Meta-Amortized Variational Inference and Learning

Mike Wu,[*†] **Kristy Choi,**[*,†] **Noah Goodman,**[†‡] **Stefano Ermon**[†]
Department of Computer Science[†] and Psychology[‡]
Stanford University
{wumike, kechoi, ngoodman, ermon}@stanford.edu

Abstract

Despite the recent success in probabilistic modeling and their applications, generative models trained using traditional inference techniques struggle to adapt to new distributions, even when the target distribution may be closely related to the ones seen during training. In this work, we present a doubly-amortized variational inference procedure as a way to address this challenge. By sharing computation across not only a set of query inputs, but also a set of different, related probabilistic models, we learn transferable latent representations that generalize across several related distributions. In particular, given a set of distributions over images, we find the learned representations to transfer to different data transformations. We empirically demonstrate the effectiveness of our method by introducing the MetaVAE, and show that it significantly outperforms baselines on downstream image classification tasks on MNIST (10-50%) and NORB (10-35%).

Introduction

A wide variety of problems in machine learning (ML) can be framed as probabilistic inference in generative models. In particular, latent variable models learn representations of data that capture salient characteristics of its underlying distribution, which can then be used for downstream tasks such as classification (Klingler et al. 2017). While traditional inference techniques can be slow or even computationally intractable, the advent of *amortized (variational) inference* allowed such methods to scale to large datasets, bringing about significant progress in generative modeling applications such as image and audio synthesis (Brock, Donahue, and Simonyan 2018; Oord et al. 2016), molecule generation (Segler et al. 2017), and more.

However, as the problem domains we face become increasingly more complex and multimodal, a technical challenge arises: generative models trained using traditional inference techniques struggle to adapt to new data distributions, even when these new distributions may be *closely related* to distributions seen during training. For example, variational autoencoders (VAEs) trained on the original image distributions have difficulty generalizing to small visual transformations such as changing the position or quantity of objects in the scene. However, we would expect the true generative model, such as those of humans (Yildirim 2014), to be invariant to these slight modifications. Therefore, we aim to address: how do we design an amortized inference algorithm that generalizes across related distributions to learn *transferable* representations? Such features would capture the salient characteristics necessary to allow for better generalization to related, but unseen distributions at test time.

To address this question, we propose a *doubly-amortized* inference procedure that amortizes computation across not only a set of query inputs, but also a *set* of different, related target probabilistic models. More precisely, we derive a new objective called the MetaELBO which serves as a variational lower bound across multiple distributions, while also incorporating a prior regularization term encouraging each generative model to match its respective data marginal. We note that this inference model is not intended to be universal, but rather tailored to a specific family where each probabilistic model is similar in structure. Inspired by meta-learning, we denote this "doubly-amortized" inference problem as *meta-inference* and let a *meta-distribution* refer to the probability distribution over the family of probabilistic models.

As an instantiation of our method, we introduce the MetaVAE, a VAE trained with the MetaELBO. Empirically, we first show three demonstrations to build intuition for meta-inference: 1) clustering, 2) compiled inference, and 3) learning sufficient statistics on exponential families. Then, we study image transformations (e.g. rotations, shearing) on MNIST digits where the MetaVAE learns representations that transfer to unseen transformations, outperforming baselines by 10-50%. Finally, we showcase similar improvements of 10-35% on real-world images (NORB). While the representations learned from other generative models quickly decay in quality under more severe transformations, those of the MetaVAE preserve relevant information about the image while abstracting away unnecessary differences induced by visual manipulation.

Preliminaries

Exact and Approximate Inference

Let $p(\boldsymbol{x}, \boldsymbol{z})$ be a joint distribution over a set of latent variables $\boldsymbol{z} \in \mathcal{Z}$ and observed variables $\boldsymbol{x} \in \mathcal{X}$. An *inference query* involves computing posterior beliefs after incorporat-

[*]Denotes equal contribution.

ing evidence into the prior: $p(\boldsymbol{z}|\boldsymbol{x}) = p(\boldsymbol{x}, \boldsymbol{z})/p(\boldsymbol{x})$. This quantity is often intractable to compute as the marginal likelihood $p(\boldsymbol{x}) = \int_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}$ requires integrating or summing over a potentially exponential number of configurations for $\boldsymbol{z}$. Thus, we are forced to seek approximations.

Approximate inference techniques such as Markov Chain Monte Carlo (MCMC) sampling (Hastings 1970) and variational inference (VI) (Jordan et al. 1999; Wainwright and Jordan 2008) are widely used to approximate the posterior $p(\boldsymbol{z}|\boldsymbol{x})$. In VI, we introduce a family of tractable distributions $\mathcal{Q}$ parameterized by ψ over the latent variables and find the member (called the approximate posterior), $q_{\psi^*} \in \mathcal{Q}$ that minimizes the Kullback-Leibler (KL) divergence between itself and the exact posterior:

$$q_{\psi^*}(\boldsymbol{z}) = \arg\min_{q_\psi} D_{\mathrm{KL}}(q_\psi(\boldsymbol{z}) || p(\boldsymbol{z}|\boldsymbol{x})) \tag{1}$$

This $q_{\psi^*}(\boldsymbol{z})$ can serve as a proxy for the true posterior distribution. We note that the solution depends on the specific value of the observed (evidence) variables $\boldsymbol{x}$ we are conditioning on. For notational clarity, we rewrite the variational parameters as $\psi_{\boldsymbol{x}}$ to make explicit their dependence on $\boldsymbol{x}$.

One commonly needs to solve multiple inference queries of the same kind, conditioning on different values of the observed variables $\boldsymbol{x}$ (evidence). Let $p_{\mathcal{D}}(\boldsymbol{x})$ be an empirical distribution over the observed variables $\boldsymbol{x} \in \mathcal{X}$. Note $p_{\mathcal{D}}(\boldsymbol{x})$ can be different from the marginal $p(\boldsymbol{x})$ when the model is mis-specified. The average quality of the variational approximations can then be quantified by:

$$\mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\max_{\psi_{\boldsymbol{x}}} \mathbb{E}_{q_{\psi_{\boldsymbol{x}}}(\boldsymbol{z})} \log \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q_{\psi_{\boldsymbol{x}}}(\boldsymbol{z})}\right] \tag{2}$$

where $q_{\phi_{\boldsymbol{x}}}(\boldsymbol{z})$ can be viewed as an importance distribution. In practice, $p_{\mathcal{D}}(\boldsymbol{x})$ is unknown but we assume access to a training dataset $\mathcal{D}$ of examples i.i.d. sampled from $p_{\mathcal{D}}(\boldsymbol{x})$ that can be used to evaluate Eq. 2.

Amortized Variational Inference

An alternative formulation leverages a technique known as *amortization* (Gershman and Goodman 2014), which reduces the computational cost of Eq. 2 by casting the per-sample optimization process as a supervised *regression* task. Rather than solving for an optimal $q_{\psi^*_{\boldsymbol{x}}}(\boldsymbol{z})$ for every $\boldsymbol{x}$, we learn a single deterministic mapping $f_\phi : \mathcal{X} \to \mathcal{Q}$ to *predict* $\psi^*_{\boldsymbol{x}}$, or equivalently $q_{\psi^*_{\boldsymbol{x}}}(\boldsymbol{z}) \in \mathcal{Q}$, as a function of $\boldsymbol{x}$. Often, we choose to represent f_ϕ as a conditional distribution, denoted by $q_\phi(\boldsymbol{z}|\boldsymbol{x}) = f_\phi(\boldsymbol{x})(\boldsymbol{z})$ when scoring a value $\boldsymbol{z}$.

This procedure introduces an *amortization gap*, in which the less flexible parameterization of the inference model replaces the objective in Eq. 2 with the following lower bound:

$$\max_{\phi} \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \log \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right] \tag{3}$$

This gap refers to the suboptimality caused by amortizing the variational parameters over the entire training set, as opposed to optimizing for each example individually (pulling the $\max$ out of the expectation in Eq. 2). This tradeoff in expressiveness, however, enables significant speedups.

Learning Latent Variable Models

So far, we have assumed that the true generative model $p(\boldsymbol{x}, \boldsymbol{z})$ is given. However, we often only possess a family of possible models, $p_\theta(\boldsymbol{x}, \boldsymbol{z})$ parameterized by θ and the data set of observations, $\mathcal{D}$. The challenge then, is to choose θ whose model best explains the evidence. To do so, we maximize the log marginal likelihood of the data:

$$\mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x})\right] = \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\log \int_{\boldsymbol{z}} p_\theta(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}\right] \tag{4}$$

As mentioned, Eq. 4 is intractable to evaluate. Instead, we derive the Evidence Lower Bound (ELBO) to Eq. 4 using $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ as a tractable amortized inference model:

$$\mathbb{E}_{p_{\mathcal{D}}}[\log p_\theta(\boldsymbol{x})] \geq \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p_\theta(\boldsymbol{x}, \boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right]\right] \tag{5}$$

With Eq. 5 as an objective, we jointly optimize the parameters of the inference and generative models: ϕ and θ.

We may derive an alternative formulation of Eq. 5:

$$\begin{aligned}\mathcal{L}(\phi, \theta) &= -D_{\mathrm{KL}}(q_\phi(\boldsymbol{x}, \boldsymbol{z}) \| p_\theta(\boldsymbol{x}, \boldsymbol{z})) &(6)\\ &= -D_{\mathrm{KL}}(p_{\mathcal{D}}(\boldsymbol{x}) \| p_\theta(\boldsymbol{x})) \\ &\quad - \mathbb{E}_{p_{\mathcal{D}}}[D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x}) \| p_\theta(\boldsymbol{z}|\boldsymbol{x}))] &(7)\end{aligned}$$

where $q_\phi(\boldsymbol{x}, \boldsymbol{z}) = f_\phi(\boldsymbol{x})(\boldsymbol{z}) p_{\mathcal{D}}(\boldsymbol{x})$. Eq. 7 is comprised of a maximum likelihood term with a regularization penalty that encourages the generative model to have posteriors that can be easily approximated by the inference model. We will revisit this intuition once we introduce meta-amortization.

Often, $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ and $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ are parameterized by deep neural networks, which is known as a variational autoencoder, or VAE (Kingma and Welling 2013). The latent variables $\boldsymbol{z}$ are learned "features" inferred by $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ that can be used in downstream tasks, such as clustering or classification. The VAE is popular in many real-world domains: in medical diagnosis, for example, one can infer the identity of a disease ($\boldsymbol{z}$) from observed symptoms ($\boldsymbol{x}$). Given a set of symptoms from a population of patients, we can fit a VAE tailored to a disease, e.g. thoracic disease (Mao et al. 2018).

Meta-Amortized Variational Inference

But in practice, physicians often work with several patient populations that vary across a wide range of socioeconomic factors. For a new population, clinicians draw on prior experience from patients with similar symptoms, lowering their chances of misdiagnosis. We can similarly construct a generative model that captures this intuition. Instead of training a VAE on a new population, which would be equivalent to the physician re-learning how to diagnose an illness, we aim to share statistical strength between different patient groups to infer latent features that transfer to similar, but previously unseen populations.

We formalize this idea into a new algorithm that we call *meta-amortized inference*. Recall a (singly)-amortized inference model for $p_\theta(\boldsymbol{x}, \boldsymbol{z})$

$$\max_{\phi} \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\mathbb{E}_{f_\phi(\boldsymbol{x})} \log \frac{p_\theta(\boldsymbol{x}, \boldsymbol{z})}{f_\phi(\boldsymbol{x})(\boldsymbol{z})}\right] \tag{8}$$

which approximates $p_\theta(\boldsymbol{z}|\boldsymbol{x})$ for various choices of the observed variables, $\boldsymbol{x} \sim p_{\mathcal{D}}(\boldsymbol{x})$. Unlike Eq. 3, we have written $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ in its alternate form, $f_\phi(\boldsymbol{x})(\boldsymbol{z})$.

We are now interested in not one but a set of models, $\mathcal{J}_\mathcal{I} = \{p_{\theta_i}(\boldsymbol{x}, \boldsymbol{z}), i \in \mathcal{I}\}$ where $\mathcal{I}$ is a finite set of indices. Crucially, (like the example above) we make a few simplifying assumptions. First, we assume that the random variables in each model have the same domains (e.g. $\mathcal{X}, \mathcal{Z}$), but the relationships between the random variables may be different. Second, we assume that for each model, we care about the same inference query $p_{\theta_i}(\boldsymbol{z}|\boldsymbol{x})$. Finally, we assume to have some knowledge of typical values of the observed variables for each model in $\mathcal{J}_\mathcal{I}$: formally, we desire a set $\mathcal{M}_\mathcal{I} = \{p_{\mathcal{D}_i}(\boldsymbol{x}), i \in \mathcal{I}\} \subseteq \mathcal{M}$ of marginal distributions over the observed variables. Here, $\mathcal{M}$ denotes the set of all possible marginal distributions over $\mathcal{X}$. Let $p_\mathcal{M} : \mathcal{M}_\mathcal{I} \to [0, 1]$ denote a distribution over $\mathcal{M}_\mathcal{I}$. For example, $p_\mathcal{M}$ may be uniform over a finite number of marginals. As $p_\mathcal{M}$ is a distribution over distributions, we refer to it as a *meta-distribution*.

The naive approach to amortize over a set of models is:

$$\mathbb{E}_{p_{\mathcal{D}_i} \sim p_\mathcal{M}} \left[\max_\phi \mathbb{E}_{p_{\mathcal{D}_i}(\boldsymbol{x})} \left[\mathbb{E}_{f_\phi(\boldsymbol{x})} \log \frac{p_{\theta_i}(\boldsymbol{x}, \boldsymbol{z})}{f_\phi(\boldsymbol{x})(\boldsymbol{z})} \right] \right] \tag{9}$$

where we separately fit an amortized inference model for each $p_{\theta_i}(\boldsymbol{x}, \boldsymbol{z})$. However, this approach is prohibitively expensive as the size of $\mathcal{M}_\mathcal{I}$ increases, and training across models is decoupled. We instead propose to doubly-amortize the inference procedure as follows (we move the $\max$ out once more):

$$\max_\phi \mathbb{E}_{p_{\mathcal{D}_i} \sim p_\mathcal{M}} \left[\mathbb{E}_{p_{\mathcal{D}_i}(\boldsymbol{x})} \left[\mathbb{E}_{g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})} \log \frac{p_{\theta_i}(\boldsymbol{x}, \boldsymbol{z})}{g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})(\boldsymbol{z})} \right] \right] \tag{10}$$

where the original regressor $f_\phi(\boldsymbol{x})$ is replaced by a doubly-amortized regressor $g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})$ that takes *both* the marginal distribution $p_{\mathcal{D}_i}(\boldsymbol{x})$ and an observation $\boldsymbol{x}$ to return a posterior distribution. Formally, we call such a mapping, $g_\phi : \mathcal{M} \times \mathcal{X} \to \mathcal{Q}$, a *meta-inference model*. This doubly-amortized inference procedure must be robust across varying marginals and evidence, generalizing over $\mathcal{M}$: a large set of sufficiently similar, previously *unseen* models.

We note that the choice of $p_{\mathcal{D}_i}(\boldsymbol{x})$ as input to g_ϕ is critical in practice. As in Eq. 7, a successful learning algorithm will learn generative models such as $p_{\theta_i}(\boldsymbol{x})$ or $p_{\theta_i}(\boldsymbol{x}, \boldsymbol{z})$ that match $p_{\mathcal{D}_i}(\boldsymbol{x})$. But similarly to the recent progress in wake-sleep (Hinton et al. 1995; Bornschein and Bengio 2014; Le et al. 2018), we found that using observations from the true marginal $p_{\mathcal{D}_i}(\boldsymbol{x})$ led to significantly more stable training. One may also consider alternate combinations of inputs for $p_{\mathcal{D}_i}(\boldsymbol{x})$, which we leave as future work.

Meta-Amortized Variational Bayes and Learning In certain settings, we are given a set of generative models $\{p_{\theta_i^*}(\boldsymbol{x}, \boldsymbol{z}), i \in \mathcal{I}\}$, where each model $p_{\theta_i^*}(\boldsymbol{x}, \boldsymbol{z})$ with known parameters captures a marginal distribution, $p_i(\boldsymbol{x}) \in \mathcal{M}_\mathcal{I}$. We can then immediately optimize Eq. 10 to obtain the optimal meta-inference model.

But in many cases the generative models are not known ahead of time, and therefore we must jointly learn $\{\theta_i, i \in \mathcal{I}\}$ along with the parameters of the meta-inference model, ϕ. To do so, we consider the objective,

$$\max_\phi \mathbb{E}_{p_{\mathcal{D}_i} \sim p_\mathcal{M}} \left[\max_{\theta_i} \mathcal{L}_{\phi, \theta_i}(p_{\mathcal{D}_i}) \right] \tag{11}$$

where the inner loss function is defined as:

$$\mathcal{L}_{\phi, \theta_i}(p_{\mathcal{D}_i}) = -D_{\mathrm{KL}}(p_{\mathcal{D}_i}(\boldsymbol{x}) g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x}) || p(\boldsymbol{z}) p_{\theta_i}(\boldsymbol{x}|\boldsymbol{z}))$$

and $p_{\mathcal{D}_i}(\boldsymbol{x}) g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})$ denotes the distribution defined implicitly by first sampling $\boldsymbol{x} \sim p_i(\boldsymbol{x})$, then sampling $\boldsymbol{z} \sim g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})$. We refer to this lower bound as the MetaELBO, and a VAE trained with this objective as the MetaVAE.

Lastly, as we did in Eq. 7, we can rewrite the MetaELBO to a more interpretable form. Similar to $f_\phi(\boldsymbol{x})$, our regressor $g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})$ can be represented as a conditional distribution, denoted $q_\phi(\boldsymbol{z}|p_{\mathcal{D}_i}, \boldsymbol{x}) = g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})(\boldsymbol{z})$. Then,

$$\begin{aligned} \mathcal{L}_{\phi, \theta}(p_{\mathcal{D}_i}) &= -D_{\mathrm{KL}}(p_{\mathcal{D}_i}(\boldsymbol{x}) q_\phi(\boldsymbol{z}|p_{\mathcal{D}_i}, \boldsymbol{x}) || p(\boldsymbol{z}) p_{\theta_i}(\boldsymbol{x}|\boldsymbol{z})) \\ &= -D_{\mathrm{KL}}(p_{\mathcal{D}_i}(\boldsymbol{x}) || p_{\theta_i}(\boldsymbol{x})) \\ &\quad - \mathbb{E}_{\boldsymbol{x} \sim p_{\mathcal{D}_i}(\boldsymbol{x})}[D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|p_{\mathcal{D}_i}, \boldsymbol{x}) || p_{\theta_i}(\boldsymbol{z}|\boldsymbol{x}))]. \end{aligned}$$

This form has a penalty term for each distribution $p_{\mathcal{D}_i}(\boldsymbol{x})$, encouraging the meta-amortized inference model to perform well across $p_{\mathcal{D}_i}(\boldsymbol{x})$ sampled from the meta-distribution $p_\mathcal{M}$. We note that if $\mathcal{M} = \{p_\mathcal{D}\}$, then $g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x}) = f_\phi(\boldsymbol{x})$, and the MetaELBO is equivalent to ELBO.

Interestingly, we find that the MetaVAE's learned representations transfer well to unseen downstream tasks at test time. We provide some intuition as to why this is the case. Samples from the corresponding marginal $p_{\mathcal{D}_i}$ help to lower the variance in the meta-inference network's inferred $\boldsymbol{z}$'s for each query point $\boldsymbol{x}$, regularizing the model's behavior to yield more robust representations.

Representing the Meta-Distribution

In Eq. 11, it is not clear how to represent a distribution $p_{\mathcal{D}_i}(\boldsymbol{x})$ as input if we parameterize $g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})$ as a neural network. One of the main insights from this work is to represent the marginal distribution as a finite set of samples,

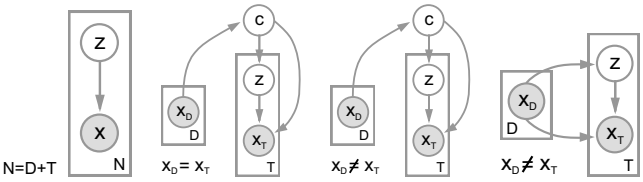
$$\mathcal{D}_i = \{\boldsymbol{x}_j \sim p_{\mathcal{D}_i}(\boldsymbol{x}) | j = 1, ..., N\} \tag{12}$$

or a *data set*. We can then use $\mathcal{D}_i$ to define an empirical analogue to $g_\phi(p_i, \boldsymbol{x})$, denoted as $\hat{g}_\phi : \mathcal{X}^N \times \mathcal{X} \to \mathcal{Q}$, which maps a data set with N samples and an observation to a posterior. Then, there is an equivalent analogue of Eq. 11 where a marginal, $p_{\mathcal{D}_i}(\boldsymbol{x})$ is replaced by a data set, $\mathcal{D}_i$.

Implementation Details In practice, for some dataset $\mathcal{D}_i$ and input $\boldsymbol{x}$, we implement the meta-inference model $g_\phi(\mathcal{D}_i, \boldsymbol{x}) = r_{\phi_2}(\mathrm{CONCAT}(\boldsymbol{x}, h_{\phi_1}(\mathcal{D}_i))$ where $\phi = \{\phi_1, \phi_2\}$. The "summary network" $h_{\phi_1}(\cdot)$ is a two-layer perceptron (MLP) that ingests each element in D_i independently and computes a summary representation using the mean. The "aggregation network" $r_{\phi_2}(\cdot)$ is a second two layer MLP that takes as input the concatenated summary and input. The corresponding i-th generative model $p_{\theta_i}(x|z)$ is parameterized by an MLP with identical architecture as $r_{\phi_2}(\cdot)$. ReLU nonlinearities were used between layers. For more complex image domains (such as NORB), we use three-layer convolutional networks instead of MLPs.

Related work

Rapid Adaptation through Meta-Learning. Among the rich body of work on meta-learning (Vinyals et al. 2016; Snell, Swersky, and Zemel 2017; Gordon et al. 2018), a common goal is to train models such that they will rapidly adapt to new, unseen classification tasks. Although the Neural Process (NP) (Garnelo et al. 2018; Kim et al. 2019) is similar to our work in that it derives predictions for new targets by conditioning the encoder network on a relevant *context set*, it models uncertainty over a distribution of *functions*. Another line of research formulates proper initialization as the workhorse of successful meta-learning (Finn, Abbeel, and Levine 2017; Grant et al. 2018). In many ways, our meta-amortized inference procedure can be thought of as learning a good initialization for an inference model on a new target distribution. However, these approaches are not directly comparable to ours because of their supervised nature.



(a) VAE (b) NS (c) VHE (d) MetaVAE

Figure 1: Plate diagrams comparing the MetaVAE to existing generative models. Critically, MetaVAE does not include a latent variable over models, $\boldsymbol{c}$.

Few-shot Generative Modeling. This branch of research aims to train generative models such that they will generalize to unseen distributions at test time given only a few examples. The focus has been on few-shot density estimation, with approaches ranging from the use of conditioning (Bartunov and Vetrov 2016) to nested optimization (Reed et al. 2017). Meta-inference however is not few-shot, and instead aims to learn transferable *representations* for downstream tasks rather than density estimation alone.

The most relevant prior works include the Neural Statistician (Edwards and Storkey 2016) (NS) and the Variational Homoencoder (Hewitt et al. 2018) (VHE), two very similar models that study inference over sets of observations. The VHE optimizes the following objective,

$$\mathbb{E}_{\boldsymbol{x},\mathcal{D}\sim p_{\mathcal{D}}}[\mathbb{E}_{q_\phi(\boldsymbol{c}|\mathcal{D})}[\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{c},\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{c})]] - D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{c},\boldsymbol{x})||p(\boldsymbol{z}|\boldsymbol{x})) - \frac{1}{N} D_{\mathrm{KL}}(q_\phi(\boldsymbol{c}|\mathcal{D})||p(\boldsymbol{c}))] \tag{13}$$

where $\mathcal{D} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\}$ is a set of N samples and $\boldsymbol{c}$ is a global latent variable. We note that if we view $\mathcal{D}$ as an approximation for a marginal distribution, then NS and VHE also serve as baselines that can perform doubly-amortized inference. Similar to our proposed inference model $\hat{g}_\phi(\mathcal{D}, \boldsymbol{x})$, the distribution $q(\boldsymbol{c}|\mathcal{D})$ in Eq. 13 ingests a data set. However, both the VHE and NS utilize a global variable $\boldsymbol{c}$ (isotropic Gaussian). We believe this constraint is *overly restrictive* in settings which require transferring to a diverse set of distributions, hurting generalization performance. Instead, the MetaVAE does not impose a distributional assumption on the different generative models, and shares a fixed meta-encoder network among separate decoders for each dataset. We find that this semi-parametric approach yields consistently better performance.

Demo: Clustering Mixtures of Gaussians

First, we present a simple clustering example to build intuition for meta-inference. Consider a standard VAE trained to capture a single mixture of two Gaussian (MoG) distributions $p_{\mathcal{D}}(\boldsymbol{x})$. Each component has isotropic covariance of 0.1 and mean drawn from the uniform distribution, $U(-5, 5)$. The two components are mixed evenly and assigned a label of 0 or 1. Then, inference $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ with $\boldsymbol{z} \in \{0, 1\}$ as a 1-D binary latent variable amounts to predicting which component $\boldsymbol{x}$ belongs to, of which the true cluster label is recoverable up to a permutation.

Now we introduce meta-inference for this task. Given that an inference model $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ of a VAE can learn to cluster data from a *specific* MoG, a meta-inference model $g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})$ should correspond to *a general-purpose clustering algorithm* that can separate out the components of any related, but previously unseen mixture distribution $p_{\mathcal{D}_i}$.

Concretely, we let each distribution $p_{\mathcal{D}_i}(\boldsymbol{x}) \sim p_{\mathcal{M}}$ be a MoG and train a MetaVAE amortized over N mixtures to assess how well it can predict $\boldsymbol{z} \in \{0, 1\}$ for a given $\boldsymbol{x}$ for an *unseen test distribution*. We measure this clustering accuracy on 1000 unseen but related MoGs sampled from the same meta-train distribution.

While the VAE has a clustering error of 27.9% due to cases where there is extreme overlap in mixture components, the MetaVAE has an error of 9.9% when $N = 50$. Moreover, larger N improved the model's performance (21.2% error with $N = 10$ and 15.8% error with $N = 20$) as expected. We include more details and a second study on clustering MNIST digits in the Appendix[1].

Demo: Inference for Classical Mechanics

For a second demonstration, we consider an introductory problem in classical mechanics: objects sliding down inclined planes. Here, we are given a physics simulator that models a box that faces friction with the plane. Each time the simulator runs, we see a new box with a different friction coefficient. The simulator then records the time it takes for the box to descend to the bottom of the plane. Each simulator has a different incline plane of length L and incline angle A, and our task is to infer the coefficient of friction ($\boldsymbol{z}$) from the observed descent time ($\boldsymbol{x}$) given a new simulator. Building on (Le, Baydin, and Wood 2016), we tackle this problem with "meta-compiled inference" and optimize:

$$\mathcal{L}_\phi = \mathbb{E}_{p_{\theta_i^*}\sim p_{\mathcal{M}}} \mathbb{E}_{\boldsymbol{x}\sim p_{\theta_i^*}(\boldsymbol{x})} \left[-g_\phi(\boldsymbol{z}|p_{\theta_i^*}, \boldsymbol{x})\right] \tag{14}$$

The meta-distribution $\mathcal{M}$ represents all possible simulators of planes with $L \in [1, 20]$ and $A \in [5, 85]$ degrees, and

[1] https://arxiv.org/pdf/1902.01950.pdf

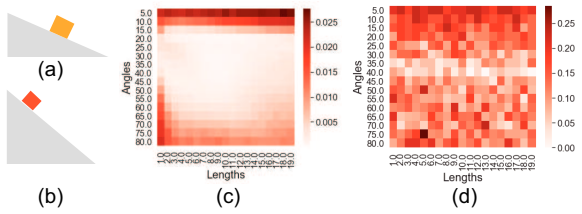


Figure 2: (a,b) Examples of planes with two lengths and angles. MSE between true and inferred friction for 304 simulators (lighter is better) using (c) MetaVAE and (d) VAE.

$p_{\theta_i^*}(\boldsymbol{x}, \boldsymbol{z})$ represents a fixed simulator. The marginal distribution, $p_{\theta_i^*}(\boldsymbol{x})$ is obtained by repeatedly simulating to build a data set $\mathcal{D}_i = \{\boldsymbol{x}\}$. Thus the empirical meta-inference model $\hat{g}_\phi(\mathcal{D}_i, \boldsymbol{x})$ takes the data set and the output of a single simulation $\boldsymbol{x}$ as input. We amortize over 25 simulators with $L \in \{2, 4, 6, 8, 10\}$ and $A \in \{20, 30, 40, 50, 60\}$, and model $\boldsymbol{z}$ as a continuous 1-D random variable (interpreted as friction). After training the MetaVAE, we measure the mean squared error between the true and inferred friction for unseen simulators from $\mathcal{M}$.

Despite seeing only 25 out of 304 simulators, the MetaVAE transfers well: we get less than 0.001 MSE for $A \in [20, 70]$ and $L \in [2, 20]$. A standard VAE trained on a single simulator ($L = 10$, $A = 45$) exhibits both much worse generalization performance and greater error overall (notice the scale in the legends).

Demo: Learning Distribution Statistics

Next, we explore whether the MetaVAE is capable of "meta-learning" the concept of a sufficient statistic for exponential families (Wainwright and Jordan 2008). Given a set of random samples, a sufficient statistic is a function that maps this set to a vector in $\mathbb{R}^d$. For the exponential families, where each family member has the form $p(x) \propto \exp(\theta \cdot \phi(x))$ for some parameter θ, this vector can be used to estimate the parameters of the distribution. In other words, the random samples (dataset) can be fully summarized by the sufficient statistic, without any loss of information. Now consider a *vector* of random variables $(x_1, \cdots, x_k)$, each distributed i.i.d from the same distribution with sufficient statistic $\phi(x_i)$. For exponential families, the sum $\sum_{i=1}^{k} \phi(x_i)$ is a sufficient statistic for the random vector. As an example, the number of successes is a sufficient statistic for a vector of i.i.d. Bernoulli, and the sample mean and variance are for a vector of Gaussians. With this intuition, we ask the following: having seen many realizations of random vectors from different exponential family distributions, can we learn a sufficient statistic for a new random vector that will be sufficient for estimating the parameters of its unseen, underlying distribution? We aim to use the MetaVAE's meta-inference network to learn this mapping. More precisely, the meta inference model $g_\phi(p_{\mathcal{D}_i}, \boldsymbol{x})$ should act (as a function of $\boldsymbol{x}$) as a sufficient statistic for an unseen distribution $p_{\mathcal{D}_i}$.

Data and Model Setup

In this experiment, we use Gaussian (fixed variance), log-normal (fixed variance), exponential, symmetric beta, Laplace (fixed location), and Weibull (fixed scale) as exponential families. We then construct a set $\mathcal{M}_\mathcal{I}$ of 20-D vectors of random variables where each component is i.i.d. distributed according to the same distribution. By construction, a random variable in this set will have only one free parameter, which can be found using the statistic learned by the meta-inference network. We further restrict $\mathcal{M}_\mathcal{I}$ by bounding the free parameter to be within a range (e.g. Gaussians with mean between -5 and 5).

After training, we measure how well we can infer the distributional parameters using the meta-inference model as a learned statistic for observations from unseen distributions. We compute the mean squared error (MSE) between the inferred and true parameters. We refer the reader to the Appendix[2] for more details.

Experiment Results

Single Exponential Family Each $p_{\mathcal{D}_i}(\boldsymbol{x}) \in \mathcal{M}$ is Gaussian with a mean sampled from $U(-5, 5)$. At test time, we measure inference quality on (1) new random vectors from $\mathcal{M}$ whose entries are distributed as Gaussians with unseen means sampled from $U(-5, 5)$, and (2) a larger meta-distribution by sampling means from $U(-20, 20)$. We find the MetaVAE successfully learns the mean of the underlying Gaussians. Interestingly, in Fig. 3(a), we find that the inference quality only decays near the boundary of the meta-distribution. We compare the MetaVAE to a VAE trained on one Gaussian distribution and find that doubly-amortizing increases the inference quality dramatically. Then we move to two new exponential families: we similarly construct 30 log-normal random vectors with means from $U(-2, 2)$ and 30 Exponential random vectors with rates sampled from $U(0, 3)$. Like above, Fig. 3(b,c) shows good performance of meta-inference over $\mathcal{M}$ in each case.

(a) Gaussian (b) Log-Normal (c) Exponential

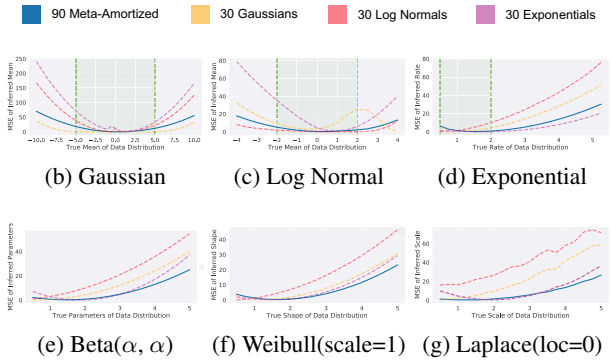
Figure 3: (a) MSE between the true and inferred mean as the true mean of $p_{\mathcal{D}_i}$ spans $[-10, 10]$. The green region shows the meta-distribution. The orange (dashed) line shows a singly-amortized VAE trained on a single $p_{\mathcal{D}_i}(\boldsymbol{x})$ with mean $[-1.2, 1.1]$ (randomly chosen) and the blue (solid) line shows the MetaVAE. (b,c) show the MSE between the true and inferred parameters. The orange line is a singly-amortized VAE trained on a randomly chosen distribution ($[-0.5, 1.8]$ for log-normal; $[1.4, 2.8]$ for exponential).

Many Exponential Families Finally, we amortize over many types of distributional families simultaneously: we

[2] https://arxiv.org/pdf/1902.01950.pdf

construct sets of 30 Gaussian, 30 log-normal, and 30 exponential random vectors (same bounds as above) to train a MetaVAE. This setup raises an interesting question: can we do inference for new random vectors comprised of *unseen members of the exponential family* (e.g. Weibull)?

We compare the performance a MetaVAE amortized over the 90 random vectors to 3 different (baseline) MetaVAEs, each of which is amortized over only 30 random vectors from one family (e.g. Gaussian). Below, Fig. 4(a-c) plot the MSE of inferred and true parameters for Gaussian, log-normal, and exponential (all of which are in $\mathcal{M}$). Due to the double-amortization gap, the best performing model is the MetaVAE amortized on random vectors only from that family. However, the 90-amortized MetaVAE only performs slightly worse, beating the remaining two baselines dramatically. Next, Fig. 4(d-f) show MSEs for three distributions not in $\mathcal{M}$: Weibull, Laplace, and Beta. The 90-amortized MetaVAE consistently outperforms all baselines.



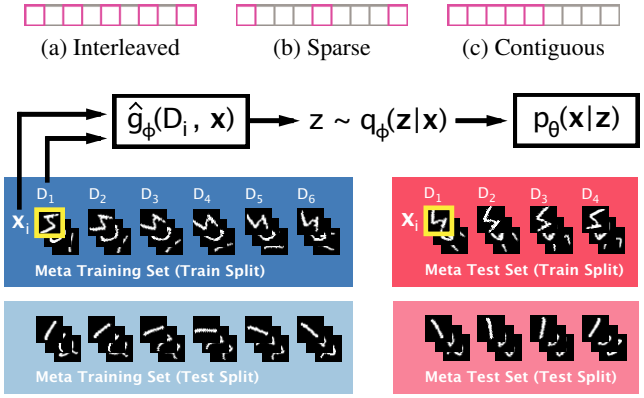
(b) Gaussian (c) Log Normal (d) Exponential

(e) Beta(α, α) (f) Weibull(scale=1) (g) Laplace(loc=0)

Figure 4: Comparison of a MetaVAE amortized over three members of the exponential family to MetaVAEs amortized over only a single member. Each subplot shows an unseen distribution from either the meta-distribution (b,c,d) or another exponential family (e,f,g).

Transformation-Invariance Experiments

To motivate the next set of experiments, imagine designing a scene understanding algorithm for a self-driving car. The video datasets used to train deep learning agents are typically collected in isolated settings, such as in large cities during favorable weather conditions. However, an agent deployed in the real world may face a variety of new settings such as paved roads in poorly-lit suburban areas. In such cases, we would hope the agent could abstract away unnecessary sources of variation, such as different lighting conditions, and act upon more salient characteristics in the scene (e.g. pedestrians) that it has seen previously during training. Inference in this scenario would mean learning representations that are "transferable," or invariant to nuisance transformations such as time of day. We take a step towards this goal as we study the MetaVAE for image distributions with explicit transformations, such as rotations or lighting.



(d) Meta-Inference Pipeline

Figure 5: (a-c) Three ways of defining the meta-training and meta-test splits; (b,c) pose a more difficult generalization challenge. (d) Overview of the doubly-amortized inference procedure. The meta-training set is used to train the MetaVAE (the test portion is to used to choose best parameters). The meta-test set is for evaluating the learned features, where the training portion is used to fit a linear classifier and the test portion is used to compute accuracy.

Datasets We study MNIST and NORB (LeCun et al. 2004), where we amortize over three axes of variation each (e.g. a range of camera angles or background lighting). Further, we vary how different variations are split into meta-training and meta-test sets, summarized in Fig. 5(a-c). For instance, we may train the MetaVAE only on images with bright backgrounds and evaluate on darker images. We consider three meta-splits: *interleaved*, where every other value in the range of possible transformations is selected; *sparse*, where half the number of values are chosen as in interleaved; *contiguous*, where we split the range in two "contiguous" halves and train only over the first half. Each meta-split is a different measure of transfer-ability.

Evaluation Metric We evaluate the latent representations on a downstream classification task. Having trained the empirical meta-inference model $\hat{g}_\phi(\mathcal{D}, \boldsymbol{x})$ using the meta-train set, we then embed observations from a distribution in the meta-test set. Each time we "embed" a test observation $\boldsymbol{x}$, we feed in a data set $\mathcal{D}$ of samples *from the meta-test set*. This way we construct a data set of latent features.

This feature set is split into a training and test subset. For both MNIST and NORB, each image has a corresponding label (e.g. digit or object class). Using the training portion (darker red in Fig. 5d) , we fit a logistic regression classifier on the representations to predict the labels and compute accuracy on the test subset (lighter red in Fig. 5d). Critically, logistic regression seeks the best linear split between classes in the latent space. For it to achieve good accuracy, such a linear division must already exist. Thus, we treat a higher classification accuracy as a more transferable, invariant representation, as in (Berthelot et al. 2018).

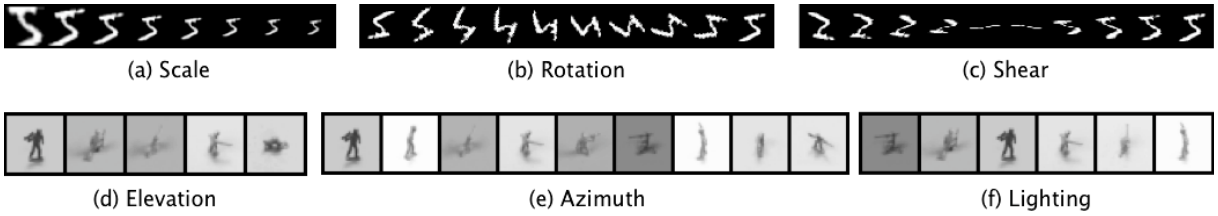


Figure 6: Examples of interpolating across three transformations each for MNIST and Small NORB. Notice that for NORB (unlike MNIST), other transformations are not held constant as we vary an individual axis.

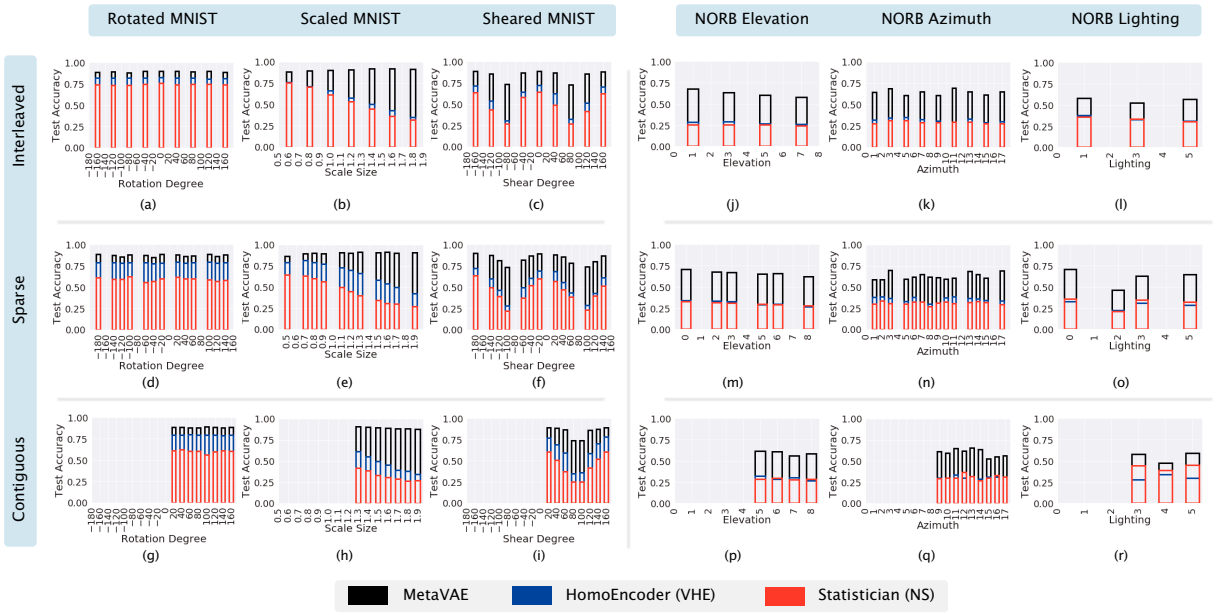


Figure 7: Classification Accuracy on Transformed MNIST and Small NORB for three different splits: interleaved, sparse, and contiguous. Each subfigure shows the prediction accuracy on the test set of held out transformations — gaps represent the values used in training the amortized generative model. We compare the performance of MetaVAE (black), the homoencoder (blue) and the statistician (red) and find appealing results for our proposed model.

Baselines We compare the performance of MetaVAE against two baselines: the Neural Statistician (NS), a hierarchical VAE which models sets of observations with a global latent variable; and the Variational HomoEncoder (VHE), a more computationally-efficient variant of NS. To ensure a fair comparison, we use the same hyperparameters and architectures across all models. See Appendix[3] for details.

Transformed MNIST

Dataset Construction We artificially impose three axes of variations on MNIST digits. We transform each image with 18 *rotations* (-180 to 180 by 20 degrees), 15 *scales* (50% to 200% original size by 10%), and 18 *skews* (-180 to 180 by 20 degrees). See Fig. 6(a-c) for an example for a single digit. For each axes of variation, the other two are held constant e.g. skew and size are constant when varying rotation.

[3]https://arxiv.org/pdf/1902.01950.pdf

Results We find consistent evidence that MetaVAE features outperform both VHE and NS features across all settings, often by a significant margin. In particular, VHE and NS have decaying performance as scale increases to 2.0. Similarly, for extreme shear values near -80 and 80 degrees where the image is nearly flat (see Fig. 6c), VHE and NS again suffer greatly in performance. However, MetaVAE features transfer better: we do not notice a drop in accuracy as scale increases and the effect of significant shearing is more gradual. This suggests that MetaVAE has learned some invariances to transformations that NS and VHE lack.

Small NORB

Dataset Construction The NORB dataset contains grayscale images of real world toys belonging to five classes: animals, humans, airplanes, trucks, and cars. The objects were imaged under 6 *lighting* conditions, 9 *elevations* (30 to 70 degrees every 5 degrees), and 18 *azimuths* (0 to 340 every 20 degrees). Unlike the MNIST dataset, extraneous transformations are *not* held constant as one

transformation is varied. For example, as Fig. 6(f) shows, the azimuth and elevation (randomly) change as we vary lighting. This design, while more difficult to amortize, is more realistic in real world datasets where it is too expensive to collect data holding all other variables constant.

Results The MetaVAE representations outperform those of VHE and NS by 10 to 35% accuracy. Overall, we notice accuracies are much lower in NORB than in MNIST, which is likely due to the complexity of learning real world image distributions and randomness introduced by variations in extraneous transformations. We note that the strong performance of the MetaVAE despite varying transformations is promising support for our approach to meta-amortization, suggesting that the MetaVAE is able to ignore irrelevant signals while capturing the principal axes of variation.

Discussion

Experimental Analysis

We aim to quantitatively measure the intuition that amortizing over a family of transformations should yield representations that are invariant to that transformation. For example, how much does the representation change as we alter the rotation in MNIST from -180 to 180, or interpolate the background from dark to light in NORB?

To investigate, we use a MetaVAE amortized over a family of transformations (e.g. interleaved rotations) and compare the average L_2 distance between the learned representation of a base (default) image and those of every rotated image. As a baseline, we compare this distance to the average L_2 distance of a separate family of transformations (e.g. scale) that this MetaVAE was not amortized over (e.g. having only seen different rotations during training). Table 1 shows the distances for MNIST and NORB. Consistently, the lowest distances belong to the class of transformations that the MetaVAE was amortized over, which supports the intuition about learning invariances.

Model Dataset	Rotation	Scale	Skew
Rotated MNIST	**1.65**	4.44	4.09
Scaled MNIST	5.44	**2.16**	4.92
Skewed MNIST	3.79	4.89	**1.47**
Model Dataset	Elevation	Azimuth	Lighting
NORB Elevation	**0.39**	1.16	1.27
NORB Azimuth	1.42	**0.44**	1.26
NORB Lighting	1.69	1.27	**0.26**

Table 1: L_2 distances between MetaVAE representations. Each row indicates the datasets used for training; each column indicates the datasets used to compute representations.

Role of Flexible Global Prior

Next, we investigate the hypothesis that a more flexible prior over the global latent variable may give the model the expressivity necessary for better performance on downstream classification tasks. Specifically, we compare the MetaVAE against the VHE equipped with the VampPrior (VP) (Tomczak and Welling 2017), which is a learned prior $p(\boldsymbol{c})$, on additional MNIST and NORB experiments. We use default settings from the reference VP implementation (500 components, 0.05 mean, and 0.01 std)[4]:

Dataset	MetaVAE	VHE+VP	VHE
Rotated MNIST	**0.885**	0.830	0.793
Scaled MNIST	**0.893**	0.767	0.463
Sheared MNIST	**0.844**	0.679	0.602
Dataset	MetaVAE	VHE+VP	VHE
NORB Elevation	**0.601**	0.337	0.309
NORB Azimuth	**0.592**	0.313	0.286
NORB Lighting	**0.548**	0.357	0.306

Table 2: Downstream classification accuracy on MNIST and NORB datasets. The MetaVAE outperforms all relevant baselines, including the VHE with a learned prior, $p(\boldsymbol{c})$.

Table 2 shows that while VHE+VP outperforms VHE, its performance is consistently lower than MetaVAE. Additionally, we note that VP incurs a large computational cost – VHE+VP uses 5.1M more parameters than the VHE due to parameterizing "pseudoinputs", whereas the MetaVAE achieves model flexibility with no additional parameters. This highlights our primary contribution: learning without an explicit prior is important for the meta-inference problem where test tasks can be quite different than training tasks.

Conclusion

In summary, we developed an inference algorithm for a *family* of probabilistic models. We introduced a meta-amortized inference paradigm and a new generative model, the MetaVAE. Through experiments on MNIST and Small NORB, we showed that the MetaVAE learned transferable representations that generalize well across similar data distributions in downstream tasks. We provide reference implementations in PyTorch, and the codebase for this work is open-sourced at https://github.com/mhw32/meta-inference-public. Future work could consider applications of meta-inference in video prediction (Ramanathan et al. 2015).

Acknowledgements

We are thankful to Aditya Grover, Daniel Levy, Michael Xie for insightful discussions and feedback. KC is supported by NSF GRFP, Stanford Graduate Fellowship, and Qualcomm. MW is supported by NSF GRFP. This research was funded by NSF(#1651565, #1522054, #1733686), ONR (N00014-19-1-2145, N00014-16-1-2007), AFOSR (FA9550-19-1-0024), AFRL (FA8650-18-C-7826) and Amazon AWS.

[4] https://github.com/jmtomczak/vae_vampprior

References

Bartunov, S., and Vetrov, D. P. 2016. Fast adaptation in generative models with generative matching networks. *arXiv preprint arXiv:1612.02192*.

Berthelot, D.; Raffel, C.; Roy, A.; and Goodfellow, I. 2018. Understanding and improving interpolation in autoencoders via an adversarial regularizer. *arXiv preprint arXiv:1807.07543*.

Bornschein, J., and Bengio, Y. 2014. Reweighted wake-sleep. *arXiv preprint arXiv:1406.2751*.

Brock, A.; Donahue, J.; and Simonyan, K. 2018. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*.

Edwards, H., and Storkey, A. 2016. Towards a neural statistician. *arXiv preprint arXiv:1606.02185*.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 1126–1135. JMLR. org.

Garnelo, M.; Schwarz, J.; Rosenbaum, D.; Viola, F.; Rezende, D. J.; Eslami, S.; and Teh, Y. W. 2018. Neural processes. *arXiv preprint arXiv:1807.01622*.

Gershman, S., and Goodman, N. 2014. Amortized inference in probabilistic reasoning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 36.

Gordon, J.; Bronskill, J.; Bauer, M.; Nowozin, S.; and Turner, R. E. 2018. Decision-theoretic meta-learning: Versatile and efficient amortization of few-shot learning. *arXiv preprint arXiv:1805.09921*.

Grant, E.; Finn, C.; Levine, S.; Darrell, T.; and Griffiths, T. 2018. Recasting gradient-based meta-learning as hierarchical bayes. *arXiv preprint arXiv:1801.08930*.

Hastings, W. K. 1970. Monte carlo sampling methods using markov chains and their applications.

Hewitt, L. B.; Nye, M. I.; Gane, A.; Jaakkola, T.; and Tenenbaum, J. B. 2018. The variational homoencoder: Learning to learn high capacity generative models from few examples. *arXiv preprint arXiv:1807.08919*.

Hinton, G. E.; Dayan, P.; Frey, B. J.; and Neal, R. M. 1995. The" wake-sleep" algorithm for unsupervised neural networks. *Science* 268(5214):1158–1161.

Jordan, M. I.; Ghahramani, Z.; Jaakkola, T. S.; and Saul, L. K. 1999. An introduction to variational methods for graphical models. *Machine learning* 37(2):183–233.

Kim, H.; Mnih, A.; Schwarz, J.; Garnelo, M.; Eslami, A.; Rosenbaum, D.; Vinyals, O.; and Teh, Y. W. 2019. Attentive neural processes. *arXiv preprint arXiv:1901.05761*.

Kingma, D. P., and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Klingler, S.; Wampfler, R.; Käser, T.; Solenthaler, B.; and Gross, M. H. 2017. Efficient feature embeddings for student classification with variational auto-encoders. In *EDM*.

Le, T. A.; Baydin, A. G.; and Wood, F. 2016. Inference compilation and universal probabilistic programming. *arXiv preprint arXiv:1610.09900*.

Le, T. A.; Kosiorek, A. R.; Siddharth, N.; Teh, Y. W.; and Wood, F. 2018. Revisiting reweighted wake-sleep. *arXiv preprint arXiv:1805.10469*.

LeCun, Y.; Huang, F. J.; Bottou, L.; et al. 2004. Learning methods for generic object recognition with invariance to pose and lighting. In *CVPR (2)*, 97–104.

Mao, C.; Yao, L.; Pan, Y.; Luo, Y.; and Zeng, Z. 2018. Deep generative classifiers for thoracic disease diagnosis with chest x-ray images. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 1209–1214. IEEE.

Oord, A. v. d.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; and Kavukcuoglu, K. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Ramanathan, V.; Tang, K.; Mori, G.; and Fei-Fei, L. 2015. Learning temporal embeddings for complex video analysis. In *Proceedings of the IEEE International Conference on Computer Vision*, 4471–4479.

Reed, S.; Chen, Y.; Paine, T.; Oord, A. v. d.; Eslami, S.; Rezende, D.; Vinyals, O.; and de Freitas, N. 2017. Few-shot autoregressive density estimation: Towards learning to learn distributions. *arXiv preprint arXiv:1710.10304*.

Segler, M. H.; Kogej, T.; Tyrchan, C.; and Waller, M. P. 2017. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science* 4(1):120–131.

Snell, J.; Swersky, K.; and Zemel, R. 2017. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, 4077–4087.

Tomczak, J. M., and Welling, M. 2017. Vae with a vampprior. *arXiv preprint arXiv:1705.07120*.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D.; et al. 2016. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, 3630–3638.

Wainwright, M. J., and Jordan, M. I. 2008. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning* 1(1–2):1–305.

Yildirim, I. 2014. From perception to conception: learning multisensory representations.