A Decoder Suffices for Query-Adaptive Variational Inference

Sakshi Agarwal[1*] **Gabriel Hope**[1*] **Ali Younis**[1] **Erik B. Sudderth**[1]

[1]Department of Computer Science, University of California, Irvine
[*]Equal contribution

Abstract

Deep generative models like variational autoencoders (VAEs) are widely used for density estimation and dimensionality reduction, but infer latent representations via amortized inference algorithms, which require that all data dimensions are observed. VAEs thus lack a key strength of probabilistic graphical models: the ability to infer posteriors for test queries with arbitrary structure. We demonstrate that many prior methods for imputation with VAEs are costly and ineffective, and achieve superior performance via query-adaptive variational inference (QAVI) algorithms based directly on the generative decoder. By analytically marginalizing arbitrary sets of missing features, and optimizing expressive posteriors including mixtures and density flows, our non-amortized QAVI algorithms achieve excellent performance while avoiding expensive model retraining. On standard image and tabular datasets, our approach substantially outperforms prior methods in the plausibility and diversity of imputations. We also show that QAVI effectively generalizes to recent hierarchical VAE models for high-dimensional images.

1 INTRODUCTION

Structured probabilistic models, like graphical models (Koller and Friedman, 2009), are the basis for many machine learning applications. Formalizing the data generation process enables incorporation of domain knowledge and allows unsupervised learning from unlabeled data. Probabilistic models also enable diverse inference queries by users, for instance *imputation* queries (or, inpainting in the image domain) where arbitrary subsets of features are observed and the values of missing features are predicted. A number of inference algorithms (Pearl, 1988; Koller and Friedman, 2009) have been developed for models with discrete or Gaussian latent variables, which efficiently compute marginals of query variables given heterogeneous observations. While exact inference is intractable for many complex models, optimization-based variational methods (Wainwright and Jordan, 2008) often provide effective approximations.

Variational bounds were classically optimized via *coordinate ascent variational inference* (CAVI, Jordan et al. (1999)) algorithms that iteratively update posterior approximations for individual variables. CAVI updates are effective for many parametric models composed from conjugate priors, and can have efficient message-passing implementations (Ghahramani and Beal, 2001; Winn and Bishop, 2005). But, CAVI updates are based on integrals that often lack closed forms, requiring Monte Carlo approximations (Paisley et al., 2012; Kucukelbir et al., 2017) of uncertain quality.

Stochastic subsampling of data helps scale variational learning to big datasets (Hoffman et al., 2013), but iterative CAVI updates may still be slow for complex models. *Amortized* variational inference (Mnih and Gregor, 2014) seeks to boost training efficiency by determining variational posteriors via an inference (or recognition) network, which is shared (or amortized) across many similar inference tasks. *Variational autoencoders* (VAEs, Kingma and Welling (2014); Rezende et al. (2014)) are deep generative models that utilize amortized inference to jointly train a generative "decoder" and inference "encoder". Sophisticated generalizations to the encoder and decoder networks (Kingma and Welling, 2019; Sønderby et al., 2016; Vahdat and Kautz, 2020; Child, 2021) have produced hierarchical VAEs that realistically model complex image data via dozens of stochastic layers.

While amortized inference has enabled the learning of impressive deep generative models, it sacrifices the flexibility of CAVI to handle arbitrary inference queries. Because VAEs are typically trained from fully-observed data, the encoder assumes *complete* and *uncorrupted* observation of every data dimension (e.g., pixel). Simple heuristics (see Sec. 2) are sometimes used for learning VAEs with missing

Proceedings of the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023), PMLR 216:33–44.

data (Mattei and Frellsen, 2019; Nazábal et al., 2020; Collier et al., 2020), such as filling missing features with zeros. However, we show that in addition to requiring expensive encoder retraining for peak performance, these approaches are inaccurate unless the test inference queries are simple or known in advance (during model training). Amortized inference also produces sub-optimal variational bounds, and this "amortization gap" may be significant (Cremer et al., 2018; Krishnan et al., 2017). While quick-and-approximate inference may be sufficient to provide a noisy gradient signal in the midst of a long training process, it is problematic when applied to test queries, especially in domains like medicine where accurate uncertainty quantification is crucial.

To address these challenges, we propose *query-adaptive variational inference* (QAVI) methods that approximate the posterior of missing data with arbitrary patterns, given only a trained generative decoder and sparse observations. Our QAVI approach has the same inferential flexibility as classic CAVI algorithms, and critically does not require a database with many examples of the missing-data pattern of interest. But unlike classic CAVI algorithms, QAVI is applicable to any (differentiable) model with continuous latent variables, including deep generative models like hierarchical VAEs. While some prior work has improved VAE training by reducing amortization gaps (Kim et al., 2018; Marino et al., 2018), our application of non-amortized inference to missing-data queries for deep generative models is novel.

We begin in Sec. 2 by reviewing prior work on handling missing data with (hierarchical) VAEs. Sec. 3.1 then develops QAVI algorithms that optimize variational bounds for the missing feature values, rather than filling them via greedy heuristics. In Sec. 3.2, we develop an alternative QAVI algorithm that directly optimizes the posterior of the latent data encoding, without amortization. Doing this allows exact marginalization of missing feature values, and enables flexible posterior approximations including mixture models (Jaakkola and Jordon, 1999; Gershman et al., 2012) and normalizing flows (Rezende and Mohamed, 2015). Results in Sec. 4 then show substantial qualitative and quantitative improvements in capturing multimodal posterior uncertainty for VAE models of tabular data, as well as state-of-the-art hierarchical VAE models of images (Child, 2021).

2 BACKGROUND AND RELATED WORK

2.1 THE VARIATIONAL AUTOENCODER (VAE)

VAEs model the distribution of typically high-dimensional observed data x using continuous, lower-dimensional latent variables z, via the following generative model:

$$z \sim p(z), \qquad x \sim p_\theta(x \mid z). \tag{1}$$

For standard VAEs, the latent code $z \in \mathbb{R}^d$ has a factorized Gaussian prior $p(z)$. Given z, data is generated via a *decoder* (deep) neural network with weights θ. The decoder maps z to a likelihood $p_\theta(x|z)$ such as a factorized Gaussian.

The VAE log-likelihood $\log p_\theta(x) = \log \int p_\theta(x|z)p(z)\,dz$ is intractable. Learning thus typically employs *amortized* VI, where an *encoder* with parameters ϕ approximates the posterior over latent codes $q_\phi(z|x) \approx p_\theta(z|x)$. We jointly learn θ, ϕ by maximizing the *evidence lower-bound* (ELBO):

$$\mathcal{L}(\theta, \phi; x) = E_{q_\phi(z|x)}[\log p_\theta(x|z)] - \mathrm{KL}(q_\phi(z|x)||p(z)). \tag{2}$$

Here, KL is the Kullback–Leibler divergence, calculated analytically when $q_\phi(z|x)$ and $p(z)$ are both Gaussian. The ELBO provides a lower bound on the evidence $\log p_\theta(x)$ that is tight when the variational posterior $q_\phi(z|x)$ is exact. This expectation can be approximated via Monte Carlo samples from $q_\phi(z|x)$. Gradients with respect to θ, ϕ can then be estimated by the reparameterization "trick" of sampling from $q_\phi(z|x)$ via linear transforms of standard normal variables (Kingma and Welling, 2014; Rezende et al., 2014).

2.2 HIERARCHICAL VAES

Hierarchical VAEs (HVAEs, Sønderby et al. (2016); Klushyn et al. (2019)) extend the VAE by partitioning the latent code into L disjoint groups $z = (z_1, z_2, ..., z_L)$, increasing model expressiveness for complex data like images (Vahdat and Kautz, 2020; Child, 2021). HVAEs generate these stochastic codes sequentially as $p_\theta(x|z) = p_\theta(z_1)(\prod_{\ell=2}^{L} p_\theta(z_\ell|z_{<\ell}))p_\theta(x|z_L)$, with a similar encoder: $q_\phi(z|x) = q_\phi(z_1|x)\prod_{\ell=2}^{L} q_\phi(z_\ell|z_{<\ell}, x)$. Each conditional in the decoder $p_\theta(z_\ell|z_{<\ell})$, and the encoder $q_\phi(z_\ell|z_{<\ell}, x)$, is typically Gaussian with mean and variance determined by (non-linear) neural networks. The HVAE ELBO equals

$$\mathcal{L}_H(\theta, \phi; x) = E_{q_\phi(z|x)}[\log p_\theta(x|z)] - \mathrm{KL}(q_\phi(z_1|x)||p_\theta(z_1)) - \sum_{\ell=2}^{L} E_{q_\phi(z_{<\ell}|x)}[\mathrm{KL}(q_\phi(z_\ell|z_{<\ell}, x)||p_\theta(z_\ell|z_{<\ell}))], \tag{3}$$

where $q_\phi(z_{<\ell}|x) = \prod_{i=1}^{\ell-1} q(z_i|z_{<i}, x)$ is the approximate posterior up to latent group $(\ell - 1)$. Reparameterization is then used to provide Monte Carlo gradient estimates.

We can rewrite the conditional prior and approximate posterior for layer ℓ to make the set of relevant networks explicit:

$$\begin{aligned} p_\theta(z_\ell|z_{<\ell}) &= \mathcal{N}(z_\ell|\ \mu_{\theta_\ell}(z_{<\ell}),\ \sigma_{\theta_\ell}(z_{<\ell})), \\ q_\phi(z_\ell|z_{<\ell}, x) &= \mathcal{N}(z_\ell|\ \mu_{\phi_\ell}(f_{\phi_\ell}(x), g_{\phi_\ell}(z_{<\ell})),\ \sigma_{\phi_\ell}(...)). \end{aligned} \tag{4}$$

Here, f_{ϕ_ℓ} and g_{ϕ_ℓ} are networks that extract feature representations of the observation x and the previous layers $z_{<\ell}$, respectively. These features determine the mean and scale of the conditional Gaussian posterior via $\mu_{\phi_\ell}, \sigma_{\phi_\ell}$. Networks $\mu_{\theta_\ell}, \sigma_{\theta_\ell}$ similarly generate the prior parameters for layer ℓ.

2.3 METHODS FOR INFERRING MISSING DATA

After training, synthetic data may be easily generated from $p_\theta(x)$ by sampling z from the learned VAE or HVAE prior, and $x \sim p_\theta(x|z)$. However, the learned encoder does *not* provide a direct mechanism for conditional queries about missing features, given partial or corrupted test data.

Let $x = (x_O, x_M)$ be a test data point with observed features x_O and missing features x_M. We are particularly interested in scenarios where the specific feature dimensions that are missing and observed will vary across test instances. Given a trained (hierarchical) VAE, for which x_O and x_M are conditionally independent given z, missing data is optimally predicted via the conditional distribution:

$$p_\theta(x_M \mid x_O) = \int p_\theta(x_M \mid z) p_\theta(z \mid x_O)\, dz. \quad (5)$$

This approach (and QAVI) are valid as long as data is *missing-at-random* (MAR, Little and Rubin (2019); Mattei and Frellsen (2019)); the mechanism that removes features must be independent of x_M. Like most related work, our experiments use *missing-completely-at-random* (MCAR) feature masks whose distribution is independent of x. Exactly evaluating the predictive distribution (5) is infeasible due to the non-linear decoder, and intractable code posterior.

Heuristic Preprocessing. Imputation heuristics, such as replacing missing features with statistical summaries like their mean or mode, are widely used. Some work on training VAEs given partially missing data (Mattei and Frellsen (2019); Nazábal et al. (2020); Collier et al. (2020)) propose a *Fill-Zeros* heuristic, simply replacing missing features with zeros as input to the VAE encoder. While Fill-Zeros may be effective for learning models of MNIST digits (where zeros are common) when pixels are missing uniformly at random, our experiments show that in even slightly more complex scenarios, its performance is very poor.

Monte Carlo methods. Rezende et al. (2014) first proposed a simple scheme to approximately sample from $p_\theta(x_M \mid x_O)$ by starting with a random imputation, which is then stochastically encoded and decoded (or autoencoded) several times. Because the encoder only approximates the true posterior distribution $p_\theta(z \mid x)$, this *pseudo-Gibbs* sampler will not sample from the true posterior of missing features, and encoder inaccuracies may cause its equilibrium distribution to be far from $p_\theta(x_M \mid x_O)$. This approach was improved by Mattei and Frellsen (2018), who proposed a Metropolis-Hastings correction to each step of the pseudo-Gibbs sampler, inducing a *Metropolis-in-Gibbs* sampler that asymptotically samples from $p_\theta(x_M \mid x_O)$. While Metropolis-in-Gibbs converges to the true posterior, it does so at a rate that may be impractically slow.

Amortized Inference for Imputation. Heuristic preprocessing may be avoided by learning a new "partial" encoder approximating $p_\theta(z \mid x_O)$. Collier et al. (2020) concatenates zero-filled data with a binary mask indicating missing features, and optimizes the standard VAE ELBO of Eq. (2), but with the log-likelihood term calculated on only the observed data x_O. While this approach was developed to *train* VAEs with missing data, it is trivially applicable to test-time missing data imputation by retraining the encoder with the masked test data, generating a *Re-tuned Encoder*.

Posterior Matching (Strauss and Oliva, 2022) instead artificially masks the complete training data x as x_O, and tunes a partial-encoder $q_\psi(z \mid x_O)$ to "match" the pre-trained encoder by maximizing $E_{z\sim q_\phi(z|x)}[\log q_\psi(z \mid x_O)]$. In concurrent work, Harvey et al. (2022) introduced an equivalent approach, calling it *Inference in a Pretrained Artifact* (*IPA*). Ivanov et al. (2019) optimize the partial-encoder as in Posterior Matching, but simultaneously retrain the encoder and decoder to produce a conditional model $p(x_M \mid x_O)$. Their model includes skip connections between the partial-encoder and decoder networks, as in the HVAE feature representations f_{ϕ_l} in Eq. (4). We adapt their *VAEAC* (arbitrarily conditioned) training to HVAEs by fine-tuning all three networks, starting from a pre-trained HVAE.

These amortized approaches to missing data imputation with VAEs have several drawbacks compared to our QAVI method. *1)* They incur a substantial initial overhead for training the partial-encoder. *2)* Partial-encoder training requires access to a relatively large set of partially-observed examples, and/or continued access to the training set along with a known missing-feature distribution. *3)* They are sensitive to shifts in the distribution of queries (missing-feature patterns) between training and evaluation. If evaluated on previously unseen missing-feature patterns, performance may suffer substantially (see Fig. 4). *4)* As we will demonstrate, even without distribution shifts, performance can be sub-optimal.

3 QUERY ADAPTIVE VI

Our *query adaptive variational inference* (QAVI) utilizes the pre-trained VAE decoder, defines variational free parameters for each inference query, and refines them. We give an overview in Fig. 1 and show that it is adaptive across different queries without the need for additional partial-encoder networks. QAVI defines an explicit variational posterior over the unobserved variables $q_\lambda(z, x_M)$ with query-specific parameters λ. Fig. 1 shows two possible factorizations of the latent codes z and missing features x_M, leading to a pair of QAVI algorithms that we elaborate below.

3.1 VI VIA MISSING FEATURE POSTERIORS

Suppose first that the variational posterior factorizes as

$$q_\lambda(x_M, z \mid x_O) = q_\lambda(x_M) q_\lambda(z \mid x_M, x_O).$$

We fix $q_\lambda(z \mid x_M, x_O)$ to the amortized encoder $q_\phi(z|x)$, which has been pre-trained to approximate the relationship

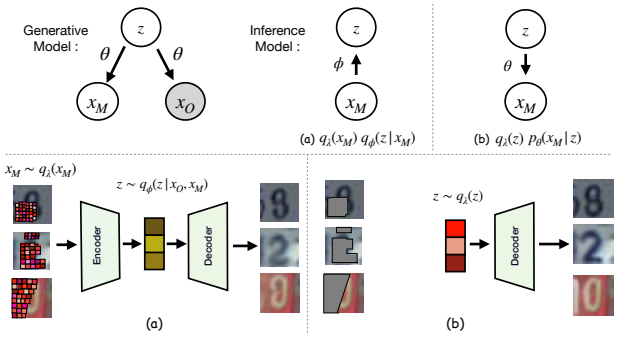

Figure 1: Overview of Feat. QAVI (*a*, Sec. 3.1) and non-amortized QAVI (*b*, Sec. 3.2). *Top:* Generative model for unobserved variables x_M, z and observed data x_O (shaded). We show two possible inference models for the joint variational distribution $q_\lambda(x_M, z)$ on latent variables. In (*a*), missing features are modeled directly by specifying $q_\lambda(x_M)$. In (*b*), queries are modeled indirectly via a tunable latent code distribution $q_\lambda(z)$. QAVI is easily adapted to existing VAEs by reusing generative (decoder) networks p_θ, and possibly inference (encoder) networks q_ϕ. *Bottom:* Computational flow of QAVI for each posterior factorization. In Feat. QAVI (*a*), samples from $q_\lambda(x_M)$ are autoencoded by the pre-trained VAE to compute the ELBO. In non-amortized QAVI (*b*), samples from $q_\lambda(z)$ are only passed through the decoder; the pre-trained encoder is not needed.

between x and z. Our *Feat. QAVI* method then defines an explicit posterior $q_\lambda(x_M)$ on missing features. This approach directly captures uncertainty in the posterior of missing features (see Fig. 1) and their impact on the latent code.

In experiments, we found that fitting $q_\lambda(x_M)$ via the standard ELBO (see supplement for derivation) resulted in posteriors with unrealistically small variance. Following Higgins et al. (2017), we thus employ hyperparameters β to more strongly encourage the latent-code posterior to align with the prior, and γ to increase the entropy of the missing-feature posterior. The variational objective $\mathcal{L}_M$ is then:

$$\mathcal{L}_M(\lambda; x) = E_{q_\lambda(x_M)}\big[E_{q_\phi(z|x_O,x_M)}[\log p_\theta(x_O, x_M|z)] - \beta \mathrm{KL}(q_\phi(z|x_M, x_O)\,||\,p(z))\big] + \gamma H(q_\lambda(x_M)), \quad (6)$$

where H is the entropy. We approximate $\mathcal{L}_M$ with S Monte Carlo samples as in standard VAE training:

$$\mathcal{L}_M(\lambda; x) \approx \frac{1}{S}\sum_{s=1}^{S}\big[\log p_\theta(x_O, x_M^{(s)}|z^{(s)}) - \beta \mathrm{KL}(q_\phi(z|x_O, x_M^{(s)})\,||\,p(z))\big] + \gamma H(q_\lambda(x_M)), \quad (7)$$

where $x_M^{(s)} \sim q_\lambda(x_M)$, $z^{(s)} \sim q_\phi(z|x_O, x_M^{(s)})$, and KL and H are calculated in closed form. Automatic differentiation is used to compute gradients with respect to λ, tuning the missing-feature distributions to observed data indirectly.

While Feat. QAVI directly models the uncertainty in the posterior distribution of missing features, its optimization requires repeated computation of the encoder and decoder. It also inherits any suboptimalities of the trained encoder. Perhaps surprisingly, we will show that in this context, a fully non-amortized inference method can have both greater computational efficiency and greater accuracy.

3.2 NON-AMORTIZED VI

We can construct an alternative variational posterior via the following factorization (see Fig. 1):

$$q_\lambda(x_M, z) = q_\lambda(z) q_\lambda(x_M|z),$$

By defining a variational posterior on z, we no longer use the encoder (except possibly for initialization), and re-use the pre-trained decoder by fixing $q_\lambda(x_M \mid z) = p_\theta(x_M \mid z)$. As derived in the supplement, this leads to the following *non-amortized QAVI* variational objective:

$$\mathcal{L}_N(\lambda; x) = E_{q_\lambda(z)}[\log p_\theta(x_O|z)] - \beta \mathrm{KL}(q_\lambda(z)||p(z)). \quad (8)$$

To evaluate $\mathcal{L}_N$ we must only explicitly sample from the latent code; missing data may be *analytically* marginalized. Non-amortized QAVI seeks latent-code distributions that assign high likelihood to the observed features x_O, and are aligned with the prior via weight $\beta > 1$. In the simplest case where $q_\lambda(z)$ is a diagonal-covariance Gaussian, Eq. (8) is approximated via S samples from $q_\lambda(z)$, and the code mean and variance λ optimized via stochastic gradient ascent.

While VAE training implicitly encourages approximately Gaussian posteriors given *complete* observations, for queries given missing data, posteriors are often multi-modal and poorly approximated by a Gaussian $q_\lambda(z)$. To address this, we extend non-amortized QAVI to more expressive variational distributions that better capture the true posterior.

Flow Posteriors. The flow-based variational posterior aims to construct a complex distribution by transforming a simple Gaussian through a series of invertible mappings. We let $z_t = \mathcal{T}_t(z_{t-1}, \lambda_t)$ for $t = 1, \ldots, T$, where z_0 is sampled from a Gaussian base distribution with parameters λ_0. λ_t is the set of parameters specifying flow layer $\mathcal{T}_t$, T is the total number of flow transformations, and $\lambda = \{\lambda_0, \lambda_1, ..., \lambda_T\}$.

The idea of improving amortized variational inference in VAEs with the help of *normalizing flows* (Tabak and Turner, 2013; Tabak and Vanden-Eijnden, 2010) was first proposed by Rezende and Mohamed (2015). We instead employ autoregressive transformations in each layer $\mathcal{T}_t$ to capture high-dimensional dependencies in the latent space, producing *inverse autoregressive flows* (IAF, Kingma et al. (2016)). We approximate both terms in $\mathcal{L}_N(\lambda; x)$ of Eq. (8) via S samples from $q_\lambda(z_T)$; each Gaussian sample from the base distribution is transformed by T flow layers. Query-specific parameters λ are optimized by stochastic backpropagation.

Gaussian Mixture Posteriors. Parameterizing the latent space distribution $q_\lambda(z)$ as a *mixture of Gaussians* enables

us to explicitly model different hypotheses in the latent space. Let $q_\lambda(z) = \sum_{t=1}^T w_t \mathcal{N}(z \mid \mu_t, \Lambda_t)$, where μ_t and Λ_t are the means and (diagonal) covariances of the T mixture components (posterior modes), w_t are mixture weights ($\sum_{t=1}^T w_t = 1$), and $\lambda = \{w_t, \mu_t, \Lambda_t\}_{t=1}^T$. Increasing T allows inference of more accurate posterior approximations.

Optimizing mixture parameters λ is not straightforward as discrete resampling from mixture weights cannot be continuously reparameterized. We use *implicit reparameterization gradients* (IRG, Figurnov et al. (2018)) to efficiently compute gradients of the mixture component means and covariances. While in principle IRG could also be used to estimate gradients of mixture weights (Graves, 2016), in practice this estimator has enormous variance when posterior modes are widely separated. We instead adapt an importance-sampling gradient estimator (Ścibior et al., 2021) for mixture weights; see supplement for details. Multiple samples S from the variational posterior are *necessary* to capture the impact of multiple posterior modes on the ELBO (8).

Hierarchical VAE Posteriors. For the hierarchical VAEs (Sønderby et al., 2016) introduced in Sec. 2, neither Gaussians nor Gaussian-mixtures are flexible enough to capture the non-linear dependencies between latent variables at different levels of the hierarchy. We therefore propose a new, more expressive (non-amortized) variational family for HVAEs that removes dependency on the observation x, while retaining the expressive and non-linear dependencies of the HVAE model. Our variational posterior factorizes as:

$$q_\lambda(z) = q_\lambda(z_1) \prod_{\ell=2}^{L} q_\lambda(z_\ell | z_{<\ell}). \tag{9}$$

As in the HVAE decoder, we let $q_\lambda(z_\ell | z_{<\ell})$ be conditionally Gaussian for all l. More complex conditional distributions (such as flows or mixtures) could also be used, but this conditionally Gaussian structure alone allows for expressive, multi-modal posterior approximations.

We propose a simple, generic strategy for constructing a family of non-amortized distributions given a pre-trained HVAE. Our approach applies to many recent hierarchical VAE architectures, including the "very-deep" HVAE (Child, 2021) that we use in experiments. To specify the non-amortized QAVI posterior, we begin with the amortized posterior defined in Eq. (4). Holding $g_{\phi_\ell}, \mu_{\phi_\ell}, \sigma_{\phi_\ell}$ fixed, we replace the features extracted from the observation with a new tunable parameter λ_ℓ. A further set of weighting parameters $\gamma_\ell, \gamma'_\ell \in [0, 1]$ interpolate these output parameters with those of the prior. Thus our hierarchical QAVI posterior for layer ℓ becomes:

$$\begin{aligned} \mu_\ell &= \gamma'_\ell \mu_{\phi_\ell}(\lambda_\ell, g_{\phi_\ell}(z_{<\ell})) + (1 - \gamma'_\ell)\mu_{\theta_\ell}(z_{<\ell}), \\ \sigma_\ell &= \gamma_\ell \sigma_{\phi_\ell}(\lambda_\ell, g_{\phi_\ell}(z_{<\ell})) + (1 - \gamma_\ell)\sigma_{\theta_\ell}(z_{<\ell}), \\ q_\lambda(z_\ell \mid z_{<\ell}) &= \mathcal{N}(z_\ell \mid \mu_\ell, \sigma_\ell). \end{aligned} \tag{10}$$

Re-using the pre-trained networks $g_{\phi_\ell}, \mu_{\phi_\ell}, \sigma_{\phi_\ell}$ allows the full amortized encoder to be used for initialization of λ_ℓ by simply setting $\lambda_\ell \leftarrow f_{\phi_\ell}(x_O, \tilde{x}_M)$. $\tilde{x}_M$ may be any initialization for the missing features, even Gaussian noise.

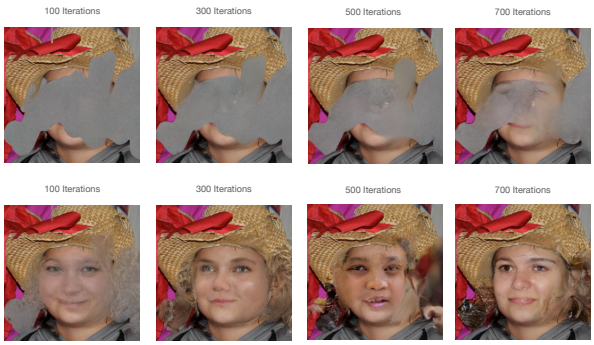

Figure 2: Comparison of QAVI optimization for HVAEs without (*top*) and with (*bottom*) our KL-balanced warmup.

Our approach of interpolating posterior ($\mu_{\phi_\ell}, \sigma_{\phi_\ell}$) and prior ($\mu_{\theta_\ell}, \sigma_{\theta_\ell}$) network outputs is vital when reusing μ_{ϕ_ℓ} and σ_{ϕ_ℓ} from the original inference model. In the original, fully-observed training phase, posterior variances may become extremely small for the highly overparameterized HVAE model. But with missing data, latent variables corresponding to unobserved features should have distributions close to the prior. Expressing the variational posterior as a weighted combination of prior and posterior network outputs allows our variational family to easily produce appropriate posteriors for latent variables corresponding to both observed and missing features, without needing to re-train μ_{ϕ_ℓ} and σ_{ϕ_ℓ}.

Hierarchical VAE Warmup. For hierarchical VAEs with a multi-scale architecture, we find that a warmup phase of optimization with a modified objective greatly accelerates posterior fitting. This idea is broadly proposed by Vahdat and Kautz (2020), and refined as follows: $\mathcal{L}_H(\theta, \phi; x_O) =$

$$\begin{aligned} &E_{q_\lambda(z)}[\log p_\theta(x_O|z)] - \frac{1}{d_1}\mathrm{KL}(q_\lambda(z_1) \,||\, p_\theta(z_1)) \\ &- \sum_{\ell=2}^{L} \frac{1}{d_\ell} E_{q_\lambda(z_{<\ell})}[\mathrm{KL}(q_\lambda(z_\ell|z_{<\ell}) \,||\, p_\theta(z_\ell|z_{<\ell}))], \end{aligned} \tag{11}$$

where d_ℓ is proportional to the size of the latent space at layer ℓ. Intuitively, when inpainting large segments of an image, high-level structures should be determined first and details refined later. But for the unmodified ELBO, higher resolution latent layers contribute substantially more to the loss, leading to slow convergence. Fig. 2 illustrates the dramatic effect of this change during QAVI optimization.

4 EXPERIMENTS & RESULTS

4.1 EXPERIMENTAL SETUP

We evaluate QAVI[1] using six tabular datasets from the UCI Machine Learning Database (Kelly et al., 2017). For tab-

[1]Code available: `https://github.com/SakshiAgarwal/QAVI`

ular data, we follow the experimental setup of Mattei and Frellsen (2019) to train our VAEs, but use a Gaussian variational posterior instead of Student's t.

We also consider three image datasets: real-valued MNIST (LeCun et al. (2010)), Street View House Numbers (SVHN) (Netzer et al., 2011), and FFHQ-256 (Karras et al., 2019). We use a single-stochastic-layer VAE for MNIST and SVHN, with a WideResNet architecture (Zagoruyko and Komodakis, 2016) that is known to work well with images. We train them with fully-observed images from the training set (70,000 MNIST images and 73,257 SVHN images) and maximize the ELBO of Eq. (2). For FFHQ-256, we adopt the "very-deep" hierarchical architecture of Child (2021), but for efficient comparison we re-trained a smaller variant of their original HVAE (5.9M vs. 115M parameters).

While QAVI handles missing-at-random (MAR) data naturally, we setup our experiments to be consistent with most prior work on imputation with VAEs. For MNIST and SVHN we consider two missing-completely-at-random (MCAR) patterns: 1) two randomly placed patches (each of size 10x10 for MNIST, 15x15 for SVHN); 2) a randomly rotated mask of half of the image. We use a more challenging random mask distribution (Zhao et al., 2021) for FFHQ-256. For tabular datasets, we corrupt the test set by removing half of the features in each row uniformly at random.

Baselines. We compare QAVI with several methods from the literature: *i)* The *Fill Zeros* heuristic (Nazábal et al., 2020); *ii)* Monte Carlo methods: *pseudo-Gibbs* (Rezende et al., 2014) and *Metropolis-in-Gibbs* (Mattei and Frellsen, 2018); *iii)* Amortized inference methods: *Re-tuned Encoder* (Collier et al., 2020) and *Posterior Match[ing]* (Strauss and Oliva, 2022).We consider three variants of posterior matching to evaluate the importance of knowing the missing-feature pattern during training. *Posterior Match (True)* trains the posterior matching encoder with exactly the same query distribution used for test evaluation. *Posterior Match (Rand.)* assumes that only the fraction of missing features is known; features are removed at random with this probability. The generic *Posterior Match* uses the image masking distribution of Zhao et al. (2021), which assumes contiguous masked regions without specific knowledge of queries to be evaluated.

We compare also QAVI to state-of-the-art approaches to inpainting with HVAEs: *Posterior Match[ing]* and *VAEAC* (Ivanov et al., 2019). Both *Posterior Match* and *VAEAC* benefitted from training with the same random mask distribution (Zhao et al., 2021) used in evaluation. As our "very-deep" HVAE architecture already links the encoder and decoder, we do not include extra deterministic skip connections as in the original VAEAC architecture.

Hyperparameters. QAVI optimization for VAE models uses $S = 100$ Monte Carlo samples to estimate our variational objective and gradients, and optimizes variational parameters λ for 300 steps using Adam (Kingma and Ba, 2015). We refer to our posterior on missing features (Sec. 3.1) as *Feat. QAVI*, and our Gaussian/Flow/Mixture non-amortized variational posteriors (Sec. 3.2) on latent codes z as *Gaus./Flow/Mix. QAVI*. For HVAE models, QAVI optimization uses $S = 28$ samples to estimate the ELBO, and optimizes for 1000 total steps (including 500 warmup steps as in Fig. 2). See supplement for additional details.

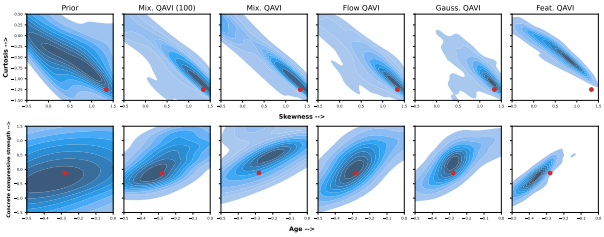

Figure 3: Kernel density visualizations of the prior and approximate posteriors for five QAVI variants, fit using $S = 1000$ samples from the optimized variational distribution. We show one test sample chosen from two UCI tabular datasets, Banknote (*top*) and Concrete (*bottom)*, containing two missing features (labeled on axes). We observe that the true data point (*red*) is often enclosed by the different posteriors, but sometimes missed by Feat. QAVI. We approximate the true posterior via a highly expressive QAVI mixture of 100 Gaussians, fit by extended optimization.

Metrics. For a quantitative analysis, we estimate the marginal log-likelihood of the missing features x_M using the importance sampling estimator of Burda et al. (2016):

$$\log p_\theta(x_M) \geq \mathbb{E}_{z^{(s)}\sim q(z)}\left[\log \frac{1}{S}\sum_{s=1}^{S}\frac{p_\theta(x_M, z^{(s)})}{q(z^{(s)})}\right]$$
$$\approx \log \frac{1}{S}\sum_{s=1}^{S}\frac{p_\theta(x_M, z^{(s)})}{q(z^{(s)})}, \qquad z^{(s)} \sim q(z). \quad (12)$$

The true log-likelihood of missing features is constant for all inference methods, since the generative model $p_\theta(x)$ is fixed, but better posterior approximations lead to tighter lower bounds for a fixed number of samples S. Fig. 4 shows estimated log-likelihoods versus the number of samples.

As the importance-weighted likelihood estimator becomes expensive and unreliable to evaluate for high-dimensional models like HVAEs, we use perceptual metrics to evaluate inpainting results for HVAEs on the FFHQ dataset. Table 2 reports three metrics on a test set of 1000 images: FID (Heusel et al., 2017) as well as P-IDS and U-IDS (Zhao et al., 2021). We modify P-IDS and U-IDS slightly to reduce sensitivity to the test set size; see supplement for details.

4.2 RESULTS

QAVI improves imputation quality. Fig. 4 and Table 1 compare the log-likelihood of missing features across MNIST, SVHN, and tabular datasets. We see that heuristic

Table 1: Test missing data log-likelihoods (LL, higher is better) and normalized root mean-square error (NRMSE, lower is better) for 6 tabular datasets from the UCI repository, estimated using S =10,000 samples. NRMSE per test row is the minimum across S samples. QAVI variants have superior performance (highlighted in bold) for almost all data.

	Breast Cancer		Red wine		White wine		Banknote		Concrete		Yeast	
	LL	NRMSE	LL	NRMSE	LL	NRMSE	LL	NRMSE	LL	NRMSE	LL	NRMSE
Mix. QAVI	**-9.16**	0.39	**-6.35**	**0.26**	-7.03	0.35	**-2.25**	**0.10**	-2.70	**0.17**	+2.42	**0.46**
Flow QAVI	**-9.14**	0.39	-6.47	**0.26**	-7.03	0.35	**-2.24**	**0.10**	**-2.65**	**0.17**	+2.55	**0.46**
Gaus. QAVI	-9.21	0.39	-6.56	0.30	**-6.94**	**0.30**	-2.35	**0.10**	-2.82	**0.17**	**+2.82**	**0.46**
Feat. QAVI	-16.32	0.37	-13.20	0.28	-8.80	0.38	-8.27	0.15	-15.14	0.23	-7.81	0.47
Posterior Match	-15.14	**0.33**	-14.83	0.42	-9.19	0.39	-4.42	0.14	-15.05	0.40	-430.36	0.47
Re-tuned Encoder	-12.82	0.40	-12.96	0.40	-9.59	0.39	-4.26	0.14	-11.48	0.35	-33.62	0.50
Metropolis-in-Gibbs	-23.94	0.44	-74.22	0.68	-19.83	0.58	-49.51	0.68	-265.62	0.62	-1.86	0.54
pseudo-Gibbs	-12.61	**0.33**	-33.53	0.36	-10.68	0.36	-44.16	0.40	-249.06	0.36	-5.11	0.48
Fill Zeros	-32.56	0.40	-21.97	0.36	-10.09	0.37	-17.46	0.32	-23.33	0.32	-11.27	0.48

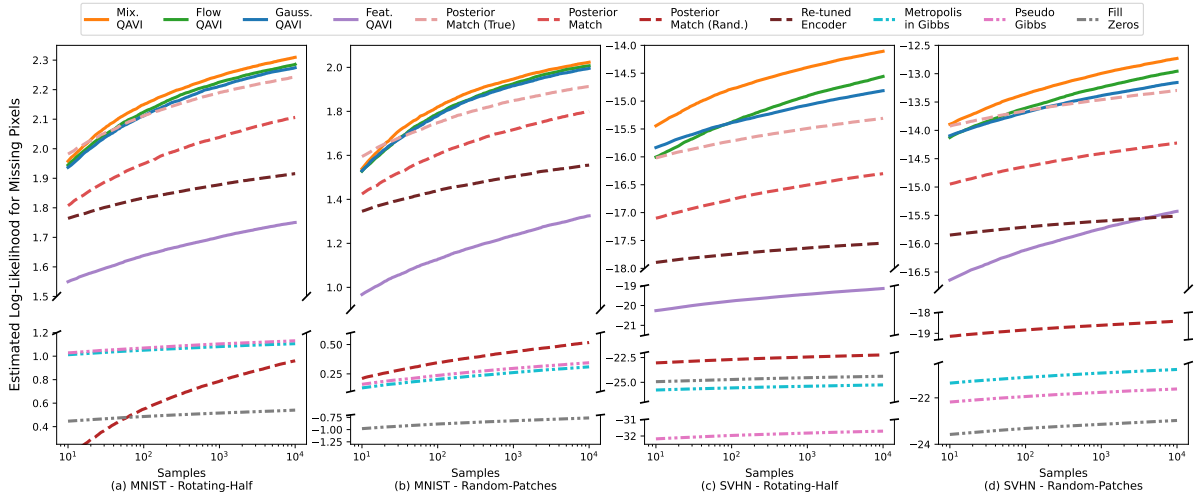

Figure 4: For two image mask distributions and several inference methods (*top*), we plot importance-weighted log-likelihood estimates for missing pixels and varying samples S. We average over 1000 MNIST (*left*) or SVHN (*right*) test images.

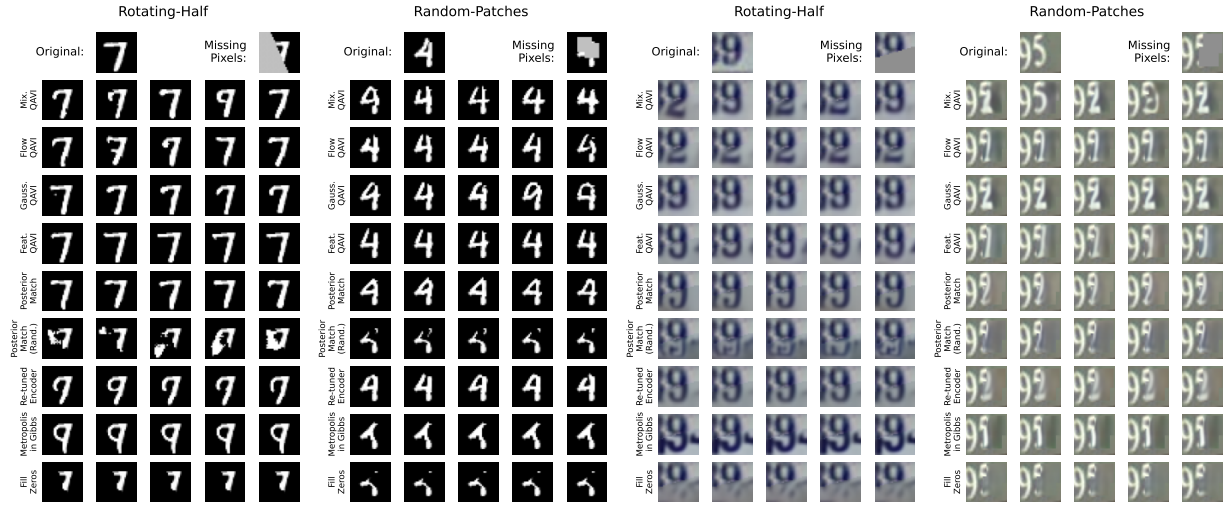

Figure 5: Digit completion results on MNIST (*left*) and SVHN (*right*) images for inference queries with pixels obscured by the Rotating-Half or Random-Patches distributions. We show 5 samples from each inferred posterior. Monte Carlo and amortized inference methods propose one sometimes-valid digit completion; amortized VI is typically more accurate. The performance of Posterior Match varies widely depending on the missing-feature distribution it is trained on. In contrast, QAVI automatically adapts to queries, and proposes multiple valid imputations that effectively capture posterior uncertainty.

Figure 6: Inpainting results on the FFHQ-256 dataset, comparing our non-amortized deep QAVI inpainting with VAEAC and Posterior Matching. We also compare QAVI results for the reduced-size models used in Table 1 to inpaintings from the original "very-deep" HVAE of Child (2021). We show the true and masked images, and 5 posterior samples for each method.

imputation and Monte Carlo methods perform poorly. Retuned Encoder and Posterior Match show relatively higher likelihoods, but do not match QAVI across any dataset or missingness pattern. Feat. QAVI is competitive for tabular data, but for high-dimensional images it is susceptible to local optima. Fig. 7 similarly shows that QAVI provides reliably strong performance for downstream tasks.

QAVI can capture multi-modal posterior uncertainty. We show imputations for four test examples in Fig. 5 to highlight differences in inference methods, and to explore the uncertainty in the inferred posterior over the missing features. We see that Gaussian and Flow QAVI defined on the latent space are capable of capturing uncertainty in the missing features. With a mixture of Gaussians variational family, QAVI produces multiple visually-plausible imputations. The classification performance in Fig. 7 shows high relative classification accuracy for expressive posteriors despite the increased variance in samples.

QAVI benefits extend to HVAEs. We find that QAVI also integrates well with hierarchical VAE models. The results in Table 2 show that our QAVI approach for HVAEs produces imputations with higher perceptual scores than prior methods for leveraging HVAEs for inpainting. Figure 6 shows that samples produced by QAVI are qualitatively more visually plausible, and also capture substantial diversity in possible feature imputations.

Amortized imputation is sensitive to training queries. Fig. 4 compares the performance for Posterior Matching when trained with different masking distributions. We see

Table 2: Quantitative comparison of perceptual inpainting quality on the FFHQ-256 dataset. We compare QAVI against two state-of-the-art adaptations of HVAEs to inpainting, using the same base "very-deep" HVAE architecture.

Method	FID ↓	P-IDS* ↑	U-IDS* ↑
QAVI	**21.21**	**6.20**	**24.98**
Posterior Match	23.68	3.36	21.55
VAEAC	26.41	2.31	18.19

that with the absence of any prior knowledge of the true masking distribution at train time, the performance of Posterior Matching can be as poor as simple heuristics like Fill Zeros. Even in the unrealistic case where the exact distribution of missingness is known at train time, posterior matching does not outperform QAVI. This sensitivity to the choice of missingness for training is significant: adaptation to new patterns of missingness require full retraining. In sensitive domains such as medicine, access to the original training set may be restricted or even impossible. QAVI is indifferent to the structure of queries, requires no retraining, and *still* outperforms the "best case" amortized imputation.

QAVI smoothly trades off time and performance. Fig. 8 illustrates the tradeoff between performance and optimization time for three variants of non-amortized QAVI, for 100 images from the MNIST dataset. We see that the performance of Posterior-Matching is far lower than QAVI, *and* requires substantial overhead to train the partial encoder (over 3.5 hours). Gaussian and Flow QAVI converge in about one minute. Mixture QAVI converges a bit more slowly, but ultimately reaches the best solutions of any method. Posterior-Match amortization would have computational advantages for very-large query sets (thousands of images), but would still have inferior inference accuracy.

5 CONCLUSION

We have presented a simple and a general framework that has been unexplored in prior work employing VAEs for the imputation of missing data. Previous state-of-the-art approaches make use of a restrictive inference network as an imputation strategy. We instead take an existing VAE generative model (decoder), allocate variational parameters for the latent code of each missing data point, and train the parameters stochastically to optimize the induced variational bound. The simple structure of our bounds enables efficient and accurate approximation of the posterior distribution of missing features, given any pattern of observed features.

We evaluated QAVI on a variety of VAEs, including current state-of-the-art hierarchical VAEs, and several datasets. We found that non-amortized QAVI with Gaussian, and especially Flow or Mixture, posterior approximations outperforms previous heuristic and amortized inference methods data imputation with VAEs. Importantly, we find that a

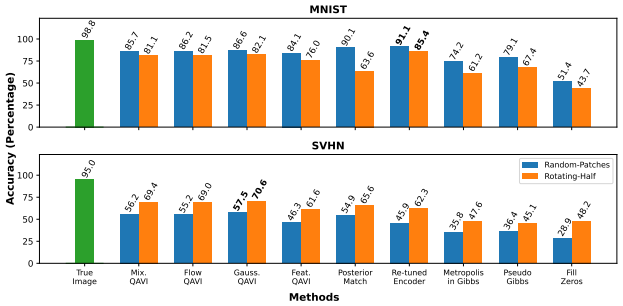

Figure 7: Classification accuracies using 100 samples from the inferred posterior for randomized missing queries on MNIST (*top*) and SVHN (*bottom*). We use a trained discriminative model, with WRN-28-2 architecture (Zagoruyko and Komodakis, 2016), to predict class labels.

Gaussian Mixture posterior is able to effectively capture the multi-modality that often arises given missing data.

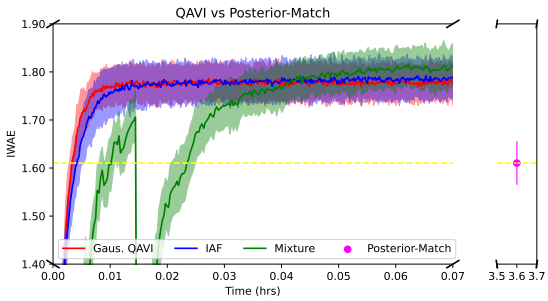

Figure 8: Importance-weighted log-likelihood (IWAE) estiamates for missing pixels versus wall-clock training time (in hours). Likelihoods are estimated at each step of optimization for 100 MNIST images with pixels missing via Random-Patches. We plot the mean and standard deviation across 10 runs of QAVI methods. We compare to Posterior-Match, whose amortized inference network requires over 3.5 hours to train. The dip in mixture log-likelihoods occurs at random re-initialization of mixture parameters to avoid local optima; see supplement Sec. 2.4 for details.

In this work, we do not consider *missing-not-at-random* (MNAR) data (Ipsen et al., 2021), but we conjecture that QAVI will provide a foundation for future advances in MNAR inference. QAVI provides a simple, effective, and general approach for inference of missing data with arbitrary patterns, that is attractive when queries are unknown during training and uncertainty in missing data is high.

Acknowledgements

This research supported in part by NSF Robust Intelligence Award No. IIS-1816365, and by the HPI Research Center in Machine Learning and Data Science at UC Irvine.

References

Aeberhard, S. and Forina, M. (1991). Wine. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5PC7J.

Burda, Y., Grosse, R. B., and Salakhutdinov, R. (2016). Importance weighted autoencoders. In Bengio, Y. and LeCun, Y., editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

Child, R. (2021). Very deep {vae}s generalize autoregressive models and can outperform them on images. In *International Conference on Learning Representations*.

Collier, M., Nazabal, A., and Williams, C. (2020). VAEs in the presence of missing data. In *ICML Workshop on the Art of Learning with Missing Values (Artemiss)*.

Cremer, C., Li, X., and Duvenaud, D. (2018). Inference suboptimality in variational autoencoders. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1078–1086. PMLR.

Figurnov, M., Mohamed, S., and Mnih, A. (2018). Implicit reparameterization gradients. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Gershman, S., Hoffman, M. D., and Blei, D. M. (2012). Nonparametric variational inference. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress.

Ghahramani, Z. and Beal, M. J. (2001). Propagation algorithms for variational Bayesian learning. In *NeurIPS 13*, pages 507–513. MIT Press.

Graves, A. (2016). Stochastic backpropagation through mixture density distributions. *arXiv preprint arXiv:1607.05690*.

Harvey, W., Naderiparizi, S., and Wood, F. (2022). Conditional image generation by conditioning variational autoencoders. In *International Conference on Learning Representations*.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, 14(1):1303–1347.

Ipsen, N. B., Mattei, P.-A., and Frellsen, J. (2021). not-MIWAE: Deep generative modelling with missing not at random data. In *International Conference on Learning Representations*.

Ivanov, O., Figurnov, M., and Vetrov, D. (2019). Variational autoencoder with arbitrary conditioning. In *International Conference on Learning Representations*.

Jaakkola, T. S. and Jordon, M. I. (1999). *Improving the Mean Field Approximation via the Use of Mixture Distributions*, page 163–173. MIT Press, Cambridge, MA, USA.

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

Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410.

Kelly, M., Longjohn, R., and Nottingham, K. (2017). UCI machine learning repository.

Kim, Y., Wiseman, S., Miller, A., Sontag, D., and Rush, A. (2018). Semi-amortized variational autoencoders. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2678–2687. PMLR.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. (2016). Improved variational inference with inverse autoregressive flow. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 4743–4751, Red Hook, NY, USA. Curran Associates Inc.

Kingma, D. P. and Welling, M. (2014). Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Kingma, D. P. and Welling, M. (2019). An introduction to variational autoencoders. *Foundations and Trends in Machine Learning*, 12(4):307–392.

Klushyn, A., Chen, N., Kurle, R., Cseke, B., and van der Smagt, P. (2019). Learning hierarchical priors in vaes. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Koller, D. and Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. MIT press.

Krishnan, R. G., Liang, D., and Hoffman, M. D. (2017). On the challenges of learning with inference networks on sparse, high-dimensional data. *ArXiv*, abs/1710.06085.

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D. M. (2017). Automatic differentiation variational inference. *Journal of Machine Learning Research*, 18(1):430–474.

LeCun, Y., Cortes, C., and Burges, C. (2010). Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2.

Little, R. J. A. and Rubin, D. B. (2019). *Statistical analysis with missing data*, volume 793. John Wiley & Sons.

Lohweg, V. (2013). banknote authentication. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C55P57.

Marino, J., Yue, Y., and Mandt, S. (2018). Iterative amortized inference. In *Proceedings of the 35th International Conference on Machine Learning*, pages 3403–3412.

Mattei, P.-A. and Frellsen, J. (2018). Leveraging the exact likelihood of deep latent variable models. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Mattei, P.-A. and Frellsen, J. (2019). MIWAE: Deep generative modelling and imputation of incomplete data sets. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4413–4423. PMLR.

Mnih, A. and Gregor, K. (2014). Neural variational inference and learning in belief networks. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, page II–1791–II–1799. JMLR.org.

Nakai, K. (1996). Yeast. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5KG68.

Nazábal, A., Olmos, P. M., Ghahramani, Z., and Valera, I. (2020). Handling incomplete heterogeneous data using vaes. *Pattern Recognition*, 107:107501.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*.

Paisley, J. W., Blei, D. M., and Jordan, M. I. (2012). Variational Bayesian inference with stochastic search. In *International Conference on Machine Learning*.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufman, San Mateo.

Rezende, D. and Mohamed, S. (2015). Variational inference with normalizing flows. In Bach, F. and Blei, D., editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1530–1538, Lille, France. PMLR.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In Xing, E. P. and Jebara, T., editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, Bejing, China. PMLR.

Ścibior, A., Masrani, V., and Wood, F. (2021). Differentiable particle filtering without modifying the forward pass. *International Conference on Probabilistic Programming (PROBPROG)*.

Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K., and Winther, O. (2016). Ladder Variational Autoencoders. In *Advances in Neural Information Processing Systems*.

Strauss, R. and Oliva, J. (2022). Posterior matching for arbitrary conditioning. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K., editors, *Advances in Neural Information Processing Systems*.

Tabak, E. G. and Turner, C. V. (2013). A family of nonparametric density estimation algorithms. *Communications on Pure and Applied Mathematics*, 66.

Tabak, E. G. and Vanden-Eijnden, E. (2010). Density estimation by dual ascent of the log-likelihood. *Communications in Mathematical Sciences*, 8:217–233.

Vahdat, A. and Kautz, J. (2020). NVAE: A deep hierarchical variational autoencoder. In *Neural Information Processing Systems (NeurIPS)*.

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

Winn, J. and Bishop, C. M. (2005). Variational message passing. *JMLR*, 6:661–694.

Wolberg, W., Mangasarian, O., Street, N., and W., S. (1995). Breast Cancer Wisconsin (Diagnostic). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5DW2B.

Yeh, I.-C. (2007). Concrete Compressive Strength. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5PK67.

Zagoruyko, S. and Komodakis, N. (2016). Wide residual networks. In *BMVC*.

Zhao, S., Cui, J., Sheng, Y., Dong, Y., Liang, X., Chang, E. I., and Xu, Y. (2021). Large scale image completion via co-modulated generative adversarial networks. In *International Conference on Learning Representations (ICLR)*.