EARLY-STOPPING FOR META-LEARNING: ESTIMATING GENERALIZATION FROM ACTIVATION DYNAMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Early-stopping, a fundamental element of machine learning practice, aims to halt the training of a model when it reaches optimal generalization to unseen examples, right before the overfitting regime on the training data. Meta-Learning algorithms for few-shot learning aim to train neural networks capable of adapting to novel tasks using only a few labelled examples, in order to achieve good generalization. However, current early-stopping practices in meta-learning are problematic since there may be an arbitrary large distributional shift between the meta-validation set coming from the training data, and the meta-test set. This is even more critical in few-shot transfer learning where the meta-test set comes from a different target dataset. To this end, we empirically show that as meta-training progresses, a model's generalization behaviour on a target distribution of novel tasks can be estimated by analysing the dynamics of its neural activations. We propose a method for estimating optimal early-stopping time from the neural activation dynamics of just a few unlabelled support examples from the target distribution, and we demonstrate its performance with various meta-learning algorithms, few-shot datasets and transfer regimes.

1 INTRODUCTION

Deep Learning research has been successful at producing algorithms and models that, when optimized on a distribution of training examples, generalize well to previously unseen examples drawn from that same distribution. Meta-Learning is in a way, a natural extension of this aim, where the model has to generalize to not only new data points, but entirely new tasks. Important practical progress has been made in this direction over the past few years. Yet it remains sparsely understood what are the underlying phenomena behind the transitioning of a neural network's generalization to novel tasks, from the underfitting to the overfitting regime, with the optimal generalization happening in between. Early-stopping, a fundamental element of machine learning practice, maximizes generalization by aiming to halt the training at the frontier between those two regimes, when generalization is optimal. It is computed on a validation set, made of held out examples from the training data, which serves as a proxy for the test data. As a regularizer, "*Early-stopping should almost be used universally. [...] It is probably the most commonly used form of regularization in deep learning. [...] a very unobtrusive form of regularization, in that it requires almost no change in the underlying training procedure*" (Goodfellow et al., 2016). However in meta-learning, implementing early-stopping is problematic since there may be an arbitrarily large distributional shift between the meta-validation tasks (drawn from the training data) and the meta-test tasks. Moreover, meta-learning typically involves learning a new task from very few labelled examples, too few to allow constituting a validation set from it.

In this work, we study the relation between generalization in Meta-Learning and neural activation dynamics : Given a neural network and a set of input examples, the network's responses measured at all of its hidden-layers are what we define as the neural activations, and the evolution of those responses during the learning time (meta-training) is what we define as the neural activation dynamics. The main contributions of our work can be summarized as follows :

1. We empirically show that in Meta-Learning, a simple function of the neural activation dynamics, for just a few unlabelled target examples, can reveal the variation of generalization to a distribution of novel target tasks (Sec.2.2), and how this function can be learned (Sec.3).

2. We propose a novel method for early-stopping in Meta-Learning, applied in many settings of Few-Shot Learning and Few-Shot Transfer Learning (Sec.5).

2 META-LEARNING AND FEW-SHOT CLASSIFICATION

Meta-Learning algorithms generally aim to train a model $f(\mathbf{x};\theta)$ on a set of source problems, often presented as a distribution over tasks $p(\mathcal{T}_{train})$, in such a way that the model is capable of generalizing to new, previously unseen tasks from a target distribution $p(\mathcal{T}_{target})$. When applied to classification, meta-learning has often been formulated in the past by defining a task $\mathcal{T}$ that involves the m-way classification of input examples $\mathbf{x}$ among m distinct classes. The tasks from $p(\mathcal{T}_{train})$ and $p(\mathcal{T}_{target})$ are made of classes drawn from two disjoint sets $\mathcal{C}_{train}$ and $\mathcal{C}_{target}$. A novel task thus involves new classes not seen during training.

In *few-shot learning*, the inputs $\mathbf{x}$ of the training and target tasks come from a same input distribution $p(\mathbf{x})$ (e.g., an image dataset) but conditioned on their respective classes, i.e. $p(\mathbf{x}_{train}) = p(\mathbf{x}|\mathbf{y} \in \mathcal{C}_{train})$ and $p(\mathbf{x}_{target}) = p(\mathbf{x}|\mathbf{y} \in \mathcal{C}_{target})$. The *few-shot* aspect means that for a given novel task $\mathcal{T}_{target}$, only a very few labelled examples are available, typically k examples per class, and the model uses this *support set* of examples $\mathcal{S} = \{(\mathbf{x},\mathbf{y})\}_{1..k}$ to adapt its parameters θ to the task, then its accuracy is evaluated on new *query* examples from $\mathcal{T}_{target}$. The meta-learning *generalization* Acc_{target}, for a model $f(\mathbf{x};\theta_t)$ at *time* t (after t training iterations) to a distribution $p(\mathcal{T}_{target})$, is thus the query accuracy averaged over multiple target tasks:

$$Acc_{target} \doteq \mathbb{E}_{\mathcal{T}_i \sim p(\mathcal{T}_{target})} \left[\mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim \mathcal{T}_i \setminus \mathcal{S}_i} \left[\mathbb{1}\{\text{argmax}(f(\mathbf{x};\theta_t^i)) = \mathbf{y}\} \right] \right] \quad (1)$$

where for each new task $\mathcal{T}_i$ the adapted solution θ_t^i is often obtained by performing T steps of gradient descent (full-batch) on the cross-entropy loss $\mathcal{L}(f, \mathcal{S}_i)$ with respect to θ_t.

In *few-shot transfer learning*, not only are the class sets $\mathcal{C}_{train}$ and $\mathcal{C}_{target}$ disjoint, but the marginal $p(\mathbf{x}_{target})$ can be arbitrarily different from $p(\mathbf{x}_{train})$ (e.g. from a different image dataset).

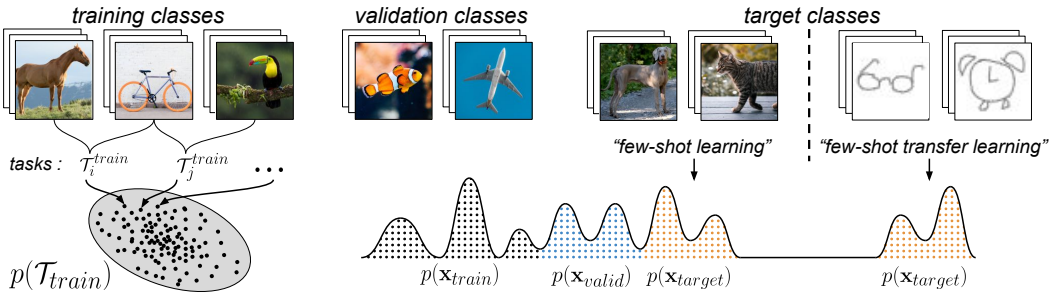


Figure 1: A *task* is created by randomly picking m classes from a set $\mathcal{C}$, and belongs to a *task distribution* $p(\mathcal{T})$ (e.g. $\mathcal{T}_i^{train} \sim p(\mathcal{T}_{train})$: classify between "horse" and "bicycle"). Training, validation and target tasks are made of different classes. *Few-Shot Learning :* target inputs (e.g. images) come from the same distribution (dataset) as for training, but conditioned on different classes. *Few-Shot Transfer Learning :* Different target classes, and target inputs come from a different dataset.

2.1 EARLY-STOPPING BASED ON VALIDATION SET PERFORMANCE CAN LEAD TO SUB OPTIMAL GENERALIZATION IN META-LEARNING

In a standard supervised learning setup, a subset of examples is held out from the training data to constitute a validation set. Since the validation accuracy is a good proxy for the test accuracy, early-stopping is performed by halting training when the validation accuracy reaches its maximum. In Meta-Learning for few-shot classification, the validation set is made of held out classes from the training data to constitute the validation task distribution $p(\mathcal{T}_{valid})$, and early-stopping happens at t^*_{valid}=$\text{argmax}_t Acc_{valid}$. But this can lead to a sub-optimal generalization (see Fig.2) because of the potential distributional shift between $p(\mathcal{T}_{target})$ and $p(\mathcal{T}_{valid})$ especially in few-shot transfer learning where it can be arbitrarily large. Estimating the out-of-distribution generalization Acc_{target} in Meta-Learning thus requires some minimal amount of information about $p(\mathcal{T}_{target})$. However, the

few-shot paradigm severely restricts the availability of data from $p(\mathcal{T}_{target})$. The support examples from target tasks are accessible, but the model doesn't control how many new tasks will actually be presented, there could be several thousands or very few. However, if there is a need to early-stop and generalize to some target task distribution $p(\mathcal{T}_{target})$, then the model will need to solve, at the very least, a single task from $p(\mathcal{T}_{target})$, and thus has access to at least a single support set $\mathcal{S}$. We thus propose to only use a few examples, typically the support set of a single new task (e.g. 5 images). This also implies that any algorithm estimating the optimal early-stopping time t^* should have a very low sample-wise (and task-wise) variance for its estimate of t^*.

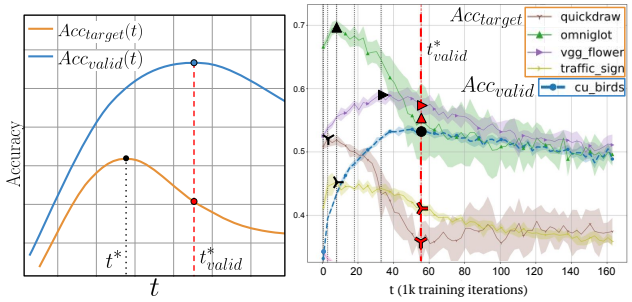
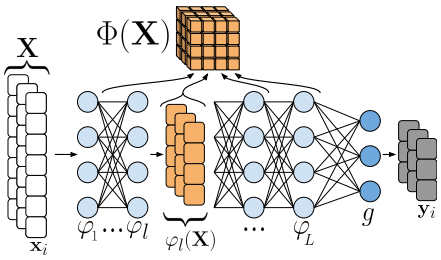


Figure 2: *(Left)* Early-stopping based on validation accuracy is problematic in Meta-Learning as there can be an arbitrarily large time gap between the optimal stopping time t^* for the target task distribution and t^*_{valid}. This can lead to sub optimal target generalization. For example, on the (*Right*) we show accuracy vs. training interations for a CNN trained with MAML on the Birds dataset with multiple target datasets, each accuracy is averaged over 500 tasks (5-way 1-shot). Markers represent t^* (black); t^*_{valid} (red). More settings in App.***.

2.2 Can neural activation dynamics for a few target inputs allow us to make inferences about generalization?

Figure 3: **Neural Activation Dynamics** : A neural network f composed of a feature extractor φ (light blue) of L hidden-layers, followed by a classifier g. For a set $\mathbf{X}$ of input vectors $\mathbf{x}_i$ the activation vectors at the l-th layer are denoted as $\varphi_l(\mathbf{X})$. The set of activations of all layers from 1 to L constitute the *neural activations* $\Phi(\mathbf{X})$ of the model and their evolution through learning time constitute the *neural activation dynamics* $\Phi(\mathbf{X}, t)$. See Eq. 3, 4.



.

In this work we search for an observable property of deep neural networks that can help us make inferences about meta-learning generalization to a given target problem as training time t progresses. We thus hypothesize the existence of a function ψ of $f(\mathbf{x}; \theta_t)$ and $p(\mathcal{T}_{target})$ such that $\psi(f, p(\mathcal{T}_{target}), t) \propto Acc_{target}(t)$, and set on to find ψ. More specifically, we want to estimate $t^* = \text{argmax}_t\, Acc_{target}(t)$ using only a few target examples (a single support set) when approximating ψ. To support a general statement on generalization in Meta-Learning and the nature of ψ, we conduced experiments across a wide range of meta-learning settings. We used different meta-learning algorithms, three of the most pivotal ones of the field : MAML (Finn et al., 2017), Prototypical Networks (Snell et al., 2017), and Matching Networks (Vinyals et al., 2016). We considered both the few-shot learning and few-shot transfer learning regimes, with 1-shot and 5-shot experiments, and various few-shot datasets for $p(\mathcal{T}_{train})$ and $p(\mathcal{T}_{target})$, such as MiniImagenet and Omniglot, but also many others included in Meta-Dataset (Triantafillou et al., 2020). We also used different architectures : the standard 4-layer CNN proposed by (Vinyals et al., 2016), as well as a ResNet as used in (Triantafillou et al., 2020). For full experimental details, refer to Appendix A. Here we present the experimental results that progressively suggest that variation of generalization can be efficiently estimated from simple metrics on the neural activation dynamics :

Observation 1: *For a deep neural network, the variation of target generalization, as a function of training time, frequently correlates with simple statistics characterizing how its feature extractor responds to the target input distribution:* In many meta-learning settings we observed that Acc_{target} is proportional to relatively simple metrics (denoted as ψ_1). One such metric is the *expected inner product between representations*:

$$\psi_1(\varphi(\mathbf{X})) \doteq \mathbb{E}_{\mathbf{x}_i, \mathbf{x}_j \sim p(\mathbf{x})}[\varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j)], \tag{2}$$

where ψ_1 is measured at the output of the feature extractor φ, where $f(\mathbf{x}) = g(\varphi(\mathbf{x}))$, and captures both the similarity among representation vectors and their norm. Moreover, we measure ψ_1 on the representations of the target inputs $\mathbf{X}_{target}$, before adapting the model to new tasks. This relation seems approximately independent of the target class identities $\mathbf{Y}_{target}$, and predominantly depends on how the feature extractor represents the marginal distribution over the input $\mathbf{X}_{target}$ of the target problem, i.e. $\psi_1(\varphi(\mathbf{X}_{target}), t) \propto Acc_{target}(t)$. Example in Fig.4, complete results in App.B.2.1.

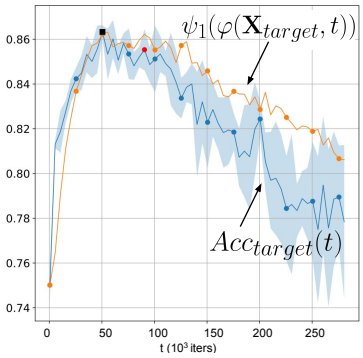


Figure 4: Average target task accuracy as a function of training iteration: $Acc_{target}(t)$. **Observation 1:** The variation of generalization (Acc_{target}), for a deep neural network, frequently correlates with simple statistics characterizing how its feature extractor φ responds to the target input distribution $p(\mathbf{x}_{target})$. For example, here we show ψ_1, which is simply the expected inner product between individual representation vectors, which follows the same trend as $Acc_{target}(t)$ and peaks roughly at the same time. Here ψ_1 is vertically rescaled to match the range of the target accuracy. See App.B.2.1 for full experiments across multiple settings.

Observation 2: *A simple statistic on the neural activations, if computed at the right layer of a network, can often strongly correlate with generalization, but this layer may change depending on the setting.* In a deep neural network a feature extractor is composed of L hidden-layers: $\varphi(\mathbf{x}) = (\varphi_L \circ \varphi_{L-1} \circ ... \circ \varphi_1)(\mathbf{x})$. In many settings, ψ_1 measured at the last layer φ_L isn't proportional to Acc_{target}, but the relation instead occurs at a lower layer φ_l. Thus, rather than just examining the last layer representation dynamics, we often need to consider the *neural activations* of the whole feature extractor (See Fig. 3 and Eq. 3). More precisely, we shall consider the evolution throughout time or the *neural activation dynamics* of a network, i.e.: $\psi(\Phi(\mathbf{X}_{target}, t)) \propto Acc_{target}(t)$, or expressed in the form of Eq. 4. See an example at Fig.5, or App.B.2.2 for full experiments across multiple settings.

$$\Phi(\mathbf{X}) \doteq \{\varphi_l(\mathbf{X}) \mid l \in [1..L]\} \tag{3}$$

$$\Phi(\mathbf{X}, t) \doteq \Phi(\mathbf{X} \mid \theta_t) \tag{4}$$

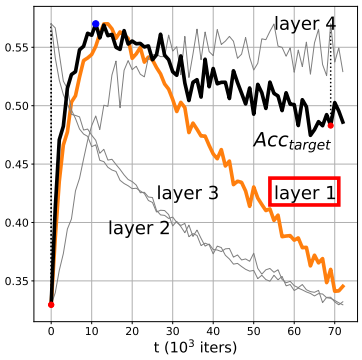


Figure 5: **Observation 2:** Simple statistics of neural activation dynamics which best correlate to generalization may be observed at different layers of the feature extractor. We therefore consider the *neural activation dynamics* of all layers in a network in our work here.

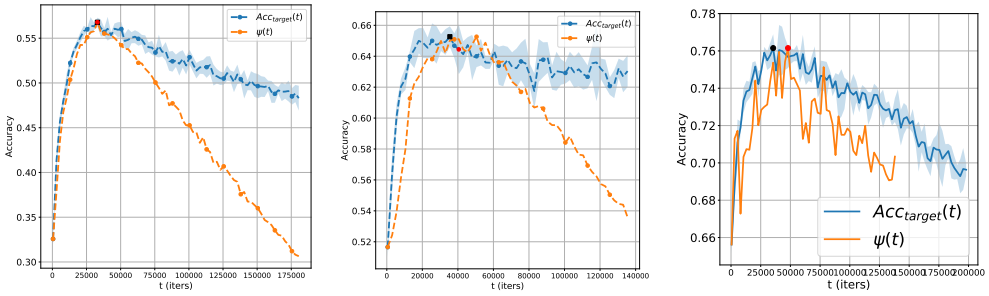
Observation 3: *Simple statistics of the activations correlate to generalization, but they may change. One needs to find the right statistic depending on the setting.* In our experiments we observe that for many settings $Acc_{target}(t)$ doesn't consistently correlate with $\psi_1(\varphi_l(\mathbf{X}_{target}))$ at any specific layer l. From this we conjectured that perhaps ψ_1 is a special case in a more general *function space* Ψ, a hypothesis space or set of functions that predict generalization, or more formally : $\Psi \doteq \{\psi \mid \psi(\Phi(\mathbf{X}_{target}, t)) \propto Acc_{target}(t)\}$ where $|\Psi| > 1$. From this perspective we ultimately care about finding the function ψ in Ψ that, given the meta-learning setting involved, minimizes the *true objective* d defined below :

$$d = \max_t(Acc_{target}(t)) - Acc_{target}(\operatorname{argmax}_t \psi(\Phi(\mathbf{X}_{target}, t))) \tag{5}$$

The natural question that follows is, what may be the characteristics of such function space Ψ ? We address this by formulating a few inductive biases and assumptions which then inform our subsequent experiments. We first note that the complexity of Ψ must be large enough so that, in most meta-learning settings in the few-shot regime, Ψ contains a good solution function ψ^* such that d is low. The complexity shouldn't be too large either, since to find ψ^* we will optimize an indirect empirical objective $\hat{d}$ (Sec.3). This is especially important in few-shot transfer learning. Furthermore, since $\Phi(\mathbf{X}_{target})$ itself has a probability distribution, our hypothesis space Ψ should be a set of functions ψ that are sample estimators of some population statistics of the distribution of $\Phi(\mathbf{X}_{target})$. However, since we only have access to a very few samples, those statistics should be relatively simple so as to keep down the standard error of their estimators. We propose to use descriptive statistics based on moments, and limit them up to the second-order (higher-order moments are harder to estimate accurately). Finally, since we ultimately need to find a one-dimensional curve $\psi^*(t)$ to

compare to $Acc_{target}(t)$, our hypothesis space Ψ should contain scalar-valued functions, which we get by computing moments on norms of the activation vectors. In our experiments we observe that when $Acc_{target}(t)$ doesn't consistently correlate with $\psi_1(\varphi_l(\mathbf{X}_{target}))$ at any specific layer l, it does typically correlate with one of the following alternative metrics : the *norm of activations* ψ_2; the *dispersion of activations* ψ_3; or the *feature-wise variance of activations* ψ_4. We have observed that generalization sometimes actually correlate with the negative (i.e. $-\psi$) of either of ψ_1 to ψ_4.

$$\psi_2 \doteq \mathbb{E}_{\mathbf{x}}[\|\varphi_l(\mathbf{x})\|_2^2] \quad (6) \quad \psi_3 \doteq \mathbb{E}_{\mathbf{x}}[\|\varphi_l(\mathbf{x}_i) - \varphi_l(\mathbf{x}_j)\|_2^2] \quad (7) \quad \psi_4 \doteq \mathbb{E}_{\mathbf{x}}[\mathrm{Var}_k(\varphi_l(\mathbf{x}_{i,k}))] \quad (8)$$



(a) Generalization here correlates with ψ_2 - the *norm of activations*. MAML, CNN, Aircraft

(b) Generalization here correlates with ψ_3 - *dispersion of activations*. ProtoNet, CNN, VGG Flower

(c) Generalization here correlates with ψ_4 - *feature-wise variance*. Matching Net,CNN,MiniImagenet

Figure 6: **Observation 3:** Generalization may correlate to different properties of the neural activation dynamics, depending on the setting. While Acc_{target} may correlate to ψ_1 (Eq.2) in some settings, in other settings it may instead correlate to ψ_2, ψ_3 or ψ_4 (Eq.7,6,8) measured on the activation dynamics at depth l of the feature extractor. See App.B.2.3 for full experiments across multiple settings.

All the metrics ψ_1 to ψ_4 and their negatives can actually be expressed by a linear combination of the following moments (assuming ReLU activation functions) :

$$m_1 = \frac{1}{n}\sum_{i=1}^{n}\|\varphi_l(\mathbf{x}_i)\|_1^2 \quad m_2 = \frac{1}{n}\sum_{i=1}^{n}\|\varphi_l(\mathbf{x}_i)\|_2^2 \quad m_3 = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\|\varphi_l(\mathbf{x}_i) - \varphi_l(\mathbf{x}_j)\|_2^2 \quad (9)$$

such that those moments define the function space $\Psi = \{\psi(\varphi_l(\mathbf{X}); \mathbf{w}) \mid \mathbf{w} \in \mathbb{R}^3, l \in [1..L]\}$ where $\psi(\varphi_l(\mathbf{X}); \mathbf{w}) = w_1 m_1 + w_2 m_2 + w_3 m_3$ and $\mathbf{w} = [w_1, w_2, w_3] \in \mathbb{R}^3$. This parametric function space Ψ, while being relatively simple, can express a variety of properties of activations, such as their norm, dispersion, feature-wise variance, inner product, positively or negatively, or even a combination of properties. In Tab.3 of App. B.3 we have experimentally verified that Ψ has enough complexity to contain a good solution function ψ^*, for many meta-learning settings, both in few-shot learning and few-shot transfer learning.**Observation 4:** *The Variation of generalization can be estimated by using just a few target input examples:* Given a function $\psi(\varphi_l(\mathbf{X}_{target}))$ which correlates with generalization $Acc_{target}(t)$. when ψ is measured on the activation dynamics for just a single unlabelled support set $\mathcal{S}$, the estimated early-stopping time $t^*_\psi = \mathrm{argmax}_t\, \psi(t)$ typically shows very low variance with respect to which task is used for the estimation (Fig. 7). We conjecture that this might be due lack of dependency of ψ on $\mathbf{Y}_{target}$, where ψ a more general property of the activations for $p(\mathbf{x}_{target})$. This makes early-stopping from such function ψ practical.

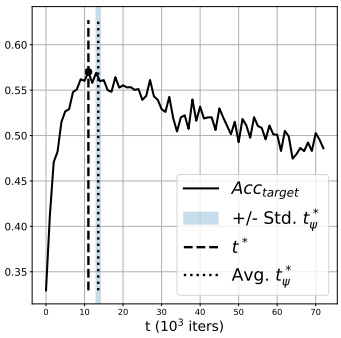


Figure 7: **Observation 4:** The variation of generalization can be estimated from a simple property ψ the neural activation dynamics $\Phi(\mathbf{X}_{target})$, from just a few input examples $\mathbf{x}_{target}$. Here use a fixed set of 50 target tasks, approximate ψ_2 with their respective support set. The approximations will peak roughly at the same time t^*_ψ. Their average peak time is denoted as $avg.t^*_\psi$, and the variance of those times is shown in blue (Std.t^*_ψ) The estimated stopping-time t^*_ψ typically shows very little variance. Setting : MAML, CNN, 5-way 1-shot, Aircraft. We make the same observations across other settings, as shown in App B.2.4.

3 INFERRING WHICH FUNCTION OF THE NEURAL ACTIVATION DYNAMICS CORRELATES TO GENERALIZATION, AND AT WHICH LAYER TO MEASURE IT

Our results in Sec.2.2 suggest that in Meta-Learning there exists a function ψ that, when measured on the neural activation dynamics $\Phi(\mathbf{X}_{target}, t)$, closely relates to the target generalization $Acc_{target}(t)$. However since this function is not unique and depends on the meta-learning setting involved (meta-learning algorithm, neural architecture, training and target distributions, etc), we propose to cast the discovery of ψ as a machine learning problem. See Fig. 8, which schematizes our framework.

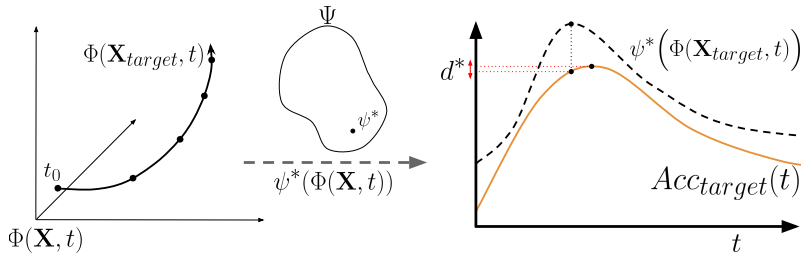


Figure 8: Our framework : Learning the relation function ψ between the neural activation dynamics $\Phi(\mathbf{X}_{target}, t)$ and the generalization to a novel task distribution $p(\mathcal{T}_{target})$). Having defined an hypothesis function space Ψ, we search, given a meta-learning setting, the optimal function ψ^* in Ψ which minimize the *true objective* d or equivalently, achieve high generalization Acc_{target}. Since we don't have access to Acc_{target}, we optimize an *empirical objective* $\hat{d}$ to find ψ^*.

At this point we know that, given a meta-learning setting, our function space Ψ should contain a good solution ψ^* such the true objective d is low. Now we need a way to actually find ψ^*. We can do so by optimizing an indirect, *empirical objective* $\hat{d}$, defined below.

3.1 Few-Shot Learning (FSL) : Inferring ψ^* and l^* from the validation dynamics and accuracy

In *few-shot learning*, novel tasks from $p(\mathcal{T}_{target})$ involve previously unseen classes but the input domain of $\mathbf{X}_{target}$ can be assumed to be similar to that of $\mathbf{X}_{train}$, and therefore to that of $\mathbf{X}_{valid}$. We thus use the dynamics $\Phi(\mathbf{X}_{valid}, t)$ and the validation accuracy Acc_{valid} (as a proxy for Acc_{target}) in order to learn the optimal function ψ^* and the layer l^* where it should be measured, and we do so by minimizing the empirical objective $\hat{d}_{FSL}$ (Eq.10). We then compute our actual early-stopping time estimate $\hat{t}^*_{FSL}$ when $\psi^*(\varphi_{l^*}(\mathbf{X}_{target}, t))$, measured on the few support input examples of a single target tasks, reaches its peak (Eq.11).

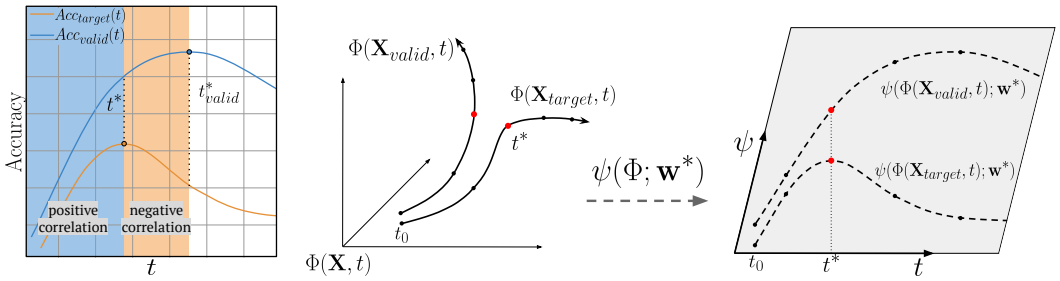
$$\hat{d}_{FSL} = \max_t Acc_{valid}(t) - Acc_{valid}(\operatorname*{argmax}_t \psi(\varphi_l(\mathbf{X}_{valid}, t); \mathbf{w}))) \tag{10}$$

$$\hat{t}^*_{FSL} = \operatorname*{argmax}_t \psi(\varphi_{l^*}(\mathbf{X}_{target}, t); \mathbf{w}^*) \quad \text{where} \quad \mathbf{w}^*, l^* = \operatorname*{argmin}_{\mathbf{w}, l} \hat{d}_{FSL} \tag{11}$$

3.2 Few-shot transfer learning (FSTL) : Meta-overfitting often happens when the target dynamics diverge from those of the source input domain

When the target problem is from an entirely new dataset, we can't use Acc_{valid} as a proxy for Acc_{target}, and we need another objective function to learn ψ^*. However, we can learn ψ^* by analyzing $\Phi(\mathbf{X}_{target}, t)$, the neural activation dynamics of the target domain, and comparing them with $\Phi(\mathbf{X}_{valid}, t)$. Assume that for a given target problem, optimal generalization doesn't happen at the same time as for the source domain, i.e., $t^* \neq t^*_{valid}$, and more precisely, assume $t^* < t^*_{valid}$. Typically, a generalization curve is generally increasing between t_0 and its maximum, whereas it is generally decreasing after the maximum. This implies that the curves of $Acc_{target}(t)$ and $Acc_{valid}(t)$ are positively correlated between t_0 and t^*, as they are both increasing, whereas they are negatively correlated between t^* and t^*_{valid}, since $Acc_{target}(t)$ is decreasing while $Acc_{valid}(t)$ is still increasing. In a sense, the two generalization behaviors "diverge" at t^*, since at that moment their correlation goes from positive to negative (See Fig.9a). Since here we assume the neural activation dynamics can characterize the generalization behavior of a model, we conjecture that $\Phi(\mathbf{X}_{target}, t)$ and $\Phi(\mathbf{X}_{valid}, t)$ might also "diverge" at t^*, under some function $\psi(\varphi_{l^*}(\mathbf{X}, t); \mathbf{w}^*)$, such that the *sample Pearson*

correlation r, of $\psi(\Phi(\mathbf{X}_{target}, t), \mathbf{w}^*)$ and $\psi(\Phi(\mathbf{X}_{valid}, t), \mathbf{w}^*)$ also goes from positive to negative near t^* (See Fig.9b).



(a) (b)

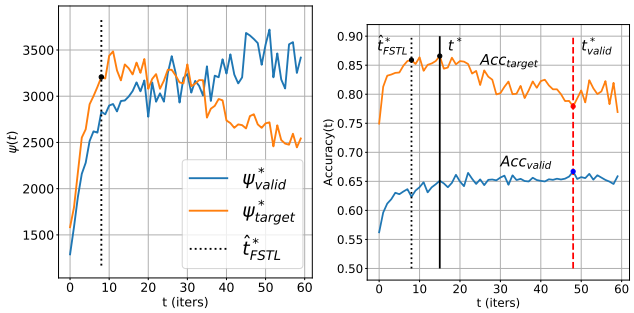
Figure 9: a) In few-shot transfer learning if overfitting on the target domain happens at a different time then on the source domain (e.g. $t^* < t^*_{valid}$) then the validation and target accuracies "diverge" at t^* : $Acc_{target}(t)$ and $Acc_{valid}(t)$ are correlated positively on $[t_0, t^*]$ and negatively on $[t^*, t^*_{valid}]$. b) Assuming that the neural activation dynamics can characterize the generalization behavior, there might be a function under which the target dynamics diverge from the validation dynamics at t^*.

Our experiments indeed suggest that functions ψ exhibiting more divergence are more likely to capture generalization to the target problem. This analysis can be found in App.***. We thus search for the weights $\mathbf{w}^*$ and hidden-layer l^* so as to observe the most negative correlation between $\psi(\varphi_l(\mathbf{X}_{target}, t); \mathbf{w})$ and $\psi(\varphi_l(\mathbf{X}_{valid}, t); \mathbf{w})$ in the time interval $[t_0, t^*_{valid}]$ (Eq.12). We then estimate t^* by finding the time $\hat{t}^*_{FSTL}$ when $\psi^*_{target}(t)$ and $\psi^*_{valid}(t)$ diverge (Eq.13). See Fig.10a,10b for a demonstration.

$$\hat{d}_{FSTL} = r(\psi_{target}(t), \psi_{valid}(t)) = \frac{\sum_t (\psi_{target}(t) - \bar{\psi}_{target}(t))(\psi_{valid}(t) - \bar{\psi}_{valid}(t))}{\sum_t (\psi_{target}(t) - \bar{\psi}_{target}(t))^2 \sum_t (\psi_{valid}(t) - \bar{\psi}_{valid}(t))^2} \tag{12}$$

$$\hat{t}^*_{FSTL} = \underset{t}{\text{argmax}} \left(t \times r\left(\psi^*_{target}, \psi^*_{valid}, [t_0, t < t^*_{valid}]\right)\right) \tag{13}$$

with shorthand notations $\psi_{target}(t) \doteq \psi(\varphi_l(\mathbf{X}_{target}, t); \mathbf{w})$ and $\psi_{valid}(t) \doteq \psi(\varphi_l(\mathbf{X}_{valid}, t); \mathbf{w})$ and $\bar{\psi}(t)$ denotes an average over t, and $\psi^* \doteq \psi(\varphi_{l^*}(\cdot); \mathbf{w}^*)$. Here again we minimize an empirical objective, and $\mathbf{w}^*, l^* = \text{argmin}_{\mathbf{w},l} .\hat{d}_{FSTL}$.



(a) (b)

Figure 10: Inferring when to stop in few-shot transfer learning. Setting shown: MAML, CNN, Quickdraw to Omniglot, 5-way 1-shot. a) The optimal function ψ^* and layer l^* are those where the neural activation dynamics of the target inputs diverge the most from those of the source domain, i.e. where the objective of Eq.12 is minimized. b) Once we have identified ψ^* and l^*, we stop at $\hat{t}^*_{FSTL}$ of Eq.13, i.e. when the Pearson correlation of ψ^*_{target} and ψ^*_{valid} flips from positive to negative. Here $\hat{t}^*_{FSTL}$ drastically outperforms t^*_{valid}.

4 RELATED WORK

In recent years, some works have started to analyze theoretical aspects of gradient-based meta-learning. (Finn et al., 2019) examine the online Meta-Learning setting, where in online learning the agent faces a sequence of tasks, and they provide a theoretical upper bound for the regret of MAML. (Denevi et al., 2019) study meta-learning through the perspective of biased regularization, where the model adapts to new tasks by starting from a biased parameter vector, which we refer in this work as the meta-training solution. For simple tasks such as linear regression and binary classification, they prove the advantage of starting from the meta-training solution, when learning new tasks via SGD. They use an assumption on the task similarity where the weight vectors parameterizing the tasks are assumed to be close to each other. Working in the framework for Online Convex Optimization where the model learns from a stream of tasks, (Khodak et al., 2019) make an assumption that the optimal solution for each task lies in a small subset of the parameter space and use this assumption to design an algorithm such that the "Task-averaged-regret (TAR)" scales with the diameter of this small subset of the parameter space, when using Reptile (Nichol et al., 2018), a first-order meta-learning algorithm. Bearing a stronger relation to our approach, (Guiroy et al., 2019) empirically study the objective landscapes of gradient-based meta-learning, with a focus on few-shot classification. They notably observed that average generalization to new tasks appears correlated with the average inner product between their gradient vectors. In other words, as gradients appear more similar in inner product, the model will, on average, better generalize to new tasks, after following a step of gradient descent. More recently, a few works have studied the properties of the feature extractor φ in the context of Meta-Learning. Notably, the authors of (Raghu et al., 2019) showed empirically that when neural networks adapting to novel task, in the few-shot setting with MAML and MiniImagenet, the feature extractor network is approximately invariant, while the final linear classifier undergoes significant functional changes. They then performed experiments where φ is frozen at meta-test time, while only the classifier g is fine-tuned, and observed very similar generalization performance to the regular fine-tuning procedure. Intuitively, these results suggest that the variation, of generalization along meta-training time t, might be predominantly driven by some evolving but unknown property of the feature extractor. The authors of (Goldblum et al., 2020) observed that generalization in few-shot learning was related to how tightly embeddings from new tasks were clustered around their respective classes. However, the authors of (Dhillon et al., 2019) observed that the embeddings at the output of φ_L were poorly clustered around their classes, but that clustering was important when measuring the logit outputs of g. This is similar to what the authors (Frosst et al., 2019) observed when dealing with new Out-of-Distribution examples. This suggests that if generalization is related to a property of the feature extractor, this property might be class agnostic. This is also something that we observed in our very early experiments (expected inner product between representation vectors strongly correlated with generalization, irrespective of taking the class identities into account). But in our work we observed that this property might not only depend on the output of the feature extractor. Earlier works demonstrated that in transfer learning, intermediate layers of φ might be critical in the ability of the model to transfer knowledge (Yosinski et al., 2014).

5 EARLY-STOPPING FOR META-LEARNING BY ANALYZING THE NEURAL ACTIVATION DYNAMICS OF A FEW TARGET INPUT EXAMPLES

Here we present experimental results on the performance of our early-stopping method. For each experiment, we only use the unlabelled input examples from the support set of a single target task to evaluate the neural activation dynamics. At the beginning of an experiment, we thus randomly sample a task $\mathcal{T}_i$ from $p(\mathcal{T}_{target})$ and only keep its set of support input examples. We repeat the experiment for multiple (50) independently and identically distributed support sets from $p(\mathcal{T}_{target})$, and take the average performance. Each such experiment is then repeated for 5 independent training runs. As a baseline for comparison, we use the validation early-stopping approach. Since ψ_1 to ψ_4 work in practice, we will use them as our function space Ψ but the method that we develop applies as well to the continuous function space defined above, and we present some experimental results in App.B.5 where we apply our early-stopping method with the continuous function space.

We begin by demonstrating our proposed early-stopping method in few-shot transfer learning, across various target dataset, and present the results in Tab.1. We use the standard 4-layer CNN architecture, with MAML, trained on MiniImagenet 5-way 1-shot. When the target dataset is Omniglot, the

performance of the validation baseline (51%) is significantly lower than the optimal generalization (76%) presumably because of the distributional shift between MiniImagenet and Omniglot. In such scenario our method appears to offer a significant advantage over the baseline, since we obtain 75% in target accuracy, quite close to the optimal generalization. In scenarios where the target domain is arguably more similar to that of the source domain, e.g. transfer from MiniImagenet to Imagenet, the early-stopping from the validation accuracy yields a performance (35.0%) closer to the optimal generalization (35.6%), and in such case our method performs only slightly worse (34.8%) than the validation baseline. We observe a similar trend when the model is trained on the Quickdraw dataset : When transferring to Omniglot, the validation baseline leads to sub-optimal generalization, but estimating the target accuracy from the neural activation dynamics allows us to halt the training close to the optimal time. When transferring to Traffic Sign, the baseline performance yields reasonable performance, and our method is roughly on par with it. From this point, we will focus on settings where there is a significant gap in performance between the validation baseline and optimal generalization, for example the transfer from Birds to Quickdraw, the present in our illustration of Sec.2.1. Next we present similar experiments with two other meta-learning algorithms : Prototypical Networks and Matching Networks, which are shown in Tab.2.

Source dataset	MiniImagenet		Quickdraw		Birds
Target dataset	Omniglot	Imagenet	Omniglot	Traffic Sign	Quickdraw
Optimal Generalization	76%	35.6%	86.7%	40.6%	52.7%
Validation Baseline	51%	35.0%	77.7%	38.4%	38.8%
Our method	75%	34.8%	84.8%	38.6%	50.7%

Table 1: MAML based early-stopping using neural activation dynamics : Few-Shot Transfer Learning 5-way 1-shot, CNN based classifications. Optimal Generalization is the maximum of the target accuracy averaged over 50 target tasks, each task accuracy being averaged over query 15 shots (75 labelled examples). The performance of the Validation Baseline is the target accuracy evaluated when the validation accuracy is maximum, itself computed in the same fashion as for the target accuracy, but here using the validation data.

Meta-Learning algorithm	Matching Networks		Prototypical Networks	
Training dataset	MiniImagenet	MiniImagenet	MiniImagenet	Omniglot
Target dataset	Omniglot	Birds	Omniglot	Quickdraw
Optimal Generalization	77.3%	39.8%	69.3%	56.4%
Validation Baseline	72.9%	37.9%	64.4%	53.2%
Our method	76.0%	38.3%	65.8%	54.8%

Table 2: Early-Stopping based on the neural activation dynamics : Results with other meta-learning algorithms : Matching Networks, Prototypical Networks.

6 CONCLUSION

In this work we have presented empirical evidence that the overfitting point of Meta-Learning for deep neural networks for few-shot classification can often be estimated from simple statistics of neural activations and how they evolve throughout meta-training time. Our results suggest that key properties, or statistics of how feature extractors respond to the target input distribution can be found which are simple enough to be estimated from just a few *unlabelled* target input examples. However, the specific function of the activations, and the layer at which to measure them, need to be inferred. We demonstrate that these functions and layers of interest can be inferred and used to guide early stopping – leading to a new, and effective method for early stopping which represents a significant departure for the *de facto* standard practice of using a validation set. In few-shot learning these ingredients can be inferred from how the neural activation dynamics of the validation data relate to the validation accuracy. In few-shot transfer learning, they are inferred through searching for which function (in a given function space) and at which layer, that the activation dynamics of the target input domain "diverge" the most from those of the source domain. Finally, we have demonstrated how this approach can be used to optimize for target generalization in practice to perform early-stopping and thus improve overall generalization to distributions of novel few-shot classification tasks, while only using unlabelled support examples from a single target task.

REFERENCES

Giulia Denevi, Carlo Ciliberto, Riccardo Grazzi, and Massimiliano Pontil. Learning-to-learn stochastic gradient descent with biased regularization. In *International Conference on Machine Learning*, pp. 1566–1575. PMLR, 2019.

Guneet S. Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. *CoRR*, abs/1909.02729, 2019. URL http://arxiv.org/abs/1909.02729.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *CoRR*, abs/1703.03400, 2017. URL http://arxiv.org/abs/1703.03400.

Chelsea Finn, Aravind Rajeswaran, Sham M. Kakade, and Sergey Levine. Online meta-learning. *CoRR*, abs/1902.08438, 2019. URL http://arxiv.org/abs/1902.08438.

Nicholas Frosst, Nicolas Papernot, and Geoffrey Hinton. Analyzing and Improving Representations with the Soft Nearest Neighbor Loss. *arXiv e-prints*, art. arXiv:1902.01889, February 2019.

Micah Goldblum, Steven Reich, Liam Fowl, Renkun Ni, Valeriia Cherepanova, and Tom Goldstein. Unraveling meta-learning: Understanding feature representations for few-shot tasks. *CoRR*, abs/2002.06753, 2020. URL https://arxiv.org/abs/2002.06753.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Simon Guiroy, Vikas Verma, and Christopher J. Pal. Towards understanding generalization in gradient-based meta-learning. *CoRR*, abs/1907.07287, 2019. URL http://arxiv.org/abs/1907.07287.

Mikhail Khodak, Maria-Florina Balcan, and Ameet Talwalkar. Provable guarantees for gradient-based meta-learning. *CoRR*, abs/1902.10644, 2019. URL http://arxiv.org/abs/1902.10644.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *CoRR*, abs/1803.02999, 2018. URL http://arxiv.org/abs/1803.02999.

Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid Learning or Feature Reuse? Towards Understanding the Effectiveness of MAML. *arXiv e-prints*, art. arXiv:1909.09157, September 2019.

Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. *CoRR*, abs/1703.05175, 2017. URL http://arxiv.org/abs/1703.05175.

Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, and Hugo Larochelle. Meta-dataset: A dataset of datasets for learning to learn from few examples. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rkgAGAVKPr.

Oriol Vinyals, Charles Blundell, Timothy P. Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *CoRR*, abs/1606.04080, 2016. URL http://arxiv.org/abs/1606.04080.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *CoRR*, abs/1411.1792, 2014. URL http://arxiv.org/abs/1411.1792.

A EXPERIMENTAL DETAILS

CNN : We use the architecture proposed by Vinyals et al. (2016) which is used by Finn et al. (2017), consisting of 4 modules stacked on each other, each being composed of 64 filters of of 3 $\times$ 3 convolution, followed by a batch normalization layer, a ReLU activation layer, and a 2 $\times$ 2 max-pooling layer. With Omniglot, strided convolution is used instead of max-pooling, and images are downsampled to 28 $\times$ 28. With MiniImagenet, we used fewer filters to reduce overfitting, but used 48 while MAML used 32. As a loss function to minimize, we use cross-entropy between the predicted classes and the target classes.

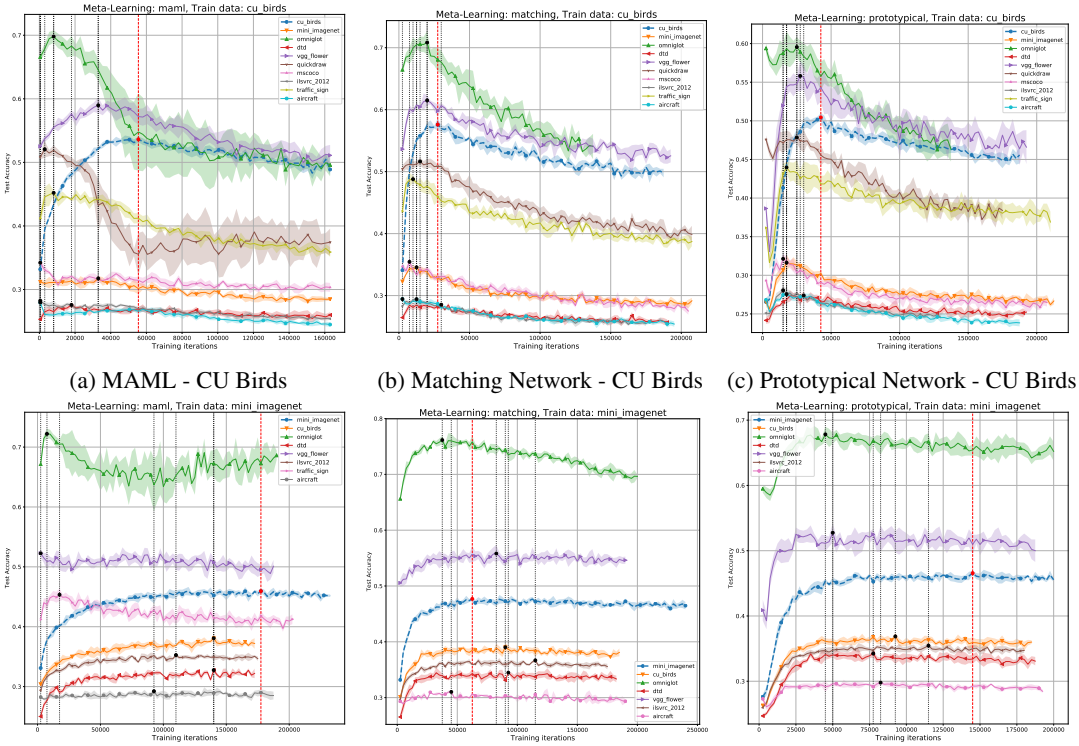
ResNet-18 : We use the same implementation of the Residual Network as in (Triantafillou et al., 2020).

For most of the hyperparameters, we follow the setup of (Triantafillou et al., 2020), but we set the main few-shot learning hyperparameters so as to follow the original MAML setting more closely, and in each setting, we consider a single target dataset at a time, with a fixed number of shots and classification ways. We use 5 steps of gradient descent for the task adaptations, 15 shots of query examples to evaluate the test accuracy of tasks. We don't use any learning rate decay during meta-training, and step-size of 0.01 when finetuning the models to new tasks.

Datasets : We use the MiniImagenet and Omniglot datasets, as well as the many datasets included in the Meta-Dataset benchmark (Triantafillou et al., 2020).

B COMPLETE EXPERIMENTAL RESULTS

B.1 THE ISSUE OF USING A VALIDATION SET FOR EARLY-STOPPING IN META-LEARNING



(a) MAML - CU Birds (b) Matching Network - CU Birds (c) Prototypical Network - CU Birds

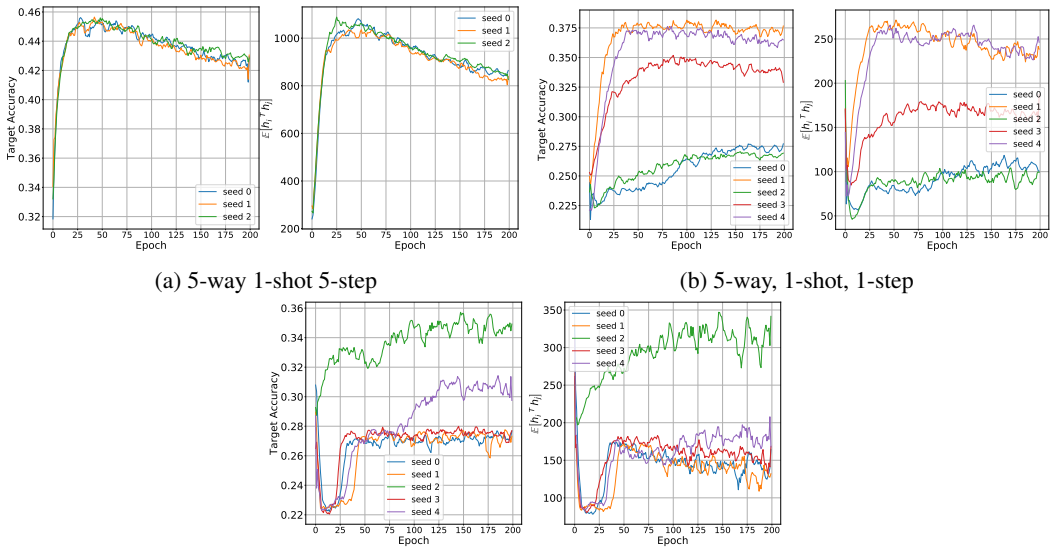
(d) MAML - MiniImagenet (e) Matching Network - MiniImagenet (f) Prototypical Network - MiniImagenet

Figure 11: Illustrating the issue of using meta-validation for early-stopping in Meta-Learning: MAML, Matching Network and Prototypical Network, trained on both MiniImagenet and CU Birds, with various target datasets from the Meta-Dataset benchmark. The validation early-stopping time t^*_{valid} (red dashed line) leads to sub-optimal generalization performances for the various target datasets (from the Meta-Dataset benchmark), each having their own optimal early-stopping time t^*_{target} (black dashed lines) at different times. We also observe $t^*_{target} \leq t^*_{valid}$ across the different settings.

B.2 THE RELATION BETWEEN THE NEURAL ACTIVATION DYNAMICS AND GENERALIZATION TO NOVEL TASKS

B.2.1 RELATION BETWEEN THE REPRESENTATION SPACE OF THE FEATURE EXTRACTOR AND TARGET GENERALIZATION

Here we present experimental results to support the **observation 1** that we make in 2.2, showing that the variation of generalization along meta-training time can be captured by a function of the neural activation dynamics that is independent of class labels.

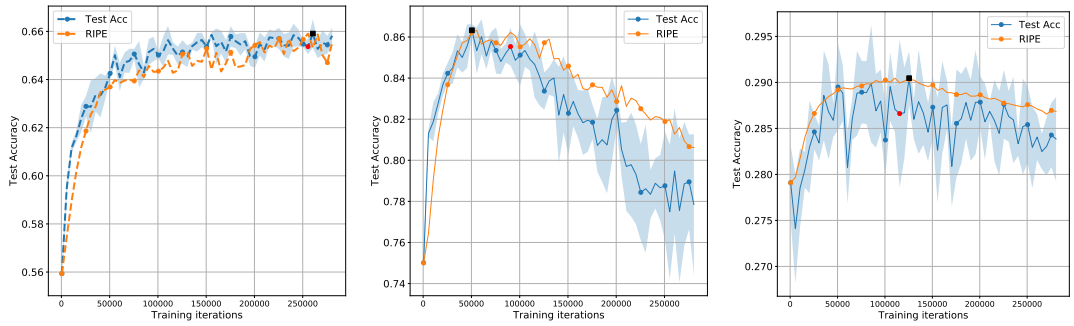


(a) 5-way 1-shot 5-step

(b) 5-way, 1-shot, 1-step

(c) 5-way, 5-shot, 1-step

Figure 12: Comparison between average inner product between representation vectors, and average target accuracy on target tasks in few-shot learning, for different regimes of MAML and First-Order MAML on MiniImagenet. Here the expression $[\mathbf{h}_i^T \mathbf{h}_j]$ is the expected inner product between representations for the target inputs, i.e. ψ_1 defined in Eq.2.



(a) Quickdraw to Quickdraw

(b) Quickdraw to Omniglot

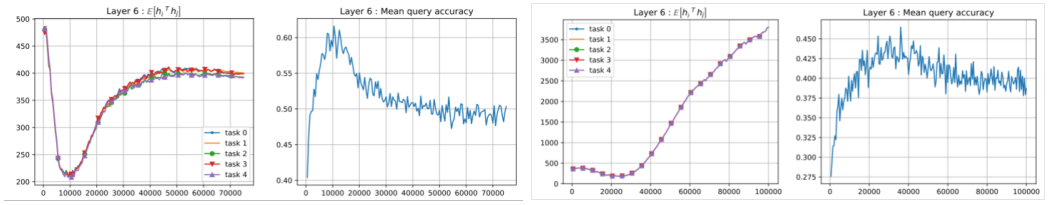
(c) Quickdraw to Imagenet

Figure 13: Measuring the expected representational inner product in Few-Shot Transfer Learning. MAML, 5-way 1-shot, training dataset : Quickdraw. The estimated early-stopping time of the metric t^*_ψ shows coincides well with the true optimal early-stopping time t^* and measuring the correlation (Pearson) between t^*_ψ and t^* gives $R = 0.925$ with a p-value near 0. Considering the gap between 1) the average performance of validation early-stopping across the three settings (58.69%); and 2) the maximum generalization across the three settings (61%), the average performance of the metric is at 59.7%, closing nearly half of the gap (43.74% of the gap).

B.2.2 Neural Activation Dynamics : Different levels of the feature extractor can reveal the variation of generalization

B.2.3 Different functions of the neural activation dynamics can reveal the variation of generalization

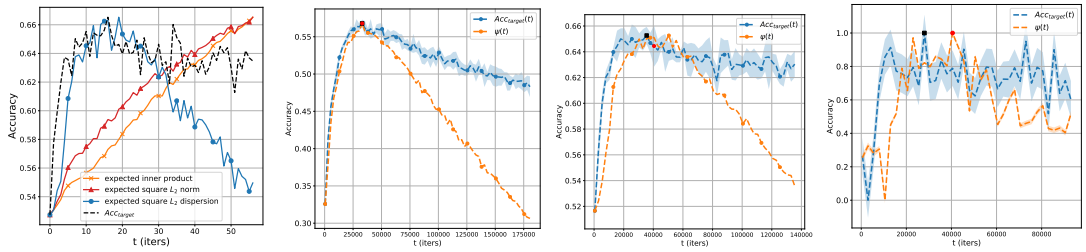
By expending the experimental setup further, we observed instances where a given metric had strong correlation with generalization but in a negative sense, i.e. that it was actually its *argmin* that coincided with optimal early-stopping time t^*. See Fig. 14 for examples of this phenomenon.



(a) Matching Network, ResNet-18, 5-way 5-shot (b) Prototypical Network, ResNet-18, 5-way 1-shot

Figure 14: Strong negative correlation between generalization and metrics on the representation space, where the minimum of the metric coincides with the maximum of generalization : Few-Shot Learning settings with MiniImagenet, 5-way 1-shot, with a ResNet-18. The metric is the expected inner product (left subfigures, represented by $\mathbb{E}[\mathbf{h}_i{}^T\mathbf{h}_j]$ where $\mathbf{h}_i$ stands for the representation vector for an input example $\mathbf{x}_i$). The metric is measured at the output of the feature extractor (6th block of the ResNet), and we show its measurement on 5 distinct tasks. The generalization (right subfigures) is averaged over 50 tasks.

We later observed that other statistical estimators can correlate with generalization.

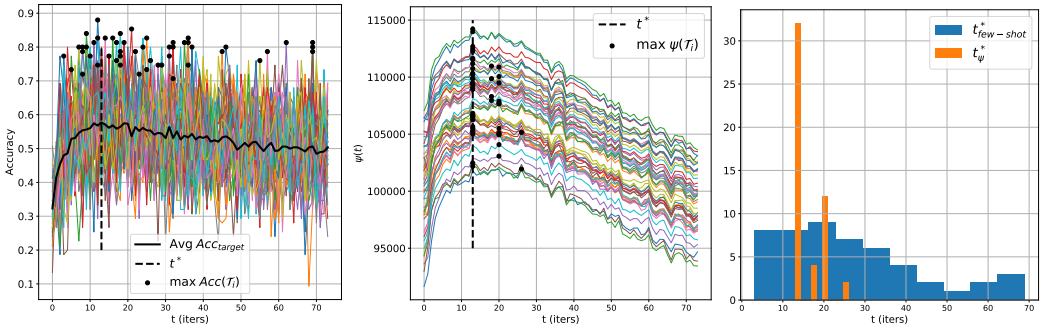


(a) Exclusive correlation between a specific metric and generalization (b) Expected l_2 Norm (c) Expected l_2 Dispersion (d) Expected Feature-Wise Variance

Figure 15: Different metrics of the representation space may have strong correlation with generalization, other than the expected inner product of Eq. 2). a) Prototypical, VGG Flower, 5-way 1-shot : out of three metrics which in other cases may be related with generalization (as in b), c), d) and Sec. **??**), here only the expected l_2 dispersion has a strong relation with generalization. b) Expected l_2 norm (Eq. 6); c) Expected square l_2 dispersion (Eq. 7, Prototypical Network, VGG Flower; d) Expected feature-wise variance (Eq. 8), Prototypical Network, Omniglot to Quickdraw. These results motivate our approach of considering a family of functions Ψ in which we must find the optimal function ψ^* given the setting, rather than trying to discover a single universal metric that would correlate to generalization in all scenarios. Even if such metric exists, it may not be estimated with enough efficiency to satisfy the requirement of using only a single support set to estimate t^*.

B.2.4 Functions of the neural activation dynamics : Task-wise variance of the estimate t^*_ψ

Here we present empirical results on the task-wise variance as discussed in the **observation 4** of Sec. 2.2. We begin by showing the task-wise variance for few-shot accuracy when evaluated with a single target task and assuming access to the query examples (15 shots). Few-shot accuracy exhibits a high variance as different tasks will peak at much different times, making it unfit to estimate t^*. On the other hand, for the metrics from Sec. 2.2, which are based on small order statistics (mean and variance) the estimated early-stopping time exhibits drastically lower variance. See Fig. 16 for an example, where we use MAML in few-shot learning (5-way 1-shot) with the Aircraft dataset, and where we use the expected square l_2 norm for the metric. As we can see, in Fig. 16, measuring the metric on different tasks merely offsets the response curve but bears almost no change on the trend of the curve itself. This also relates to our assumption that the variation of target generalization in Meta-Learning might be linked to a function of the neural activation dynamics that is class agnostic.



(a) Few-Shot Accuracy (b) Expected Square l_2 Norm (c) Histogram of $t^*_{few-shot}$ vs. t^*_ψ

Figure 16: Task-wise variance : The maximum of the true average target accuracy (generalization) is at 57.7% with optimal early-stopping time t^* at $t = 14$. a) Displaying the query accuracy of 50 target tasks. The estimated early-stopping time $t^*_{few-shot}$ from the query accuracy of a single task has a task-wise standard deviation of 17.3 and a mean at $t = 26.3$. The resulting average generalization is 54.5%; b) expected square l_2 norm, 50 tasks. The metric exhibits very low task-wise standard deviation of its estimated early-stopping time $t^*\psi$ (3.7) and a mean at $t = 15.6$. Resulting average generalization is 57.4%, outperforming validation-based early-stopping. c) Histogram : $t^*_{few-shot}$ shows high variance while $t^*\psi$ is mostly concentrated near t^*. Setting used : MAML, CNN, Aircraft, 5-way 1-shot.

B.3 Capacity of the continuous function space Ψ (defined by the three moments of Eq.9) to contain good solutions ψ^*

The moments m_1, m_2 and m_3 of Eq.9 define the parametric function space $\Psi = \{\psi(\varphi_l(\mathbf{X}); \mathbf{w}) \mid \mathbf{w} \in \mathbb{R}^3, l \in [1..L]\}$ where $\psi(\varphi_l(\mathbf{X}); \mathbf{w}) = w_1 m_1 + w_2 m_2 + w_3 m_3$ and $\mathbf{w} = [w_1, w_2, w_3] \in \mathbb{R}^3$. This parametric function space Ψ. Here we have experimentally observed that Ψ has enough complexity to contain a good solution function ψ^*, for different meta-learning settings, both in few-shot learning and few-shot transfer learning, as shown in Tab.3.

$d^* / \max Acc_{target}$	Omniglot	VGG Flower	Birds	Aircraft	Quickdraw
MAML	0%	0%	0%	0%	0%
Matching Net	0%	0%	0.17%	0.41%	–
Proto Net	0.17%	0%	1.03%	0.43%	–

(a) Few-Shot Learning

train algo	MAML	Prototypical Network	Matching Network	Matching Network
$p(\mathcal{T}_{train})$	Quickdraw	Omniglot	Omniglot	MiniImagenet
$p(\mathcal{T}_{target})$	Omniglot	Quickdraw	Quickdraw	Omniglot
$d^* / \max Acc_{target}$	0%	0%	0.07%	0.66%

(b) Few-Shot Transfer Learning

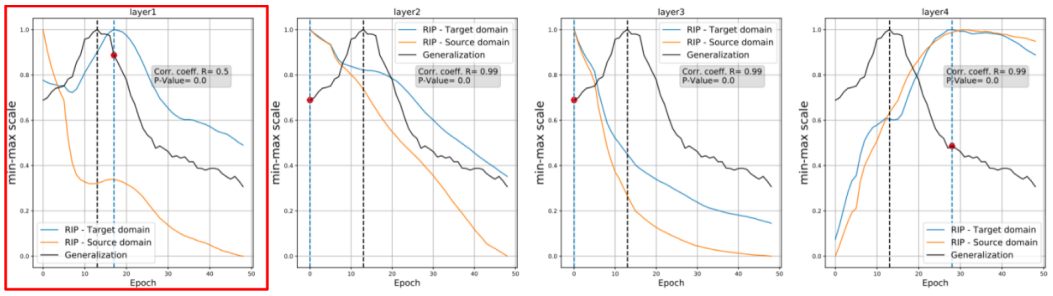
Table 3: Our function space Ψ is rich enough to contain good solution functions, both in few-shot learning and few-shot transfer learning. For multiple settings, we present the relative minimal gap $\max_t Acc_{target}(t) - Acc_{target}(t^*_\psi) / \max_t Acc_{target}(t)$ achieved by ψ^*.

B.4 Learning ψ^* in Few-Shot Transfer Learning

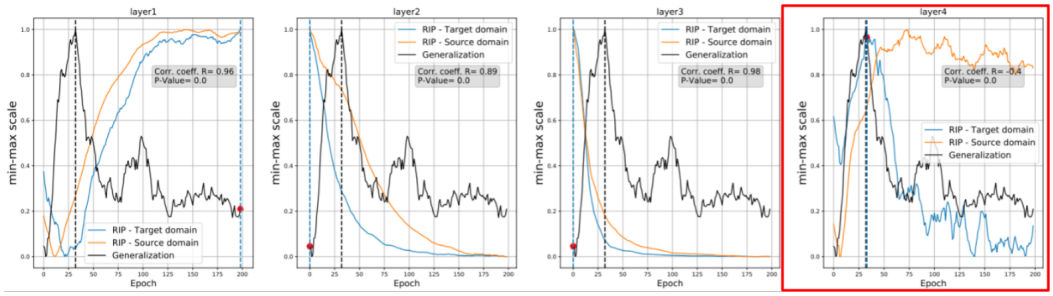
B.4.1 Finding $\mathbf{w}^*$ and l^* where $\psi_{target}(t)$ and $\psi_{valid}(t)$ "diverge" the most

As we added more experimental settings for Few-Shot Transfer Learning, we observed instances where, for a given metric measured at the representation space φ_L, there was no strong link with generalization, but when measuring the metric at lower hidden-layers than φ_L, then we observed a strong correlation with generalization. We illustrate this in Fig. 17. Theses results motivated our

approach of considering the whole neural activation dynamics (all layers), rather than only those the final layer of the feature extractor alone, in our search for functions linked to generalization. Then in Fig.18 we conducted a more systematic analysis, but concerned with identifying the right functions ψ^*, with results suggesting that functions ψ showing stronger "divergence" (negative correlation) between the target and validation dynamics, will more likely lead to a higher target accuracy if we stop at their peak time.



(a) Transfer : Omniglot to MiniImagenet, MAML. The critical depth, i.e. the one where measuring the expected inner product (here marked as RIP) predicts generalization on the target domain, is at layer 1, even though the critical depth for the source domain was at layer 4.



(b) Transfer : MiniImagenet to Omniglot, MAML. The critical depth for the target domain is at layer 4, the same as for the source domain.

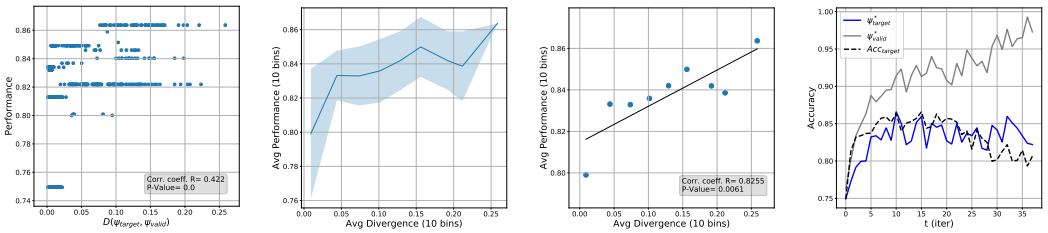
Figure 17: Generalization can correlate to a metric at different levels of neural activations. Here the critical layer l^* (squared in red) is identified by searching for the highest divergence between the validation and target neural activation dynamics.

Correlation between $D(\psi_{target}, \psi_{target},)$ and Generalization		
MAML Quickdraw ↓ Omniglot	Prototypical Network Omniglot ↓ Quickdraw	Matching Network MiniImagenet ↓ Omniglot
0.82	0.75	0.81

Table 4: Correlation between $D(\psi_{target}, \psi_{target},)$ and Generalization, for different few-shot learning settings. The correlation is computed as in the analysis of Fig. 18c. The results show that functions exhibiting high divergence between the validation and target neural activation dynamics are likely to lead to good generalization performance on the target distribution.

B.5 Evaluating the performance of our early-stopping method when using the continuous function space Ψ defined by the three moments of Eq.9

Here we present a few experimental results where we apply our early-stopping methof in the continuous function space Ψ. Since there are only three weights to tune, namely w_1, w_2 and w_3, we don't suffer from the *curse of dimensionality*, which is the classic motivation for using gradient-based optimization of neural networks with many parameters. This allows for a search based optimization of $\mathbf{w}$.



(a) Functions with high *divergence* between valid and target dynamics are more likely to achieve higher generalization

(b) Average *divergence* vs. average performance

(c) Strong correlation between average *divergence* and average performance

(d) Solution function measured on the valid and target examples

Figure 18: Divergence of the neural activation dynamics as an indication for early-stopping : MAML, Quickdraw to Omniglot, 5-way 1-shot. For 50 pairs of target and validation tasks, we measure all the four metrics (expected inner product; expected square l_2 norm; expected square l_2 dispersion; feature-wise variance), at all the layers of the feature extractor. For each measurement (layer and metric), we plot the divergence $D(\psi_{target}, \psi_{valid}) = 1 - r(\psi_{target}, \psi_{valid})$ (where r is the sample Pearson correlation through time) against the obtained generalization *performance* (here for this example we simply use $t^*_\psi = \text{argmax}_t\ \psi_{target}$). a) while functions with low divergence may lead from very poor to very good performance, functions showing high divergence between the validation and target dynamics very likely lead to good performance; b) We illustrate this by dividing the points into 10 bins along the divergence axis, and in each bin we average the divergence and performance, and plot the averages from the bins (blue curve) along with the standard deviations (blue dashed area). c) correlation between the bin averaged divergences and performances d) We display ψ^* that achieved the highest divergence, which we can observe by its two response curves on its pair of validation and target tasks.

Algorithm	MAML	
Source dataset	Quickdraw	MiniImagenet
Target dataset	Omniglot	Omniglot
Baseline	77.9%	54.6%
Our method	**81%**	**75%**

Table 5: Performance of our method - Few-Shot Transfer Learning, MAML

Algorithm	Prototypical Network	
Source dataset	Omniglot	MiniImagenet
Target dataset	Quickdraw	Omniglot
Baseline	53.2%	60.1%
Our method	**54.6%**	**63.3%**

Table 6: Performance of our method - Few-Shot Transfer Learning, Prototypical Network

Algorithm	Matching Network
Source dataset	MiniImagenet
Target dataset	Omniglot
Baseline	73.75%
Our method	**75%**

Table 7: Performance of our method - Few-Shot Transfer Learning, Matching Network