Meta-Learning Divergences of Variational Inference

Ruqi Zhang
Cornell University

Yingzhen Li*
Imperial College London

Christopher De Sa
Cornell University

Sam Devlin
Microsoft Research, Cambridge

Cheng Zhang
Microsoft Research, Cambridge

Abstract

Variational inference (VI) plays an essential role in approximate Bayesian inference due to its computational efficiency and broad applicability. Crucial to the performance of VI is the selection of the associated divergence measure, as VI approximates the intractable distribution by minimizing this divergence. In this paper we propose a meta-learning algorithm to learn the divergence metric suited for the task of interest, automating the design of VI methods. In addition, we learn the initialization of the variational parameters without additional cost when our method is deployed in the few-shot learning scenarios. We demonstrate our approach outperforms standard VI on Gaussian mixture distribution approximation, Bayesian neural network regression, image generation with variational autoencoders and recommender systems with a partial variational autoencoder.

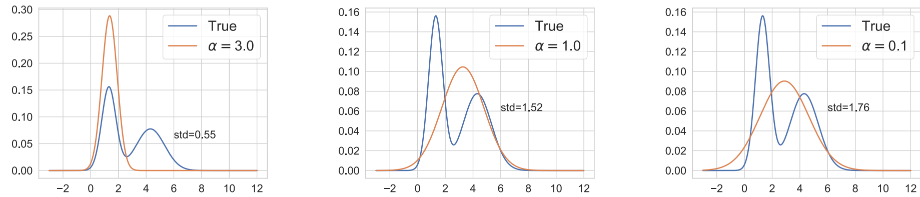
1 Introduction

Approximate inference is a powerful tool for probabilistic modelling of complex data. Among these inference methods, variational inference (VI) (Jordan et al., 1999; Zhang et al., 2018) approximates the intractable target distribution through optimizing a tractable distribution. This optimization-based inference makes VI computationally efficient, thus suitable to large-scale models in deep learning, such as Bayesian neural networks (Blundell et al., 2015) and deep generative models (Kingma and Welling, 2014). The objective function in VI is a divergence which measures the discrepancy between the approximate distribution and the target distribution. As an objective function, this divergence significantly affects the inductive bias of the VI algorithm. By selecting a divergence, we encode our preference to the approximate distribution, such as whether it should be mass-covering or mode-seeking. The Kullback-Leibler (KL) divergence is one of the most widely used divergence metrics. However, it has been criticized for under-estimating uncertainty, leading to poor results when uncertainty estimation is essential (Bishop, 2006; Blei et al., 2017; Wang et al., 2018a). Many alternative divergences have been proposed to alleviate this issue (Bamler et al., 2017; Csiszár et al., 2004; Hernández-Lobato et al., 2016; Li and Turner, 2016; Minka et al., 2005; Wang et al., 2018a).

Although prior work has enriched the divergence family, the optimal divergence metric usually depends on tasks (Minka et al., 2005; Li and Turner, 2016). As illustrated by Figure 1, different divergence metrics can lead to very different inference results. Unfortunately, choosing a divergence for a specific task is challenging as it requires a thorough understanding of (i) the shape of the target distribution; (ii) the desirable properties of the approximate distribution; and (iii) the bias-variance trade-off of the variational bound. A crucial question remains to be addressed in order to make VI a success: how can we automatically choose a suitable divergence tailored to specific types of task?

To answer this question, we propose meta-learning divergences of variational inference which utilizes meta-learning, or learning to learn, to refine VI's divergence automatically. In a nutshell, we leverage the fact that various real-world applications consist of many small tasks (e.g. personalized recommendations for different user groups in recommender systems), and it is important to design a meta-learning algorithm to learn a good inference algorithm for new tasks from previous tasks. We summarize our contributions as follows:

Proceedings of the 24$^{\text{th}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130. *Work done in Microsoft Research.



(a) $\alpha = 3.0$ (b) $\alpha = 1.0$ (c) $\alpha = 0.1$

Figure 1: An illustration of approximate distributions on a Gaussian mixture by different α-divergences (defined in Eq.(3)). "std" is the standard deviation of the Gaussian approximation.

- We develop a general framework for meta-learning variational inference's divergence (Section 3.2), which chooses the desired divergence objective automatically given a type of tasks. In this way, we meta-learn the VI algorithm.
- Besides meta-learning the divergence objective, we further meta-learn the parameters for the variational distribution without additional cost (Section 3.3), enabling meta-learning VI in few-shot setting.
- We demonstrate VI with meta-learned divergences outperforms standard VI on Gaussian mixture distribution approximation, Bayesian neural network regression, image generation with variational autoencoders, and recommender systems with a partial variational autoencoder (Section 4).

2 Preliminaries

Consider a dataset $\mathcal{D} = \{x_n\}_{n=1}^N$ and a probabilistic model with parameters θ. Bayesian inference requires computing the posterior over θ given the dataset $\mathcal{D}$: $p(\theta|\mathcal{D}) = p(\mathcal{D}|\theta)p(\theta)/p(\mathcal{D})$. The exact posterior is generally intractable, so it needs to be approximated with a tractable posterior $q_\phi(\theta) \approx p(\theta|\mathcal{D})$. Typically the approximate posterior $q_\phi(\theta)$ is obtained by minimizing a divergence, e.g. variational inference (VI) often minimizes $\mathrm{KL}(q_\phi(\theta)\|p(\theta|\mathcal{D}))$. This turns Bayesian inference into an optimization task (divergence minimization). In practice, due to the intractability of $p(\mathcal{D})$, VI alternatively maximizes an equivalent objective called the *variational lower bound*:

$$\mathcal{L}_{\mathrm{VI}} = \mathbf{E}_{\theta\sim q_\phi}\left[\log \frac{p(\mathcal{D},\theta)}{q_\phi(\theta)}\right] = \log p(\mathcal{D}) - \mathrm{KL}(q_\phi\|p). \quad (1)$$

Renyi's α-divergence α-divergence is a generalization of KL divergence (Hernández-Lobato et al., 2016; Li and Turner, 2016; Minka, 2001). There are different definitions of α-divergence and their equivalences are shown in Cichocki and Amari (2010). Here we focus on Renyi's definition (Li and Turner, 2016; Rényi et al., 1961) instead of others (Amari, 2012; Tsallis, 1988) as it allows our meta-learning framework to be differentiable in α (Section 3.2). Renyi's α-divergence is defined on $\alpha > 0, \alpha \neq 1$

$$D_\alpha(p\|q) = \frac{1}{\alpha-1}\log\int p(\theta)^\alpha q(\theta)^{1-\alpha}d\theta, \quad (2)$$

and for $\alpha = 1$ it is defined by continuity: $D_1(p\|q) = \lim_{\alpha\to 1} D_\alpha(p\|q) = \mathrm{KL}(p\|q)$. Similar to the variational lower bound, one can maximize the *variational Renyi bound* (VR bound) (Li and Turner, 2016):

$$\begin{aligned}\mathcal{L}_\alpha(q_\phi;\mathcal{D}) &= \frac{1}{1-\alpha}\log\mathbf{E}_{\theta\sim q_\phi}\left[\left(\frac{p(\theta,\mathcal{D})}{q_\phi(\theta)}\right)^{1-\alpha}\right] \quad (3)\\ &= \log p(\mathcal{D}) - D_\alpha(q_\phi\|p).\end{aligned}$$

The expectation is usually computed by Monte Carlo (MC) approximation. To allow gradient backpropagation, the VR bound uses the reparameterization trick (Kingma and Welling, 2014; Salimans et al., 2013), where sampling $\theta \sim q_\phi(\theta)$ is conducted by first sampling $\epsilon \sim p(\epsilon)$ from a simple distribution independent with the variational distribution (e.g. Gaussian) then parameterizing $\theta = r_\phi(\epsilon)$. It follows that the gradient of the VR bound w.r.t. the variational parameter ϕ after MC approximation with K particles is

$$\nabla_\phi L_\alpha(q_\phi;x) = \sum_{k=1}^{K}\left[w_{\alpha,k}\nabla_\phi \log\frac{p(r_\phi(\epsilon_k),x)}{q(r_\phi(\epsilon_k))}\right], \quad (4)$$

where $w_{\alpha,k} = \left(\frac{p(r_\phi(\epsilon_k),x)}{q(r_\phi(\epsilon_k))}\right)^{1-\alpha} \Big/ \sum_{k=1}^K\left[\left(\frac{p(r_\phi(\epsilon_k),x)}{q(r_\phi(\epsilon_k))}\right)^{1-\alpha}\right]$. When $\alpha = 1$ the weights $w_{\alpha,k} = 1/K$ and the gradient Eq.(4) becomes an unbiased estimate of the gradient of the variational lower bound Eq.(1).

As shown in Figure 1, approximate inference with different α-divergences results in distinct variational distributions. Prior work (Li and Turner, 2016; Minka et al., 2005) also showed the optimal α-divergence varies for different tasks and datasets, and in practice it is difficult to choose an optimal α-divergence *a priori*.

f-divergence f-divergence defines a more general family of divergences (Csiszár et al., 2004; Minka et al., 2005). Given a twice differentiable convex function $f : \mathbb{R}_+ \to \mathbb{R}$, the f-divergence is defined as (Csiszár et al., 2004):

$$D_f(p\|q_\phi) = \mathbf{E}_{\theta\sim q_\phi}\left[f\left(p(\theta)/q_\phi(\theta)\right) - f(1)\right]. \tag{5}$$

This family includes KL-divergences in both directions, by taking $f(t) = -\log t$ for $\mathrm{KL}(q\|p)$ and $f(t) = t\log t$ for $\mathrm{KL}(p\|q)$. It also includes α-divergences by setting $f(t) = t^\alpha/(\alpha(\alpha-1))$ for $\alpha \in \mathbb{R}\backslash\{0,1\}$. Although the f-divergence family is very rich due to its parameterization by an arbitrary twice-differentiable convex function, it requires significant expertise to design a suitable f function for a specific task. Thus the potential of f-divergence has not been fully leveraged.

3 Meta-Learning Divergences of Variational Inference

3.1 Problem Set-Up

The goal of meta-learning VI algorithm is to learn, from a set of tasks, a VI algorithm that produces an approximate distribution with desired properties on new similar tasks. We approach this goal by learning the divergence in use for VI. We formalize the problem setups as follows.

Assume we have a task distribution $p(\mathcal{T})$. Each task $T_i \sim p(\mathcal{T})$ has its own dataset $\mathcal{D}_{T_i}$ and its own probabilistic model $p_{T_i}(\theta_i, \mathcal{D}_{T_i})$. Let $D_\eta(\cdot\|\cdot)$ denote a learnable divergence parameterized by η; then for each task T_i the approximate posterior $q_{\phi_i}(\theta_i)$ is computed by minimizing $D_\eta(p_{T_i}(\theta_i|\mathcal{D}_{T_i})\|q_{\phi_i}(\theta_i))$. In the rest of the paper we write $D_\eta(q_{\phi_i}, T_i) = D_\eta(p_{T_i}(\theta_i|\mathcal{D}_{T_i})\|q_{\phi_i}(\theta_i))$ for brevity. To do meta-training, in each step we first sample a minibatch of tasks $T_i, i = 1, \ldots, M$ from $p(\mathcal{T})$. Then we define a meta-loss function $\mathcal{J}(q_{\phi_i}, T_i)$, and optimize the total meta-loss across all training tasks in the minibatch $\sum_{i=1}^M \mathcal{J}(q_{\phi_i}, T_i)$ over the divergence parameter η. This meta-loss function is designed to evaluate the desired properties of the approximate distribution for these tasks, e.g. negative log-likelihood. During meta-testing, a new task is sampled from $p(\mathcal{T})$, and the learned divergence D_η is used to optimize the variational distribution q_ϕ.

We also consider (in Section 3.3) a few-shot learning setup similar to the model-agnostic meta-learning (MAML) framework (Finn et al., 2017). In this case, each task only has a few training data, therefore it is crucial to learn a good model initialization to avoid overfitting and adapt fast on unseen tasks. The goal of meta-learning VI algorithm in this setting is to obtain a divergence as well as an initialization of the variational parameters ϕ for unseen tasks. During meta-testing, we will train the model with the learned divergence and the learned initialization of variational parameters on new tasks.

The above two meta-learning settings are practical as demonstrated in many previous works (Finn et al., 2017, 2018; Gong et al., 2019; Kim et al., 2018), showing that attaining common knowledge from previous tasks is valuable for future tasks.

3.2 Meta-Learning Divergences (meta-D)

We consider the first setting of learning a divergence. We assume for now D_η is given in some parametric form; later on we will provide the details of parameterization of two divergence families (α- and f-divergence) and show how they fit in this framework. The general idea is to first optimize the approximate posterior by minimizing the current divergence, then update the divergence using the feedback from the meta-loss. Concretely, for each task T_i we perform B gradient descent steps on the variational parameters ϕ_i using VI with the current divergence D_η:

$$\phi_i \leftarrow \phi_i - \beta\nabla_{\phi_i} D_\eta(q_{\phi_i}, T_i). \tag{6}$$

By doing so the updated variational parameters ϕ_i are a function of the divergence parameter η, which we then update by one-step gradient descent using the meta-loss $\mathcal{J}$:

$$\eta \leftarrow \eta - \gamma\nabla_\eta \frac{1}{M}\sum_i \mathcal{J}(q_{\phi_i}, T_i). \tag{7}$$

We call this algorithm *meta-D* for meta-learning divergences, which is outlined in Algorithm 1. Our algorithm is different from MAML in that MAML's inner and outer loop losses are designed to be the same, prohibiting it to meta-learn the inner loop loss function which is the divergence in VI. The key insight of our approach is that the updated variational parameters are dependant on the inner loop divergence. This dependency enables meta-D to update the divergence by descending the meta-loss with back-propagation through the variational parameters.

Meta-learning within α-divergence family To make α-divergence learnable by the meta-D framework (in this case $\eta = \alpha$), it requires the inner-loop updates (Eq.(6)) to be continuous in α. This means a naive solution which relies on automatic differentiation of existing α-divergences will fail, due to the fact that these α-divergences are not twice differentiable everywhere (Li and Turner, 2016; Minka et al., 2005). Instead, we propose to manually compute the gradient of Renyi's α-divergence (Eq.(4)) which is continuous in $\alpha \in (0, +\infty)$.

Algorithm 1 Meta-D

Input: $p(\mathcal{T})$: distribution over tasks; β, γ: learning rate hyperparameters; initialize η
loop
  Sample M tasks $T_i \sim p(\mathcal{T})$
  for all T_i **do**
    if ϕ_i does not exist **then**
      initialize ϕ_i (can have different architectures)
    end if
    Update ϕ_i with the current divergence:
    for $b = 1 : B$ **do**
      $\phi_i \leftarrow \phi_i - \beta \nabla_{\phi_i} D_\eta(q_{\phi_i}, T_i)$
    end for
  end for
  Update $\eta \leftarrow \eta - \gamma \nabla_\eta \frac{1}{M} \sum_i \mathcal{J}(q_{\phi_i}, T_i)$
end loop
Output: η

Algorithm 2 Meta-D&ϕ

Input: $p(\mathcal{T})$: distribution over tasks; β, γ, τ: learning rate hyperparameters
Initialize ϕ, η
loop
  Sample M tasks $T_i \sim p(\mathcal{T})$
  for all T_i **do**
    Update ϕ_i with the current divergence:
    for $b = 1 : B$ **do**
      $\phi_i \leftarrow \phi - \beta \nabla_\phi D_\eta(q_\phi, T_i)$
    end for
  end for
  Update $\phi \leftarrow \phi - \tau \nabla_\phi \frac{1}{M} \sum_i \mathcal{J}(q_{\phi_i}, T_i)$;
  $\eta \leftarrow \eta - \gamma \nabla_\eta \frac{1}{M} \sum_i \mathcal{J}(q_{\phi_i}, T_i)$
end loop
Output: η, ϕ

Specifically we parameterize α-divergence by parameterizing its gradient (Eq.(4)) and set $\nabla_{\phi_i} D_\eta = -\nabla_{\phi_i} \mathcal{L}_\alpha$ in Algorithm 1. We denote meta-learning a divergence within α-divergences family as *meta-α*.

Meta-learning within f-divergence family We wish to parameterize the f-divergence Eq.(5) by parameterizing the convex function f using a neural network, since neural networks are known to be universal approximators and thus can cover diverse f-divergences. However, it is less straightforward to specify the convexity constraint for neural networks. Fortunately, Proposition 1 below indicates that the f-divergence and its gradient can be specified through its second derivative f'' (Wang et al., 2018a).

Proposition 1 *If $\nabla_\theta \log\left(\frac{p(\theta)}{q_\phi(\theta)}\right)$ exists, then by setting $g_f(t) = t^2 \cdot f''(t)$, we have (with $\theta = r_\phi(\epsilon)$)*

$$\begin{aligned}&\nabla_\phi D_f(p \| q_\phi)\\ &= -\mathbf{E}_{\epsilon \sim p(\epsilon)}\left[g_f\left(\frac{p(\theta)}{q_\phi(\theta)}\right)\nabla_\phi r_\phi(\epsilon) \nabla_\theta \log\left(\frac{p(\theta)}{q_\phi(\theta)}\right)\right].\end{aligned} \tag{8}$$

Therefore it remains to specify g (or f''), and the following Proposition 2 guarantees that using non-negative functions as g is sufficient for parameterizing the f-divergence family.

Proposition 2 *For any non-negative function g on $\mathbb{R}_+$, there exists a function f such that $g(t) = g_f(t) = t^2 \cdot f''(t)$. If $g_f(1) > 0$, then $D_f(p \| q_\phi) = 0$ implies $p = q_\phi$.*

See Wang et al. (2018a) for the proofs. Given these guarantees, we propose to parameterize f implicitly by parameterizing $g(t) = g_f(t)$ which can be any non-negative function. We turn the problem into using a neural network to express a non-negative function that is strictly positive at $t = 1$. For convenience, we further restrict the form of the function to be

$$g(t) = \exp(h_\eta(t)) \tag{9}$$

where $h_\eta(t)$ is a neural network with parameter η. This definition of g is strictly positive for all t, satisfying the assumption of Proposition 2. By doing so, the f-divergence is now learnable through Algorithm 1, by computing the gradient $\nabla_{\phi_i} D_\eta = \nabla_{\phi_i} D_{f_\eta}$ with Eq. (8).

With dataset $\mathcal{D}$, the density ratio in Eq. (8) becomes $\frac{p(\theta|D)}{q_\phi(\theta)} = \frac{p(\mathcal{D}|\theta)p(\theta)}{q_\phi(\theta)p(\mathcal{D})}$. We estimate $p(\mathcal{D})$ through importance sampling and MC approximation. After doing this, $\frac{p(\theta_k|\mathcal{D})}{q_\phi(\theta_k)} = \frac{p(\mathcal{D}|\theta_k)p(\theta_k)}{q_\phi(\theta_k)} \Big/ \frac{1}{K}\sum_{k=1}^K \frac{p(\mathcal{D}|\theta_k)p(\theta_k)}{q_\phi(\theta_k)}$ which can be regarded as a self-normalized estimator (see Appendix A for details).

Our method is different from Wang et al. (2018a) in the way that we use deep neural networks parameterization and enable learning the f-divergence through standard optimization. We denote meta-learning a divergence within f-divergences family as *meta-f*.

3.3 Meta-Learning Divergences and Variational Parameters (meta-D&ϕ)

In addition to learning the divergence objective, we also consider the few-shot setting where fast adaptation of the variational parameters to new tasks is desirable. Similar to MAML, the probabilistic models

$\{p_{T_i}(\theta_i, \mathcal{D}_{T_i})\}$ share the same architecture, and the goal is to learn an initialization of variational parameters $\phi_i \leftarrow \phi$. On a specific task, ϕ is adapted to be ϕ_i according to the learnable divergence D_η (which can be $-\mathcal{L}_\alpha$ or D_{f_η}):

$$\phi_i \leftarrow \phi - \beta \nabla_\phi D_\eta(q_\phi, T_i). \tag{10}$$

The updated ϕ_i is a function of both η and ϕ. For meta-update, besides updating divergence parameter η with Eq.(7), we also use the same meta-loss to update

$$\phi \leftarrow \phi - \tau \nabla_\phi \frac{1}{M} \sum_i \mathcal{J}(q_{\phi_i}, T_i). \tag{11}$$

We call this algorithm *meta-D&ϕ* which meta-learns both the divergence objective and variational parameters' initialization. It is summarized in Algorithm 2. Similar to the previous section, the divergence families in consideration are α- and f-divergence (denoted as meta-α&ϕ and meta-f&ϕ respectively).

4 Experiments

We evaluate the proposed approaches on a variety of tasks. For the mixture of Gaussians task, we perform distribution approximation (no data) and use different meta-losses to directly demonstrate the ability of meta-D (meta-learning divergences) and meta-D&ϕ (meta-learning divergences and variational parameters) to learn the optimal divergence. For all other experiments, we use negative log-likelihood as the meta-loss. For meta-D, we use standard VI (KL divergence) and VI with $\alpha = 0.5$ divergence which is a comonly used α-divergence (Li et al., 2015; Wang et al., 2018a) as baselines. For meta-D&ϕ, we test it in few-shot setup (i.e. few training data), and compare it to learning ϕ only which is obtained by Algorithm 2 without updating η. During meta-testing, we test this learned ϕ with KL divergence (denoted by VI&ϕ). We also include results of VI without learning initialization in the few-shot setup as a reference to show the gain of meta-learning initialization. Unless otherwise specified, we set $B = 1$. We discussed the effect of this hyperparameter in Appendix B and put details of experimental setting in Appendix C.

4.1 Approximate Mixture of Gaussians (MoG)

We first verify the ability of our methods on learning good divergences using a 1-d distribution approximation problem. Each task includes approximating a mixture of two Gaussians p by a Gaussian distribution q_{ϕ^*} attained from $\min_\phi D_\eta(p \| q_\phi)$. The mixture of Gaussian distribution $p(\theta) = 0.5\mathcal{N}(\theta; \mu_1, \sigma_1^2) + 0.5\mathcal{N}(\theta; \mu_2, \sigma_2^2)$ is generated by

$$\mu_1 \sim \text{Unif}[0, 3], \quad \sigma_1 \sim \text{Unif}[0.5, 1.0];$$
$$\mu_2 = \mu_1 + 3, \quad \sigma_2 = \sigma_1 * 2.$$

Table 1: Meta-D on MoG: learned value of α. BO (8 iters) has similar runtime as meta-α.

Methods	$\alpha = 0.5$	TV
meta-α	0.52±0.01	0.31±0.01
BO (8 iters)	0.81±0.03	0.69±0.08
BO (16 iters)	0.54±0.07	0.32±0.03

Table 2: Meta-D on MoG: rank of meta-loss over 10 test tasks.

Methods	$\alpha = 0.5$	TV
meta-α	**2.10**±0.70	2.10±0.30
meta-f	**2.10**±1.37	**1.00**±0.00
BO (8 iters)	3.50±0.67	4.00±0.00
BO (16 iters)	2.30±0.90	2.90±0.30

Therefore each task has a different target distribution but with similar properties (the same $\mu_2 - \mu_1$ and σ_2/σ_1). As shown in Figure 1, the divergence choice has significant impact on the approximation.

We test our methods with two types of meta-loss $\mathcal{J}$: $D_{0.5}(q\|p)$ and total variation (TV). If $D_{0.5}(q\|p)$ is the metric we care about when evaluating the quality of approximation q, then a good divergence will be $D_{0.5}(q\|p)$ itself. This case is to verify our method is able to learn the preferred divergence given a rich enough family $\{D_\eta\}$. In practice, the desired evaluation metric for approximation quality (e.g. log-likelihood) typically does not belong to α- or f-divergence family; to test this scenario we use the total variation distance (TV) to evaluate the performance of our method when meta-loss is beyond the divergence family.

We first test meta-D (meta-learning the divergences, Algorithm 1). As a baseline, we treat α as a hyperprameter and use Bayesian optimization (BO) (Snoek et al., 2012) to optimize it. Note that BO is not applicable when the divergence set is f-divergence which is parameterized by a neural network, therefore BO is only used as a baseline for meta-α.

We report the learned values of α from meta-α and BO in Table 1. When the meta-loss is $D_{0.5}$, the learned α from meta-α is very close to 0.5, confirming that our method can pick up a desired divergence. Note that BO is less computationally efficient, as it needs to train a model from scratch every single time when evaluating a new value of α, while our method can update α based on the current model.[1] We test learning f-divergence and visualize the learned $h_\eta(t)$ (Eq.(9)) in Figure 2(a)&(b).

[1] We also considered BO in later sections but found it

Table 3: Meta-$D\&\phi$ on MoG: rank of meta-loss over 10 test tasks.

Method	$\alpha = 0.5$ (20 iters)	TV (20 iters)	$\alpha = 0.5$ (100 iters)	TV (100 iters)
VI&ϕ	2.70±0.46	2.70±0.46	2.40±0.49	2.50±0.50
meta-α&ϕ	2.10±0.54	1.80±0.60	2.20±0.75	**1.40**±0.66
meta-f&ϕ	**1.20**±0.60	**1.50**±0.81	**1.40**±0.80	2.10±0.83

When the meta-loss is $D_{0.5}(q\|p)$, the corresponding $h_{0.5}(t)$ for $D_{0.5}$ is analytical (Appendix C.5), and we see from Figure 2(a) that the learned $h_\eta(t) \approx h_{0.5}(t) + 1.25$. This means meta-$D$ has learned the optimal divergence $D_{0.5}$, since $f(t)$ and $af(t)$ define the same divergence for $\forall a > 0$.

When the meta-loss is TV, the optimal divergence is not analytic. Therefore, we instead report the averaged rank of meta-losses on 10 test tasks in Table 2 (see Table 8 in Appendix for averaged value of meta-losses). It clearly shows that meta-α and meta-f are superior over BO. Moreover, meta-f outperforms meta-α when the meta-loss is TV. From Figure 2(b), we can see that the learned f-divergence is not inside α-divergence, showing the benefit of using a larger divergence family. It also indicates that our f-divergence parameterization using a neural network is flexible and can lead to new f-divergences that are not used before.

Next we test meta-$D\&\phi$ (meta-learning divergences and variational parameters, Algorithm 2). During training, we perform $B = 20$ inner loop gradient updates. The learned α is 0.88 and 0.77 for meta-loss $D_{0.5}$ and TV respectively, which is different from those reported in Table 1. We conjecture that this is related to the learned ϕ and B (the horizon length). During meta-testing, we start from the learned ϕ and train the variational parameters with the learned divergence for 20 and 100 iterations, corresponding to short and long horizons respectively. Table 3 summarizes the rankings. Our methods are better than VI&ϕ (which uses KL and only meta-learns ϕ) in all cases, demonstrating the benefit of learning a task-specific divergence instead of using the conventional VI for all. To further elaborate, we visualize in Figure 2(c)&(d) the approximate distributions after 20 steps. The q distributions obtained by meta-$D\&\phi$ tend to fit the MoG more globally (mass-covering), resulting in better meta-losses when compared with VI&ϕ. Compared to Algorithm 1, Algorithm 2 helps shorten the training time on new tasks (100 v.s. 2000 iterations). Notably, meta-$D\&\phi$ is able to provide this initialization along with divergence learning without extra cost.

very inefficient (e.g. on the experiment in Section 4.2, BO can only conduct two searches given similar runtime as our methods) thus omitted the results.

4.2 Regression Tasks with Bayesian Neural Networks

The second test considers Bayesian neural network regression. The distribution of ground truth regression function is defined by a sinusoid function with heteroskedastic noise (which is a function of x, see Figure 3(a)): $y = A\sin(x+b) + A/2\,|\cos((x+b)/2)|\,\epsilon$, where the amplitude $A \in [5, 10]$, the phase $b \in [0, 1]$ and $\epsilon \sim \mathcal{N}(0, 1)$. The heteroskedastic noise makes the uncertainty estimate more crucial comparing with the sinusoid function fitting task in prior work (Finn et al., 2017; Kim et al., 2018).

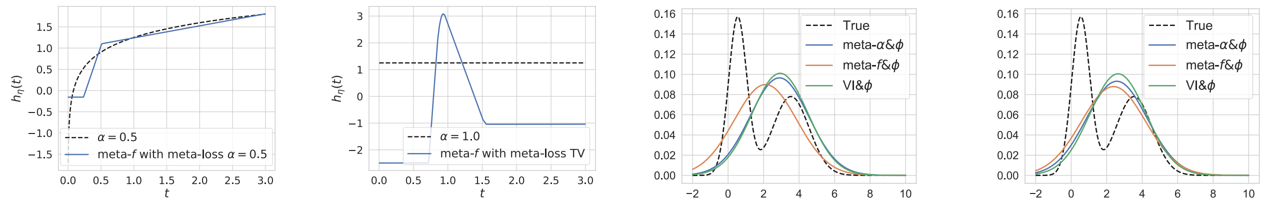
For Meta-D (meta-learning divergences, Algorithm 1), the quantitative results are summarized in Table 4. We can see that the test log-likelihood (LL) of both meta-α and meta-f are significantly better than VI and VI ($\alpha = 0.5$), while the root mean square error (RMSE) are similar for all methods. We visualize the predictive distribution on an example sinusoid function in Figure 3. All methods fit the mean well which is consistent with the RMSE results. Meta-α and meta-f can reason about the heteroskedastic noise whereas VI and VI ($\alpha = 0.5$) used homoskedastic noise to fit the data resulting in bad test LL.

For Meta-$D\&\phi$ (meta-learning divergences and variational parameters, Algorithm 2), during meta-testing, we fine-tune the learned ϕ with learned divergence on 40 datapoints for 300 epochs. Again meta-$\alpha\&\phi$ and meta-$f\&\phi$ are able to model heteroskedastic predictive distribution while VI&ϕ cannot. The quantitative results are reported in Table 5, and an example of predictive distribution is visualised in Figure 7 (see Appendix). Meta-$D\&\phi$ achieves similar results as meta-D with only 40 training data and 300 epochs. Methods without learning initialization for this setup significantly under-perform, indicating that learning model initialization is essential when data is scarce.

4.3 Image Generation with Variational Auto-encoders

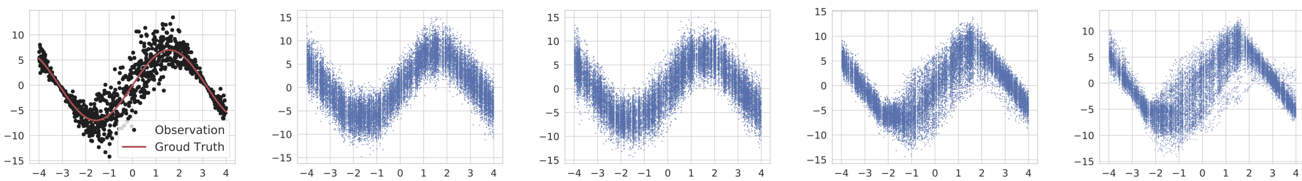
We also evaluate the image generation task with variational auto-encoders (VAEs). Specifically, we train VAEs to generate MNIST digits with different divergences. Generating each digit is regarded as a task and we use the first 5 digits (0-4) as the training tasks and the last 5 digits (5-9) as the test tasks.

We report the test marginal log-likelihood for each test digit in Table 6 and 7. Overall, these results align with other experiments that the meta-D and meta-$D\&\phi$ are both better than their counterparts. Meta-D and meta-$D\&\phi$ are better than VAE with common divergences on all 5 test tasks, indicating our methods have learned



(a) Meta-loss: $\alpha = 0.5$ (b) Meta-loss: TV (c) Meta-loss: $\alpha = 0.5$ (d) Meta-loss: TV

Figure 2: Visualization of (a)-(b) learned h_η and (c)-(d) approximate distribution after 20 updates. Meta-f refers to meta-learning divergences (Algorithm 1) within f-divergences. Meta-α&ϕ and Meta-f&ϕ refer to meta-learning divergences and variational parameters (Algorithm 2) within α- and f-divergences respectively. VI&ϕ refers to meta-learning varitional parameters only.



(a) Ground Truth (b) VI (c) VI ($\alpha = 0.5$) (d) meta-α (e) meta-f

Figure 3: Meta-D for BNN regression: visualizing the predictive distributions on sinusoid data. With our proposed method to meta-learn the divergence (panels (d) and (e)), the learned distribution can accurately capture the uncertainty in different regions while with vanilla VI (panel (b)) or VI with typical $\alpha = 0.5$ fails to capture the varying uncertainty in different regions.

Table 4: Meta-D on sin: 10 test tasks and each task has 1000 training data (1000 epochs).

	Test LL	RMSE
VI	-0.59±0.01	0.44±0.01
VI ($\alpha = 0.5$)	-0.57±0.02	0.43±0.01
meta-α	**-0.39**±0.04	0.43±0.00
meta-f	-0.40±0.04	0.42±0.02

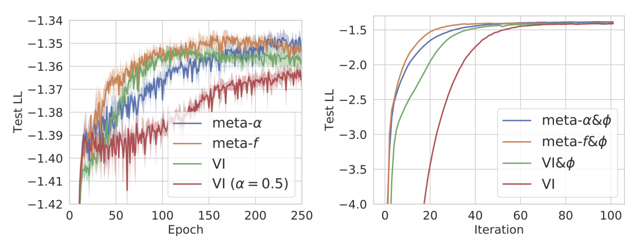
Table 5: Meta-D&ϕ on sin: 10 test tasks and each task has 40 training data (300 epochs).

	Test LL	RMSE
VI	-3.94±0.18	0.51±0.02
VI&ϕ	-0.69±0.04	0.44±0.02
meta-α&ϕ	**-0.43**±0.05	0.42±0.03
meta-f&ϕ	-0.46±0.04	0.43±0.02

a suitable divergence.

4.4 Recommender System with a Partial Variational Autoencoder

We test our method on recommender systems with a Partial Variational Auto-encoder (p-VAE) (Ma et al., 2019). P-VAE is proposed to deal with partially observed data and has been shown to achieve state-of-the-art level performance on user rating prediction in recommender system (Ma et al., 2018). We consider MovieLens 1M dataset (Harper and Konstan, 2016) which contains 1,000,206 ratings of 3,952 movies from 6,040 users. We select four age groups as training tasks, and use the remaining three groups as test tasks. During meta-testing, we use 90%/10% and 60%/40% training-test split for Meta-D and Meta-D&ϕ, respectively. From Figure 4(a), we see that when applied to learning p-VAEs, meta-D outperforms standard VI (KL divergence) and VI with $\alpha = 0.5$ divergence in



(a) Meta-D (b) Meta-D&ϕ

Figure 4: Test log-likelihood on MovieLens. Panel (a) shows the results of meta-learning divergences only (Meta-D), and panel (b) shows the results of meta-learning both divergences and variational parameters (Meta-D&ϕ).

Table 6: Meta-D (meta-learning divergences) on MNIST: marginal log-likelihood on 5 test tasks. Each task has 6000 training data. We train the model for 1000 epochs during meta-testing.

Digit	5	6	7	8	9
VI	-133.69± 0.23	-121.80±0.15	-92.25±0.40	-145.14±0.19	-119.64±0.23
VI ($\alpha = 0.5$)	-133.24±0.16	-121.90±0.71	-91.52±0.72	-144.90±0.31	-119.59±0.90
meta-α	**-132.74**±0.33	**-120.67**±0.36	**-90.62** ±0.45	-145.13±0.96	**-119.42**±0.36
meta-f	-133.21±0.44	-121.10± 0.20	-91.80±0.28	**-144.85**±0.31	**-119.42**±0.15

Table 7: Meta-$D\&\phi$ (meta-learning divergences and variational parameters) on MNIST: marginal log-likelihood on 5 test tasks. Each task has 100 training data. We train the model for 200 epochs during meta-testing.

Digit	5	6	7	8	9
VI	-177.92±0.46	-182.93±0.06	-125.57±0.41	-182.63±0.55	-161.68±0.27
VI&ϕ	-174.32±0.18	-176.17±0.26	-123.20±0.12	-177.96±0.23	-147.25±0.32
meta-$\alpha\&\phi$	-163.31±0.61	-163.19±0.36	**-115.52**±0.16	-173.35±0.38	-142.76±0.33
meta-$f\&\phi$	**-160.16**±0.16	**-154.16**±0.67	-122.61±0.43	**-165.83**±0.48	**-138.90**±0.10

terms of test LL, showing that meta-D has learned a suitable divergence that leads to better test performance. Figure 4(b) implies that all methods with learned ϕ can converge quickly on the new task with only 100 iterations. Both meta-$\alpha\&\phi$ and meta-$f\&\phi$ learn faster than VI&ϕ in meta-test time, indicating that the learned divergence can help fast adaptation.

5 Related Work

Variational Inference Variational inference (VI) has advanced rapidly in recent years (Zhang et al., 2018). These advances can be grouped into three categories: (1) introduction of new divergences for VI (Bamler et al., 2017; Hernández-Lobato et al., 2016; Li and Turner, 2016); (2) introduction of more expressive approximate families (e.g. Rezende and Mohamed, 2015; Ranganath et al., 2016); (3) improvement of sampling estimates for model evidence (Burda et al., 2015) and gradient (Rainforth et al., 2018); (4) stochastic optimization to scale VI (Dehaene and Barthelmé, 2018; Hoffman et al., 2013; Li et al., 2015). Our work is related to the work that improves the variational objective with alternative divergence measures; the difference is that our divergence measure is learnable and can be selected in an automatic fashion for a certain type of tasks.

Meta-Learning/few-shot learning Recent work has applied Bayesian modelling techniques to enhance uncertainty estimate for meta-learning/few-shot learning (Finn et al., 2018; Grant et al., 2018; Kim et al., 2018; Ravi and Beatson, 2019). They view the framework of MAML (Finn et al., 2017) as hierarchical Bayes and conduct Bayesian inference on meta-parameters and/or task-specific parameters. Grant et al. (2018) and Kim et al. (2018) applied approximate Bayesian inference to task-specific parameters, while Finn et al. (2018) kept point estimate for task-specific parameters and conducted variational inference over the meta-parameters instead. Ravi and Beatson (2019) obtained posteriors over both meta and task-specific parameters with variational inference. Our focus is distinct from this line of work in that our research is in the opposite direction: leveraging the idea of meta-learning to advance Bayesian inference. Additionally, our meta-$D\&\phi$ without learning divergence (VI&ϕ) can be viewed as a different Bayesian MAML method other than hierarchical Bayes, which directly trains the variational parameters so that it can quickly adapt to new tasks.

Meta-Learning for loss functions Our meta-learning method is also related to meta-learning a loss function. In reinforcement learning, Houthooft et al. (2018) meta-learned the loss function for policy gradients where the parameters of the loss function is updated using evolutionary strategies. Xu et al. (2018) meta-learned the hyperparameters of the loss functions in TD(λ) and IMPALA. Our work extends the idea of a learnable loss function to Bayesian inference.

Meta-Learning for Bayesian inference algorithms A recent attempt to meta-learning stochastic gradient MCMC (SG-MCMC) is presented by Gong et al. (2019), which proposed to meta-learn the diffusion and curl matrices of the SG-MCMC's underlying stochastic differential equation. Also Wang et al. (2018b) applied meta-learning to build efficient and generalizable block-Gibbs sampling proposals. Our work is distinct from previous work in that we apply meta-learning to improve VI, which is a more scalable inference method than MCMC. To the best of our knowledge, we are the first to study the automatic choice and design of VI algorithms.

6 Conclusion

We propose meta-learning divergences of VI which automates the selection of divergence objective in VI via meta-learning. It further allows meta-learning of variational parameter initialization for fast adaptation on new tasks. Within our meta-learning divergences framework, we consider two divergence families, α- and f-divergence, and design parameterizations of divergences to enable learning via gradient descent. Experimental results on Gaussian mixture approximation, regression with Bayesian neural networks, generative modeling and recommender systems demonstrate the superior performance of meta-learned divergences over standard divergences.

References

Shun-ichi Amari. *Differential-geometrical methods in statistics*, volume 28. Springer Science & Business Media, 2012.

Ron Amit and Ron Meir. Meta-learning by adjusting priors based on extended PAC-Bayes theory. In *International Conference on Machine Learning*, pages 205–214. PMLR, 2018.

Robert Bamler, Cheng Zhang, Manfred Opper, and Stephan Mandt. Perturbative black box variational inference. In *Advances in Neural Information Processing Systems*, pages 5079–5088, 2017.

Christopher M Bishop. *Pattern recognition and machine learning*. Springer Science+ Business Media, 2006.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112 (518):859–877, 2017.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. *International Conference on Machine Learning*, 2015.

Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*, 2015.

Andrzej Cichocki and Shun-ichi Amari. Families of alpha-beta-and gamma-divergences: Flexible and robust measures of similarities. *Entropy*, 12(6):1532–1568, 2010.

Imre Csiszár, Paul C Shields, et al. Information theory and statistics: A tutorial. *Foundations and Trends® in Communications and Information Theory*, 1(4): 417–528, 2004.

Guillaume Dehaene and Simon Barthelmé. Expectation propagation in the large data limit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(1):199–217, 2018.

Stefan Depeweg, José Miguel Hernández-Lobato, Finale Doshi-Velez, and Steffen Udluft. Learning and policy search in stochastic dynamical systems with Bayesian neural networks. *International Conference on Learning Representations*, 2017.

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*, pages 1126–1135. JMLR. org, 2017.

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

Sebastian Flennerhag, Pablo G Moreno, Neil D Lawrence, and Andreas Damianou. Transferring knowledge across learning processes. *International Conference on Learning Representations*, 2019.

Sebastian Flennerhag, Andrei A Rusu, Razvan Pascanu, Hujun Yin, and Raia Hadsell. Meta-learning with warped gradient descent. *International Conference on Learning Representations*, 2020.

Gustavo L Gilardoni. On Pinsker's and Vajda's type inequalities for Csiszár's f-divergences. *IEEE Transactions on Information Theory*, 56(11):5377–5386, 2010.

Wenbo Gong, Yingzhen Li, and José Miguel Hernández-Lobato. Meta-learning for stochastic gradient MCMC. *International Conference on Learning Representations*, 2019.

Erin Grant, Chelsea Finn, Sergey Levine, Trevor Darrell, and Thomas Griffiths. Recasting gradient-based meta-learning as hierarchical Bayes. *International Conference on Learning Representations*, 2018.

F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):19, 2016.

José Miguel Hernández-Lobato, Yingzhen Li, Mark Rowland, Daniel Hernández-Lobato, Thang Bui, and Richard Turner. Black-box α-divergence minimization. *International Conference on Machine Learning,*, 2016.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1): 1303–1347, 2013.

Rein Houthooft, Yuhua Chen, Phillip Isola, Bradly Stadie, Filip Wolski, OpenAI Jonathan Ho, and

Pieter Abbeel. Evolved policy gradients. In *Advances in Neural Information Processing Systems*, pages 5400–5409, 2018.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

Taesup Kim, Jaesik Yoon, Ousmane Dia, Sungwoong Kim, Yoshua Bengio, and Sungjin Ahn. Bayesian model-agnostic meta-learning. *Advances in Neural Information Processing Systems*, 2018.

Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *International Conference on Learning Representations4*, 2014.

Yingzhen Li and Richard E Turner. Rényi divergence variational inference. In *Advances in Neural Information Processing Systems*, pages 1073–1081, 2016.

Yingzhen Li, José Miguel Hernández-Lobato, and Richard E Turner. Stochastic expectation propagation. In *Advances in neural information processing systems*, pages 2323–2331, 2015.

Chao Ma, Wenbo Gong, José Miguel Hernández-Lobato, Noam Koenigstein, Sebastian Nowozin, and Cheng Zhang. Partial VAE for hybrid recommender system. In *NIPS Workshop on Bayesian Deep Learning*, 2018.

Chao Ma, Sebastian Tschiatschek, Konstantina Palla, Jose Miguel Hernandez Lobato, Sebastian Nowozin, and Cheng Zhang. Eddi: Efficient dynamic discovery of high-value information with partial VAE. *International Conference on Machine Learning*, 2019.

Thomas P Minka. Expectation propagation for approximate Bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 362–369. Morgan Kaufmann Publishers Inc., 2001.

Tom Minka et al. Divergence measures and message passing. Technical report, Technical report, Microsoft Research, 2005.

Tom Rainforth, Adam R Kosiorek, Tuan Anh Le, Chris J Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. *International Conference on Machine Learning*, 2018.

Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. Meta-learning with implicit gradients. In *Advances in Neural Information Processing Systems*, pages 113–124, 2019.

Rajesh Ranganath, Dustin Tran, and David Blei. Hierarchical variational models. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 324–333, 2016.

Sachin Ravi and Alex Beatson. Amortized Bayesian meta-learning. *International Conference on Learning Representations*, 2019.

Alfréd Rényi et al. On measures of entropy and information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*. The Regents of the University of California, 1961.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1530–1538, 2015.

Tim Salimans, David A Knowles, et al. Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis*, 8(4):837–882, 2013.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012.

Constantino Tsallis. Possible generalization of Boltzmann-Gibbs statistics. *Journal of statistical physics*, 52(1-2):479–487, 1988.

Dilin Wang, Hao Liu, and Qiang Liu. Variational inference with tail-adaptive f-divergence. In *Advances in Neural Information Processing Systems*, pages 5737–5747, 2018a.

Tongzhou Wang, Yi Wu, Dave Moore, and Stuart J Russell. Meta-learning MCMC proposals. In *Advances in Neural Information Processing Systems*, pages 4146–4156, 2018b.

Zhongwen Xu, Hado P van Hasselt, and David Silver. Meta-gradient reinforcement learning. In *Advances in neural information processing systems*, pages 2396–2407, 2018.

Cheng Zhang, Judith Butepage, Hedvig Kjellstrom, and Stephan Mandt. Advances in variational inference. *IEEE transactions on pattern analysis and machine intelligence*, 2018.