Online gradient-based mixtures for modulating transfer in meta-learning

Ghassen Jerfel
Department of Electrical and Computer Engineering, Duke University

Erin Grant
Berkeley Artificial Intelligence Research (BAIR) Lab, UC Berkeley

Thomas Griffiths
Departments of Psychology and Computer Science, Princeton University

Katherine Heller
Department of Statistical Science, Duke University

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 miniImageNet 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 few-shot regression tasks.

Keywords: meta-learning, hierarchical Bayes, latent variable model, mixture model, clustering, non-parametric

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 can improve generalization and sample efficiency (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 and Larochelle, 2017; Finn et al., 2017), these meta-learning algorithms make the assumption that tasks are equally related and therefore.

© 2018 G. Jerfel, E. Grant, T. Griffiths & K. Heller.
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 (e.g., Finn et al., 2017; 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). However, the consolidation of all inductive biases into a single set of hyperparameters cannot flexibly account for variability in the task distribution. In contrast, 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 and Frank, 2013, for a review).

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 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.

We extend our latent variable model to the non-parametric setting and leverage stochastic point estimation for scalable inference 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 modelling techniques to high-dimensional parameter spaces such as those of deep neural networks. The non-parametric extension allows the com-

---

**Figure 1:** (a) The standard hierarchical Bayesian model for multi-task learning. A set of global parameters $\theta$ provides an inductive bias for the estimation of task-specific parameters $\phi_j$. (b) In a mixture of hierarchical Bayesian models, the cluster assignment of each task-specific parameter set $\phi_j$ is represented with a latent Categorical variable $z_j$. (c) Allowing an unbounded number of mixture components instantiates a non-parametric model that has the potential to grow with the data in complexity.
Online gradient-based mixtures for modulating transfer in meta-learning

Complexity of a meta-learner 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 parametric counterpart.

2. Gradient-based meta-learning as empirical Bayes

The goal of a meta-learner is to extract task-general knowledge from the experience of solving a number of related tasks. By leveraging this acquired prior knowledge, the meta-learner can 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 gradient-based optimization algorithm with learned parameters (e.g., Ravi and Larochelle, 2017).

Model-agnostic meta-learning (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 hierarchical 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 $\phi_j$ given a batch of task data $x_{j_1:N}$ and a prior over $\phi_j$ that is induced by early stopping of an iterative descent procedure. A few steps of gradient descent on the negative log-likelihood $-\log p(x_{j_1:N} | \phi_j)$, starting from $\phi_j = \theta$ can be then understood as mode estimation of the posterior $p(\phi_j | x_{j_1:N}, \theta)$. The mode estimates $\hat{\phi}_j = \theta + \alpha \nabla_{\theta} \log p(x_{j_1:N} | \theta)$ are then combined to evaluate the marginal likelihood as

$$p\left(\{x_{j_1:M}\}_{j=1}^{J} | \theta\right) = \prod_{j} \int p\left(x_{j_1:M} | \phi_j\right) p\left(\phi_j | \theta\right) d\phi_j \approx \prod_{j} p\left(x_{j_1:M} | \hat{\phi}_j\right),$$

where $x_{j_1:M}$ is another batch of data from each task. A training dataset can then be summarized in an empirical Bayes point estimate of $\theta$ computed by gradient descent in $-\log p(\{x_{j_1:M}\}_{j=1}^{J} | \theta)$ (Lehmann and Casella, 2006). As such, the likelihood of a datapoint $x$ sampled from a new task depends only on the empirical Bayes estimate of $\theta$.

3. Learning latent task structure with gradient-based meta-learning

If the task distribution is heterogeneous, assuming a single parameter initialization $\theta$ 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 $\phi$ to a diversity of tasks. Moreover, explicitly estimating relatedness between tasks has the potential to aid the efficacy of a meta-learning algorithm by modulating both positive and negative transfer (Thrun; Zhang and Schneider, 2010; Rothman et al., 2010; Zhang and Yeung, 2014; Xue et al., 2007).

Nonetheless, defining an appropriate notion of task relatedness is a difficult problem in the high-dimensional parameter or activation space of models such as neural networks.
Algorithm 2: Stochastic expectation maximization via gradient-based meta-learning for clustering

\begin{algorithm}
\begin{algorithmic}
\State Initialize $L \leftarrow L_0$ and $\{\theta^{(1)}, \ldots, \theta^{(L)}\}$ as $\theta^{(0)} \sim G_0$ for $\ell$ in $1, \ldots, L$
\While {not converged}
\State Draw tasks $T_1, \ldots, T_J \sim p(\mathcal{T})$
\For {$j$ in $1, \ldots, J$}
\State Draw task-specific datapoints $x_{j1}, \ldots, x_{jN}, x_{jN+1}, \ldots, x_{jN+M} \sim p_{T_j}(x)$
\State (Draw a parameter initialization for a new cluster $\theta^{(L+1)} \sim G_0$)
\EndFor
\For {$\ell$ in $1, \ldots, L, (L+1)$}
\State Set $\hat{\phi}^{(\ell)}_j \leftarrow \theta^{(\ell)}$ as the initialization for gradient-based fast adaptation
\State Compute mode estimate via $\hat{\phi}^{(\ell)}_j \leftarrow \hat{\phi}^{(\ell)}_j + \alpha \nabla \phi \sum_i \log p(x_{ij} | \hat{\phi}^{(\ell)}_j)$ for $K$ steps
\State Compute assignment of tasks to clusters as $\gamma_j \leftarrow E$-STEP ($\{x_{j1}\}_{i=1}^N, \{\hat{\phi}^{(\ell)}_j\}_{\ell=1}^L$)
\State Update $\theta^{(\ell)} \leftarrow \theta^{(\ell)} + M$-STEP ($\{\{x_{jN+1}\}_{i=1}^M, \hat{\phi}^{(\ell)}_j, \gamma_j \}_{j=1}^J$) for $\ell$ in $1, \ldots, L$
\EndFor
\State (Update global prior $G_0$ with statistics based on $G_0$ and $\{\theta_1, \ldots\}$)
\EndWhile
\Endalgorithmic
\end{algorithm}

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

\begin{align*}
E$-STEP & ($\{x_{ij}\}_{i=1}^N, \{\hat{\phi}^{(\ell)}_j\}_{\ell=1}^L$) \quad \text{return} \quad \tau$-softmax($\sum_i \log p(x_{ij} | \hat{\phi}^{(\ell)}_j)$) \\
M$-STEP & ($\{x_{ij}\}_{i=1}^M, \hat{\phi}^{(\ell)}_j, \gamma_j$) \quad \text{return} \quad \beta \nabla \theta \sum_{i,j} \gamma_{ij} \log p(x_{ij} | \hat{\phi}^{(\ell)}_j)$
\end{align*}

Using the probabilistic interpretation of Section 2, we may deal with the variability in the tasks by assuming that each set of task-specific parameters $\phi_j$ is drawn from a mixture of base distributions each of which is parameterized by a hyperparameter $\theta^{(\ell)}$. Accordingly, we capture task relatedness by estimating the likelihood of assigning each task to a mixture component based only on the task loss itself after a single step of fast adaptation (Finn et al., 2017). The result is a scalable end-to-end meta-learning algorithm that jointly learns task-specific cluster assignments and network parameters. This algorithm, further detailed in the following section, is capable of modulating the transfer of information across tasks to better generalize to heterogeneous or evolving task distributions.

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

Let $z_j$ be the categorical latent variable indicating the cluster assignment of each task-specific parameter $\phi_j$. A direct maximization of the mixture model likelihood is a combinatorial optimization problem that can grow intractable. This intractability is equally problematic for the posterior distribution over the cluster assignment variables $z_j$ and the task-specific parameters $\phi_j$, which are both treated as latent variables in the probabilistic formulation of meta-learning.

A standard approach for estimation in latent variable models such as probabilistic mixtures is to represent the distribution using samples with a sampler. The most widely used is the Gibbs sampler (Neal, 2000; Gershman and Blei, 2012), which draws from the condi-
Online gradient-based mixtures formodulating transfer inmeta-learning

concentration distribution of each latent variable given the others until convergence to the posterior distribution over all the latents. However, in the setting of latent variables defined over high-dimensional parameter spaces such as those of neural network models, a sampling approach such as Gibbs sampling is prohibitively expensive (Neal, 2012; Müller and Insua, 1998).

Instead of maintaining samples to represent the distribution over the latent variables, a scalable approximation involves representing the conditional distribution for each latent variable with either a maximum a posteriori (MAP) value or an expectation. In our meta-learning setting of a mixture of hierarchical Bayesian models, this suggests an augmented expectation maximization (EM) procedure (Dempster et al., 1977) alternating between an E-STEP that computes an expectation of the task-to-cluster assignments, which itself involves the computation of a MAP estimate for the task-specific parameters, and an M-STEP that computes a local maximum of the hyperparameters \( \theta^{(1:L)} \).

To ensure scalability, we use the minibatch variant of stochastic optimization (Robbins and Monro, 1951) to compute both the E-step and M-steps; such 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 and Hinton, 1998). In particular, for each task \( j \) and cluster \( \ell \), we follow the gradients to minimize the negative log-likelihood on the training data points using the cluster parameters \( \theta^{(\ell)} \) as initialization. This allows us to obtain a modal point estimate of the task parameters \( \hat{\phi}^{(\ell)}_j \).

The E-step in Subroutine 3 leverages the connection between gradient-based meta-learning and hierarchical Bayes (HB) (Grant et al., 2018) to employ the task-specific parameters to compute the posterior probability of cluster assignment. Accordingly, based on the likelihood of the same training data points under the model parameterized by \( \hat{\phi}^{(\ell)}_j \), we compute the cluster assignment probabilities as

\[
\gamma^{(\ell)}_j := p(z_j = \ell \mid x_{j1:N}, \theta^{(1:L)}) \propto \int p(x_{j1:N} \mid \phi_j) p(\phi_j \mid \theta^{(\ell)}) d\phi_j \approx p(x_{j1:N} \mid \hat{\phi}^{(\ell)}_j).
\]

The cluster means \( \theta^{(\ell)} \) are then updated by gradient descent on the validation loss in the M-step, given in Subroutine 3; this M-step is similar to the MAML algorithm in Finn et al. (2017). Note that, unlike other recent approaches to probabilistic clustering (e.g., Bauer et al., 2017) we adhere to the episodic meta-learning setup for both training and testing since only the task support set \( x_{j1:N} \) is used to compute both the point estimate \( \hat{\phi}^{(\ell)}_j \) and the cluster responsibilities \( \gamma^{(\ell)}_j \). See Algorithm 2 for the full algorithm, whose high-level structure is shared with the non-parametric variant of our method.

4. Standard few-shot classification with miniImageNet

Clustering task-specific parameters provides a way for a meta-learner to deal with task heterogeneity since each cluster can be associated with a subset of the tasks that would benefit most from inductive transfer. We apply Algorithm 2 with \( L = 5 \) components to the 1-shot 5-way classification few-shot classification benchmark miniImageNet (Vinyals et al., 2016) using the same data split, architecture, and hyperparameter values as in Finn et al. (2017). We additionally use \( \tau = 1 \) for the softmax temperature and the same initialization as Finn et al. (2017) for our global prior \( G_0 \) (which reduces to a fixed initialization in the parametric case).
Online gradient-based mixtures formodulating transfer inmeta-learning

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

- Results reported by Ravi and Larochelle (2017).
- We report test accuracy for a comparable architecture.
- We report test accuracy for models matching train and test “shot” and “way”.

While we do not expect the standard miniImageNet dataset to present much heterogeneity given the uniform sampling assumptions behind its design, we demonstrate in Table 1 that our parametric meta-learner can improve the generalization of gradient-based meta-learning on this task. This result suggests some level of cluster differentiation even on this non-heterogeneous benchmark.

5. Scalable online non-parametric mixtures for evolving tasks

The mixture of meta-learners developed in Section 3 addresses a drawback of meta-learning approaches such as MAML that consolidate task-general information into a single set of hyperparameters. However, the method adds another dimension to model validation in the form of identifying the correct number of mixture components. While this may be resolved by cross-validation if the dataset is static and therefore the number of components can remain fixed, adhering to a fixed number of components throughout training is not appropriate in the nonstationary regime, where the underlying task distribution changes since different types of tasks are presented sequentially. In this regime, it is important to add mixture components sequentially to enable specialization of the component meta-learners to the different types of tasks that constitute the dataset.

To address this, we derive a scalable stochastic estimation procedure to compute the expectation of task-to-cluster assignments (E-STEP) for a growing number of task clusters using an infinite or non-parametric mixture model (Rasmussen, 2000) called the Dirichlet process mixture model (DPMM). This approach 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.

The formulation of the DPMM that is most appropriate for incremental learning is the sequential draws formulation that corresponds to an instantiation of the Chinese restaurant process (CRP) (Rasmussen, 2000). A CRP prior over \( z_j \) allows some probability to be assigned to a new mixture component while the task identities are inferred in a sequential manner, and has therefore been key to recent online and stochastic learning of the
DPMM (Lin, 2013). A draw from a CRP proceeds as follows: For a sequence of tasks, the first task is assigned to the first cluster and the $j$th subsequent task is then assigned to the $\ell$th cluster with probability

$$p\left(z_j = \ell \mid z_{1:j-1}, \zeta\right) = \begin{cases} \frac{n^{(\ell)}}{n^{(\ell)} + \zeta} & \text{for } \ell \leq L \\ \frac{\zeta}{n^{(\ell)} + \zeta} & \text{for } \ell = L + 1, \end{cases}$$

where $L$ indicates the number of non-empty clusters, $n^{(\ell)}$ indicates the number of tasks already occupying a cluster $\ell$, and $\zeta$ is a fixed positive concentration parameter. The prior probability associated with a new mixture component is therefore $p(z_j = L+1 \mid z_{1:j-1}, \zeta)$, and the joint log-likelihood of the data and parameters for a specific task $j$ in our mixture of hierarchical models can be written as

$$\log p(x_{j1:M}, \phi_j \mid \theta^{(1:L)}) = \sum_\ell \gamma^{(\ell)}_j \log p(x_{j1:M}, \phi_j \mid \theta^{(\ell)}) + L \log \zeta + \sum_\ell \log \Gamma(n^{(\ell)}) ,$$

where $\gamma^{(\ell)}_j$ is the responsibility of cluster $\ell$ for task-specific parameters $\phi_j$. We refer the reader to Rasmussen (2000) for more details on the likelihood function associated with the DPMM.

In a similar spirit to Section 3, we develop a stochastic EM procedure for the estimation of the latent task-specific parameters $\phi_{1:j}$ and the meta-level parameters $\theta^{(1:L)}$. While the computation of the mode estimate of the task-specific parameters $\phi_j$ is mostly unchanged from the finite variant, the estimation of the cluster assignment variables $z$ in the E-STEP requires revisiting the Gibbs conditional distributions due to the potential addition of a new cluster at each step. For a DPMM, the conditional distributions for $z$ are

$$p\left(z_j = \ell \mid x_{j1:M}, z_{1:j-1}\right) \propto \begin{cases} n^{(\ell)} \int p(x_{j1:M} \mid \theta) dG_\ell(\theta) & \text{for } \ell \leq L \\ \zeta \int p(x_{j1:M} \mid \theta) dG_0(\theta) & \text{for } \ell = L + 1, \end{cases}$$

with $G_0$ as the base measure over the components of the CRP, also known as the global prior. $G_\ell$ is the prior over each cluster’s parameters, initialized with a draw from a Gaussian centered at $G_0$ with a fixed variance and updated over time.

Using a mode estimate $\hat{\phi}^{(\ell)}_j$ for task-specific $\phi_j$ as drawn from the $\ell$th cluster, the distribution over task-to-cluster assignments $z_j$ can be expressed as

$$\log p\left(z_j = \ell \mid x_{j1:M}, z_{1:j-1}\right) \approx \begin{cases} \log n^{(\ell)} + \log p(x_{j1:M} \mid \hat{\phi}^{(\ell)}_j) + \log p(\hat{\phi}^{(\ell)}_j \mid \theta^{(\ell)}) & \text{for } \ell \leq L \\ \log \zeta + \log p(x_{j1:M} \mid \hat{\phi}^{(\ell)}_j) + \log p(\hat{\phi}^{(\ell)}_j \mid \theta^{(0)}) & \text{for } \ell = L + 1. \end{cases}$$

We thus obtain a set of local objective functions amenable to gradient-based optimization in a similar fashion to the parametric M-STEP of Subroutine 3. We can also omit the prior term $\log p(\hat{\phi}^{(\ell)}_j \mid \theta^{(\ell)})$ as it arises as an implicit prior resulting from truncated gradient descent, as explained in Section 3 of Grant et al. (2018).

One marked difference between the objective of the M-STEP in Subroutine 3 and that in Subroutine 4 is the penalty term of $\log n^{(\ell)}$ or $\log \zeta$ which incentivizes larger clusters
E-STEP (\( \{ \hat{\phi}_j^{(\ell)} \}_{i=1}^N, \{ \phi_j^{(\ell)} \}_{j=1}^{L+1}, \) concentration \( \zeta \), spawn threshold \( \epsilon \))

DPMM log-likelihood, \( \rho_j^{(\ell)} \leftarrow \sum_i \log p(\hat{x}_{ji} | \phi_j^{(\ell)}) + \log n^{(\ell)} \) for all \( \ell \) in 1, . . . , \( L \)

DPMM log-likelihood for new component, \( \rho_j^{(L+1)} \leftarrow \sum_i \log p(\hat{x}_{ji} | \hat{\phi}_j^{(L+1)}) + \log \zeta \)

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

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 \( \gamma_j \)

M-STEP (\( \{ \hat{x}_{ji} \}_{i=1}^M, \{ \phi_j^{(\ell)} \}_{j=1}^L, \gamma_j, \) concentration \( \zeta \))

return \( \beta \nabla_{\theta} [\sum_{j,i} \gamma_j \log p(\hat{x}_{ji} | \hat{\phi}_j^{(\ell)}) + \log n^{(\ell)}] \)

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

in order to deter over-spawning. Accordingly, this approximate inference routine still preserves the preferential attachment (“rich-get-richer”) dynamics of Bayesian nonparametrics (Raykov et al., 2016). Another difference that is not immediate from the Gibbs conditionals is the use of a threshold on the cluster responsibilities to account for noise from stochastic optimization when spawning a cluster on the basis of a single batch. This threshold is necessary for our stochastic mode estimation of Algorithm 4, as it ensures that a new cluster’s responsibility needs to exceed a certain value before being permanently added to the set of components (see the E-STEP in Subroutine 4 for more details). Intuitively, if a cluster has close to an equal share of responsibilities as compared to existing clusters after accounting for the CRP penalty (\( \log n^{(L)} \) vs. \( \log \zeta \)), it is spawned. A sequential approximation for nonparametric mixtures with a similar threshold was proposed in Lin (2013) and Tank et al. (2015), where variational Bayes was used instead of point estimation in a DPMM.

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. Instead, we incrementally infer model parameters and add components during episodic training based on noisy estimates but unbiased of the log-likelihood gradients. A practical reason for the stochastic setting is that it would be extremely expensive to store the assignments for an entire dataset in memory. Moreover, preserving task assignments is potentially harmful due to stale parameter values since the task assignments in our framework are meant to be easily reconstructed on-the-fly using the E-STEP with updated parameters \( \theta \). To the best of our knowledge, no previous work has considered a scalable stochastic point estimation of the parameters in a non-parametric mixture model. Furthermore, we are not aware of prior work applying non-parametric mixture models to high-dimensional function approximators such as modern neural networks with end-to-end learning.

6. “Task-agnostic” few-shot learning with an evolving dataset

A non-parametric mixture model should be able to detect and adapt to a changing distribution of tasks, without any external information to signal the start of a new task type (i.e., in
Figure 5: Mixture component validation log-likelihoods (MSE losses) on an evolving dataset that generates even polynomial regression tasks for 700 iterations, then odd polynomial regression tasks until iteration 1400 at which point it generates sinusoidal regression tasks. We plot the validation negative log-likelihood of the data for each task. Note, for example, the change in loss to the second (red) cluster at 700 iterations when the odd polynomial tasks are introduced.

6.1. Synthetic regression tasks: Improved Generalization

Experimental Setup To demonstrate the unconstrained and adaptive capacity of our non-parametric meta-learning model, we consider alternating sinusoidal, even-degree polynomial, and odd-degree polynomial regression tasks with input $x$ sampled uniformly from $[-5, 5]$ during the meta-training procedure. The sinusoidal regression tasks involve regressing to the output of a sine wave with phase sampled uniformly from $[0, \pi]$ and with amplitudes sampled from $a_1 \sim \mathcal{N}(2, 1)$ in a similar fashion to the synthetic regression tasks in Finn et al. (2017). Similarly, the polynomial regression involves regressing to the output of a polynomial of degree $d$ with output given by $y = \sum_{i=1}^{d} c_i x^i$ for $c_i$ sampled uniformly from $\mathcal{U}(-5, 5)$. For the experiment in Figure 5, we presented even-degree polynomial regression tasks for 700 iterations, followed by odd-degree polynomial regression tasks until 1400 iterations, before switching to sinusoidal regression tasks. We use the mean-squared error loss function for each task as the inner loop error loss function for each task as the inner loop and meta-level objectives.

Hyperparameter choices Our architecture is a feedforward neural network with 2 hidden layers with ReLU nonlinearities, each of size 40. We use a meta-batch size of 25 tasks (both for the inner updates and the meta-gradient updates) 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). 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.

We also compute the cluster sizes using a moving window of size 20 (which is a dataset-dependent hyperparameter) to accommodate the order of magnitude in comparison to the small training losses used for cluster responsibility evaluation. This is 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. In Figure 5, we report the mean-squared error (MSE) validation loss for each task as the 3 different tasks are introduced sequentially and disjoint (only one task for each training phase) to the non-parametric meta-learner. Overall, 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 does not perfectly preserve old inductive biases, we will see in the following experiment (Figure 6) 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 continual learning.

6.2. Synthetic few-shot regression tasks: Task Differentiation

In this experiment, we turn our focus to task differentiation. We investigate the cluster responsibilities on validation data from each of the 3 tasks: quadratic regression, sinusoidal regression, and logistic regression on data from the same input range as specified in Section 6.1.

In Figure 6, we notice a clear 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 and the motivation for a longer period of training. Regression losses can differ greatly in scale based on the function to be regressed, and thus pose difficulties to any optimization-based continual learner; most continual learning datasets, such as Permuted MNIST (Kirkpatrick et al., 2017; Zenke et al., 2017; Nguyen et al., 2017) or Split MNIST (Zenke et al., 2017; Nguyen et al., 2017), consist of classification tasks that make use of a cross-entropy error.
Online gradient-based mixtures for modulating transfer in meta-learning

Figure 6: Each panel represents task-specific per-cluster validation responsibilities on an evolving dataset that generates quadratic regression tasks for 700 iterations, sinusoidal regression until iteration 5700, and logistic regression tasks afterwards. Note the change in responsibilities to the second (red) cluster at 700 iterations when the sinusoidal tasks are introduced. At 5700 iterations, the logistic regression tasks cause a third (brown) cluster to be spawned. At the end of training, both the second (red) and third (brown) clusters are specialized to their corresponding datasets. The lack of specialization of the first (blue) and the initial large responsibility values for the second cluster (red) on the first task is a consequence of the larger magnitude of the sinusoidal regression losses.

7. Related Work

Multi-task learning. 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 and Platt, 2004; Yu et al., 2005; Gao et al., 2008; Daumé III, 2009; Wan et al., 2012). Closest to our proposed approach is early work on hierarchical Bayesian multi-task learning with neural networks (Heskes, 1998; Bakker and Heskes, 2003; Salakhutdinov et al., 2013; Srivastava and Salakhutdinov, 2013). These approaches are different from ours in that they place a prior, which could be non-parametric as in Salakhutdinov et al. (2013) and Srivastava and Salakhutdinov (2013), only on the output layer. Furthermore, none of these approaches were applied to the episodic training setting of meta-learning. Heskes (1998) and Srivastava and 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.

Continual learning. Techniques developed specifically to address the catastrophic forgetting problem in continual learning, such as elastic weight consolidation (EWC) (Kirkpatrick et al., 2017), synaptic intelligence (SI) (Zenke et al., 2017), and variational continual learning (VCL) (Nguyen et al., 2017) require access to an explicit delineation between tasks that acts as a catalyst to grow model size, which we refer to as “task-aware.” In contrast, our
Online gradient-based mixtures formodulating transfer inmeta-learning

A nonparametric algorithm tackles the “task-agnostic” setting of continual learning, where the meta-learner does not receive information about task changes but instead learns to recognize a shift in the task distribution and adapt accordingly.

**Clustering.** Incremental or stochastic clustering has been considered in the EM setting in *Neal and Hinton* (1998) and in the *K*-means setting in *Sculley* (2010). Online learning of nonparametric mixture models is also a way of performing clustering in the incremental 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 a hierarchical model and gradient-based meta-learning (*Grant et al.*, 2018) to use the MAML (*Finn et al.*, 2017) objective as a log posterior surrogate. This allows our algorithm to scale and easily integrate with mini-batch stochastic gradient-based meta-learning instead of alternating a maximization step with an inference pass over the full dataset (*Srivastava and Salakhutdinov*, 2013; *Bauer et al.*, 2017). Our approach is distinct from recent work on gradient-based clustering (*Greff et al.*, 2017) since we employ the episodic batching of *Vinyals et al.* (2016) for both training and testing. This can be a challenging setting for a clustering algorithm, as the assignments need to be computed using, for example, $K = 1$ examples per class in the 1-shot setting.

### 8. Conclusion

Meta-learning is a source of learned inductive bias. Occasionally, the inductive bias is harmful because the experience gained from solving one task does not transfer well to another. On the other hand, if tasks are closely related, they can benefit from a greater amount of inductive transfer. Here, we present an approach that allows a gradient-based meta-learner to explicitly modulate the amount of transfer between tasks, as well as to adapt its parameter dimensionality when the underlying task distribution evolves. We formulate this as probabilistic inference in a mixture model that defines a clustering of task-specific parameters. To ensure scalability, we make use of the recent connection between gradient-based meta-learning and hierarchical Bayes (*Grant et al.*, 2018) to perform approximate maximum a posteriori (MAP) inference in both a finite and an infinite mixture model. This approach admits non-conjugate likelihoods parameterized with a black-box function approximator such as a deep neural network, and therefore learns to identify underlying genres of tasks using the standard gradient descent learning rule. 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.
References

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 Światakowski, 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*, pages 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)*, pages 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)*, pages 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)*, pages 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.

Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. URL https://arxiv.org/abs/1804.09458.

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*, pages 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)*, pages 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.

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)*, page 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 (NIPS)*, pages 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.

Peter Müller and David Rios Insua. Issues in bayesian analysis of neural network models. *Neural Computation*, 10(3):749–770, 1998.

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. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

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*, pages 355–368. Springer, 1998.

Cuong V Nguyen, Yingzhen Li, Thang D Bui, and Richard E Turner. Variational continual learning. *arXiv preprint arXiv:1710.10628*, 2017.

Boris N Oreshkin, Alexandre Lacoste, and Pau Rodriguez. Tadam: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2018. URL https://arxiv.org/abs/1805.10123.
Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan 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, pages 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, pages 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), pages 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, pages 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*, pages 2094–2102, 2013.

Alex Tank, Nicholas Foti, and Emily Fox. Streaming variational inference for bayesian nonparametric mixture models. In *Artificial Intelligence and Statistics*, pages 968–976, 2015.

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*, pages 3630–3638, 2016.

Jing Wan, Zhilin Zhang, Jingwen Yan, Taiyong Li, Bhaskar D Rao, Shiaoofen 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)*, pages 940–947. IEEE, 2012.

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)*, pages 1012–1019, 2005.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. *arXiv preprint arXiv:1703.04200*, 2017.

Yi Zhang and Jeff G Schneider. Learning multiple tasks with a sparse matrix-normal penalty. In *Advances in Neural Information Processing Systems*, pages 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.
## Appendix A. Extended miniImageNet benchmarking

| Model                                      | 5-way acc. (%) | 1-shot | 5-shot |
|--------------------------------------------|----------------|--------|--------|
| fine-tuning                               | 28.86 ± 0.54   | 49.79  | ± 0.79 |
| nearest neighbor                          | 41.08 ± 0.70   | 51.04  | ± 0.65 |
| matching network FCE (Vinyals et al., 2016) | 43.56 ± 0.84   | 55.31  | ± 0.73 |
| meta-learner LSTM (Ravi and Larochelle, 2017) | 43.44 ± 0.77   | 60.60  | ± 0.71 |
| SNAIL (Mishra et al., 2018)                | 45.1 ± -----   | 55.2   | ± ----- |
| matching network FCE (Snell et al., 2017) | 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 and Bruna, 2017) | 49.44 ± 0.28   | 64.02  | ± 0.51 |
| GNN (Garcia and Bruna, 2017)              | 50.33 ± 0.36   | 66.41  | ± 0.63 |
| fwCNN (Hebb) (Munkhdalai and Trischler, 2018) | 50.21 ± 0.37   | 64.75  | ± 0.49 |
| Our method (clustering all layers)        | 50.8 ± 1.70    | -----  | ± ----- |

### Table 2: One-shot classification accuracy on the miniImageNet test set, with comparison methods ordered by one-shot performance. The second section in the chart represents methods that use significantly more powerful base architectures than Vinyals et al. (2016). All results are averaged over 600 test episodes, and we report 95% confidence intervals when available. 

- Results reported by Ravi and Larochelle (2017).
- We report test accuracy for a comparable architecture.
- We report test accuracy for models matching train and test “shot” and “way”.
- We report test accuracy for a non-standard architecture with more parameters.
- Results reported by Rusu et al. (2018).