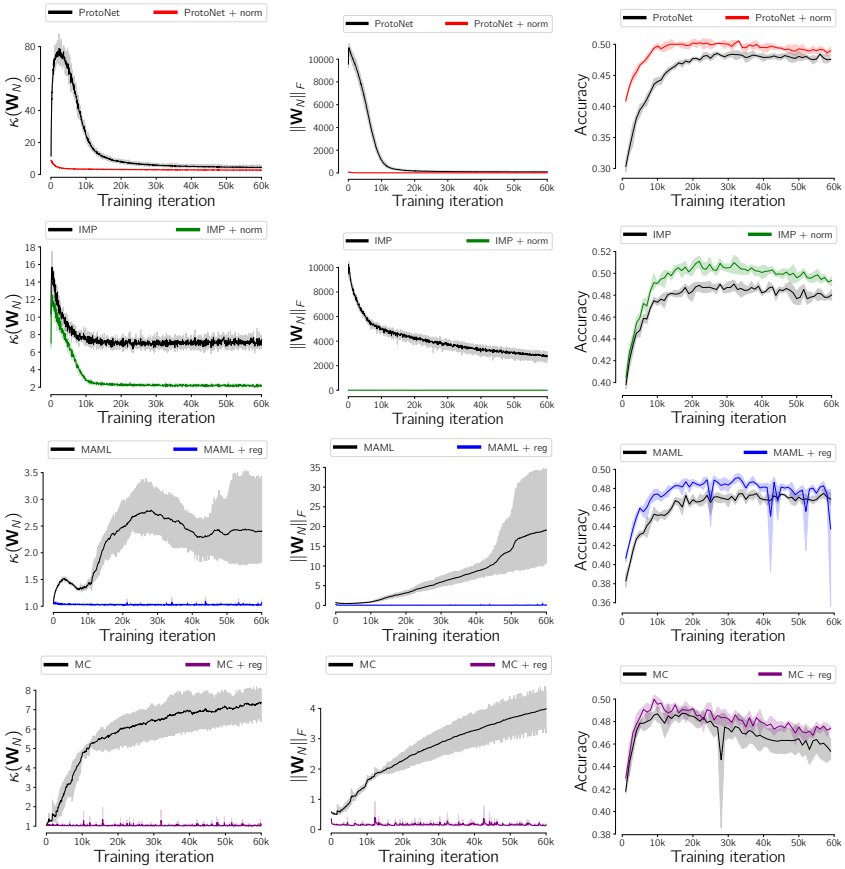

Fig. 3: Evolution of $\kappa(\mathbf{W}_N)$ (*left*), $\|\mathbf{W}_N\|_F$ (*middle*) and the accuracy (*right*) on 5-way 1-shot episodes from *miniImageNet*, for ProtoNet, IMP, Maml, MC (*from top to bottom respectively.*) and their regularized or normalized counterparts. All results are averaged over 4 different random seeds. The shaded areas show 95% confidence intervals.

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

To guide the model into learning an encoder with the lowest condition number, we consider adding $\kappa(\mathbf{W}_N)$ as a regularization term when training a normalized ProtoNet. In addition to the normalization of the prototypes, this should further enforce the assumption on the condition number. Unfortunately, this latter strategy hinders the convergence of the network and leads to numerical instabilities. It is most likely 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

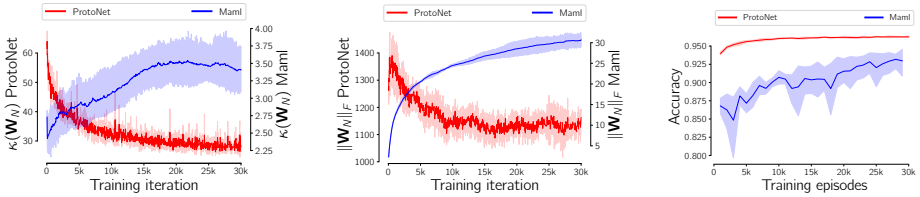

Fig. 4: Evolution of $\kappa(\mathbf{W}_N)$, $\|\mathbf{W}_N\|_F$ and $\kappa(\mathbf{W})$ (*in log scale*) during the training of ProtoNet (*red, left axes*) and Maml (*blue, right axes*) on Omniglot with 5-way 1-shot episodes.

condition number as a regularization term by the *negative entropy of the vector of singular values* 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. Since uniform distribution has the highest entropy, regularizing with $\kappa(\mathbf{W}_N)$ or $H_\sigma(\mathbf{W}_N)$ leads to a better coverage of $\mathbb{R}^k$ by ensuring a nearly identical importance regardless of the direction.
We obtain the following regularized optimization problem:

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

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

In Table 3, we report the performance of ProtoNet without normalization, with normalization and with both normalization and regularization on the entropy. Finally, 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 improve the performance.

G Ablation studies

In this Section, we present a study on the effect of each term in the proposed regularization for Maml and MTL. In Table 6, we compare the performance of Maml without regularization ($\lambda_1 = \lambda_2 = 0$), with a regularization on the condition number $\kappa(\mathbf{W}_N)$ ($\lambda_1 = 1$ 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 terms are important in the training and that using only a single term can be detrimental

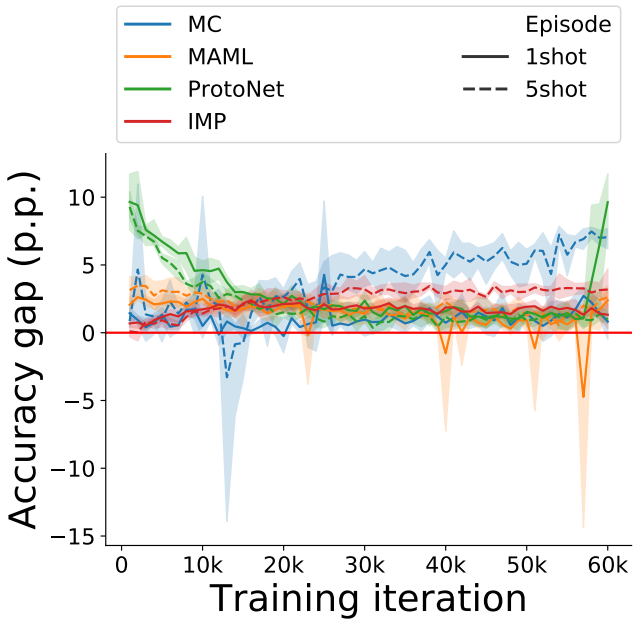

Fig. 5: Performance gap (in p.p.) when applying regularization for gradient-based and normalization for metric-based methods throughout the training process on 5-way 1-shot and 5-shot episodes on miniImageNet (*better viewed in color*). Each data point is averaged over 2400 validation episodes and 4 different seeds and shaded areas report 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.

to the performance. Table 5 presents the effect of varying independently either parameter λ_1 or λ_2 in the regularization, the other being fixed to 1. From these results, we can see that performance is much more impacted by the condition number regularization (parameter λ_1) than by the normalization (parameter λ_2). Indeed, varying the regularization weight can lead from the lowest accuracy (74.64%, for $\lambda_1 = 0$) to one of the highest accuracies (76.15% for $\lambda_1 = 0.2$).

H Out-of-Domain Analysis

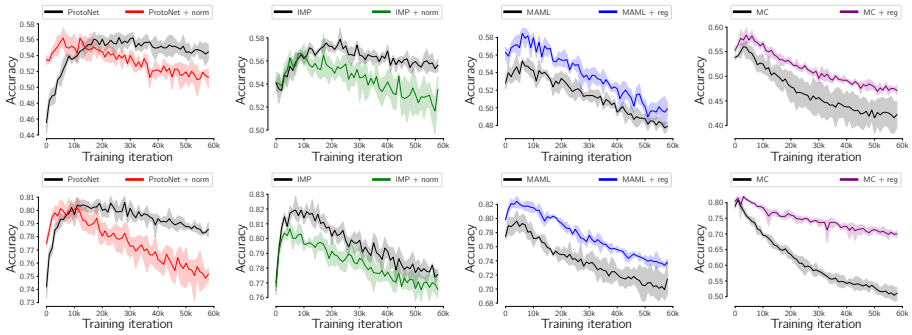

Fig. 6: Evolution of accuracy of 5-way 1-shot (*top*, resp. 5-way 5-shot, *bottom*) meta-test episodes from *CropDisease* during meta-training on 5-way 1-shot (*top*, resp. 5-way 5-shot, *bottom*) episodes from *miniImageNet*, for ProtoNet, IMP, Maml, MC (*from left to right*) and their regularized or normalized counterparts (*in red, green, blue and purple, respectively*). All results are averaged over 4 different random seeds. The shaded areas show 95% confidence intervals.

In the theoretical MTR framework, one additional critical assumption made is that the task data marginal distributions are similar (see Appendix A and assumption A2.5 for more information), which does not hold in a *cross-domain* setting, where we evaluate a model on a dataset different from the training dataset. In this setting, we do not have the same guarantees that our regularization or normalization schemes will be as effective as in *same-domain*. To verify this, we measure out-of-domain performance on the **CropDiseases** dataset [11] adopted by [5]. Following their protocol, this dataset is used only for testing purposes. In this specific experiment, evaluated models are trained on *miniImageNet*.

On the one hand, for metric-based methods, the improvement in the *same-domain* setting does not translate to the *cross-domain* setting. From Figure 6, we can see that even though the low condition number in the beginning of training leads to improved early generalization capabilities of ProtoNet, this is not the case for IMP. We attribute this discrepancy between ProtoNet and

IMP to a difference in cluster radius parameters of IMP and normalized IMP, making the encoder less adapted to *out-of-domain* features. On the other hand, we found that gradient-based models keep their accuracy gains when evaluated in cross-domain setting with improved generalization capabilities due to our regularization. This can be seen on Figure 6, where we achieve an improvement of about 2 p.p. for both MAML and MCmodels on both 1-shot and 5-shot settings.

These results confirm that minimizing the norm and condition number of the linear predictors learned improves the generalization capabilities of meta-learning models. As opposed to metric-based methods which are already implicitly doing so, the addition of the regularization terms for gradient-based methods leads to a more significant improvement of performance in *cross-domain*.

References

1. Arnold, S., Iqbal, S., Sha, F.: When MAML can adapt fast and how to assist when it cannot. In: AISTATS (2021)
2. Chen, W.Y., Wang, Y.C.F., Liu, Y.C., Kira, Z., Huang, J.B.: A closer look at few-shot classification. In: ICLR (2019)
3. Du, S.S., Hu, W., Kakade, S.M., Lee, J.D., Lei, Q.: Few-shot learning via learning the representation, provably. In: International Conference on Learning Representations (2020)
4. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: International Conference on Machine Learning (2017)
5. Guo, Y., Codella, N.C., Karlinsky, L., Codella, J.V., Smith, J.R., Saenko, K., Rosing, T., Feris, R.: A broader study of cross-domain few-shot learning. In: Computer Vision – ECCV 2020. Springer Int. Publishing (2020)
6. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
7. Lake, B.M., Salakhutdinov, R., Tenenbaum, J.B.: Human-level concept learning through probabilistic program induction. Science (2015)
8. Lee, K., Maji, S., Ravichandran, A., Soatto, S.: Meta-learning with differentiable convex optimization. In: CVPR (2019)
9. Lewis, A.S., Sendov, H.S.: Nonsmooth Analysis of Singular Values. Part I: Theory. Set-Valued Analysis (2005)
10. Maurer, A., Pontil, M., Romera-Paredes, B.: The benefit of multitask representation learning. Journal of Machine Learning Research (2016)
11. Mohanty, S.P., Hughes, D.P., Salathé, M.: Using deep learning for image-based plant disease detection. Frontiers in Plant Science p. 1419 (2016)
12. Qiao, S., Liu, C., Shen, W., Yuille, A.L.: Few-shot image recognition by predicting parameters from activations. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 7229–7238 (2018)
13. Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning. In: ICLR (2017)
14. Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J.B., Larochelle, H., Zemel, R.S.: Meta-learning for semi-supervised few-shot classification. In: ICLR (2018)
15. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. International Journal of Computer Vision (2015)

16. Rusu, A.A., Rao, D., Sygnowski, J., Vinyals, O., Pascanu, R., Osindero, S., Hadsell, R.: Meta-learning with latent embedding optimization. In: International Conference on Learning Representations (2018)
17. Tripuraneni, N., Jin, C., Jordan, M.: Provable meta-learning of linear representations. In: International Conference on Machine Learning. PMLR (2021)
18. Wang, H., Zhao, H., Li, B.: Bridging multi-task learning and meta-learning: Towards efficient training and effective adaptation. In: Proceedings of the 38th International Conference on Machine Learning. Proceedings of Machine Learning Research, PMLR (2021)
19. Wang, T., Isola, P.: Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In: ICML. pp. 9929–9939. Proceedings of machine learning research (2020)
20. Ye, H.J., Chao, W.L.: How to train your MAML to excel in few-shot classification. In: International Conference on Learning Representations (2022)
21. Ye, H.J., Hu, H., Zhan, D.C., Sha, F.: Few-shot learning via embedding adaptation with set-to-set functions. In: Computer vision and pattern recognition (CVPR) (2020)

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	Original	$\lambda_1 = 0$	$\lambda_1 = 1$	$\lambda_1 = 0.1$	$\lambda_1 = 0.01$	$\lambda_1 = 0.001$	$\lambda_1 = 0.0001$
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: Performance of MTL [18] when varying either λ_1 or λ_2, the other being fixed to 1, on the miniImageNet 5-way 5-shot benchmark. All accuracy results (in %) are averaged over 2000 test episodes on a single random seed.

λ_1	1	0.8	0.6	0.4	0.2	0.1	0.05	0.01	0
Accuracy ($\lambda_2 = 1$)	75.84	75.85	76.02	76.11	76.15	75.99	75.65	75.08	74.64
λ_2	1	0.8	0.6	0.4	0.2	0.1	0.05	0.01	0
Accuracy ($\lambda_1 = 1$)	75.84	76.09	75.81	76.28	76.23	76.1	76.25	**76.42**	76.06

Table 6: Ablative study of the regularization parameter for MAML, on Omniglot (*left*) with 20-way 1-shot (*top values*) and 20-way 5-shot (*bottom values*) episodes, and miniImageNet (*right*) with 5-way 1-shot (*top values*) and 5-way 5-shot (*bottom values*) episodes. All accuracy results (in %) are averaged over 2400 test episodes and 4 different random seeds and are reported with 95% confidence interval. We can see that in all cases, using both regularization terms is important.

(a) Omniglot (20-way 1-shot / 5-shot)

	$\lambda_1 = 0$	$\lambda_1 = 1$
$\lambda_2 = 0$	$91.72 \pm 0.29\%$	$89.86 \pm 0.31\%$
	$97.07 \pm 0.14\%$	$72.47 \pm 0.17\%$
$\lambda_2 = 1$	$92.80 \pm 0.26\%$	$\mathbf{95.67 \pm 0.20}\%$
	$96.99 \pm 0.14\%$	$\mathbf{98.24 \pm 0.10}\%$

(b) miniImageNet (5-way 1-shot / 5-shot)

	$\lambda_1 = 0$	$\lambda_1 = 1$
$\lambda_2 = 0$	$47.93 \pm 0.83\%$	$47.76 \pm 0.84\%$
	$64.47 \pm 0.69\%$	$64.44 \pm 0.68\%$
$\lambda_2 = 1$	$48.27 \pm 0.81\%$	$\mathbf{49.16 \pm 0.85}\%$
	$64.16 \pm 0.72\%$	$\mathbf{66.43 \pm 0.69}\%$