Noname manuscript No.
(will be inserted by the editor)

Arbitrary Conditional Inference in Variational Autoencoders via Fast Prior Network Training

Ga Wu · Justin Domke · Scott Sanner

Received: date / Accepted: date

Abstract Variational Autoencoders (VAEs) are a popular generative model, but one in which conditional inference can be challenging. If the decomposition into query and evidence variables is fixed, conditionally trained VAEs provide an attractive solution. However, to efficiently support arbitrary queries over pre-trained VAEs when the query and evidence are not known in advance, one is generally reduced to MCMC sampling methods that can suffer from long mixing times. In this paper, we propose an idea of efficiently training small conditional prior networks to approximate the latent distribution of the VAE after conditioning on an evidence assignment; this permits generating query samples *without* retraining the full VAE. We experimentally evaluate three variations of conditional prior networks showing that (i) they can be quickly optimized for different decompositions of evidence and query and (ii) they quantitatively and qualitatively outperform existing state-of-the-art methods for conditional inference in pre-trained VAEs.

Keywords Variational Autoencoder · Conditional Inference · Prior Network

1 Introduction

Variational Autoencoders (VAEs) [10] are a popular deep generative model with numerous extensions including variations for planar flow [15], inverse autoregressive flow [9], importance weighting [1], ladder networks [14], and discrete latent spaces [17] to name just a few. Unfortunately, existing methods for conditional inference in VAEs rely on an *a priori* fixed decomposition of evidence and query and can thus be prohibitively slow for arbitrary queries. However, the ability to make fast arbitrary queries is critical for tasks such as inference of occluded portions of an image, where one does not know the occluded portion (query) and observed portion

Ga Wu · Scott Sanner
Department of Mechanical and Industrial Engineering, University of Toronto
E-mail: {wuga, ssanner}@mie.utoronto.ca

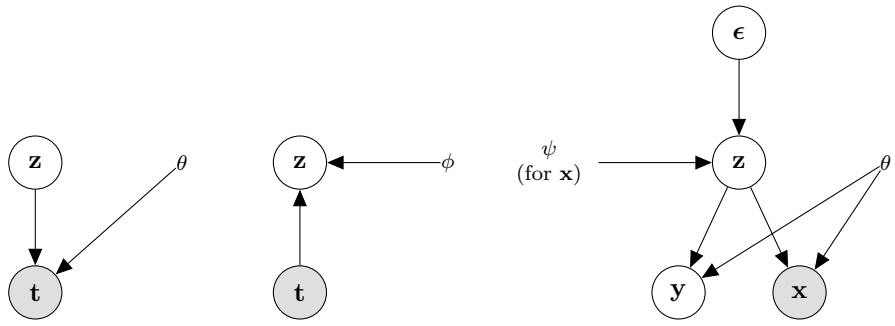
Justin Domke
College of Computing and Information Sciences, University of Massachusetts
E-mail: domke@cs.umass.edu

(evidence) in advance. Both Conditional VAEs (CVAEs) [21] as well as extensions made in Bottleneck Conditional Density Estimation (BCDE) [20] require full VAE training for a fixed decomposition of query and evidence – this is computationally impractical when VAE training alone can take over one day of computation time. Alternatively, Markov Chain Monte Carlo methods such as Hamiltonian Monte Carlo (HMC) [4, 3] are difficult to adapt to these problems and empirically suffer from long mixing times.

To remedy the limitations of existing methods for conditional inference in VAEs, we aim to approximate the distribution over the latent variables after conditioning on an evidence assignment through a variational Bayesian methodology. In doing this, we reuse the decoder of the VAE and show that the error of the distribution over query variables is controlled by that over latent variables. This avoids the computational expense of re-training the decoder as done by the CVAE and BCDE approaches. We term the network that generates the conditional latent distribution the *conditional prior network* as it only takes Gaussian noise as input. We remark that the conditional prior networks that generate the conditional latent distribution correctly approximate the query density simply by "warping" the standard Gaussian distribution of a VAE through a small set of parameters and optimization epochs.

We experiment with two conditional prior network alternatives: Gaussian variational inference via a linear transform (GVI) and Normalizing Flows (NF). We also provide comparison to a fully connected network (FCN), which suffers from some technical and computational issues but provides a useful point of reference for experimental comparison purposes. Overall, our results show that the GVI and NF variants of conditional prior networks can be optimized quickly for arbitrary decompositions of query and evidence and compare favorably against a ground truth provided by rejection sampling for low latent dimensionality. For high dimensionality, we observe that HMC often fails to mix despite our systematic efforts to tune its parameters and hence demonstrates poor performance compared to conditional prior networks in both quantitative and qualitative evaluation.

In summary, an outline of our novel contributions follows. While previous works have examined conditional training in VAEs [21, 20], no paper has currently taken our fast and simple approach of freezing the decoder of a pre-trained VAE and efficiently training a relatively small conditional prior network given evidence; we remark that doing so requires us to derive a novel Conditional ELBO (C-ELBO) training objective that extends the well-known ELBO for training VAEs. We compare the performance of various classes of conditional prior networks ranging from GVI to NF to FCNs (as outlined above) on a variety of datasets vs. MCMC and a fixed-point alternation approach suggested by Rezende *et al* [16]. Our results first show that Rezende's method simply does not work well with sparse evidence ($< 40\%$ of variables observed), whereas MCMC easily outperforms it, hence we focus on MCMC methods for further comparison. We then proceed to our main results showing that conditional training of conditional prior networks is very fast (a few seconds compared to 24 hours or more for full VAE training in some cases) and demonstrate that conditional prior networks based on GVI and NF generally outperform Hamiltonian MCMC across a variety of quantitative and qualitative metrics. Overall, our work suggests that our simple, intuitive, and fast conditional prior network training allows high-performance conditional inference for arbitrary queries in pre-trained VAEs and offers a novel and efficient alternative to state-of-the-art methods including MCMC.



(a) $p_\theta(\mathbf{z}, \mathbf{t})$ (b) $q_\phi(\mathbf{z}|\mathbf{t})$ (c) $q_{\psi,\theta}(\boldsymbol{\epsilon}, \mathbf{z}, \mathbf{x}, \mathbf{y})$, $\mathbf{t} = (\mathbf{x}, \mathbf{y})$

Fig. 1: Graphical model of the proposed framework. (a) Decoder $p_\theta(\mathbf{z}, \mathbf{t}) = p(\mathbf{z})p_\theta(\mathbf{t}|\mathbf{z})$, representing the generative model. This is exactly the standard VAE model. (b) Encoder $q_\phi(\mathbf{z}|\mathbf{t})$. Since exact maximum-likelihood learning is intractable, VAE training uses this to bound the likelihood using the ELBO (Eq. 2). (c) Inference with conditional prior network: $q_{\psi,\theta}(\boldsymbol{\epsilon}, \mathbf{z}, \mathbf{x}, \mathbf{y}) = q(\boldsymbol{\epsilon})q_\psi(\mathbf{z}|\boldsymbol{\epsilon})p_\theta(\mathbf{x}, \mathbf{y}|\mathbf{z})$. This re-uses the decoder parameters θ, but ψ is optimized (for the particular input $\mathbf{x}$) by the C-ELBO (Eq. 5). Here, $\mathbf{t}$ is "split" as $\mathbf{t} = (\mathbf{x}, \mathbf{y})$.

2 Background

2.1 Variational Auto-encoders

One way to define an expressive generative model $p_\theta(\mathbf{t})$ is to introduce latent variables $\mathbf{z}$ as outlined in the latent generative model of Fig. 1(a). Variational Auto-Encoders (VAEs) [10] model $p(\mathbf{z})$ as a simple fixed Gaussian distribution. Then, for real $\mathbf{t}$, $p_\theta(\mathbf{t}|\mathbf{z})$ is a Gaussian with the mean determined by a "decoder" network as

$$p_\theta(\mathbf{t}|\mathbf{z}) = \mathcal{N}(\mathbf{t}; \text{Decoder}_\theta(\mathbf{z}), \sigma^2 I). \tag{1}$$

If $\mathbf{t}$ is binary, a product of independent Bernoulli's is parameterized by a sigmoidally transformed decoder. If the decoder network has high capacity, the marginal distribution $p_\theta(\mathbf{t})$ can represent a wide range of distributions. In principle, one might wish to train such a model by (regularized) maximum likelihood. Unfortunately, the marginal $p_\theta(\mathbf{t})$ is intractable. However, a classic idea [19] is to use variational inference to lower-bound it. For any distributions p_θ and q_ϕ,

$$\log p_\theta(\mathbf{t}) = \log \int_{\mathbf{z}} p_\theta(\mathbf{t}, \mathbf{z}) d\mathbf{z} = \underbrace{\mathbb{E}_{q_\phi(\mathbf{Z})} \log \frac{p_\theta(\mathbf{Z}, \mathbf{t})}{q_\phi(\mathbf{Z})}}_{\text{ELBO}[q_\phi(\mathbf{Z}) \| p_\theta(\mathbf{Z}, \mathbf{t})]} + KL[q_\phi(\mathbf{Z}) || p_\theta(\mathbf{Z}|\mathbf{t})]. \tag{2}$$

Since the KL-divergence is non-negative, the "evidence lower bound" (ELBO) lower bounds $\log p_\theta(\mathbf{t})$. Thus, as a surrogate to maximizing the likelihood over θ one can maximize the ELBO over θ and ϕ simultaneously.

VAEs define $q_\phi(\mathbf{z})$ as the marginal of $q(\mathbf{t})q_\phi(\mathbf{z}|\mathbf{t})$ where $q(\mathbf{t})$ is simple and fixed and $q_\phi(\mathbf{z}|\mathbf{t}) = \mathcal{N}(\mathbf{z}; \text{Encoder}_\phi(\mathbf{t}))$ is a Gaussian with a mean and covariance both determined by an "encoder" network; this is depicted in Figure 1(b).

Algorithm 1 Conditional Inference via Conditional Prior Networks.

Input (a) Pre-trained VAE $p(\mathbf{z})p_\theta(\mathbf{t}|\mathbf{z})$ with $p_\theta(\mathbf{t}|\mathbf{z})$ based on $\text{Decoder}_\theta(\mathbf{z})$. (Encoder ignored.) (b) Single evidence $\mathbf{x}$ (any subset of $\mathbf{t}$) for which to predict query $\mathbf{y}$. (Rest of $\mathbf{t}$.)

Optimize Define $q(\boldsymbol{\epsilon})q_\psi(\mathbf{z}|\boldsymbol{\epsilon})$ with $q_\psi(\mathbf{z}|\boldsymbol{\epsilon})$ based on $\text{Prior}_\psi(\boldsymbol{\epsilon})$. Find ψ to maximize C-ELBO$[q_\psi(\mathbf{Z})\|p_\theta(\mathbf{Z},\mathbf{x})]$ (Defined in Theorem 2). Estimate stochastic gradients by drawing random $\boldsymbol{\epsilon} \sim q(\boldsymbol{\epsilon})$ and using the reparameterization trick.

Predict Draw a sample $\{\mathbf{z}_m\}_{m=1}^M \sim q_\psi(\mathbf{z})$ by setting $\mathbf{z}_m = \text{Prior}_\psi(\boldsymbol{\epsilon}_m)$ for $\boldsymbol{\epsilon}_m \sim q(\boldsymbol{\epsilon})$. Predict $p_\theta(\mathbf{y}|\mathbf{x}) \approx \frac{1}{M}\sum_{m=1}^M p_\theta(\mathbf{y}|\mathbf{z}_m)$. (Justified since the optimization phase tightened a bound (Lemma 1) on the divergence between $\int q_\psi(\mathbf{z})p_\theta(\mathbf{y}|\mathbf{z})d\mathbf{z}$ and $p_\theta(\mathbf{y}|\mathbf{x})$.)

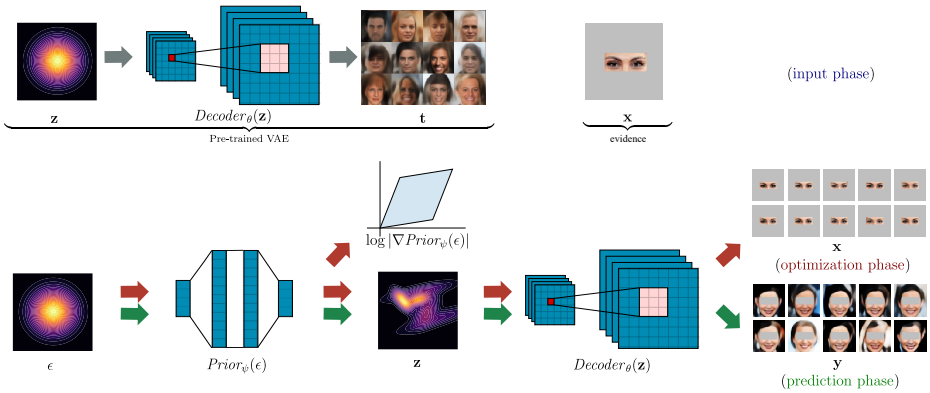


Fig. 2: Proposed conditional Prior network framework for conditional inference with Variational Auto-encoders. Arrow and text colors are aligned with the description in Algorithm 1, where the full image $\mathbf{t} = (\mathbf{x}, \mathbf{y})$. (**input phase**) We are provided with a pre-trained VAE and evidence $\mathbf{x}$ of observed image pixels. (**optimization phase**) We optimize the Prior network via stochastic gradient descent to learn a modulated $\mathbf{z}$ predictive of evidence $\mathbf{x}$. (**prediction phase**) Once the Prior network has been learned, we use it to generate samples of the query $\mathbf{y}$ conditioned on $\mathbf{x}$.

2.2 The conditional inference problem

In this paper, we assume a VAE has been pre-trained. Then, at test time, some arbitrary subset $\mathbf{x}$ of $\mathbf{t}$ is observed as evidence, and the goal is to predict the distribution of the non-observed query $\mathbf{y}$ where the decomposition $\mathbf{t} = (\mathbf{x}, \mathbf{y})$ is unpredictable. If this decomposition of $\mathbf{t}$ into evidence and query variables is *fixed* and known ahead of time, a natural solution is to train an explicit conditional model, the approach previously mentioned that is taken by CVAEs [21] as well as BCDE [20]. However, methods that would train a full conditional CVAE or BCDE model for each possible query decomposition $\mathbf{t} = (\mathbf{x}, \mathbf{y})$ can be infeasible considering that large VAEs (such as the ones we work with in this article) can take longer than a day to train. In this work, we focus on supporting queries with *arbitrary* evidence for pre-trained VAEs, where conditional training and inference completes in seconds *per* query. We next describe the framework that allows us to achieve such fast conditional inference.

2.3 Image completion as a conditional inference task

While our overall conditional inference approach is intended for any VAE (image-oriented or not), one particularly relevant application for conditional inference is the task of image completion. Specifically, the image completion task aims to restore the missing parts of occluded images. Many powerful algorithms [23, 6, 27, 26] can produce image reconstructions that humans cannot distinguish in quality from the original uncorrupted images. For example, work on the Deep Image Prior (DIP) [23] conducts image inpainting through a U-net architecture [18] with parameter prior optimization, whereas GLCIC [6] use Generative Adversarial Networks (GANs) [5].

All of these aforementioned algorithms implicitly assume that there exists a single prediction for each restoration task, which is ideal for image in-painting tasks where we aim to edit or remove objects from images. However, this assumption becomes questionable when most of the image is inaccessible – in this setting, multiple diverse completions may visually cohere with the observed evidence and thus it may be desirable to produce a variety of samples of such completions. In this sense, conditional probabilistic inference that can estimate the query (i.e., occluded image) distribution can be highly advantageous in this scenario. This motivates our proposed conditional inference VAE model for fast generation of diverse completion samples given a high quality pre-trained deep generative model.

3 Conditional Inference on Variational Auto-encoders

We now turn to the details of our conditional inference framework. We assume we have pre-trained a VAE $p_\theta(\mathbf{t}|\mathbf{z})$ and we now wish to approximate the distribution $p_\theta(\mathbf{y}|\mathbf{x})$, where $\mathbf{x}$ is some arbitrary new "test" input (i.e., evidence) not known at VAE training time.

Unfortunately, exact inference is difficult, since computing $p_\theta(\mathbf{y}|\mathbf{x})$ exactly would require marginalizing out $\mathbf{z}$. Consequently, for each decomposition of variables $\mathbf{t}$ into query $\mathbf{y}$ and evidence $\mathbf{x}$, we propose the idea of efficiently training a small conditional prior network $q_\psi(\mathbf{z}|\boldsymbol{\epsilon})$ to approximate $p_\theta(\mathbf{z}|\mathbf{x})$ by leveraging the pre-trained VAE with frozen weights for $p_\theta(\mathbf{t}|\mathbf{z})$. Technical details of this Conditional ELBO (C-ELBO) training method will follow.

Then, given this small conditional prior network $q_\psi(\mathbf{z}|\boldsymbol{\epsilon})$ efficiently trained *for* evidence $\mathbf{x}$, we can easily generate samples of the intended conditional distribution $p_\theta(\mathbf{y}|\mathbf{x})$. Specifically, we first use $q_\psi(\mathbf{z}|\boldsymbol{\epsilon})$ to sample $\mathbf{z}$ and then use the part of the original pre-trained VAE decoder for $p_\theta(\mathbf{y}|\mathbf{z})$ to sample $\mathbf{y}$ given $\mathbf{z}$.

The overall graphical model for this framework is shown in Figure 1(c). However, to make this abstract framework more concrete, we summarize the approach in Algorithm 1 and Figure 2. In the following subsections, we proceed to show the detailed technical derivation of this proposed framework.

3.1 Exploiting Factorization in the Output

To begin our derivation, we first need to establish conditional independence of $\mathbf{x}$ and $\mathbf{y}$ given $\mathbf{z}$. One helpful property comes from the fact that in a VAE, the

conditional distribution over the output (Eq. 1) has a diagonal covariance, which leads to the following decomposition:

Observation 1 The distribution of a VAE can be factorized as $p_\theta(\mathbf{x}, \mathbf{y}, \mathbf{z}) = p(\mathbf{z})p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{y}|\mathbf{z})$.

Since $\mathbf{x}$ and $\mathbf{y}$ are conditionally independent given $\mathbf{z}$, the conditional of $\mathbf{y}$ given $\mathbf{x}$ can be written as

$$p_\theta(\mathbf{y}|\mathbf{x}) = \int_{\mathbf{z}} p_\theta(\mathbf{z}, \mathbf{y}|\mathbf{x})p_\theta d\mathbf{z} = \int_{\mathbf{z}} p_\theta(\mathbf{z}|\mathbf{x})p_\theta(\mathbf{y}|\mathbf{z})d\mathbf{z}. \tag{3}$$

Here, $p_\theta(\mathbf{y}|\mathbf{z})$ can easily be evaluated or simulated. However $p_\theta(\mathbf{z}|\mathbf{x})$ is much more difficult to work with since it involves "inverting" the decoder. This factorization can also be exploited by Markov chain Monte Carlo methods (MCMC), such as Hamiltonian Monte Carlo (HMC) [4,3]. In this case, it allows the Markov chain to be defined over $\mathbf{z}$ alone, rather than $\mathbf{z}$ and $\mathbf{y}$ together. That is, one can use MCMC to attempt sampling from $p_\theta(\mathbf{z}|\mathbf{x})$, and then draw exact samples from $p_\theta(\mathbf{y}|\mathbf{z})$ just by evaluating the decoder network at each of the samples of $\mathbf{z}$. The experiments using MCMC in Section 4 use this strategy.

3.2 Variational Inference Bounds

The basic idea of variational inference (VI) is to posit some distribution q_ψ, and optimize ψ to make it match the target distribution as closely as possible. So, in principle, the goal of VI would be to minimize $KL[q_\psi(\mathbf{Y})\|p_\theta(\mathbf{Y}|\mathbf{x})]$. For an arbitrary distribution q_ψ this divergence would be difficult to work with due to the need to marginalize out $\mathbf{z}$ in p_θ as in Eq. 3.

However, if q_ψ is chosen carefully, then the above divergence can be upper-bounded by one defined directly over $\mathbf{Z}$. Specifically, we will choose q_ψ so that the dependence of $\mathbf{y}$ on $\mathbf{z}$ under q_ψ is the same as under p_θ (both determined by the "decoder").

Lemma 1. *Suppose we choose* $q_\psi(\mathbf{z}, \mathbf{y}) = q_\psi(\mathbf{z})p_\theta(\mathbf{y}|\mathbf{z})$. *Then*

$$KL[q_\psi(\mathbf{Y})\|p_\theta(\mathbf{Y}|\mathbf{x})] \leq KL[q_\psi(\mathbf{Z})\|p_\theta(\mathbf{Z}|\mathbf{x})]. \tag{4}$$

Proof of Lemma 1. To show this, we first note that the joint divergence over $\mathbf{Y}$ and $\mathbf{Z}$ is equivalent to one over $\mathbf{Z}$ only.

$$\begin{aligned} KL[q_\psi(\mathbf{Y}, \mathbf{Z})\|p_\theta(\mathbf{Y}, \mathbf{Z}|\mathbf{x})] &= KL[q_\psi(\mathbf{Z})\|p_\theta(\mathbf{Z}|\mathbf{x})] + KL[q_\psi(\mathbf{Y}|\mathbf{Z})\|p_\theta(\mathbf{Y}|\mathbf{Z}, \mathbf{x})] \\ &\quad \text{by the chain rule of KL-divergence} \\ &= KL[q_\psi(\mathbf{Z})\|p_\theta(\mathbf{Z}|\mathbf{x})] + KL[q_\psi(\mathbf{Y}|\mathbf{Z})\|p_\theta(\mathbf{Y}|\mathbf{Z})] \\ &\quad \text{since } \mathbf{Y} \perp \mathbf{X}|\mathbf{Z} \text{ in both } q_\psi \text{ and } p_\theta \\ &= KL[q_\psi(\mathbf{Z})\|p_\theta(\mathbf{Z}|\mathbf{x})] \\ &\quad \text{since } q_\psi(\mathbf{y}|\mathbf{z}) = p_\theta(\mathbf{y}|\mathbf{z}) \end{aligned}$$

Then, the result follows just from observing (again by the chain rule of KL-divergence) that

$$KL[q_\psi(\mathbf{Y})\|p_\theta(\mathbf{Y}|\mathbf{x})] \leq KL[q_\psi(\mathbf{Y}, \mathbf{Z})\|p_\theta(\mathbf{Y}, \mathbf{Z}|\mathbf{x})].$$

□

The result follows from using the chain rule of KL-divergence [2] to bound the divergence over $\mathbf{y}$ by the divergence jointly over $\mathbf{y}$ and $\mathbf{z}$. Then the common factors in q_ψ and p_θ mean this simplifies into a divergence over $\mathbf{z}$ alone.

Given this Lemma, it makes sense to seek a distribution q_ψ such that the divergence on the right-hand side of Eq. 4 is as low as possible. To minimize this divergence, consider the decomposition

$$\log p_\theta(\mathbf{x}) = \underbrace{\mathbb{E}_{q_\phi(\mathbf{Z})} \log \frac{p_\theta(\mathbf{Z}, \mathbf{x})}{q_\psi(\mathbf{Z})}}_{\text{C-ELBO}[q_\psi(\mathbf{Z}) \| p_\theta(\mathbf{Z}, \mathbf{x})]} + KL[q_\psi(\mathbf{Z}) || p_\theta(\mathbf{Z}|\mathbf{x})], \tag{5}$$

which is analogous to Eq. 2. Here, we call the first term the "conditional ELBO" (C-ELBO) to reflect that maximizing it is equivalent to minimizing an upper bound on $KL[q_\psi(\mathbf{Y}) \| p_\theta(\mathbf{Y}|\mathbf{x})]$.

3.3 Inference via Conditional Prior Networks

The previous section says that we should seek a distribution q_ψ to approximate $p_\theta(\mathbf{z}|\mathbf{x})$ as depicted in Figure 1(c). Although the latent distribution $p(\mathbf{z})$ may be simple, the conditional distribution $p_\theta(\mathbf{z}|\mathbf{x})$ is typically complex and often multimodal (cf. Fig. 4).

To define a variational distribution satisfying the conditions of Lemma 1, we propose to draw $\boldsymbol{\epsilon}$ from some fixed base density $q(\boldsymbol{\epsilon})$ and then use a network with parameters ψ to map to the latent space $\mathbf{z}$ so that the marginal $q_\psi(\mathbf{z})$ can represent a complex output distribution. The conditional of $\mathbf{y}$ given $\mathbf{z}$ is exactly as in p. The full variational distribution is therefore

$$q_\psi(\boldsymbol{\epsilon}, \mathbf{z}, \mathbf{y}) = q(\boldsymbol{\epsilon}) q_\psi(\mathbf{z}|\boldsymbol{\epsilon}) p_\theta(\mathbf{y}|\mathbf{z}) \qquad \text{with} \qquad q_\psi(\mathbf{z}|\boldsymbol{\epsilon}) = \delta(\mathbf{z} - \text{Prior}_\psi(\boldsymbol{\epsilon})), \tag{6}$$

where δ is a multivariate delta function. We call the network Prior_ψ a "conditional prior network" to emphasize that the parameters ψ are fit so that $q_\psi(\mathbf{Z})$ matches $p_\theta(\mathbf{Z}|\mathbf{x})$, and so that $\mathbf{z}$, when "decoded" using θ, will predict $\mathbf{y}$ given $\mathbf{x}$.

Theorem 2. *If q_ψ is as defined in Eq. 6 and $\text{Prior}_\psi(\boldsymbol{\epsilon})$ is one-to-one for all ψ, the C-ELBO from Eq. 5 becomes*

$$\textit{C-ELBO}[q_\psi(\mathbf{Z}) \| p_\theta(\mathbf{Z}, \mathbf{x})] = \mathbb{E}_{q(\boldsymbol{\epsilon})} \left[\log p_\theta(\text{Prior}_\psi(\boldsymbol{\epsilon}), \mathbf{x}) + \log |\nabla \text{Prior}_\psi(\boldsymbol{\epsilon})|\right] + \mathbb{H}[q(\boldsymbol{\epsilon})],$$

where $\mathbb{H}[q(\boldsymbol{\epsilon})]$ is the (fixed) entropy of $q(\boldsymbol{\epsilon})$, ∇ is the Jacobian with respect to $\boldsymbol{\epsilon}$, and $|\cdot|$ is the determinant.

Proof of Theorem 2. Firstly, since the latent density $q_\psi(\mathbf{z})$ is projected from some fixed base density $q(\boldsymbol{\epsilon})$, in order to preserve total probability, the change of $q_\psi(\mathbf{z})$ along interval $d\mathbf{z}$ must be equivalent to the change of $q(\boldsymbol{\epsilon})$ along interval $d\boldsymbol{\epsilon}$:

$$q_\psi(\mathbf{z}) d\mathbf{z} = q(\boldsymbol{\epsilon}) d\boldsymbol{\epsilon}$$

This property requires the change of variables theorem [8] such that

$$q_\psi(\text{Prior}_\psi(\boldsymbol{\epsilon})) \, |\nabla \text{Prior}_\psi(\boldsymbol{\epsilon})| = q(\boldsymbol{\epsilon})$$

where

$$\text{Prior}_\psi(\boldsymbol{\epsilon}) = \mathbf{z} \quad \textit{and} \quad \nabla\text{Prior}_\psi(\boldsymbol{\epsilon}) = \frac{\partial \mathbf{z}}{\partial \boldsymbol{\epsilon}}.$$

Thus, we can write

$$\begin{aligned}
\text{C-ELBO}[q_\psi(\mathbf{Z})\|p_\theta(\mathbf{Z},\mathbf{x})] &= \mathbb{E}_{q_\psi(\mathbf{Z})} \log \frac{p_\theta(\mathbf{Z},\mathbf{x})}{q_\psi(\mathbf{Z})} \\
&= \mathbb{E}_{q(\boldsymbol{\epsilon})} \log \frac{p_\theta(\text{Prior}_\psi(\boldsymbol{\epsilon}),\mathbf{x})}{q_\psi(\text{Prior}_\psi(\boldsymbol{\epsilon}))} \\
&= \mathbb{E}_{q(\boldsymbol{\epsilon})} \log \frac{p_\theta(\text{Prior}_\psi(\epsilon),\mathbf{x})}{q(\boldsymbol{\epsilon})/\left|\nabla\text{Prior}_\psi(\boldsymbol{\epsilon})\right|} \\
&= \mathbb{E}_{q(\boldsymbol{\epsilon})} \left[\log p_\theta(\text{Prior}_\psi(\boldsymbol{\epsilon}),\mathbf{x}) + \log\left|\nabla\text{Prior}_\psi(\boldsymbol{\epsilon})\right|\right] + \mathbb{H}_q[\boldsymbol{\epsilon}].
\end{aligned}$$

□

This objective is related to the "triple ELBO" used by [24] for a situation with a small number of *fixed* decompositions of $\mathbf{t}$ into $(\mathbf{x},\mathbf{y})$. Algorithmically, the approaches are quite different since they pre-train a single network for each subset of $\mathbf{t}$, which can be used for any $\mathbf{x}$ with that pattern, and a further product of experts approximation is used for novel missing features at test time. We assume arbitrary queries and so pre-training is inapplicable and novel missing features pose no issue. Still, our bounding justification may provide additional insight for their approach.

3.4 Conditional Prior Network Alternatives

We explore the following two candidiate conditional prior network options:

Gaussian Variational Inference (GVI): The GVI Prior_ψ linearly warps a spherical Gaussian over ϵ into an arbitrary Gaussian $\mathbf{z}$:

$$\text{Prior}_\psi(\boldsymbol{\epsilon}) = \mathbf{W}\boldsymbol{\epsilon} + \mathbf{b}, \qquad \text{where} \qquad \log\left|\nabla\text{Prior}_\psi(\boldsymbol{\epsilon})\right| = \log\left|\mathbf{W}\right|, \tag{7}$$

where $\psi = (\mathbf{W},\mathbf{b})$ for a square matrix $\mathbf{W}$ and a mean vector $\mathbf{b}$. While projected gradient descent can be used to maintain invertibility of W, we did not encounter issues with non-invertible W requiring projection during our experiments.

Normalizing Flows (NF): A normalizing flow [15] projects a probability density through a sequence of easy computable and invertible mappings. By stacking multiple mappings, the transformation can be complex. We use the special structured network called Planar Normalizing Flow:

$$\mathbf{h}_i = f_i(\mathbf{h}_{i-1}) = \mathbf{h}_{i-1} + \mathbf{u}_i g(\mathbf{h}_{i-1}^T \mathbf{w}_i + b_i), \tag{8}$$

for all i, where $h_0 = \boldsymbol{\epsilon}$, i is the layer id, w and u are vectors, and the output dimension is exactly same with the input dimension. Using $\circ$ for function composition, the conditional prior network$_\psi$ is given as

$$\text{Prior}_\psi(\boldsymbol{\epsilon}) = f_k \circ f_{k-1} \cdots f_1(\boldsymbol{\epsilon}), \qquad \text{where} \qquad \log\left|\nabla\text{Prior}_\psi(\boldsymbol{\epsilon})\right| = \sum_{i=1}^{k} \log\left|\nabla f_i\right|. \tag{9}$$

The bound in Theorem 2 requires that Prior_ψ is invertible. Nevertheless, we find **Fully Connected Networks (FCNs)** useful for comparison in low-dimensional visualizations. Here, the Jacobian must be calculated using separate gradient calls for each ouput variable, and the lack of invertibility prevents the C-ELBO bound from being correct.

We summarize our approach in Algorithm 1. In brief, we define a variational distribution $q_\psi(\boldsymbol{\epsilon}, \mathbf{z}) = q(\boldsymbol{\epsilon})q_\psi(\mathbf{z}|\boldsymbol{\epsilon})$ and optimize ψ so that $q_\psi(\mathbf{z})$ is close to $p_\theta(\mathbf{z}|\mathbf{x})$. The variational distribution includes a "Prior" as $q_\psi(\mathbf{z}|\boldsymbol{\epsilon}) = \delta(\mathbf{z} - \text{Prior}_\psi(\boldsymbol{\epsilon}))$. The algorithm uses stochastic gradient decent on the C-ELBO with gradients estimated using Monte Carlo samples of $\boldsymbol{\epsilon}$ and the reparameterization trick [10, 22, 16]. After inference, the original VAE distribution $q(\mathbf{y}|\mathbf{z}) = p_\theta(\mathbf{y}|\mathbf{z})$ gives samples over the query variables.

4 Experiments

Having defined our conditional prior networks methodology for conditional inference with pre-trained VAEs, we now proceed to empirically evaluate our three previously defined conditional prior network instantiations and compare them with (Markov chain) Monte Carlo (MCMC) sampling approaches on three different pre-trained VAEs. Below we discuss our datasets and methodology followed by our experimental results.

4.1 Datasets and Pre-trained VAEs

MNIST is the well-known benchmark handwritten digit dataset [11]. We use a pre-trained VAE with a fully connected encoder and decoder each with one hidden layer of 64 ReLU units, a final sigmoid layer with Bernoulli likelihood, and 2 latent dimensions for $\mathbf{z}$.[1] The VAE has been trained on 60,000 black and white binary thresholded images of size 28×28. The limitation to 2 dimensions allows us to visualize the conditional latent distribution of all methods and compare to the ground truth through a fine-grained discretization of $\mathbf{z}$.

Anime is a dataset of animated character faces [7]. We use a pre-trained VAE with convolutional encoder and deconvolutional decoder, each with 4 layers. The decoder contains respective channel sizes $(256, 128, 32, 3)$ each using 5×5 filters of stride 2 and ReLU activations followed by batch norm layers. The VAE has a final tanh layer with Gaussian likelihood, and 64 latent dimensions for $\mathbf{z}$.[2] The VAE has been trained on 20,000 images encoded in RGB of size $64 \times 64 \times 3$.

CelebA dataset [13] is a benchmark dataset of images of celebrity faces. We use a pre-trained VAE with a structure that exactly matches the Anime VAE provided above, except that it uses 100 latent dimensions for $\mathbf{z}$.[3] The VAE has been trained on 200,000 images encoded in RGB of size $64 \times 64 \times 3$.

[1] `https://github.com/kvfrans/variational-autoencoder`

[2] `https://github.com/wuga214/IMPLEMENTATION_Variational-Auto-Encoder`

[3] `https://github.com/yzwxx/vae-celebA`

4.2 Methods Compared

For sampling approaches, we evaluate rejection sampling **(RS)**, which is only feasible for our MNIST VAE with a 2-dimensional latent embedding for $\mathbf{z}$. We also compare to the MCMC method of Hamiltonian Monte Carlo **(HMC)** [4,3]. Both sampling methods exploit the VAE decomposition and sampling methodology described in Section 3.1.

We went to great effort to tune the parameters of HMC. For MNIST, with low dimensions, this was generally feasible, with a few exceptions as noted in Figure 5(b). For the high-dimensional latent space of the Anime and CelebA VAEs, finding parameters to achieve good mixing was often impossible, leading to poor performance. Section 7.3 of the Appendix discusses this in detail.

For the conditional prior networks methods, we use the three conditional prior network variants described in Section 3.3: Gaussian Variational Inference **(GVI)**, Planar Normalizing Flow **(NF)**, and a Fully Connected Neural Network **(FCN)**. By definition, the latent dimensionality of ϵ must match the latent dimensionality of $\mathbf{z}$ for each pre-trained VAE. Given evidence as described in the experiments, all conditional prior networks were trained as described in Algorithm 1. We could not train the FCN conditional prior network for conditional inference in Anime and CelebA due to the infeasibility of computing the Jacobian for the respective latent dimensionalities of these two VAEs.

In preliminary experiments, we considered the alternating sampling approach suggested by [16, Appendix F], but found it to perform very poorly when the evidence is ambiguous. We provide a thorough analysis of this in Section 7.2 of the Appendix comparing results on MNIST with various fractions of the input taken as evidence. In summary, Rezende's alternation method produces reasonable results when a large fraction of pixels are observed, so the posterior is highly concentrated. When less than around 40% of pixels are observed, however, performance rapidly degrades.

4.3 Evaluation Methodology

We experiment with a variety of evidence sets to demonstrate the efficiency and flexibility of our conditional prior networks methodology for arbitrary conditional inference queries in pre-trained VAEs. All conditional prior networks optimization and inference takes (typically well) under 32 seconds per evidence set for all experiments running on an Intel Xeon E5-1620 v4 CPU with 4 cores, 16Gb of RAM, and an NVIDIA GTX1080 GPU. A detailed running time comparison is provided in Section 4.6.

Qualitatively, we visually examine the 2D latent distribution of $\mathbf{z}$ conditioned on the evidence for the special case of MNIST, which has low enough latent dimensionality to enable us to obtain ground truth through discretization. For all experiments, we qualitatively assess sampled query images generated for each evidence set to assess both the coverage of the distribution and the quality of match between the query samples and the evidence, which is fixed in the displayed images.

Quantitatively, we evaluate the performance of the proposed framework and candidate inference methods through the following two metrics.

C-ELBO: As a comparative measure of inference quality for each of the conditional prior network methods, we provide pairwise scatterplots of the C-ELBO as defined in 5 for a variety of different evidence sets.

Query Marginal Likelihood: For each conditional inference evaluation, we randomly select an image and then a subset of that image as evidence $\mathbf{x}$ and the remaining pixels $\mathbf{y}$ as the ground truth query assignment. Given this, we can evaluate the marginal likelihood of the query $\mathbf{y}$ as follows:

$$\log p(\mathbf{y}) = \log E_{\mathbf{Z}}[p(\mathbf{y}|\mathbf{Z})]$$

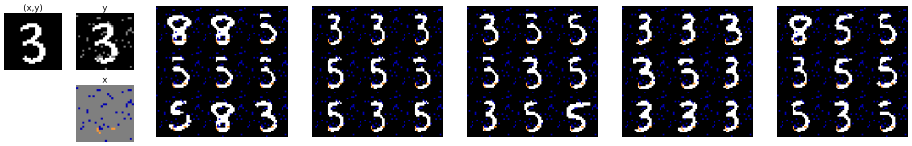
Average Structural Similarity (SSIM): The Structural Similarity Index Measure (SSIM) [25] is a perception-based model for comparing images. In our case, we would like to assess the average SSIM between the original image and the reconstructions for N samples from a conditional query over occluded portions of the image. Specifically, for each query we compute:

$$\frac{1}{N}\sum_{i}^{N} \text{SSIM}(s_i, o),$$

where s_i denotes the reconstruction of sample i and o denotes the original image.

Average Standard Deviation of Samples: The ultimate goal of our proposed inference method for VAEs is to produce a distribution over conditional queries, which we've argued previously can be advantageous over deterministic methods for image completion in the case that there are multiple plausible completions. To understand just how diverse our image completions are, we compute the per-pixel standard deviation of sampled images and report the average standard deviation over pixels for each query. Clearly, a value of 0 would indicate deterministic completion and higher values indicate more variation (diversity) in sampled images.

4.4 Conditional Inference on MNIST



(a) Data (b) GVI (c) NF (d) FCN (e) HMC (f) RS (Exact)

Fig. 3: One conditional inference example for MNIST. (a) The original digit $\mathbf{t}$, the subset selected for evidence $\mathbf{x}$, and the remaining ground truth query $\mathbf{y}$. (b–f) Nine sample queries from each of five methods. In all plots, the evidence subset has white replaced with orange and black replaced with blue.

For conditional inference in MNIST, we begin with Figure 3, which shows one example of conditional inference in the pre-trained MNIST model using the

different inference methods. While the original image used to generate the evidence represents the digit 3, the evidence is very sparse allowing the plausible generation of other digits. It is easy to see that most of the methods can handle this simple conditional inference, with only GVI producing some samples that do not match the evidence well in this VAE with 2 latent dimensions.

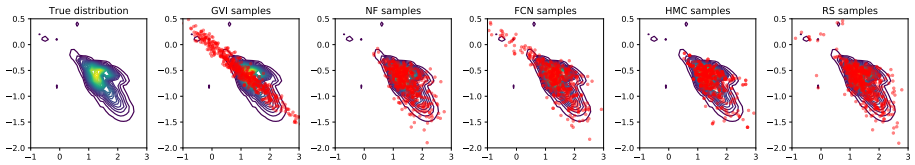
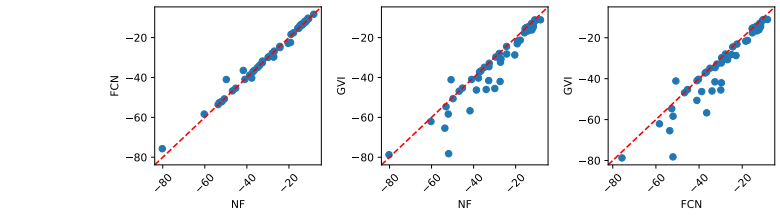


Fig. 4: $p(\mathbf{z}|\mathbf{x})$ for the MNIST example in Figure 3. The contour plot (left) shows the true distribution. The remaining plots show samples from each method overlaid on the true distribution.

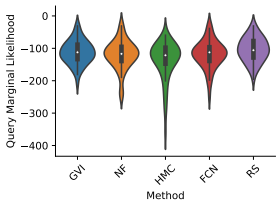
To provide additional insight into Figure 3, we now turn to Figure 4, where we visually compare the true conditional latent distribution $p(\mathbf{z}|\mathbf{x})$ (leftmost) with the corresponding distributions of each of the inference methods. At a first glance, we note that the true distribution is both multimodal and non-Gaussian. We see that GVI covers some mass not present in the true distribution that explains its relatively poor performance in Figure 3(b). All remaining methods (both conditional prior network and sampling) do a reasonable job of covering the irregular shape and mass of the true distribution.

We now proceed to a quantitative comparison of performance on MNIST over 50 randomly generated queries. We summarize our observations as follows:

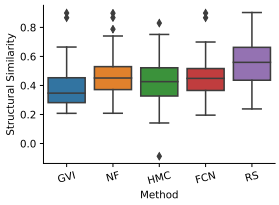
1. In Fig. 5(a), we present a pairwise comparison of the performance of each conditional prior network method on 50 randomly generated evidence sets. Noting that higher is better, we observe that FCN and NF perform comparably and generally outperform GVI.
2. In Fig. 5(b), we examine the Query Marginal Likelihood distribution for the same 50 evidence sets from (a) with each likelihood expectation generated from 500 samples. Again, noting that higher is better, here we see that RS slightly edges out all other methods with all conditional prior networks generally performing comparably. HMC performs worst here, where we remark that inadequate coverage of the latent $\mathbf{z}$ due to poor mixing properties leads to over-concentration on $\mathbf{y}$ leading to a long tail in a few cases with poor coverage.
3. In Fig. 5(c), we evaluate the structural similarity between reconstructed conditional inference samples and the original digit images. Here, a higher value indicates higher similarity of the reconstruction to the original. RS has a significant advantage over the others since it is exact (though only computable for this 2D example), whereas GVI shows worst performance in terms of recovering the original image. NF, HMC and FCN show comparable performance to each other due to their ability to capture more complex latent distributions than GVI's simple latent Gaussian model.
4. In Fig. 5(d), we estimate diversity of the generated samples for the conditional inference by evaluating the average standard deviation as previously described.



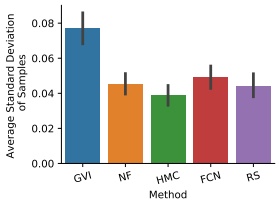
(a) Pairwise C-ELBO Comparison



(b) Query Marginal Likelihood



(c) Structural Similarity



(d) Standard Deviation

Fig. 5: Quantitative Analysis for MNIST dataset. (a) Pairwise C-ELBO comparison of different conditional prior network methods evaluated over the 50 randomly generated evidence sets for MNIST. (b) Violin (distribution) plots of the Query Marginal Likelihood for the same 50 evidence sets from (a), with each likelihood expectation generated from 500 samples. (c) Box plots of the Structural Similarity between samples of conditional inference and original image. (d) Average Standard Deviation of the conditional inference samples. *For all metrics, higher is better.*

It shows that all of the algorithms are able to provide a variety of predictions for the queried portion of the image. In terms of relative comparison, GVI shows a significant advantage over all other methods in terms of producing diverse predictions, which reflects our previous observation in Fig. 3 that GVI may occasionally produce results conflicting with the evidence due to lack of expressivity in its latent distribution model. In this sense the increased diversity may be due to this out-of-evidence generalization.

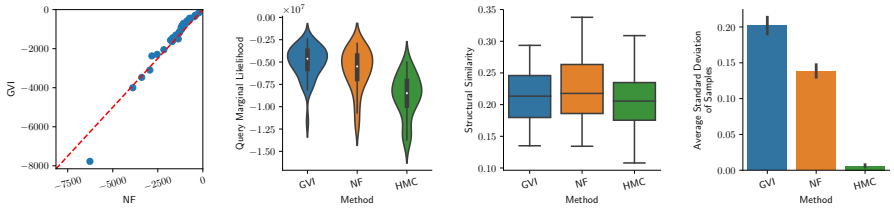
We will see that these issues with HMC mixing become much more pronounced as we move to experiments in VAEs with higher latent dimensionality in the next section.

4.5 Conditional Inference on Anime and CelebA

Now we proceed to our larger VAEs for Anime and CelebA with respective latent dimensionality of 64 and 100 that allow us to work with larger and more visually complex RGB images. In these cases, FCN could not be applied due to the infeasibilty of computing the Jacobian and RS is also infeasible for such high dimensionality. Hence, we only compare the two conditional prior networks GVI and NF with HMC.

(a) Data (b) GVI Samples (c) NF Samples (d) HMC Samples

Fig. 6: One conditional inference example for Anime. (a) The original image $\mathbf{t}$, the subset selected for evidence $\mathbf{x}$, and the remaining ground truth query $\mathbf{y}$. (b–d) 20 samples from each method with the evidence superimposed on each image. (c,d) NF and HMC demonstrate poor coverage.



(a) Pairwise C-ELBO Comparison (b) Query Marginal Likelihood (c) Structural Similarity (d) Standard Deviation

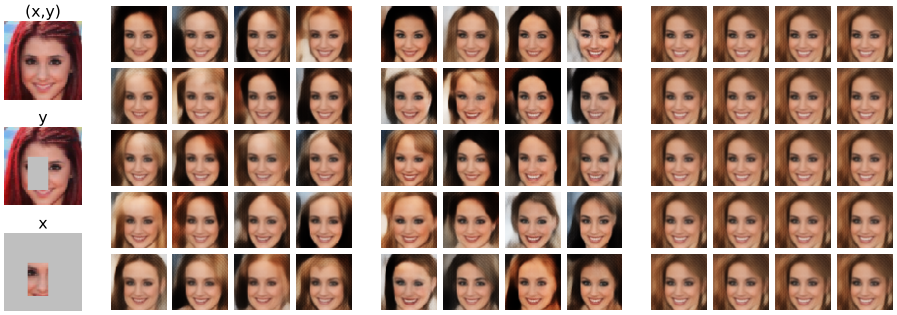
Fig. 7: (a)Pairwise C-ELBO comparison of GVI vs. FCN, (b) Violin plots of the Query Marginal Likelihood, (c) Box plots of the Structural Similarity and (d) Average Standard Deviation of Samples for Anime. Evaluation details match those of Fig. 5 except with 50 conditional inference queries. *For all metrics, higher is better.*

We start with a qualitative and quantitative performance analysis of conditional inference for the Anime dataset. Qualitatively, in Fig. 6, we see that inference for HMC shows little identifiable variation and seems to have collapsed into a single latent mode. In contrast, GVI appears to show better coverage, generating a wide range of faces that generally match very well with the superimposed evidence. NF also shows some degree of generalization ability as its examples have not collapsed although it does not show significant diversity over its examples. It is also worth noting that all of the candidate algorithms (GVI, NF, and HMC) successfully capture the observed evidence with less than 5% coverage of the original image.

Quantitatively, Fig. 7 strongly reflects the qualitative visual observations above. We summarize our observations as follows:

1. For the conditional prior networks, GVI and NF are comparable in terms of maximizing the C-ELBO.

2. In Fig. 7(b), for all methods evaluated on Query Marginal Likelihood, we observe that both GVI and NF outperform HMC on Anime due to HMC's mode collapse.
3. In Fig. 7(c), we evaluate the structural similarity between conditional inference samples and the original Anime images for each inference method. The results indicate that NF slightly outperforms GVI and HMC, which is due to GVI's lack of expressive latent modeling and HMC's mode collapse.
4. In Fig. 7(d), we estimate the diversity of the samples as the average standard deviation over pixels given each query, as measured for all inference methods. As before, GVI is most diverse, which reflects our visual intuition from Fig. 6(b); notably GVI appears to be consistent with the evidence for this dataset. HMC has lowest diversity simply due to its observed mode collapse as also reflected in Fig. 6(d). NF shows quantitative diversity closer to GVI, which is reflected in the variation of images in Fig. 6(c) as compared to HMC in Fig. 6(d).



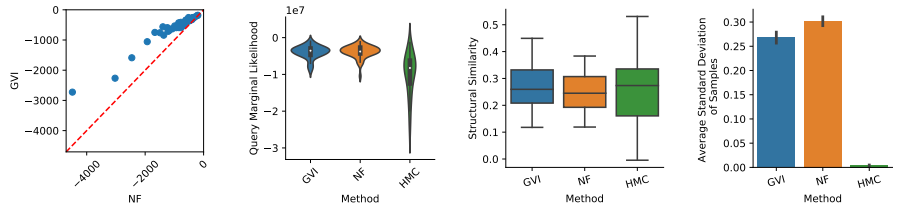
(a) Data (b) GVI Samples (c) NF Samples (d) HMC Samples

Fig. 8: One conditional inference example for CelebA. (a) The original image $\mathbf{t}$, the subset selected for evidence $\mathbf{x}$, and the remaining ground truth query $\mathbf{y}$. (b–d) 20 samples from each method with the evidence superimposed on each image. (d) HMC demonstrates poor coverage due to mode collapse.

We now continue to a qualitative and quantitative performance analysis of conditional inference for the CelebA. Qualitatively, in Fig. 8, HMC still performs poorly, but NF appears to perform much better, with both conditional prior networks GVI and NF generating a wide range of faces that match the superimposed evidence, with perhaps slightly more face diversity for GVI.

Quantitatively, Fig. 9 strongly reflects the qualitative visual observations above. In short, for the conditional prior networks, GVI solidly outperforms NF on the C-ELBO comparison. For all methods evaluated on Query Marginal Likelihood, GVI outperforms both NF and HMC on Anime, while for CelebA, GVI performs comparably to (if not slightly worse) than NF, with both solidly outperforming HMC. Finally, as observed previously, HMC suffers from mode collapse since the samples generated from it have near-zero diversity (per pixel standard deviation over samples) for all of the 50 random conditional inference queries.

We remark that GVI does empirically encounter some numerical instability in maintaining a valid log determinant as part of our objective function. As a



(a) Pairwise C-ELBO Comparison (b) Query Marginal Likelihood (c) Structural Similarity (d) Standard Deviation

Fig. 9: (a)Pairwise C-ELBO comparison of GVI vs. FCN, (b) Violin plots of the Query Marginal Likelihood, (c) Box plots of the Structural Similarity and (d) Average Standard Deviation of Samples for CelebA. Evaluation details match those of Fig. 5 except with 50 conditional inference queries. *For all metrics, higher is better.*

consequence, we observe multiple cases where GVI inference halts prematurely due to numerical instability (i.e., an NaN from the log determinant calculation). Specifically, in over 50 inference instances from a total of 5000 inference epochs, we note that only 15% of the GVI inferences finish all epochs without encountering numerical error.[4] This overall observation leads us to a strong preference for using Normalizing Flow (NF) for prior-based conditional inference in VAEs in practice due to its empirically observed strong performance and overall numerical stability.

4.6 Comparison of Running Time

We do not evaluate the time for training VAEs here since all of the methods we empirically compare rely on a pre-trained VAE. Indeed, as noted in the experimental section, we simply used pre-trained VAEs from various online repositories for our experiments. With that in mind though, it is important to note that when we experimented with training some of the larger VAEs from scratch (namely Anime and CelebA), it required over one day on the hardware used for our experimentation as described in Section 4.3.

The running time of conditional inference varies with the complexity of conditional prior networks, the optimization algorithm used, and the complexity of the pre-trained Decoder. We found that L-BFGS [12] consistently converged fastest and with the best results in comparison to SGD, Adam, Adadelta, and RMSProp.

Table 1 shows the computation time for each of the three candidate conditional prior networks (FCN is only applicable to MNIST) as well as HMC and Rejection Sampling (RS is only applicable for MNIST). Here we note that HMC burn-in can take an order of magnitude more time than conditional network optimization. HMC can also take an order of magnitude more time for prediction, with our conditional network method generating all required prediction samples in under one second in *all* cases. As a final remark, we observe that the running times here for HMC do

[4] While GVI inference is halted when an NaN log determinant is encountered, its inference is still valid in the last epoch before the numerical error cutoff. Hence, all experimental results for GVI use the last valid epoch in the event that numerical error is encountered.

Table 1: Average running Time (in seconds) of experiments. We use L-BFGS for conditional prior networks in this table. For HMC, we predefine the burn-in (optimization) iterations to be 10,000 for all datasets. For all methods, the prediction sample size is 500.

Period	MNIST					Anime			CelebA		
-	GVI	NF	FCN	HMC	RS	GVI	NF	HMC	GVI	NF	HMC
Optimization	0.36	2.79	5.26	34.92	-	2.52	4.22	81.3	31.45	9.95	224.5
Prediction	< 0.04	< 0.04	< 0.04	2	0.19	< 0.08	< 0.08	2	< 0.37	< 0.37	2

not reflect the *substantial* effort that went into tuning its parameters for Anime and CelebA as described in Section 7.3 of the Appendix.

5 Conclusion

We introduced conditional prior networks, a novel variational inference method for performing fast arbitrary conditional inference in pre-trained VAEs that does not require retraining the decoder for every decomposition of query and evidence. Using three VAEs pre-trained on different datasets used for image completion queries, we demonstrated that the Gaussian Variational Inference (GVI) and Normalizing Flows (NF) conditional prior networks generally outperform Hamiltonian Monte Carlo both qualitatively and quantitatively on a variety of evaluation metrics. Moreover, NF empirically tends to be more numerically stable than GVI for inference. As discussed, other methods proposed in the literature for the same task are either computationally prohibitive or cannot handle sparse evidence sets. In sum, our work suggests that our simple and intuitive conditional prior network training enables fast conditional inference for arbitrary queries in pre-trained VAEs and provides an efficient and effective alternative to existing state-of-the-art methods for this task.

6 Declarations

Funding: The first author (GW) was funded by an NSERC Discovery Grant to the third author (SS).
Conflicts of interest/Competing interests: None.
Ethics approval: Not applicable.
Consent to participate: Not applicable.
Consent for publication: Not applicable for publicly available data. All authors consent to listing their name, affiliation, email, and image (if required).
Availability of data and material: All data sources are public and cited in the text.
Code availability: All code will be made publicly available upon publication.
Authors' contributions: The first author (GW) led the development of the key ideas and performed all implementation and writing of the initial draft and revisions. The second author (JD) provided supervision and help with the mathematical derivation, experimental design, guidance on paper structure, and revisions. The

third author (SS) provided supervision and help with the experimental design, guidance on paper structure, and revisions.

References

1. Burda, Y., Grosse, R., Salakhutdinov, R.: Importance weighted autoencoders. International Conference on Learning Representations (ICLR) (2016)
2. Cover, T.M., Thomas, J.A.: Elements of information theory 2nd edition, thm 2.5.3 (2006)
3. Daniel Levy Matthew D. Hoffman, J.S.D.: Generalizing hamiltonian monte carlo with neural networks. The Sixth International Conference on Learning Representations (ICLR) (2018)
4. Girolami, M., Calderhead, B.: Riemann manifold langevin and hamiltonian monte carlo methods. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **73**(2), 123–214 (2011)
5. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems (NIPS), pp. 2672–2680 (2014)
6. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and locally consistent image completion. ACM Transactions on Graphics (ToG) **36**(4), 1–14 (2017)
7. Jin, Y., Zhang, J., Li, M., Tian, Y., Zhu, H., Fang, Z.: Towards the automatic anime characters creation with generative adversarial networks. arXiv preprint arXiv:1708.05509 (2017)
8. Kaplan, W.: Advanced calculus. Cambridge, Mass., Addison-Wesley Press (1952)
9. Kingma, D.P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., Welling, M.: Improved variational inference with inverse autoregressive flow. In: Advances in Neural Information Processing Systems (NIPS), pp. 4743–4751 (2016)
10. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. International Conference on Learning Representations (ICLR) (2014)
11. LeCun, Y., Cortes, C.: MNIST handwritten digit database (2010). URL `http://yann.lecun.com/exdb/mnist/`
12. Liu, D.C., Nocedal, J.: On the limited memory bfgs method for large scale optimization. Mathematical programming **45**(1-3), 503–528 (1989)
13. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (2015)
14. Maaløe, L., Sønderby, C.K., Sønderby, S.K., Winther, O.: Auxiliary deep generative models. 33rd International Conference on Machine Learning (ICML) (2016)
15. Rezende, D., Mohamed, S.: Variational inference with normalizing flows. In: International Conference on Machine Learning (ICML), pp. 1530–1538 (2015)
16. Rezende, D.J., Mohamed, S., Wierstra, D.: Stochastic backpropagation and approximate inference in deep generative models. In: Proceedings of the 31th International Conference on Machine Learning (ICML), pp. 1278–1286 (2014)
17. Rolfe, J.T.: Discrete variational autoencoders. International Conference on Learning Representations (ICLR) (2017)
18. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention, pp. 234–241. Springer (2015)
19. Saul, L.K., Jaakkola, T.S., Jordan, M.I.: Mean field theory for sigmoid belief networks. J. Artif. Intell. Res. **4**, 61–76 (1996)
20. Shu, R., Bui, H.H., Ghavamzadeh, M.: Bottleneck conditional density estimation. In: International Conference on Machine Learning, pp. 3164–3172 (2017)
21. Sohn, K., Lee, H., Yan, X.: Learning structured output representation using deep conditional generative models. In: Advances in Neural Information Processing Systems (NIPS), pp. 3483–3491 (2015)
22. Titsias, M.K., Lázaro-Gredilla, M.: Doubly stochastic variational bayes for non-conjugate inference. In: Proceedings of the 31th International Conference on Machine Learning (ICML), pp. 1971–1979 (2014)
23. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Deep image prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9446–9454 (2018)

24. Vedantam, R., Fischer, I., Huang, J., Murphy, K.: Generative models of visually grounded imagination. International Conference on Learning Representations (ICLR) (2017)
25. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
26. Yan, Z., Li, X., Li, M., Zuo, W., Shan, S.: Shift-net: Image inpainting via deep feature rearrangement. In: Proceedings of the European conference on computer vision (ECCV), pp. 1–17 (2018)
27. Yang, C., Lu, X., Lin, Z., Shechtman, E., Wang, O., Li, H.: High-resolution image inpainting using multi-scale neural patch synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6721–6729 (2017)

7 Appendix

7.1 Preliminary Check of Inference Methods

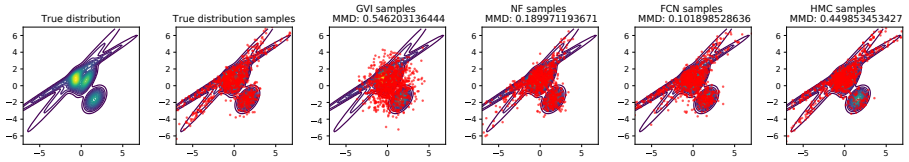


Fig. 10: Comparison of different inference methods on modeling a Gaussian mixture model distribution. The true distribution samples are directly sampled from a Gaussian mixture model. Maximum mean discrepancy (MMD) values given in the plot titles are generated relative to the true sample distribution.

In this experiment, we do not use a VAE, but instead simply model a complex latent 2D multimodal distribution over $\mathbf{z}$ as a Gaussian mixture model to evaluate the ability of each conditional inference method to accurately draw samples from this complex distribution. In general, Fig. 10 shows that while the conditional prior networks NF and FCN work well here, GVI (by definition) cannot model this multimodal distribution and HMC draws too few samples from the disconnected mode compared to the true sample distribution, indicating slight failure to mix well.

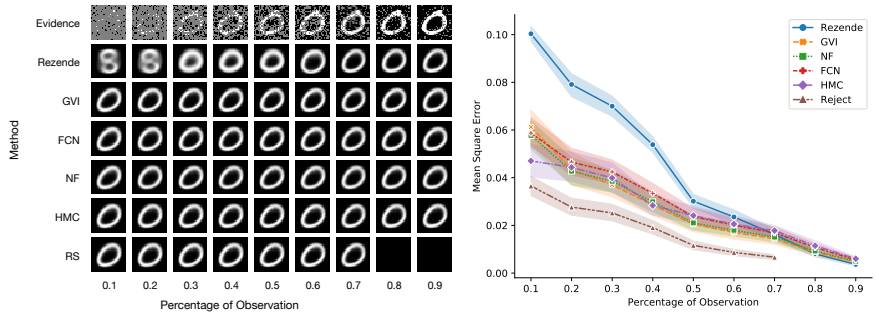
7.2 Comparison to Rezende Alternation

We compare to the alternating sampling approach of [16] (Appendix Section F) which is essentially an approximation of block Gibbs sampling. We call it the “Rezende method" in the following. This method does not asymptotically sample from the conditional distribution since the step sampling the latent variables given the query variables are approximated using the encoder.

Fig. 11(a) shows one experiment comparing all candidate algorithms including the Rezende method. We noticed that it fails to generate images that match the evidence when less than 40% of pixels are observed as evidence, while it makes reasonable predictions when the observation rate is higher. Fig. 11(b) shows this result is consistent over 50 randomly selected queries.

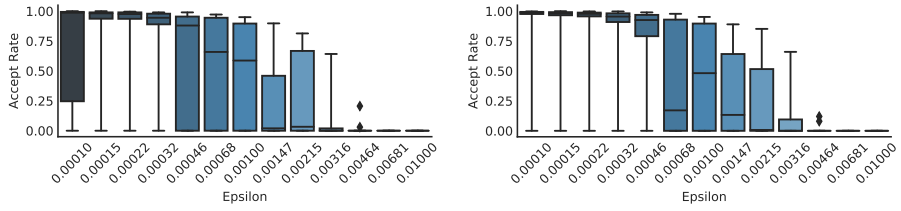
7.3 Systematic HMC Tuning Analysis for Anime and CelebA

While tuning HMC in lower dimensions was generally feasible for MNIST with a few exceptions noted in previous discussion of Fig. 5(b), we observed that HMC becomes very difficult to tune in the Anime and CelebA VAEs with higher latent dimensionality. To illustrate these HMC tuning difficulties, we present a summary of our systematic efforts to tune HMC on Anime and CelebA in Figure 12 with boxplots of the acceptance rate distribution of HMC for 30 Markov Chains vs



(a) Conditional Inference Example (b) Mean Squared Error of Query Variables

Fig. 11: Comparison of different conditional inference methods include the Rezende method on the MNIST dataset. (a) Shows one intuitive example. The first row shows the evidence observed, and the following rows show the mean of generated samples from the different algorithms. We note that with very high evidence, the posterior becomes extremely concentrated, meaning the rejection rates for rejection sampling become impractical. (b) The mean squared error between query variables of the original image and the generated samples of different algorithms. The results and standard deviations at each observation percentage come from 50 independent randomly selected queries.



(a) Anime (b) CelebA

Fig. 12: Boxplots of acceptance rate distribution of HMC for 30 Markov Chains vs different ϵ on (a) Anime and (b) CelebA. Each Markov chain ran for 10,000 burn-in samples with 10 leapfrog steps per iteration.

different ϵ on (a) Anime and (b) CelebA. We ran each Markov chain for 10,000 burn-in samples with 10 leapfrog steps per iteration; we tried 3 different standard leapfrog step settings of $\{5, 10, 30\}$, finding that 10 leapfrog steps provided the best performance across a range of ϵ and hence chosen for Fig. 12.

In short, Fig. 12 shows that only a very narrow band of ϵ lead to a reasonable acceptance rate for good mixing properties of HMC. Even then, however, the distribution of acceptance rates for any particular Markov Chain for a good ϵ is still highly unpredictable as given by the quartile ranges of the boxplot. In summary, we found that despite our systematic efforts to tune HMC for higher dimensional problems, it was difficult to achieve a good mixing rate and overall contributes to the generally poor performance observed for HMC on Anime and CelebA that we discuss next.

7.4 Quality of the Pre-trained VAE Models

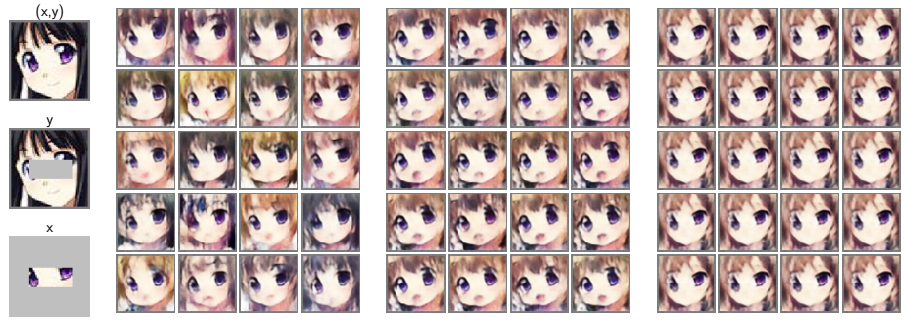
To assess the quality of the pre-trained VAE models, we show 100 samples from each in Fig. 13.

(a) MNIST (b) Anime (c) CelebA

Fig. 13: Samples from each of the pre-trained VAE models.

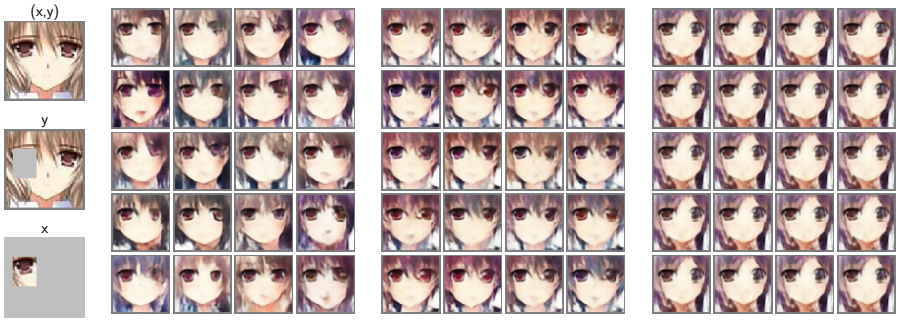
7.5 More Inference Examples

From Fig. 14 to Fig. 19, we show multiple additional examples of conditional inference matching the structure of experiments shown in Fig. 6 and 8 in the main text. Note that all of the inferences are conducted on the same trained VAE models used in the main paper. Overall, we observe the same general trends as discussed in the main text for Fig. 6 and 8.



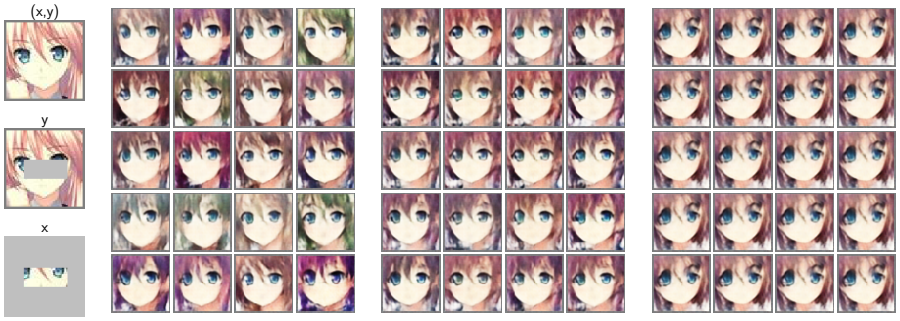
(a) Data (b) GVI Samples (c) NF Samples (d) HMC Samples

Fig. 14: Another conditional inference example on Anime dataset. Evidence includes eyeball color and face direction information.



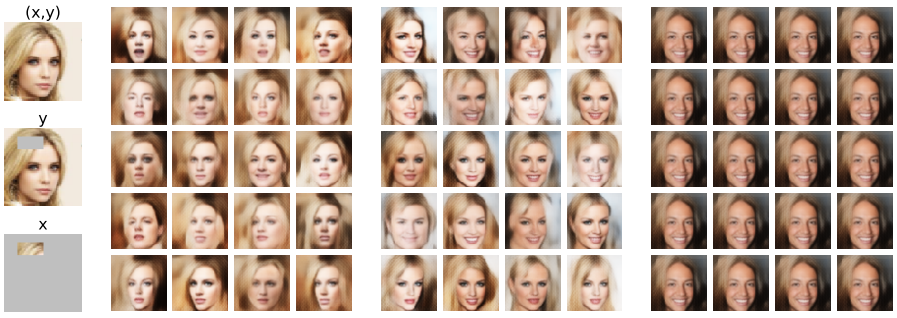
(a) Data (b) GVI Samples (c) NF Samples (d) HMC Samples

Fig. 15: Another conditional inference example on Anime dataset. Evidence includes style of eyes.



(a) Data (b) GVI Samples (c) NF Samples (d) HMC Samples

Fig. 16: Another conditional inference example on Anime dataset. Evidence includes hair style and eyeball color.

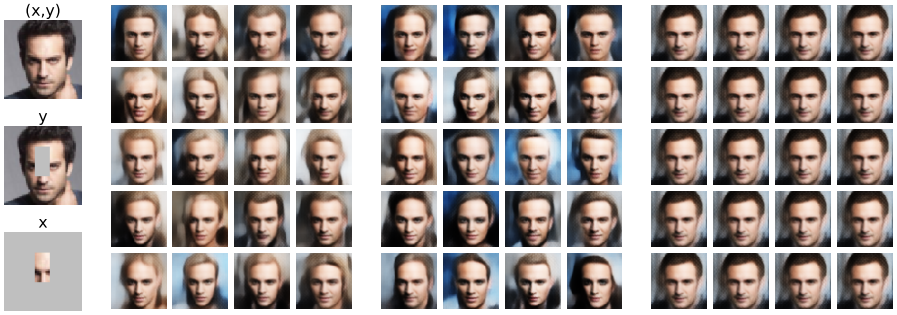


(a) Data (b) GVI Samples (c) NF Samples (d) HMC Samples

Fig. 17: Another conditional inference example on CelebA dataset. Evidence includes hair color. Note HMC inference fails to capture the evidence.

(a) Data (b) GVI Samples (c) NF Samples (d) HMC Samples

Fig. 18: Another conditional inference example on CelebA dataset. Evidence includes nose and mouth shape.



(a) Data (b) GVI Samples (c) NF Samples (d) HMC Samples

Fig. 19: Another conditional inference example on CelebA dataset. Evidence includes nose and forehead information.