LEARNING TO BALANCE: BAYESIAN META-LEARNING FOR IMBALANCED AND OUT-OF-DISTRIBUTION TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

While tasks could come with varying the number of instances and classes in realistic settings, the existing meta-learning approaches for few-shot classification assume that the number of instances per task and class is fixed. Due to such restriction, they learn to *equally* utilize the meta-knowledge across all the tasks, even when the number of instances per task and class largely varies. Moreover, they do not consider distributional difference in unseen tasks, on which the meta-knowledge may have less usefulness depending on the task relatedness. To overcome these limitations, we propose a novel meta-learning model that *adaptively balances* the effect of the meta-learning and task-specific learning within each task. Through the learning of the balancing variables, we can decide whether to obtain a solution by relying on the meta-knowledge or task-specific learning. We formulate this objective into a Bayesian inference framework and tackle it using variational inference. We validate our Bayesian Task-Adaptive Meta-Learning (Bayesian TAML) on two realistic task- and class-imbalanced datasets, on which it significantly outperforms existing meta-learning approaches. Further ablation study confirms the effectiveness of each balancing component and the Bayesian learning framework.

1 INTRODUCTION

Despite the success of deep learning in many real-world tasks such as visual recognition and machine translation, such good performances are achievable at the availability of large training data, and many fail to generalize well in small data regimes. To overcome this limitation of conventional deep learning, recently, researchers have explored meta-learning (Schmidhuber, 1987; Thrun & Pratt, 1998) approaches, whose goal is to learn a model that generalizes well over distribution of tasks, rather than instances from a single task, in order to utilize the obtained meta-knowledge across tasks to compensate for the lack of training data for each task.

However, so far, most existing meta-learning approaches (Santoro et al., 2016; Vinyals et al., 2016; Snell et al., 2017; Ravi & Larochelle, 2017; Finn et al., 2017; Li et al., 2017) have only targeted an artificial scenario where all tasks participating in the multi-class classification problem have equal number of training instances per class. Yet, this is a highly restrictive setting, as in real-world scenarios, tasks that arrive at the model may have different training instances (task imbalance), and within each task, the number of training instances per class may largely vary (class imbalance). Moreover, the new task may come from a distribution that is different from the task distribution the model has been trained on (out-of-distribution task) (See (a) of Figure 1).

Under such a realistic setting, the meta-knowledge may have a varying degree of utility to each task. Tasks with small number of training data, or close to the tasks trained in meta-training step may want to rely mostly on meta-knowledge obtained over other tasks, whereas tasks that are out-of-distribution or come with more number of training data may obtain better solutions when trained in a task-specific manner. Furthermore, for multi-class classification, we may want to treat the learning for each class differently to handle class imbalance. Thus, to optimally leverage meta-learning under various imbalances, it would be beneficial for the model to task- and class-adaptively decide how much to use from the meta-learner, and how much to learn specifically for each task and class.

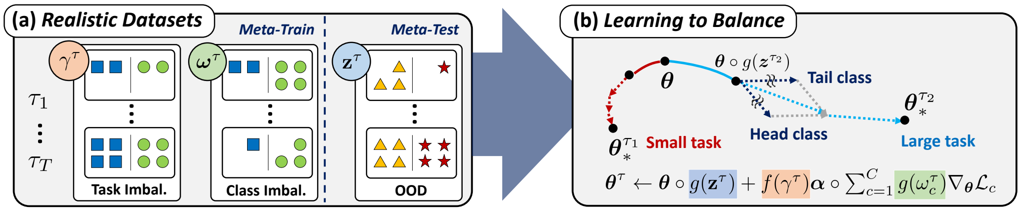


Figure 1: **Concept.** (a) To handle task imbalance (Task Imbal.), class imbalance (Class Imbal.) and out-of-distribution tasks (OOD) for each task τ, we introduce task-specific balancing variables γ^τ, $\boldsymbol{\omega}^\tau$ and $\mathbf{z}^\tau$, respectively. (b) With those variables, we learn to balance between the meta-knowledge $\boldsymbol{\theta}$ and task-specific update to handle imbalances and distributional discrepancies.

To this end, we propose a novel Bayesian meta-learning framework, which we refer to as Bayesian Task-Adaptive Meta-Learning (Bayesian TAML), that learns variables to adaptively balance the effect of meta- and task-specific learning. Specifically, we first obtain set-representations for each task, which are learned to convey useful statistics about the task or class distribution, such as mean, variance, tailedness (kurtosis), and skewness, and then learn the distribution of three balancing variables as the function of the set: 1) *task-dependent learning rate decay*, which decides how far away to deviate from the meta-knowledge, when performing task-specific learning. Tasks with higher shots could benefit from taking gradient steps afar, while tasks with few shots may need to stay close to the initial parameter. 2) *class-dependent learning rate*, which decides how much information to use from each class, to automatically handle class imbalance where the number of instances per class can largely vary. 3) *task-dependent attention mask*, which modifies the shared parameter for each task by learning a set-dependent attention mask to it, such that the task can decide how much and what to use from the initial shared parameter and what to ignore based on its set representation. This is especially useful when handling out-of-distribution task, which may need to ignore some of the meta-knowledge.

We validate our model on Omniglot and mini-ImageNet dataset, as well as a new dataset that consists of heterogeneous datasets, under a scenario where every class in each episode can have *any* number of shots, that leads to task and class imbalance, and where the dataset at meta-test time is different from that of meta-training time. The experimental results show that our Bayesian TAML obtains significantly improves over the existing approaches under these realistic scenarios. Further analysis of each component reveals that the improvement is due to the effectiveness of the balancing terms for handling task and class imbalance, and out-of-distribution tasks.

To summarize, our contribution in this work is threefold:

- We consider a novel problem of meta-learning under a realistic task distribution, where the number of instances across classes and tasks could largely vary, or the unseen task at the meta-test time is largely different from the seen tasks.
- For effective meta-learning with such imbalances, we propose a Bayesian task-adaptive meta-learning (Bayesian TAML) framework that can adaptively adjust the effect of the meta-learner and the task-specific learner, differently for each task and class.
- We validate our model on realistic imbalanced few-shot classification tasks with a varying number of shots per task and class and show that it significantly outperforms existing meta-learning models.

2 Related Work

Meta-learning Meta-learning (Schmidhuber, 1987; Thrun & Pratt, 1998) is an approach to learn a model to generalize over a distribution of task. The approaches in general can be categorized into either memory-based, metric-based, and optimization-based methods. A memory-based approach (Santoro et al., 2016) learns to store correct instance and label into the same memory slot and retrieve it later, in a task-generic manner. Metric-based approaches learn a shared metric space that defines the distance between the instance and the class prototype, such that the instances are closer to their correct prototypes than to others (Vinyals et al., 2016; Snell et al., 2017). As for optimization-based meta-learning, MAML Finn et al. (2017) learns a shared initialization parameter that is optimal for any tasks within few gradient steps from the initial parameter. Meta-SGD (Li et al., 2017) improves upon MAML by proposing to learn the learning rate differently for each parameter. For effective learning of a meta-learner, meta-learning approaches adopt the episodic training strategy (Vinyals et al., 2016) which trains and evaluates a model over a large number of tasks, which are called meta-training and meta-test phase, respectively. However, existing approaches only consider an artificial scenario where each episode considers the classification of classes with exactly the same number of training instances, both within each episode and across episodes. On the other hand, we consider a more challenging scenario where number of shots per class and task could vary at each episode, and that the task given at the meta-test time could be an out-of-distribution task.

Task-adaptive meta-learning The goal of learning a single meta-learner that works well for all tasks may be overly ambitious and leads to suboptimal performances for each task. Thus recent approaches adopt task-adaptively modified meta-learning models. Oreshkin et al. (2018) proposed to learn the temperature scaling parameter to work with the optimal similarity metric. Qiao et al. (2018) also suggested a model that generates task-specific parameters for the network layers, but it only trains with many-shot classes, and implicitly expects generalization to few-shot cases. Rusu et al. (2018) proposed a network type task-specific parameter producer, and Lee & Choi (2018) proposed to differentiate the network weights into task-shared and task-specific weights. Our model also aims to obtain task-specific parameter for each task, but is rather focused on learning how to balance between the meta-learning and task-/class-specific learning. To our knowledge, none of the existing approaches explicitly tackle this balancing problem since they only consider few-shot learning with the fixed number of instances for each class and task.

Probabilistic meta-learning Recently, a probabilistic version of MAML has been proposed (Finn et al., 2018), where they interpret a task-specific gradient update as a posterior inference process under variational inference framework. Kim et al. (2018) proposed Bayesian MAML with a similar motivation but with a stein variational inference framework and chaser loss. Gordon et al. (2018) proposed a probabilistic meta-learning framework where the paramter for a novel task is rapidly estimated under decision theoretic framework, given a set representation of a task. The motivation behind these works is to represent the inherent uncertainty in few-shot classification tasks. Our model also uses Bayesian modeling, but it focuses on leveraging the uncertainties of the meta-learner and the gradient-direction in order to balance between meta- and task- or class-specific learning.

3 Learning to Balance

We first introduce notations and briefly recap the model-agnostic meta-learning (MAML) by Finn et al. (2017). Suppose a task distribution $p(\tau)$ that randomly generates task τ consisting of a training set $\mathcal{D}^\tau = \{\mathbf{X}^\tau, \mathbf{Y}^\tau\}$ and a test set $\tilde{\mathcal{D}}^\tau = \{\tilde{\mathbf{X}}^\tau, \tilde{\mathbf{Y}}^\tau\}$. Then, the goal of MAML is to meta-learn the initial model parameter $\boldsymbol{\theta}$ to generalize over the task distribution $p(\tau)$, such that we can easily obtain the task-specific predictor $\boldsymbol{\theta}^\tau$ in a single (or a few) gradient step from the initial $\boldsymbol{\theta}$. Toward this goal, MAML optimizes the following gradient-based meta-learning objective:

$$\min_{\boldsymbol{\theta}} \sum_{\tau \sim p(\tau)} \mathcal{L}(\boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; \mathcal{D}^\tau); \tilde{\mathcal{D}}^\tau) \tag{1}$$

where α denotes stepsize and $\mathcal{L}$ denotes empirical loss such as negative log-likelihood of observations. Note that by meta-learning the initial point $\boldsymbol{\theta}$, the task-specific predictor $\boldsymbol{\theta}^\tau = \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; \mathcal{D}^\tau)$ can minimize the test loss $\mathcal{L}(\cdot; \tilde{\mathcal{D}}^\tau)$ even with $\mathcal{D}^\tau$ which only contains few samples. We can easily extend the Eq. (1), such that we obtain $\boldsymbol{\theta}^\tau$ with more than one inner-gradient steps from the initial $\boldsymbol{\theta}$.

However, the existing MAML framework has the following limitations that prevent the model from efficiently solving real-world problems involving task/class imbalance and out-of-distribution tasks.

1. **Task imbalance.** MAML has a fixed number of inner-gradient steps and stepsize α across all tasks, which prevents the model from adaptively deciding how much to use from the meta-knowledge depending on the number of the training examples per task.
2. **Class imbalance.** The model does not provide any framework to handle class imbalance within each task. Therefore, classes with large number of training instances (head classes) may dominate the task-specific learning during the inner-gradient steps, yielding low performance on classes with fewer shots (tail classes).
3. **Out-of-distribution tasks.** The model assumes that the meta-knowledge will be equally useful for the unseen tasks, but for unseen tasks that are out-of-distribution, the meta-knowledge may be less useful.

3.1 Task-Adaptive Meta-Learning (TAML)

As shown in Figure 1 for the concepts, we introduce three balancing variables $\gamma^\tau, \boldsymbol{\omega}^\tau, \mathbf{z}^\tau$ to tackle each problem mentioned above. How to compute these variables will be described in Section 4. In order to learn with realistic scenarios, we assume that the task distribution $p(\tau)$ samples some fixed number of C classes ("way"), and then sample uniform-random number of instances for each class ("shots"), thereby simulating both task and class imbalance at the same time.

Tackling task imbalance. To control whether to stay close to the initial parameter or deviate far from it, we introduce a clipping function $f(\cdot) = \max(0, \min(\cdot, 1))$ and a task-dependent learning-rate decaying factor $f(\gamma^\tau)$, such that the learning rate exponentially decays as $\alpha \rightarrow f(\gamma^\tau)\alpha \rightarrow \cdots \rightarrow f(\gamma^\tau)^{K-1}\alpha$, for step $k = 1, \ldots, K$. We expect $f(\gamma^\tau)$ to be large for large tasks, such that they rely more on task-specific updates, while small tasks use small $f(\gamma^\tau)$ to benefit from the meta-knowledge.

Tackling class imbalance. To handle class imbalance, we vary the learning rate of class-specific gradient update for each task-specific gradient update step. Specifically, for class $c = 1, \ldots, C$, we introduce a non-negative activation function $g(\cdot) = \text{SoftPlus}(\cdot)$ and a set of class-specific non-negative scalars $g(\omega_1^\tau), \ldots, g(\omega_C^\tau)$ multiplied to each of the class-specific gradients $\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}; \mathcal{D}_1^\tau), \ldots, \nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}; \mathcal{D}_C^\tau)$, where $\mathcal{D}_c^\tau$ is the set of instances and labels for class c. We expect $g(\omega_c^\tau)$ to be large for tail-classes to consider them more in task-specific gradient updates.

Tackling out-of-distribution tasks. Lastly, we introduce an additional task-dependent variable $\mathbf{z}^\tau$ with the non-negative activation function $g(\cdot)$ which weights the initial parameter $\boldsymbol{\theta}$ according to the usefulness for each task. We expect the variable $g(\mathbf{z}^\tau)$ to heavily emphasize the meta-knowledge $\boldsymbol{\theta}$ when $\mathcal{D}^\tau$ is similar to the trained dataset, and use less of it when $\mathcal{D}^\tau$ is unfamilar. This behavior can be implemented with Bayesian modeling on the latent $\mathbf{z}^\tau$, which we introduce in the next subsection.

A unified framework. Finally, we assemble all these components together into a single unified framework. The update rule for the task-specific $\boldsymbol{\theta}^\tau$ is recursively defined as follows:

$$\boldsymbol{\theta}_0 = \boldsymbol{\theta} \circ g(\mathbf{z}^\tau), \tag{2}$$

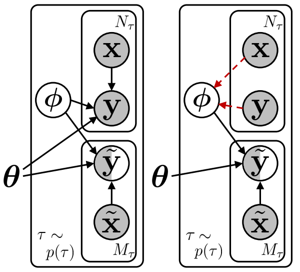
$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + f(\gamma^\tau)^{k-1}\boldsymbol{\alpha} \circ \sum_{c=1}^{C} g(\omega_c^\tau)\nabla_{\boldsymbol{\theta}_{k-1}}\mathcal{L}(\boldsymbol{\theta}_{k-1}; \mathcal{D}_c^\tau) \quad \text{for } k = 1, \ldots, K \tag{3}$$

where the last step $\boldsymbol{\theta}_K$ corresponds to the task-specific predictor $\boldsymbol{\theta}^\tau$ and $\boldsymbol{\alpha}$ is a multi-dimensional global learning rate vector that is learned such as Li et al. (2017).

3.2 Bayesian Task-Adaptive Meta-Learning

As previously mentioned, we need a Bayesian framework for modeling $\mathbf{z}^\tau$, since it needs a prior in order to prevent the posterior of $\mathbf{z}^\tau$ from overly utilizing the meta-knowledge $\boldsymbol{\theta}$ when the task is out-of-distribution. Moreover, for the learning of balancing variables γ^τ and $\boldsymbol{\omega}^\tau$, Bayesian modeling improve the quality of the inference on them, which we empirically verified through extensive experiments. We allow the three variables to share the same inference network pipeline to minimize the computational cost, and thereby effectively amortize the inference rule across variables as well.

Firstly, define $\mathbf{X}^\tau = \{\mathbf{x}_n^\tau\}_{n=1}^{N_\tau}$ and $\mathbf{Y}^\tau = \{\mathbf{y}_n^\tau\}_{n=1}^{N_\tau}$ for training, and $\tilde{\mathbf{X}}^\tau = \{\tilde{\mathbf{x}}_m^\tau\}_{m=1}^{M_\tau}$ and $\tilde{\mathbf{Y}}^\tau = \{\tilde{\mathbf{y}}_m^\tau\}_{m=1}^{M_\tau}$ for test. Let $\boldsymbol{\phi}^\tau$ denote the collection of three latent variables, γ^τ, $\boldsymbol{\omega}^\tau$ and $\mathbf{z}^\tau$ for uncluttered notation. Then, the generative process is as follows for each task τ (See Figure 2):



(a) (b)

Figure 2: Graphical model. (a) Generative process. (b) Inference.

$$p(\mathbf{Y}^\tau, \tilde{\mathbf{Y}}^\tau, \boldsymbol{\phi}^\tau | \mathbf{X}^\tau, \tilde{\mathbf{X}}^\tau; \boldsymbol{\theta}) = p(\boldsymbol{\phi}^\tau) \prod_{n=1}^{N_\tau} p(\mathbf{y}_n^\tau | \mathbf{x}_n^\tau, \boldsymbol{\phi}^\tau; \boldsymbol{\theta}) \prod_{m=1}^{M_\tau} p(\tilde{\mathbf{y}}_m^\tau | \tilde{\mathbf{x}}_m^\tau, \boldsymbol{\phi}^\tau; \boldsymbol{\theta}) \tag{4}$$

for the complete data likelihood. Note that the deterministic $\boldsymbol{\theta}$ is shared across all the tasks.

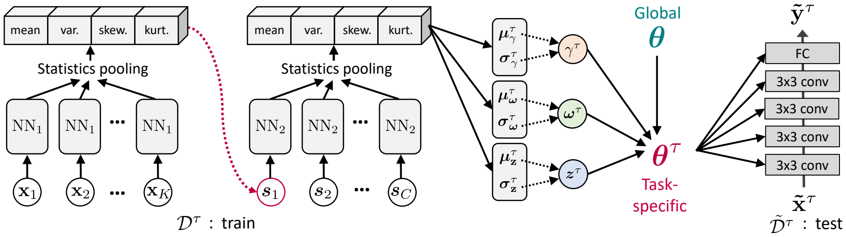


Figure 3: **Inference Network.** The proposed Set-of-Sets encoder captures the instance-wise and class-wise statistics hierarchically, from which we infer three different balancing variables.

4 Variational Inference

The goal of learning for each task τ is to maximize the log-likelihood of the joint dataset $\tilde{\mathcal{D}}^\tau$ and $\mathcal{D}^\tau$: $\log p(\tilde{\mathbf{Y}}^\tau, \mathbf{Y}^\tau | \tilde{\mathbf{X}}^\tau, \mathbf{X}^\tau; \boldsymbol{\theta})$. However, solving it involves the true posterior $p(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau, \tilde{\mathcal{D}}^\tau)$, which is intractable. Thus, we resort to amortized variational inference with a tractable form of approximate posterior $q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau, \tilde{\mathcal{D}}^\tau; \boldsymbol{\psi})$ parameterized by $\boldsymbol{\psi}$. Further, similarly to Ravi & Beatson (2018), we drop the dependency on the test dataset $\tilde{\mathcal{D}}^\tau$ for the approximate posterior, in order to make the two different pipelines consistent; one for meta-training where we observe the whole test dataset, and the other for meta-testing where the test labels are unknown. The form of our approximate posterior is now $q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau; \boldsymbol{\psi})$. It greatly simplifies the inference framework, while ensuring that the following objective is still a valid lower bound of the log evidence. Also, considering that performing the inner-gradient steps with the training dataset $\mathcal{D}^\tau$ automatically maximizes the training log-likelihood in MAML framework, we slightly modify the objective so that the expected loss term only involves the test examples. The resultant form of the lower bound that suits for our meta-learning purpose is as follows:

$$L^\tau_{\boldsymbol{\theta},\boldsymbol{\psi}} = \sum_{m=1}^{M_\tau} \mathbb{E}_{q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau; \boldsymbol{\psi})} \Big[\log p(\tilde{\mathbf{y}}^\tau_m | \tilde{\mathbf{x}}^\tau_m, \boldsymbol{\phi}^\tau; \boldsymbol{\theta}) \Big] - \mathrm{KL}[q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau; \boldsymbol{\psi}) \| p(\boldsymbol{\phi}^\tau)]. \tag{5}$$

We assume $q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau; \boldsymbol{\psi})$ fully factorizes for each variable and also for each dimension as well:

$$q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau; \boldsymbol{\psi}) = q(\gamma^\tau | \mathcal{D}^\tau; \boldsymbol{\psi}) \prod_c q(\omega^\tau_c | \mathcal{D}^\tau; \boldsymbol{\psi}) \prod_i q(z^\tau_i | \mathcal{D}^\tau; \boldsymbol{\psi}) \tag{6}$$

where we assume that each single dimension of $q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau; \boldsymbol{\psi})$ follows univariate gaussian having trainable mean and variance. We also let each dimension of prior $p(\boldsymbol{\phi}^\tau)$ factorize into $\mathcal{N}(0, 1)$. The KL-divergence between two univariate gaussians has a simple closed form (Kingma & Welling, 2014), thereby we obtain the low-variance estimator for the lower bound $L^\tau_{\boldsymbol{\theta},\boldsymbol{\psi}}$.

The final form of the meta-training minimization objective with Monte-Carlo approximation for the expection in (5) is as follows:

$$\min_{\boldsymbol{\theta},\boldsymbol{\psi}} \frac{1}{T} \sum_{\tau \sim p(\tau)} \frac{1}{M_\tau} \Big\{ \sum_{m=1}^{M_\tau} \frac{1}{S} \sum_{s=1}^{S} - \log p(\tilde{\mathbf{y}}^\tau_m | \tilde{\mathbf{x}}^\tau_m, \boldsymbol{\phi}^\tau_s; \boldsymbol{\theta}) + \mathrm{KL}[q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau; \boldsymbol{\psi}) \| p(\boldsymbol{\phi}^\tau)] \Big\}. \tag{7}$$

where $\boldsymbol{\phi}^\tau_s \sim q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau; \boldsymbol{\psi})$, and T is the number of tasks. We implicitly assume the reparameterization trick for $\boldsymbol{\phi}^\tau$ to obtain stable and unbiased gradient estimate w.r.t. $\boldsymbol{\psi}$ (Kingma & Welling, 2014). We set the number of MC samples to $S = 1$ for meta-training for computational efficiency. When meta-testing, we can set $S = 10$ or naively approximate the expectation by taking the expectation inside: $\mathbb{E}_q[p(\tilde{\mathbf{y}}^\tau_m | \tilde{\mathbf{x}}^\tau_m, \boldsymbol{\phi}^\tau; \boldsymbol{\theta})] \approx p(\tilde{\mathbf{y}}^\tau_m | \tilde{\mathbf{x}}^\tau_m, \mathbb{E}_q[\boldsymbol{\phi}^\tau]; \boldsymbol{\theta})$, which works well in practice.

4.1 Dataset Encoding

The main challenge in modeling our variational distribution $q(\boldsymbol{\phi}^\tau | \mathcal{D}^\tau; \boldsymbol{\psi})$ is how to refine the training dataset $\mathcal{D}^\tau$ into informative representation capturing the dataset as a distribution, which is not trivial. This inference network should capture all the necessary statistical information in the dataset $\mathcal{D}^\tau$ to solve both imblanace and out-of-distribution problems. DeepSets (Zaheer et al., 2017) is frequently used as a practical set-encoder, where each instance in the set is transformed by the shared nonlinearity, and then summed together to generate a single vector summarizing the set. However, for the classification dataset $\mathcal{D}^\tau$ which is the *set of (class) sets*, we cannot use DeepSets directly as it will completely ignore the label information. Therefore, we need to stack the structure of DeepSets twice according to the hierarchical set of sets structure of classification dataset.

However, there exists additional limitation of DeepSets with sum-pooling when describing the distribution. Suppose that we have a set containing a replication of single instance. Then, its representation will change based on the number of replications, although distribution-wise all sets should be the same. Mean-pooling may alleviate the problem; however, it does not recognize the number of elements in the set, which is a critical limitation in encoding imbalance. To overcome the limitations of the two pooling methods, we propose to use higher-order statistics in addition to the sample mean, namely element-wise sample variance, skewness and kurtosis. For instance, the sample variance could capture task imbalance and skewness will capture class imbalance (imbalance in the number of instances per class). Based on this intuition, we propose the following encoder network $\mathrm{StatisticsPooling}(\cdot)$ that generates the concatenation of those statistics (See Figure 3):

$$\mathbf{v}^\tau = \mathrm{StatisticsPooling}\left(\{\mathrm{NN}_2\left(\mathbf{s}_c\right)\}_{c=1}^{C}\right), \quad \mathbf{s}_c = \mathrm{StatisticsPooling}\left(\{\mathrm{NN}_1(\mathbf{x})\}_{\mathbf{x}\in\mathbf{X}_c^\tau}\right)$$

for classes $c = 1, \dots, C$, and $\mathbf{X}_c^\tau$ is the collection of class c examples in task t. NN_1 and NN_2 are some appropriate neural networks parameterized by $\boldsymbol{\psi}$. The vector $\mathbf{v}^\tau$ finally summarizes the whole classification dataset $\mathcal{D}^\tau$ and our balancing variables γ^τ, $\boldsymbol{\omega}^\tau$ and $\mathbf{z}^\tau$ are generated from it with an additional affine transformation. See Appendix B for the justification.

5 Experiments

We validate our method in imbalanced scenarios, where each task, or every class within a task can have different shots, and the tasks at the evaluation time could come from a different task distribution from the seen task distribution. Following Finn et al. (2017), we use 4-block convolutional neural networks with 64 channels for each layer for Omniglot and MNIST. We reduce the convolution filter size of the 4-block CNN network into 32 for other datasets.

5.1 Any-shot Classification

Table 1: **Any-shot classification performance.** All reported results are averaged performances over 1000 randomly selected episodes for Omniglot and MNIST and 600 randomly selected episodes for tiered-ImageNet and mini-ImageNet with standard errors for 95% confidence intervals.

Meta-train	Omniglot		tiered-ImageNet	
Meta-test	Omniglot	MNIST	tiered-ImageNet	mini-ImageNet
Prototypical Net (Snell et al., 2017)	**98.37 ± 0.05**	82.16 ± 0.19	65.25 ± 0.74	49.67 ± 0.38
MAML (Finn et al., 2017)	93.38 ± 0.28	79.63 ± 0.33	66.70 ± 0.40	49.61 ± 0.36
Meta-SGD (Li et al., 2017)	94.27 ± 0.24	81.00 ± 0.31	68.16 ± 0.92	56.57 ± 0.37
MT-NET (Lee & Choi, 2018)	95.41 ± 0.36	81.89 ± 0.54	69.84 ± 0.79	55.36 ± 0.38
ABML (Ravi & Beatson, 2018)	95.72 ± 0.20	81.48 ± 0.44	57.32 ± 0.61	53.02 ± 0.48
Bayesian TAML	96.29 ± 0.31	**84.39 ± 0.48**	**71.42 ± 1.00**	**58.37 ± 0.49**

Imbalanced Omniglot. This dataset (Lake et al., 2015) consists of 1623 hand-written character classes, with 20 training instances per class. We consider 10-way classification problems, where we have 5 instances per each class for test (queries). To generate imbalanced tasks, we modify the existing strategy for episode generation (Michalski et al., 1983), such that we randomly set the number of training instances to be sampled within the range of 1 to 15. We train our model and all baseline models with 5 inner-gradient steps on Omniglot, and evaluate on both Omniglot and MNIST, where the latter is used to evaluate the performance on out-of-distribution task.

Imbalanced tiered-ImageNet. This is sub-sampled ImageNet dataset including 608 classes (Ren et al., 2018). As similarly with the Imbalanced Omniglot dataset, we consider 5-way classification problems, while randomly setting the number of training instances per class within the range of 1 to 50, and use 15 instances per class for test (queries). We train all models with 5 inner-gradient steps on the tiered-ImageNet dataset, and evaluate the model on the test split of tiered-ImageNet and mini-ImageNet, where the latter is used to evaluate on out-of-distribution task. See the Appendix A for more details of the experimental setup.

Analysis. Table 1 shows the any-shot classification accuracies of various meta-learning models. Bayesian TAML outperforms all baseline models in all settings, including MAML variants that are directly comparable with Bayesian TAML, except for the Omniglot In-distribution experiment where Prototypical Network works the best. However, Prototypical Network shows poor performance on out-of-distribution tasks (MNIST). Especially, for the tiered-ImageNet and mini-ImageNet which are relatively difficult datasets, our model achieves largely outperforms all baseline models with significant margins. These results overall confirm the effectiveness of our learning to balance framework, that can balance the effect of the meta-knowledge and task- and class-specific knowledge.

Meta-test	OVD	Fashion MNIST
Prototypical Networks	96.87 ± 0.34	60.13 ± 0.30
Meta-SGD	93.97 ± 0.55	62.46 ± 0.73
MT-NET	94.72 ± 0.61	62.18 ± 0.90
Bayesian TAML	**97.69 ± 0.62**	**64.25 ± 0.86**

Table 2: **Multi-Dataset any-shot classification results.**

Multi-Dataset OVD. We further test our model under a more challenging setting where tasks could come from a highly heterogeneous dataset. To this end, we combine Omniglot, VGG flower (Nilsback & Zisserman, 2008), DTD (Cimpoi et al., 2014) into a single dataset OVD, and randomly sample each class from the combined dataset for every task. We train all models with 10-way any-shot tasks with 3 inner gradient steps and test on OVD and FasionMNIST(Xiao et al., 2017), where the latter is used to generate out-of-distribution tasks at evaluation time. The results in Table 2 shows that under this challenging multi-dataset setting, our Bayesian TAML outperforms all baselines, especially with larger gains on the out-of-distribution tasks (Fashion MNIST) consistent with the result of Table 1.

5.2 Effectiveness of the Balancing Variables

We now validate the effectiveness of each balancing parameter. For these experiments, we trained and evaluated the model with Omniglot 10-way 5 inner-gradient steps. We report the average performances over 1000 randomly selected episodes. Due to the limits of space, we omit standard errors for the confidence score. To correctly evaluate the effect of the individual balancing parameter, we drop all other balancing parameters when experimenting with each term.

$g(\mathbf{z}^\tau)$ for handling distributional discrepancy. $g(\mathbf{z}^\tau)$ weights initialization parameter, to decide on what and how much to use from the meta-knowledge, depending on the relateness of the unseen tasks to seen tasks. We evaluate models on both Omniglot and MNIST with 15-shot. Figure 4 shows that $g(\mathbf{z}^\tau)$ is large for in-distribution task and small with out-of-distribution task. Table 3 shows that suppressing the utilization of meta-knowledge with $g(\mathbf{z}^\tau)$ effectively handles out-of-distribution tasks and yields our model largely outperforms the baselines.

Models (K=5)	Omniglot	MNIST
MAML	98.32	89.19
Meta-SGD	98.38	90.42
ABML	98.67	90.69
Bayesian $g(\mathbf{z}^\tau)$ TAML	**98.92**	**92.06**

Table 3: Distribution discrepancy results.

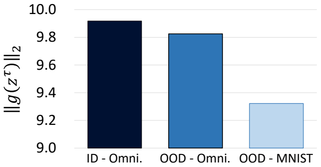


Figure 4: $\|g(\mathbf{z}^\tau)\|_2$ under ID/OOD.

$f(\gamma^\tau)$ for handling task imblance. $f(\gamma^\tau)$, which is a decaying factor for inner gradient steps, handles inter-task imbalance where each task has different number of examples. Figure 5 shows the $f(\gamma^\tau)$ to varying size of tasks, where it increases monotonically with the number of instances allowing the model to stay close to the initial parameter for few-shot cases and deviate far from it for many-shot cases. Table 4 shows that the larger gains of our model for 1-shot than 5 or 15 shots support that relying on meta-knowledge is useful to improve the performance on the small size tasks.

Models (K=5)	1-shot	5-shot	15-shot
MAML	92.93	97.13	95.52
Meta-SGD	92.38	96.60	98.04
ABML	92.46	97.27	98.27
Bayesian $f(\gamma^\tau)$ TAML	**94.41**	**97.74**	**98.62**

Table 4: Task-Imbalance results.

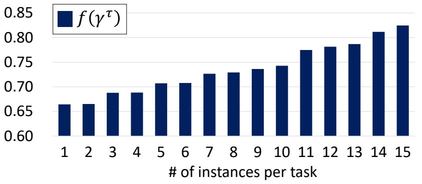


Figure 5: $f(\gamma^\tau)$ with varying number of shots.

$g(\boldsymbol{\omega}^{\tau})$ for handling class imbalance. $g(\boldsymbol{\omega}^{\tau})$ adjusts the scale of the gradient for each class to handle class imbalance where the number of instances per class largely varies. Table 5 shows the result of each model under varying degree of the number of shots imbalance between classes within each task. We observe that our model significantly outperforms baselines especially on sets with a high degree of class imbalance (x5 and x15). Figure 6 further shows that the $g(\boldsymbol{\omega}^{\tau})$ supresses the empirical loss of classes with larger shots to balance the learning for each class, and that we obtain the most improvements on classes with small number of instances.

Models (K=5)	x1	x5	x15
MAML	**98.17**	95.77	84.97
Meta-SGD	97.31	95.38	89.12
ABML	97.79	95.89	91.09
Bayesian $g(\boldsymbol{\omega}^{\tau})$ TAML	97.29	**96.82**	**91.61**

Table 5: Class-Imbalance results.

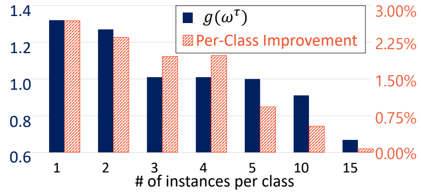


Figure 6: $g(\boldsymbol{\omega}^{\tau})$ and accuracy gains over Meta-SGD with varying number of instances per class.

5.3 More ablation study

Effectiveness of Bayesian modeling We further see the effect of the Bayesian framework by comparing the behaviors of each balancing variable on out-of-distribution tasks. Table 6 shows the performance of the models with the same setting of Table 3. The result clearly shows that the Bayesian methods greatly contribute to address imbalance problem, especially with the OOD tasks. Figure 7 further confirms the effectiveness against deterministic model (Deterministic TAML) as the balancing variables more sensitively react to the imbalance conditions with the Bayesian modeling (Bayesian TAML) of those variables.

Models (K=5)	Omniglot	MNIST
MAML	98.32	89.19
Meta-SGD	98.38	90.42
ABML	98.67	90.69
Deterministic TAML	98.75	90.77
Bayesian TAML	**99.13**	**92.38**

Table 6: Distribution discrepancy results.

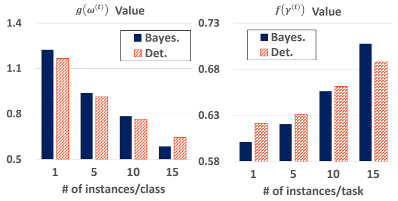


Figure 7: Balancing variables under OOD.

Dataset encoding We perform an ablation study to validate the effectiveness of the proposed dataset encoding method *Set of Sets* when generating the balancing variables. Table 7 shows the performance of various encoding schemes on the imbalanced tiered-ImageNet for 5-way classification with the same setting as Table 1. We observe the effectiveness of *Set of Sets* with higher-order statistics and hierarchical set encoding against DeepSets (Zaheer et al., 2017).

Models (K=5)	DeepSets	Set of Sets (Ours)
Mean	63.22	68.89
Mean + Var.	65.32	69.70
Mean + Skew.	66.76	70.49
Mean + Kurt.	68.46	70.75
Mean + All	69.21	**71.42**

Table 7: Ablation study on the dataset-encoding methods.

6 Conclusion

We propose Bayesian TAML that learns to balance the effect of meta-learning and task-adaptive learning, to consider meta-learning under a more realistic task distribution where each task and class can have varying number of instances. Specifically, we encode the dataset for each task into set representations, and use it to generate weight mask for the original parameter, learning rate decay, and the class-specific learning rate. We use a Bayesian framework to infer the posterior of these balancing variables, and propose a effective meta-learning variational inference framework to solve for them. Our model outperforms existing meta-learning methods when validated on imbalanced few-shot classification tasks. Further analysis of each balancing variable shows that each variable effectively handles task imbalance, class imbalance, and out-of-distribution tasks respectively. We believe that our work makes a meaningful step toward application of meta-learning to real-world problems.

REFERENCES

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3606–3613, 2014.

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-Volume 70*, pp. 1126–1135. JMLR. org, 2017.

Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic model-agnostic meta-learning. In *Advances in Neural Information Processing Systems*, pp. 9516–9527, 2018.

Jonathan Gordon, John Bronskill, Matthias Bauer, Sebastian Nowozin, and Richard E Turner. Meta-learning probabilistic inference for prediction. *arXiv preprint arXiv:1805.09921*, 2018.

Taesup Kim, Jaesik Yoon, Ousmane Dia, Sungwoong Kim, Yoshua Bengio, and Sungjin Ahn. Bayesian model-agnostic meta-learning. *arXiv preprint arXiv:1806.03836*, 2018.

Diederik P. Kingma and Max Welling. Auto encoding variational bayes. In *2nd International Conference on Learning Representations*. 2014.

Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

Yoonho Lee and Seungjin Choi. Gradient-based meta-learning with learned layerwise metric and subspace. In *International Conference on Machine Learning*, pp. 2933–2942, 2018.

Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few shot learning. *arXiv preprint arXiv:1707.09835*, 2017.

R. S. Michalski, J. G. Carbonell, and T. M. Mitchell (eds.). *Machine Learning: An Artificial Intelligence Approach, Vol. I*. Tioga, Palo Alto, CA, 1983.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *CoRR, abs/1803.02999*, 2, 2018.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pp. 722–729. IEEE, 2008.

Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 719–729. 2018.

Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L Yuille. Few-shot image recognition by predicting parameters from activations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7229–7238, 2018.

Sachin Ravi and Alex Beatson. Amortized bayesian meta-learning. 2018.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International conference on machine learning*, 2017.

Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. Meta-learning for semi-supervised few-shot classification. In *Proceedings of 6th International Conference on Learning Representations ICLR*, 2018.

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.

Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*, pp. 1842–1850, 2016.

Jürgen Schmidhuber. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, Technische Universität München, 1987.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pp. 4077–4087, 2017.

Sebastian Thrun and Lorien Pratt (eds.). *Learning to Learn*. Kluwer Academic Publishers, Norwell, MA, USA, 1998. ISBN 0-7923-8047-9.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pp. 3630–3638, 2016.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Flood Sung Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning.(2017). 2017.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan R Salakhutdinov, and Alexander J Smola. Deep sets. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 3391–3401. Curran Associates, Inc., 2017. URL `http://papers.nips.cc/paper/6931-deep-sets.pdf`.

A Experimental Setup

A.1 Baselines and Network Architecture.

We describe baseline models and our task-adaptive learning to balance model. Note that all gradient-based models can be extended to take K inner-gradient steps for both meta-training and meta-testing.

1) Meta-Learner LSTM. A meta-learner that learns optimization algorithm with LSTM (Ravi & Larochelle, 2017). The model performs few-shot classification using cosine similarities between the embeddings generated from a shared convolutional network.

2) Prototypical Networks. A metric-based few-shot classification model proposed by (Snell et al., 2017). The model learns the metric space based on Euclidean distance between class prototypes and query embeddings.

3) MAML. The Model-Agnostic Meta-Learning (MAML) model by (Finn et al., 2017), which aims to learn the global initial model parameter, from which we can take a few gradient steps to get task-specific predictors.

4) Meta-SGD. A base MAML with the learnable learning-rate vector (without any restriction on sign) element-wisely multiplied to each step inner-gradient (Li et al., 2017).

5) MT-NET. A gradient-based meta-learning model proposed by Lee & Choi (2018). The model obtains a task-specific parameter only w.r.t. a subset of the whole dimension (M-Net), followed by a linear transformation to learn a metric space (T-Net).

6) Probabilistic MAML. A probabilistic version of MAML by Finn et al. (2018), where they model task-adaptive inner-gradient steps as a posterior inference process under hierarchical Bayesian framework.

7) ABML. This model also interprets MAML under hierarchical Bayesian framework, but they propose to share and amortize the inference rules across both global initial parameters as well as the task-specific parameters.

8) Bayesian TAML. Our learning to balance model that can adaptively balance between meta- and task-specific learners for each task and class.

A.2 Realistic any-shot classification.

We describe more detailed settings for realistic any-shot classification.

Imbalanced Omniglot. We modified the episode generating strategy of C-way classification, which selects the number of shots randomly between 1 to 15 for each of the classes. The meta learning rate β and the total number of iteration are set to 1e-3 and 60000, respectively for all models, and the inner gradient step size α is set to 0.05 for MAML, MT-NET and ABML and is set to learnable parameters for Meta-SGD and our model. The number of inner-gradient steps is 5 for all models and all other components are the same as reported for fixed-way and fixed-shot few-shot classification in the paper for each model. We keep the meta-batch as 1 for all experiments to clearly see the effect of imbalance scenario. We trained models in 5-way 5 inner-gradient steps on the Omniglot, and evaluated with the test split of Omniglot and MNIST.

Imbalanced tiered-ImageNet. We modified the episode generating strategy of C-way classification, which selects the number of shots randomly between 1 to 50 for each of the classes. We set the number of query points as 15 and the meta learning rate β is set to 1e-4. Other components are set to the same as referred in **Imbalanced Omniglot.** We trained models on the tiered-ImageNet, and evaluated with the test split of tiered-ImageNet and mini-ImageNet.

A.3 Inference network architecture

We describe the network architecture of the inference network that takes a classification dataset as an input and generates three balancing variables as output. We additionally used two *average pooling* with 2×2 strides before the shared encoder NN_1 with large inputs such as tiered-ImageNet and mini-ImageNet. We empirically found that attaching *average pooling* reduces computation cost while improving performance.

Shared encoder $\mathrm{NN}_1 : \mathbf{X}_c^\tau \rightarrow \mathbf{s}_c$

conv2d 10 feature maps with 3 × 3 kernels and ReLU activation
max pooling with 2 × 2 strides
conv2d 10 feature maps with 3 × 3 kernels and ReLU activation
max pooling with 2 × 2 strides
fully-connected linear layer with 64 units
Statistics Pooling across the 64-dim. representations.
Concatenate the statistics into a single vector $\mathbf{s}_c$ for class c

Shared encoder $\mathrm{NN}_2 : \mathbf{s}_1, \ldots, \mathbf{s}_C \rightarrow \mathbf{v}^\tau$

fully-connected layer with 128 units and ReLU activation
fully-connected linear layer with 32 units
Statistics Pooling across the 32-dim. class representations
Concatenate the statistics into a single vector $\mathbf{v}^\tau$ for task τ

Shared encoder $\mathrm{NN}_3 : \mathbf{v}^\tau \rightarrow \boldsymbol{\mu}_\phi^\tau, \boldsymbol{\sigma}_\phi^\tau$

fully-connected layer with 64 units and ReLU activation
fully-connected layer to generate $\boldsymbol{\mu}_\phi^\tau$ and $\boldsymbol{\sigma}_\phi^\tau$

B JUSTIFICATION FOR SET-OF-SETS STRUCTURE.

Based on the previous justification of DeepSets (Zaheer et al., 2017), we can easily justify the Set-of-Sets structure proposed in the main paper as well, in terms of the two-level permutation invariance properties required for any classification dataset. The main theorem of DeepSets is:

Theorem 1. *A function f operating on a set $\mathbf{X} \in \mathcal{X}$ is a valid set function (i.e. permutation invariant), iff it can be decomposed as $f(\mathbf{X}) = \rho_2(\sum_{\mathbf{x}\in\mathbf{X}} \rho_1(\mathbf{x}))$, where ρ_1 and ρ_2 are appropriate nonlinearities.*

See (Zaheer et al., 2017) for the proof. Here we apply the same argument twice as follows.

1. A function f operating on a set of representations $\{\mathbf{s}_1, \ldots, \mathbf{s}_C\}$ (we assume each $\mathbf{s}_c$ is an output from a shared function g) is a valid set function (i.e. permutation invariant w.r.t. the order of $\{\mathbf{s}_1, \ldots, \mathbf{s}_C\}$), *iff* it can be decomposed as $f(\{\mathbf{s}_1, \ldots, \mathbf{s}_C\}) = \rho_2(\sum_{c=1}^C \rho_1(\mathbf{s}_c))$ with appropriate nonlinearities ρ_1 and ρ_2.
2. A function g operating on a set of examples $\{\mathbf{x}_{c,1}, \ldots, \mathbf{x}_{c,N}\}$ is a valid set function (i.e. permutation invariant w.r.t. the order of $\{\mathbf{x}_{c,1}, \ldots, \mathbf{x}_{c,N}\}$) *iff* it can be decomposed as $g(\{\mathbf{x}_{c,1}, \ldots, \mathbf{x}_{c,N}\}) = \rho_4(\sum_{i=1}^N \rho_3(\mathbf{x}_{c,i}))$ with appropriate nonlinearities ρ_3 and ρ_4.

Inserting $\mathbf{s}_c = g(\{\mathbf{x}_{c,1}, \ldots, \mathbf{x}_{c,N}\})$ into the expression of f, we arrive at the following valid composite function operating on a set of sets:

$$f\left(\{g\left(\{\mathbf{x}_{c,1}, \ldots, \mathbf{x}_{c,N}\}\right)\}_{c=1}^{C}\right) = \rho_2\left(\sum_{c=1}^{C} \rho_1\left(\rho_4\left(\sum_{i=1}^{N} \rho_3\left(\mathbf{x}_{c,i}\right)\right)\right)\right) \tag{8}$$

Let F denote the composite of f and (multiple) g and let NN_2 denote the composite of ρ_1 and ρ_4. Further define $\mathrm{NN}_1 := \rho_3$ and $\mathrm{NN}_3 := \rho_2$. Then, we have

$$F\Big(\big\{\{\mathbf{x}_{1,1}, \ldots, \mathbf{x}_{1,N}\}, \ldots, \{\mathbf{x}_{C,1}, \ldots, \mathbf{x}_{C,N}\}\big\}\Big) = \mathrm{NN}_3\left(\sum_{c=1}^{C} \mathrm{NN}_2\left(\sum_{i=1}^{N} \mathrm{NN}_1\left(\mathbf{x}_{c,i}\right)\right)\right) \tag{9}$$

where C is the number of classes and N is the number of examples per class. See Section A.3 for the correspondence between Eq. (9) and the actual encoder structure.

C Comparison between the two approximations

We provide the comparison between the two approximation schemes for evaluating the expectation of the test example predictions at meta-testing time. Naive approximation means that we take the expectation inside (i.e. we do not sample) and MC approximation means that we perform Monte-Carlo integration with sample size $S = 10$ [1]. We see from the Table 8 that MC ingetration performs better than the naive approximation, especially with OOD tasks (e.g. MNIST, mini-ImageNet). This is because the predictive distributions involve higher uncertainty for OOD tasks, hence there exists more benefit from considering the large variance than simply ignoring it.

Models (K=5)	Omniglot	MNIST	tiered-ImageNet	mini-ImageNet
Bayesian TAML (Naive approx.)	96.18 ± 0.33	83.95 ± 0.52	71.17 ± 1.09	57.68 ± 0.48
Bayesian TAML (MC approx.)	**96.29 ± 0.31**	**84.39 ± 0.48**	**71.42 ± 1.00**	**58.37 ± 0.49**

Table 8: **Classification performance under realistic scenario.** The models are trained with Omniglot(left) and tiered-ImageNet(right) with imbalanced setting. All reported results are average performances over 1000 for Omniglot and MNIST and 600 for tiered-ImageNet and mini-ImageNet randomly selected episodes with standard errors for 95% confidence interval over tasks.

D Few-shot classification with fixed-way and fixed-shot.

We further compare our model with the existing meta-learning approaches on conventional few-shot classification with fixed-way and fixed-shot.

Omniglot. We report the 20-way classification performance for this dataset. Following Finn et al. (2017), we use 4-block convolutional neural network architecture with 64 channels for each layer. We set the number of inner-gradient steps K to 5 for both meta-training and meta-testing.

Mini-ImageNet. We report the 5-way classification performance with the meta-batch 4 and 2 for 1- and 5-shot, respectively. We reduce the convolution filter size of the 4-block CNN network into 32 to prevent overfitting. We set $K = 5$ for multi-step models for both meta-training and meta-testing.

	Omniglot 20-way		mini-ImageNet 5-way	
Models	1-shot	5-shot	1-shot	5-shot
Meta-Learner LSTM Ravi & Larochelle (2017)	-	-	43.44 ± 0.77	60.60 ± 0.71
Prototypical Networks Snell et al. (2017) [2]	96.0	98.9	49.42 ± 0.78	65.77 ± 0.70
MAML (Finn et al., 2017)	95.80 ± 0.30	98.90 ± 0.20	48.70 ± 1.84	63.11 ± 0.92
Meta-SGD (Li et al., 2017)	95.93 ± 0.38	98.97 ± 0.19	50.71 ± 1.87	64.03 ± 0.94
Meta-SGD (K=3,5)	96.03 ± 0.43	98.72 ± 0.56	49.63 ± 1.41	63.74 ± 1.03
Relation Net (Yang et al., 2017)	97.6 ± 0.2	99.1 ± 0.1	50.44 ± 0.82	65.32 ± 0.70
MT-NET (Lee & Choi, 2018)	96.20 ± 0.40	-	51.70 ± 1.84	-
Probabilistic MAML (Finn et al., 2018)	-	-	50.13 ± 1.86	-
Reptile (Nichol et al., 2018)	89.43 ± 0.14	97.12 ± 0.32	49.97 ± 0.32	65.99 ± 0.58
BMAML (Kim et al., 2018)	-	-	**53.8 ± 1.46**	-
VERSA (Gordon et al., 2018)	97.66 ± 0.29	98.77 ± 0.18	53.40 ± 1.82	67.37 ± 0.86
Bayesian TAML	**98.10 ± 0.26**	**99.12 ± 0.18**	51.72 ± 1.62	**68.32 ± 1.11**

We first compare our method on conventional fixed-way fixed-shot classification task against existing meta-learning methods. Though the classification with uniformly distributed instances is not the task we aim to tackle, we find that Bayesian TAML outperforms most baseline models, except for the mini-imagenet 5-way 1-shot experiment where BMAML works the best.

[1] At meta-training time, we perform MC approximation with a single sample for computational efficiency

[2] We adopt the accuracies of the Prototypical Network in the setting which the number of shot and way are the same for training and testing phase for the consistency with other methods. In the "higher way" setting where 20-way is used during training for 5-way testing, the reported performance of the model is 68.20 ± 0.66%.