MODULATING TRANSFER BETWEEN TASKS IN GRADIENT-BASED META-LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning-to-learn or *meta-learning* leverages data-driven inductive bias to increase the efficiency of learning on a novel task. This approach encounters difficulty when transfer is not mutually beneficial, for instance, when tasks are sufficiently dissimilar or change over time. Here, we use the connection between gradient-based meta-learning and hierarchical Bayes (Grant et al., 2018) to propose a mixture of hierarchical Bayesian models over the parameters of an arbitrary function approximator such as a neural network. Generalizing the model-agnostic meta-learning (MAML) algorithm (Finn et al., 2017), we present a stochastic expectation maximization procedure to jointly estimate parameter initializations for gradient descent as well as a latent assignment of tasks to initializations. This approach better captures the diversity of training tasks as opposed to consolidating inductive biases into a single set of hyperparameters. Our experiments demonstrate better generalization performance on the standard *mini*ImageNet benchmark for 1-shot classification. We further derive a novel and scalable non-parametric variant of our method that captures the evolution of a task distribution over time as demonstrated on a set of synthetic regression tasks as well as an evolving dataset adapted from *mini*ImageNet.

1 INTRODUCTION

Meta-learning algorithms aim to increase the efficiency of learning by treating task-specific learning episodes as examples from which to generalize (Schmidhuber, 1987). The central assumption of a meta-learning algorithm is that some tasks are inherently related and so inductive transfer has the potential to be beneficial both for sample efficiency and predictive power (Caruana, 1993; 1998; Baxter, 2000). Recent meta-learning algorithms have encoded this assumption by learning global hyperparameters that provide a task-general inductive bias. In learning a single set of hyperparameters that parameterize, for example, a metric space (Vinyals et al., 2016) or an optimizer for gradient descent (Ravi & Larochelle, 2017; Finn et al., 2017), these meta-learning algorithms make the assumption that tasks are equally related and therefore mutual transfer is appropriate. This assumption has been cemented in recent few-shot learning benchmarks, which consist of a set of tasks generated in a systematic manner, for example, by varying the parameters of a periodic function to generate a family of regression tasks (*e.g.,* Finn et al., 2017) or sampling uniformly over a large vocabulary of classes to create related few-shot discrimination tasks (*e.g.,* Vinyals et al., 2016).

However, the real world often presents scenarios in which an agent must decide what degree of transfer is appropriate. In the case of positive transfer, a subset of tasks may be more strongly related to each other and so non-uniform transfer poses a strategic advantage. Negative transfer in the presence of dissimilar or outlier tasks worsens generalization performance (Rosenstein et al., 2005). Moreover, when the underlying task distribution is non-stationary, inductive transfer to initial tasks should exhibit graceful degradation to address the catastrophic forgetting problem (Kirkpatrick et al., 2017). In order to deal with this degree of task heterogeneity, extensive task-switching literature reveals that people detect and readily adapt even in the face of significantly novel contexts (see Collins & Frank, 2013, for a review), while studies of animal cognitive in combination with neural evidence suggest a mechanism for partitioning events into types that typify reactive behavior (Gershman et al., 2015; Tervo et al., 2016).

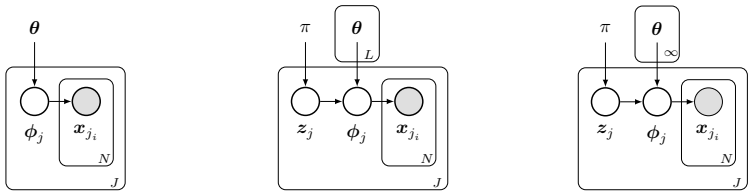

Figure 1: (a) The standard hierarchical Bayesian model for multi-task learning. A set of global parameters $\boldsymbol{\theta}$ provides an inductive bias for the estimation of task-specific parameters $\boldsymbol{\phi}_j$. (b) In a mixture of hierarchical Bayesian models, the cluster assigment of each task-specific parameter set $\boldsymbol{\phi}_j$ is represented with a latent Categorical variable $\boldsymbol{z}_j$. (c) Allowing an unbounded number of mixture components instantiates a non-parametric model that has the potential to grow with the data in complexity.

In this work, we learn a mixture of hierarchical models that allows the meta-learner to adaptively select over a set of learned parameter initializations for gradient-based fast adaptation to a new task. The method is equivalent to clustering task-specific parameters in the hierarchical model induced by recasting gradient-based meta-learning as hierarchical Bayes (Grant et al., 2018) and generalizes the model-agnostic meta-learning (MAML) algorithm introduced in Finn et al. (2017). By treating the assignment of task-specific parameters to clusters as latent variables in a probabilistic model, we can directly detect similarities between tasks on the basis of the task-specific likelihood, which may be parameterized by a black-box model such as a neural network. Our approach therefore alleviates the need for explicit geometric or probabilistic modelling assumptions about the weights of a parametric model and provides a scalable method to regulate information transfer between episodes.

Furthermore, we adapt our latent variable model to propose a scalable non-parametric extension derived from a stochastic point approximation to the posterior in a Dirichlet process mixture model (DPMM) (Rasmussen, 2000).

To the best of our knowledge, no previous work has considered a scalable stochastic point estimation in a non-parametric mixture model. Furthermore, we are not aware of prior work applying non-parametric mixture models to high-dimensional function approximations such as modern neural networks with end-to-end learning. The non-parametric extension allows the model complexity to evolve by introducing or removing clusters in alignment with the changing composition of the dataset, and preserves performance on previously encountered tasks better than a parameteric counterpart.

2 Learning to learn in situations of task variability

The goal of a meta-learner is to extract task-general knowledge through the experience of solving a number of related tasks. By leveraging this acquired prior knowledge, the meta-learner has the potential to quickly adapt to novel tasks even in the face of limited data or limited computation time (Schmidhuber, 1992). Recent approaches to meta-learning consolidate information from a set of training tasks into the parameters of a mapping to be applied at test time to a novel task. This mapping has taken the form of, for instance, a learned metric space (*e.g.,* Vinyals et al., 2016; Snell et al., 2017), a trained recurrent neural network (*e.g.,* Santoro et al., 2016), or an optimization algorithm with learned parameters (*e.g.,* Ravi & Larochelle, 2017).

The consolidation of all information garnered from the training tasks into a single set of hyperparameters does not take into account potential variability in the task distribution. In contrast, learning relationships between tasks has the potential to aid the efficacy of a meta-learning algorithm by modulating both positive and negative transfer (Thrun; Zhang & Schneider, 2010; Rothman et al., 2010; Zhang & Yeung, 2014; Xue et al., 2007). Defining an appropriate notion of task relatedness is a difficult problem that is exacerbated in the high-dimensional parameter space of models such as neural networks. Furthermore, it is not immediately clear how to define a task representation for methods that rely on computation of distance metric over the activation space of examples for the purposes of classifcation to an exemplar or prototype (Vinyals et al., 2016; Snell et al., 2017)

To address this issue, we propose an algorithm for parameter estimation a mixture of hierarchical probabilistic models. By making use of probabilistic clustering, we capture task relatedness by estimating the likelihood of assigning each task to a mixture component based only on the task

Algorithm ($\mathscr{D}$, β, α, τ, L_0, J, N, M, K, G_0)

Initialize $L \leftarrow L_0$ and $\{\boldsymbol{\theta}^{(1)}, \dots, \boldsymbol{\theta}^{(L)}\}$ as $\boldsymbol{\theta}^{(\ell)} \sim G_0$ for ℓ in $1, \dots, L$

while *not converged* **do**

  Draw tasks $\mathcal{T}_1, \dots, \mathcal{T}_J \sim p_{\mathscr{D}}(\mathcal{T})$

  for j *in* $1, \dots, J$ **do**

    Draw task-specific datapoints $\boldsymbol{x}_{j_1}, \dots, \boldsymbol{x}_{j_N}, \boldsymbol{x}_{j_{N+1}}, \dots \boldsymbol{x}_{j_{N+M}} \sim p_{\mathcal{T}_j}(\,\boldsymbol{x}\,)$

    (Draw a parameter initialization for a new cluster $\boldsymbol{\theta}^{(L+1)} \sim G_0$)

    for ℓ *in* $\{1, \dots, L, (L+1)\}$ **do**

      Set $\boldsymbol{\phi}_j^{(\ell)} \leftarrow \boldsymbol{\theta}^{(\ell)}$ as the initialization for gradient-based fast adaptation

      Compute mode estimate via $\boldsymbol{\phi}_j^{(\ell)} \leftarrow \boldsymbol{\phi}_j^{(\ell)} + \alpha \nabla_{\boldsymbol{\phi}} \sum_i \log p(\,\boldsymbol{x}_{j_i} \mid \boldsymbol{\phi}_j^{(\ell)}\,)$ for K steps

    Compute assignment of tasks to clusters as $\gamma_j \leftarrow$ E-STEP $(\{\boldsymbol{x}_{j_i}\}_{i=1}^N, \{\boldsymbol{\phi}_j^{(\ell)}\}_{\ell=1}^L)$

  Update $\boldsymbol{\theta}^{(\ell)} \leftarrow \boldsymbol{\theta}^{(\ell)} +$ M-STEP $(\{\{\boldsymbol{x}_{j_{N+i}}\}_{i=1}^M, \boldsymbol{\phi}_j^{(\ell)}, \gamma_j\}_{j=1}^J)$ for ℓ in $1, \dots, L$

  (Update global prior G_0 with statistics based on G_0 and $\{\boldsymbol{\theta}_1, \dots\}$)

return $\{\boldsymbol{\theta}^{(1)}, \dots\}$

Algorithm 2: Stochastic expectation maximization via gradient-based meta-learning for clustering of task-specific parameters in the few-shot learning setting. (Commands in parentheses are executed only for the nonparametric variant of the algorithm.)

E-STEP ($\{\boldsymbol{x}_{j_i}\}_{i=1}^N, \{\boldsymbol{\phi}_j^{(\ell)}\}_{\ell=1}^L$)

  return τ-softmax$_\ell(\sum_i \log p(\,\boldsymbol{x}_{j_i} \mid \boldsymbol{\phi}_j^{(\ell)}\,))$

M-STEP ($\{\boldsymbol{x}_{j_i}\}_{i=1}^M, \boldsymbol{\phi}_j^{(\ell)}, \gamma_j$)

  return $\beta \nabla_{\boldsymbol{\theta}}[\sum_{j,i} \gamma_j \log p(\,\boldsymbol{x}_{j_i} \mid \boldsymbol{\phi}_j^{(\ell)}\,)]$

Subroutine 3: The E-STEP and M-STEP for a finite mixture of hierarchical Bayesian models.

loss itself. This faciliates the joint estimation of the task-specific cluster assignments and network parameters in a single gradient-based fast adaptation step without defining an explicit notion of relatedness. The result is a scalable end-to-end meta-learning algorithm, further detailed in the following section, that is capable of modulating the transfer of information across tasks to better generalize to heterogenous or evolving task distributions.

3 Learning latent structure with gradient-based meta-learning

In this work, we propose a probabilistic mixture of hierarchical models for meta-learning to scalably learn the relatedness among tasks during gradient-based meta-learning. This leverages an interpretation of model-agnostic meta-learning (MAML) as a method for parameter estimation in a probabilistic model. MAML (Finn et al., 2017) is a gradient-based meta-learning approach that estimates global parameters to be shared among task-specific models as an initialization for a few steps of gradient descent. MAML also admits a natural interpretation as parameter estimation in a hierachical model, where the learned initialization acts as data-driven regularization for task-specific parameters (Grant et al., 2018).

In particular, Grant et al. (2018) cast MAML as posterior inference for task-specific parameters $\boldsymbol{\phi}_j$ given a batch of task data $\boldsymbol{x}_{j_{1:N}}$ and a prior over $\boldsymbol{\phi}_j$ that is induced by early stopping of an iterative descent procedure. In particular, a few steps of gradient descent on the negative log-likelihood $-\log p(\,\boldsymbol{x}_{j_{1:N}} \mid \boldsymbol{\phi}_j^{(\ell)}\,)$, starting from $\boldsymbol{\phi}_j = \boldsymbol{\theta}$ can be understood as mode estimation of the posterior $p(\,\boldsymbol{\phi}_j^{(\ell)} \mid \boldsymbol{x}_{j_{1:N}}, \boldsymbol{\theta}\,)$. The mode estimates $\hat{\boldsymbol{\phi}}_j = \boldsymbol{\theta} + \alpha \nabla_{\boldsymbol{\theta}} \log p(\,\boldsymbol{x}_{j_{1:N}} \mid \boldsymbol{\theta}\,)$ for each task $\mathcal{T}_j$ are then combined to evaluate the marginal likelihood as

$$p\Big(\,\{\boldsymbol{x}_{1:M}\}_{j=1}^J \mid \boldsymbol{\theta}\Big) = \prod_j \int p\,(\,\boldsymbol{x}_{1:M} \mid \boldsymbol{\phi}_j\,)\,p\,(\,\boldsymbol{\phi}_j \mid \boldsymbol{\theta}\,)\,\mathrm{d}\boldsymbol{\phi}_j \approx \prod_j p\Big(\,\boldsymbol{x}_{1:M} \mid \hat{\boldsymbol{\phi}}_j\Big)\ .$$

A training dataset can be summarized in an empirical Bayes point estimate of $\boldsymbol{\theta}$ computed by gradient descent in $-\log p(\,\{\boldsymbol{x}_{1:M}\}_{j=1}^J \mid \boldsymbol{\theta}\,)$ (Lehmann & Casella, 2006). As such, the likelihood of a datapoint $\boldsymbol{x}$ sampled from a new task depends only on the empirical Bayes estimate of $\boldsymbol{\theta}$.

Model	1-shot		
nearest neighbor[a]	41.08	±	0.70
matching network FCE (Vinyals et al., 2016)[a]	43.56	±	0.84
meta-learner LSTM (Ravi & Larochelle, 2017)	43.44	±	0.77
SNAIL (Mishra et al., 2018)[b]	45.1	±	——
prototypical networks (Snell et al., 2017)[c]	46.61	±	0.78
MAML (Finn et al., 2017)	48.70	±	1.84
LLAMA (Grant et al., 2018)	49.40	±	1.83
KNN + GNN embedding (Garcia & Bruna, 2017)	49.44	±	0.28
mAP-DLM (Triantafillou et al., 2017)	49.82	±	0.78
fwCNN (Hebb) (Munkhdalai & Trischler, 2018)	50.21	±	0.37
GNN (Garcia & Bruna, 2017)	50.33	±	0.36
Our method (clustering all layers of a neural network)	50.80	±	1.70

Table 1: Performance according to meta-test accuracy on the *mini*ImageNet 5-way, 1-shot classification benchmark from Vinyals et al. (2016). We report further comparisons in Appendix A.

If the task distribution is heterogeneous, assuming a single parameter initialization $\boldsymbol{\theta}$ is not suitable because it is unlikely that the point estimate computed by a few steps of gradient descent will sufficiently adapt the task-specific parameters $\boldsymbol{\phi}$ to a diversity of tasks. Using the probabilistic interpretation of Section 2, we may deal with the variability in the tasks $\mathcal{T}_j$ by assuming that each set of task-specific parameters $\boldsymbol{\phi}_j$ is drawn from a mixture of base distributions each of which is parameterized by a hyperparameter $\boldsymbol{\theta}^{(\ell)}$. In the case of hard assignments with a finite number of components, we may introduce a latent binary vector $\boldsymbol{z}_j = [\boldsymbol{z}_j^{(1)}, \ldots, \boldsymbol{z}_j^{(L)}]$ to depict the model in Figure 1 (b). Here, a value of $\boldsymbol{z}_j^{(\ell)} = 1$ indicates that the task-specific parameters j are drawn from cluster ℓ.

3.1 Expectation maximization for gradient-based meta-learning with latents

Under the mixture model formulation, direct maximization of the marginal likelihood requires the marginalization of the cluster assignment $\boldsymbol{z}_j$ of each task-specific parameter $\boldsymbol{\phi}_j$. This is a combinatorial optimization problem which can become intractable to solve directly. The expectation maximization (EM) algorithm (Dempster et al., 1977) is an iterative coordinate ascent method that bypasses the need for enumerating all possible cluster assignments. It alternates between the expectation step (E-STEP), which computes the expectation of a latent variable and the maximization step (M-STEP), which searches for the maximum of the likelihood with the expected value of the latent variables as evidence.

To ensure scalability as well as compatibility with stochastic gradient-based meta-learning, we use stochastic optimization (Robbins & Monro, 1951) to compute the quantities in both the E-STEP and the M-STEP. Accordingly, Algorithm 2 performs a single EM iteration per training batch by following a noisy but unbiased estimate of the gradient in order to estimate task-specific parameters $\boldsymbol{\phi}_j$, which in turn determine the cluster assignments by computation of the expectation of the latent variable $\boldsymbol{z}_j$. Stochastic optimization approaches to EM are motivated by a view of the algorithm as optimizing a single free energy at both the E-STEP and the M-STEP (Neal & Hinton, 1998).

The M-STEP in Algorithm 3 is similar to the MAML algorithm in Finn et al. (2017). The E-STEP in Algorithm 3 leverages the connection between gradient-based meta-learning and hierarchical Bayes (HB) (Grant et al., 2018) to employ the task-specific objective as a surrogate for the posterior probability of cluster assignment. Accordingly, for each task j and cluster ℓ, we follow the gradients to minimize the negative log likelihood on the training datapoints using the cluster parameters $\boldsymbol{\theta}^{(\ell)}$ as initialization. This allows us to obtain a modal point estimate of the task parameters $\hat{\boldsymbol{\phi}}_j^{(\ell)}$. Based on the likelihood of the same data under the new parameter estimate $\hat{\boldsymbol{\phi}}_j^{(\ell)}$, we compute the cluster assignment probabilities as

$$\gamma_j^{(\ell)} := p\big(\boldsymbol{z}_j = \ell \mid \boldsymbol{x}_{j_{1:N}}, \boldsymbol{\theta}^{(1:L)}\big) \propto \int p(\boldsymbol{x}_{j_{1:N}} \mid \boldsymbol{\phi}_j^{(\ell)})\, p(\boldsymbol{\phi}_j^{(\ell)} \mid \boldsymbol{\theta}^{(\ell)})\, \mathrm{d}\boldsymbol{\phi}_j^{(\ell)} \approx p(\boldsymbol{x}_{j_{1:N}} \mid \hat{\boldsymbol{\phi}}_j^{(\ell)})\ .$$

The cluster means $\boldsymbol{\theta}^{(\ell)}$ are then updated by gradient descent on the validation loss. See Algorithm 2 for more details. Note that, unlike other recent approaches to probabilistic clustering (*e.g.,* Bauer

E-STEP ($\{\boldsymbol{x}_{j_i}\}_{i=1}^{N}, \{\boldsymbol{\phi}_j^{(\ell)}\}_{\ell=1}^{L+1}$, *concentration* ζ, *spawn threshold* ϵ)

DPMM log-likelihood, $\rho_j^{(\ell)} \leftarrow \sum_i \log p(\, \boldsymbol{x}_{j_i} \mid \boldsymbol{\phi}_j^{(\ell)}\,) + \log n^{(\ell)}$ for all ℓ in $1, \ldots, L$

DPMM log-likelihood for new component, $\rho_j^{(L+1)} \leftarrow \sum_i \log p(\, \boldsymbol{x}_{j_i} \mid \boldsymbol{\phi}_j^{(L+1)}\,) + \log \zeta$

$\gamma_j \leftarrow \tau\text{-softmax}(\rho_j^{(1)}, \ldots, \rho_j^{(L+1)})$

if $\gamma_j^{(L+1)} > \epsilon$ **then**

Expand the model by incrementing $L \leftarrow L + 1$

else

Renormalize $\gamma_j \leftarrow \tau\text{-softmax}(\rho_j^{(1)}, \ldots, \rho_j^{(L)})$

return γ_j

M-STEP ($\{\boldsymbol{x}_{j_i}\}_{i=1}^{M}, \boldsymbol{\phi}_j^{(\ell)}, \gamma_j$, *concentration* ζ)

return $\beta\ \nabla_{\boldsymbol{\theta}}[\sum_{j,i} \gamma_j \log p(\, \boldsymbol{x}_{j_i} \mid \boldsymbol{\phi}_j^{(\ell)}\,) + L \log \zeta + \sum_\ell \log \Gamma(n^{(\ell)})]$

Subroutine 4: The E-STEP and M-STEP for an infinite mixture of hierarchical Bayesian models.

et al., 2017) we adhere to the episodic meta-learning setup for both training and testing, since, since only the task support set $\boldsymbol{x}_{j_{1:N}}$ is used to compute both the point estimate $\hat{\boldsymbol{\phi}}_j^{(\ell)}$ as well as the cluster responsibilities. $\gamma_j^{(\ell)}$

4 Scalable non-parametric mixtures for evolving data

Since the assignment of datapoints to components is not directly observed and could change in time, the choice of the number of components in a finite mixture analysis poses a difficult model selection problem. A natural way to regularize mixture models is to place priors over the cluster assignments to control the distribution over cluster densities. In this case, K can also be estimated as part of the clustering algorithm in a more computationally efficient manner. Therefore, we introduce an *infinite* or *non-parametric* mixture model (Rasmussen, 2000) that allows the number of components to vary during inference. We then derive a scalable stochastic estimation of task-to-cluster assignments for an unconstrained number of task clusters. This obviates the need for an *a priori* fixed number of components and enables the model to unboundedly adapt its complexity according to the observed data. Among *non-parametric* models, the Dirichlet process mixture model (DPMM) and the related Chinese restaurant process (CRP) are among the most popular in practice (Gershman & Blei, 2012). The Chinese restaurant process (CRP) distribution, over $\boldsymbol{z}$ is defined in terms of a single positive concentration parameter ζ and can be most easily described by how to draw samples from it. For a sequence of $1 : J$ tasks, the first task is assigned to the first cluster; the Jth subsequent task is then assigned a cluster drawn from the following distribution:

$$p\,(\, \boldsymbol{z}_J = \ell \mid \boldsymbol{z}_{1:J-1}, \zeta\,) = \begin{cases} \frac{N^{(\ell)}}{N+\zeta} & \text{for} \quad \ell \leq L \\ \frac{\zeta}{N+\zeta} & \text{for} \quad \ell = L+1, \end{cases}$$

where L indicates the number of occupied clusters and $N^{(\ell)}$ indicates the number of tasks already occupying a cluster ℓ.

The joint data log likelihood can be written as

$$\log p\,(\, \boldsymbol{x}_{j_{1:N}}, \boldsymbol{\phi}_{1:J}, \mid \boldsymbol{\theta}\,) = \sum_{j,i} \gamma_j \log p(\boldsymbol{x}_{j_i} \mid \boldsymbol{\phi}_j^{(\ell)}) + \sum_j \gamma_j \log p(\boldsymbol{\phi}_j^{(\ell)} \mid \boldsymbol{\theta}) + L \log \zeta + \sum_\ell \log \Gamma(n^{(\ell)})\ ,$$

where γ_j is the reponsibility of cluster ℓ for task-specific parameters j.

One simple and popular sampling algorithm for parameter estimation is Gibbs sampling (Neal, 2000) where we draw from conditional distributions on the cluster assignments until convergence to a stationary distribution. The conditional draws follow:

$$p\,(\, \boldsymbol{z}_{J+1} = \ell \mid \mathcal{T}_j, \boldsymbol{z}_{1:J}, \zeta, \,) \propto \begin{cases} N^{(\ell)} & \int P(x_i \mid \theta) dG_l(\theta) \approx N^{(\ell)} P(x_i \mid \phi) P(\phi \mid \theta_l) \quad \text{for} \quad \ell \leq L \\ \zeta & \int P(x_i \mid \theta) dG_0(\theta) \approx \zeta P(x_i \mid \hat{\phi}) P(\hat{\phi} \mid \theta_0) \quad \text{for} \quad \ell = L+1, \end{cases}$$

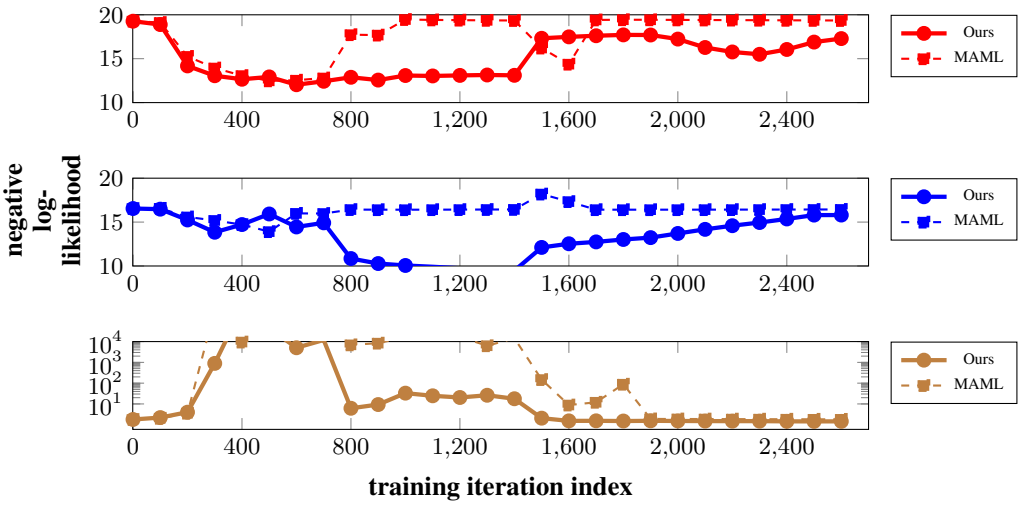

Figure 5: Mixture component likelihoods on an evolving dataset that generates even polynomial regression tasks for 700 iterations, then odd polynomial regression tasks until iteration 2100 at which point it generates sinusoidal regression tasks. We plot the negative log likelihood of the data across all tasks under each cluster indexed by ℓ (*i.e.,* $\ell = 0$ in blue (top), $\ell = 1$ in red (middle), and $\ell = 2$ in brown (bottom)). Note the change in loss to the second (red) cluster at 700 iterations when the odd polynomial tasks are introduced. At 2100 iterations, the sinusoid tasks (third row) cause a third (brown) cluster to be introduced.

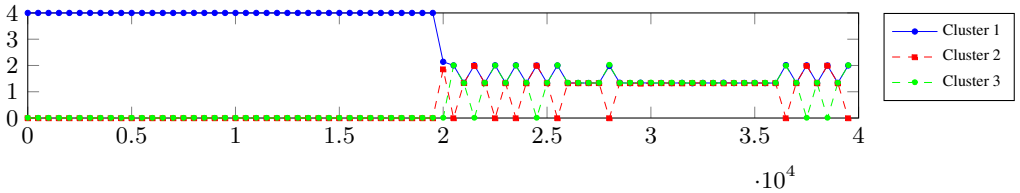

Figure 6: An evolving dataset of *mini*ImageNet few-shot classification tasks where for the first 20k iterations we train on the standard dataset, then switch to a "pencil" effect set of tasks for 10k iterations before finally switching to a "blurred" effect set of tasks until 40k. Responsibilities $\gamma^{(\ell)}$ for each cluster are plotted over time. Note the change in responsibilities as the dataset changes at iterations 20k and 30k.

with G_0 as the base measure over the components of the CRP, also known as the global prior. G_ℓ is the prior on each cluster's parameters, initialized with the global prior, that is updated as more data is observed.

Due to the high-dimensionality of the parameter set of neural networks, we consider a mode estimation procedure based on iterated conditional modes (ICM) (Welling & Kurihara, 2006; Raykov et al., 2016) instead of the expensive process of averaging over Gibbs samples. This leads to a fast point-estimation of the non-parametric mixture parameters where we only track the means of the cluster priors $\boldsymbol{\theta}^{(\ell)}$. Furthermore, the ICM approximation preserves the preferential attachment –the rich gets richer– dynamics of Bayesian nonparametrics. This ensures a sub-linear growth in the number of clusters as more data is observed during episodic training (Raykov et al., 2016).

However, unlike traditional nonparametric algorithms, our model does not refine the cluster assignments of previously observed points by way of multiple expensive passes over the whole data set. In Algorithm 4, we propose a stochastic optimization similar to Algorithm 3 for the mode estimation of $\boldsymbol{z}$ where we incrementally infer the model parameters and add components during episodic training based on noisy but unbiased estimates of the likelihood gradients. A similar sequential approximation for nonparametric mixtures was proposed in (Lin, 2013) with the use of variational Bayes instead of point estimation.

5 FEW-SHOT LEARNING WITH AN EVOLVING DATASET

5.1 SYNTHETIC REGRESSION TASKS

A non-parametric mixture model can adapt to a changing distribution of tasks by increasing its capacity to accomodate the diversity of the tasks. As we will show using synthetic few-shot regression tasks as well as mini-imagenet few-shot classification tasks, our non-parametric meta-learning algorithm is able to acquire new inductive biases as new tasks are introduced without necessarily over-writing existing ones.

Experimental Setup To confirm this attribute of non-parametric models, we considered alternating sinusoidal and polynomial regression tasks input $\boldsymbol{x}$ sampled uniformly from $[-5, 5]$ during the meta-training procedure. For the sinusoid regression, we considered a sine wave with phase sampled uniformly from $[0, \pi]$ and with amplitudes sampled from $a_1 \sim \mathcal{N}(2, 1)$.As for the polynomial regression, $y = \sum_i^d c_i x^i$ where $c_i \sim \mathcal{U}(-5, 5)$.

Hyperparameter choices We use the mean-squared error for each task as the inner loop and meta-level objectives. Our architecture is a feedforward neural network with 2 hidden layers of size 40 and ReLU nonlinearities. We use a meta-batch size of 25 as in the setup for 10-shot sinusoidal regression in Finn et al. (2017).

Our non-parametric algorithm starts with a single cluster ($L_0 = 1$ in Algorithm 4). To account for the noise when spawning clusters based on a single batch, we implemented a threshold that a new cluster's resposibility needs to exceed in order to be added. A similar threshold was proposed in (Lin, 2013) for the same reasons. In these experiments we set the threshold $\epsilon = 0.95T/(L + 1)$, with L the number of non-empty clusters and T the size of the meta-batch. Intuitively, if a cluster has an almost equal share of responsibilities after accounting for the CRP penalty of the other clusters' sizes as described in Algorithm 4, it is spawned. We also used compute the cluster sizes using moving window of size 20 (which is a dataset-dependent hyperparameter) to accomodate their order of magnitude in comparison to the small training losses used for cluster responsibility evaluation. This is also necessary since we do not keep track of the exact prior cluster assignments for our randomly selected task mini-batches nor do we re-assess our assignments at each iteration. Otherwise, non-empty clusters can accumulate an extremely large number of assignments, making cluster spawning impossible after only a few meta-learning episodes.

Results. While we found a variety of combinations of synthetic tasks to demonstrate differentiation across tasks groups, here we investigate the results in Figure 5. For this experiment, we presented even-degree polynomial regression tasks for 700 iterations, followed by odd-degree polynomial regression tasks for 1400 iterations, before switching to sinusoidal regressions.

In Figure 5, we notice a differentiation between the tasks as indicated by the cluster responsibilities. The responsibilities under the first cluster (in blue) decreases to almost zero for the 3rd task (sinusoid) while staying evenly split for the related odd and even polynomial regression tasks. Furthermore, a second cluster (in red) is spawned to account for the difference between odd and even degree polynomials. However, we also notice that the second cluster responsibilities are not zero for the first task, indicating similarities between even and odd polynomial regressions. The same behavior can be seen for the third cluster on the third task. Note that the sinusoidal regression task is the more difficult task which explains the different order of magnitude of the losses.

Overall, we notice our algorithm consistently outperforms MAML in terms of validation loss on the three tasks across the three training phases. More interestingly, our algorithm preserves its performance on old tasks when switching to a new training phase whereas MAML suffers from a clear degradation. While it seems our algorithm doesn't perfectly remember the old inductive biases, it is clear from Figure 5 that it can increase its capacity, when needed, to adjust for new training phases. This allows for better preservation of previously learnt knowledge which is key for life-long learning.

5.2 EVOLVING FEW-SHOT CLASSIFICATION

We apply Algorithm 4 to an evolving variant of the mini-imagenet few-shot classification dataset while using the same architecture as in Finn et al. (2017). In this variant, different artistic filters are applied to the images in the few-shot classification tasks over the meta-training procedure to simulate a changing distribution of classification tasks. More specifically, we first train on the standard mini-imagenet tasks for 20000 iterations then introduce "pencil" effect tasks for 10000 iterations, before finally switching to a blur effect for another 10000 iterations.

Cluster responsibilities during training can be found in Figure 6. To compare more closely to the single-cluster baseline, we restrict the number of clusters to 1 for the first phase of training (the standard mini-imagenet tasks). However, the moment we start introducing new datasets, this restriction is lifted. This allows a better evaluation of the transfer from the first cluster to the new tasks as more datasets are introduced in an online setting.

We notice some differentiation between the tasks that is not as pronounced as what we observed on the toy data. This is potentially due to the fact that all the tasks are dervied from the same core set of images. Accordingly, inductive biases learned on the unfiltered dataset for the first 20k iterations can transfer to the filtered datasets more easily than the different regression tasks we experimented with. This confirms the need for regulating the transfer between tasks, as the taks distribution changes over time, instead of coercing the inductive biases into a single set of hyperparameters that is unrealistically expected to perform well regardless of changing envrionments.

6 RELATED WORK

There is a significant amount of related work, all of which differs from what is presented here, but which is relevant and we discuss.

There are papers on generalizing to novel tasks with few examples using supervised learning on a single task (Lake et al., 2016). In contrast, a human learner can quickly generalize after seeing very few examples from a novel task.Recently, optimization-based meta-learning has seen growing interest, especially in the few-shot learning regime. Early work includes (Andrychowicz et al., 2016; Ravi & Larochelle, 2017) who trained an optimizer by gradient descent and then showed that a few training episodes can be used. More recently (Finn et al., 2017) demonstrated a procedure (MAML) to learn a single set of base parameters which can be quickly adapted to any task loss by gradient descent.

Rosenstein et al. (Rosenstein et al., 2005) demonstrated that negative transfer can worsen generalization performance. This has motivated much work on HB in transfer learning and domain adaptation (*e.g.,* Lawrence & Platt, 2004; Yu et al., 2005; Gao et al., 2008; Daumé III, 2009; Wan et al., 2012). Closest to our proposed approach here, is early work on hierarchical bayesian multi-task learning with neural networks (Heskes, 1998; Bakker & Heskes, 2003; Salakhutdinov et al., 2013; Srivastava & Salakhutdinov, 2013). These approaches are different from ours in that they place a prior, which could be nonparametric as in Salakhutdinov et al. (2013); Srivastava & Salakhutdinov (2013), only on the output layer. Furthemore, none of these approaches were applied to the episodic training setting of meta-learning. (Heskes, 1998; Srivastava & Salakhutdinov, 2013) also propose training a mixture model over the output layer weights using MAP inference. However, this approach does not scale well to all the layers in a network and performing full passes on the dataset for inference of the full set of weights can become computationally intractable.

Incremental or stochastic clustering has been considered in the EM setting in Neal & Hinton (1998), and similarly for minibatch K-means (Sculley, 2010). Online learning of nonparametric mixture models is also a way of doing clustering in the mini-batch setting using sequential variational inference Lin (2013).A key distinction between our work and these approaches is that we leverage the connection between empirical Bayes in this model and gradient-based meta-learning (Grant et al., 2018) to use the MAML (Finn et al., 2017) objective to estimate the likelihood with an implicit prior induced by early stopping. This allows our algorithm to scale and easily integrate with minibatch gradient-based meta-learning instead of alternating multiple backpropagation steps with multiple inference passes over the full dataset (Srivastava & Salakhutdinov, 2013; Bauer et al., 2017).

Our approach is distinct from recent work on gradient-based clustering (Greff et al., 2017) since we adhere to the more challenging setting of episodic meta-learning for both training and testing. Accordingly, only the task support set, for each episode, is used. to compute the cluster responsibilities. This can be a challenging setting for a clustering algorithm as we might only have access to $K < 5$ examples per class as is the case in the standard K-shot classification setting. Another related approach is ensembling. However, an ensemble of independently trained models cannot capture task diversity, and thus cannot modulate information transfer.

7 Conclusion

Meta-learning is a source of inductive bias. Occasionally, the inductive bias is harmful because the domain-agnostic information from one task does not transfer well to another. On the other hand, if tasks are closely related, they can benefit from more inductive transfer. Here, we present an approach that allows a gradient-based meta-learner to modulate the amount of transfer between tasks. We formulate this as probabilistic inference in a mixture model that defines a clustering of task-specific parameters. The probabilistic approach to clustering also permits future extensions to hierarchical clustering or overlapping clustering.To ensure scalability and compatibility with high dimensional function approximators such as deep neural networks, we perform approximate *maximum a posteriori* (MAP) inference both for a finite mixture model as well as a Dirichlet process mixture model (DPMM). As a proxy for assignment likelihood, to any given cluster, we use the empirical training loss of the task when provided with the cluster mean as initialization. This method allows learning of the latent clustering of tasks using gradient descent, and we demonstrate that this approach allows the model complexity to grow along with the evolving complexity of the observed tasks in a few-shot regression problem and a few-shot classification problem.

References

Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, and Nando de Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems (NIPS) 29*, pp. 3981–3989, 2016.

Bart Bakker and Tom Heskes. Task clustering and gating for Bayesian multitask learning. *Journal of Machine Learning Research*, 4(May):83–99, 2003.

Matthias Bauer, Mateo Rojas-Carulla, Jakub Bartłomiej Świkatkowski, Bernhard Schölkopf, and Richard E Turner. Discriminative k-shot learning using probabilistic models. *arXiv preprint arXiv:1706.00326*, 2017.

Jonathan Baxter. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12 (1):149–198, 2000.

Rich Caruana. Multitask learning. In *Learning to learn*, pp. 95–133. Springer, 1998.

Richard Caruana. Multitask learning: A knowledge-based source of inductive bias. In *Proceedings of the 10th International Conference on Machine Learning (ICML)*, 1993.

Anne GE Collins and Michael J Frank. Cognitive control over learning: Creating, clustering, and generalizing task-set structure. *Psychological review*, 120(1):190, 2013.

Hal Daumé III. Bayesian multitask learning with latent hierarchies. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 135–142, 2009.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

Jing Gao, Wei Fan, Jing Jiang, and Jiawei Han. Knowledge transfer via multiple model local structure mapping. In *Proceedings of the 14th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 283–291. ACM, 2008.

Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. *arXiv preprint arXiv:1711.04043*, 2017.

Samuel J Gershman and David M Blei. A tutorial on bayesian nonparametric models. *Journal of Mathematical Psychology*, 56(1):1–12, 2012.

Samuel J Gershman, Kenneth A Norman, and Yael Niv. Discovering latent causes in reinforcement learning. *Current Opinion in Behavioral Sciences*, 5:43–50, 2015.

Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting.

Erin Grant, Chelsea Finn, Sergey Levine, Trevor Darrell, and Thomas Griffiths. Recasting gradient-based meta-learning as hierarchical Bayes. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, 2018.

Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. Neural expectation maximization. In *Advances in Neural Information Processing Systems*, pp. 6694–6704, 2017.

Tom Heskes. Solving a huge number of similar tasks: A combination of multi-task learning and a hierarchical Bayesian approach. In *Proceedings of the 15th International Conference on Machine Learning (ICML)*, pp. 233–241, 1998.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114 (13):3521–3526, 2017.

Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, pp. 1–101, 2016.

Neil D Lawrence and John C Platt. Learning to learn with the informative vector machine. In *Proceedings of the 21st International Conference on Machine Learning (ICML)*, pp. 65, 2004.

Erich L Lehmann and George Casella. *Theory of point estimation*. Springer Science & Business Media, 2006.

Dahua Lin. Online learning of nonparametric mixture models via sequential variational approximation. In *Advances in Neural Information Processing Systems*, pp. 395–403, 2013.

Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, 2018.

Tsendsuren Munkhdalai and Adam Trischler. Metalearning with hebbian fast weights. *arXiv preprint arXiv:1807.05076*, 2018.

Tsendsuren Munkhdalai and Hong Yu. Meta networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

Radford M Neal. Markov chain sampling methods for dirichlet process mixture models. *Journal of computational and graphical statistics*, 9(2):249–265, 2000.

Radford M Neal and Geoffrey E Hinton. A view of the em algorithm that justifies incremental, sparse, and other variants. In *Learning in graphical models*, pp. 355–368. Springer, 1998.

Boris N Oreshkin, Alexandre Lacoste, and Pau Rodriguez. Tadam: Task dependent adaptive metric for improved few-shot learning. *arXiv preprint arXiv:1805.10123*, 2018.

Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L Yuille. Few-shot image recognition by predicting parameters from activations.

Carl Edward Rasmussen. The infinite gaussian mixture model. In *Advances in neural information processing systems*, pp. 554–560, 2000.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, 2017.

Yordan P Raykov, Alexis Boukouvalas, Fahd Baig, and Max A Little. What to do when k-means clustering fails: a simple yet principled alternative algorithm. *PloS one*, 11(9):e0162259, 2016.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, 22(3):400–407, 1951.

Michael T Rosenstein, Zvika Marx, Leslie Pack Kaelbling, and Thomas G Dietterich. To transfer or not to transfer. In *NIPS 2005 workshop on transfer learning*, volume 898, pp. 1–4, 2005.

Adam J Rothman, Elizaveta Levina, and Ji Zhu. Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, 19(4):947–962, 2010.

Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. *arXiv preprint arXiv:1807.05960*, 2018.

Ruslan Salakhutdinov, Joshua B Tenenbaum, and Antonio Torralba. Learning with hierarchical-deep models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1958–1971, 2013.

Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural networks. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pp. 1842–1850, 2016.

Jürgen Schmidhuber. *Evolutionary principles in self-referential learning*. PhD thesis, Institut für Informatik, Technische Universität München, 1987.

Jürgen Schmidhuber. Learning to control fast-weight memories: An alternative to dynamic recurrent networks. *Neural Computation*, 4(1):131–139, 1992.

David Sculley. Web-scale k-means clustering. In *Proceedings of the 19th international conference on World wide web*, pp. 1177–1178. ACM, 2010.

Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems (NIPS) 30*, 2017.

Nitish Srivastava and Ruslan R Salakhutdinov. Discriminative transfer learning with tree-based priors. In *Advances in Neural Information Processing Systems*, pp. 2094–2102, 2013.

D Gowanlock R Tervo, Joshua B Tenenbaum, and Samuel J Gershman. Toward the neural implementation of structure learning. *Current opinion in neurobiology*, 37:99–105, 2016.

Sebastian Thrun. Discovering structure in multiple learning tasks: The tc algorithm.

Eleni Triantafillou, Richard Zemel, and Raquel Urtasun. Few-shot learning through an information retrieval lens. In *Advances in Neural Information Processing Systems (NIPS) 30*, 2017.

Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems (NIPS) 29*, pp. 3630–3638, 2016.

Jing Wan, Zhilin Zhang, Jingwen Yan, Taiyong Li, Bhaskar D Rao, Shiaofen Fang, Sungeun Kim, Shannon L Risacher, Andrew J Saykin, and Li Shen. Sparse Bayesian multi-task learning for predicting cognitive outcomes from neuroimaging measures in Alzheimer's disease. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 940–947. IEEE, 2012.

Max Welling and Kenichi Kurihara. Bayesian k-means as a "maximization-expectation" algorithm. In *Proceedings of the 2006 SIAM international conference on data mining*, pp. 474–478. SIAM, 2006.

Ya Xue, Xuejun Liao, Lawrence Carin, and Balaji Krishnapuram. Multi-task learning for classification with dirichlet process priors. *Journal of Machine Learning Research*, 8(Jan):35–63, 2007.

Kai Yu, Volker Tresp, and Anton Schwaighofer. Learning Gaussian processes from multiple tasks. In *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pp. 1012–1019, 2005.

Yi Zhang and Jeff G Schneider. Learning multiple tasks with a sparse matrix-normal penalty. In *Advances in Neural Information Processing Systems*, pp. 2550–2558, 2010.

Yu Zhang and Dit-Yan Yeung. A regularization approach to learning task relationships in multitask learning. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(3):12, 2014.

A EXTENDED *mini*IMAGENET BENCHMARKING

Model	5-way acc. (%) 1-shot			5-shot		
fine-tuning [a]	28.86	±	0.54	49.79	±	0.79
nearest neighbor [a]	41.08	±	0.70	51.04	±	0.65
matching network FCE (Vinyals et al., 2016) [a]	43.56	±	0.84	55.31	±	0.73
meta-learner LSTM (Ravi & Larochelle, 2017)	43.44	±	0.77	60.60	±	0.71
SNAIL (Mishra et al., 2018) [b]	45.1	±	——	55.2	±	——
prototypical networks (Snell et al., 2017) [c]	46.61	±	0.78	65.77	±	0.70
MAML (Finn et al., 2017)	48.70	±	1.84	63.11	±	0.92
LLAMA (Grant et al., 2018)	49.40	±	1.83	——-	±	——
mAP-DLM (Triantafillou et al., 2017)	49.82	±	0.78	63.70	±	0.67
KNN + GNN embedding (Garcia & Bruna, 2017)	49.44	±	0.28	64.02	±	0.51
GNN (Garcia & Bruna, 2017)	50.33	±	0.36	66.41	±	0.63
fwCNN (Hebb) (Munkhdalai & Trischler, 2018)	50.21	±	0.37	64.75	±	0.49
fwResNet (Hebb) (Munkhdalai & Trischler, 2018)	56.84	±	0.52	71.00	±	0.34
SNAIL (Mishra et al., 2018)	55.71	±	0.99	68.88	±	0.92
Gidaris & Komodakis	56.20	±	0.86	73.00	±	0.64
Bauer et al. (2017)	56.30	±	0.40	73.90	±	0.30
MetaNet (Munkhdalai & Yu, 2017)	57.10	±	0.70	70.04	±	0.63
MAML (Finn et al., 2017) [d e]	58.05	±	0.10	72.41	±	0.20
TADAM (Oreshkin et al., 2018)	58.50	±	0.30	76.70	±	0.30
Qiao et al.	59.60	±	0.41	73.74	±	0.19
LEO (Rusu et al., 2018)	60.06	±	0.05	75.72	±	0.18
Our method (clustering all layers)	50.8	±	—-	——	±	—-

Table 2: One-shot classification accuracy on the *mini*ImageNet test set, with comparison methods ordered by one-shot performance. All results are averaged over 600 test episodes, and we report 95% confidence intervals when available. [a]Results reported by Ravi & Larochelle (2017). [b]We report test accuracy for a comparable architecture. [c]We report test accuracy for models matching train and test "shot" and "way". [d]We report test accuracy for a non-standard architecture with more parameters. [e]Results reported by Rusu et al. (2018).