A MINIMUM-VARIANCE PERSPECTIVE ON THE ENCODER AND ELBO IN VAES

Anonymous authors
Paper under double-blind review

ABSTRACT

Variational Autoencoders (VAEs) are widely used for generative modeling, denoising, super-resolution, and compression, and are characterized by their encoder–decoder architecture. Yet, from the perspective of the generative model, the encoder is not required: the marginal likelihood is defined solely by the decoder and the latent prior, while the encoder introduces additional trainable parameters. Then, *why should an encoder be part of a generative model?* In this paper, we show that the encoder–decoder structure of VAEs can be derived from a minimum-variance principle and it is not only an inherited architectural choice. Starting from weak assumptions and considering the finite-budget Monte Carlo estimation of marginal likelihoods, we use a classical result from importance-sampling theory to derive the variance-minimizing proposal distribution. This proposal is input-dependent, and its amortized implementation necessarily induces an encoder that maps observations x to a distribution over latent variables z, followed by a decoder that generates x from z. Thus, the encoder–decoder structure emerges as a variance-optimal construction for likelihood estimation.

We further show that the ELBO is consistent with the same principle: beyond providing a tractable lower bound on the marginal log-likelihood, its optimization is aligned with reducing the variance of the corresponding importance-sampling estimator and with approaching the variance-minimizing proposal. This establishes a common statistical foundation for both the VAE architecture and its training objective. Experiments on controlled latent-variable benchmarks confirm the predicted effect: encoder-based models achieve lower importance-sampling estimator variance and higher likelihood than encoder-free alternatives, with the advantage increasing as the latent dimensionality grows.

1 INTRODUCTION

Variational Autoencoders (VAEs) (Kingma & Welling, 2014) are a foundational class of latent-variable models with applications spanning image generation, data compression, denoising, super-resolution, and representation learning (Doersch, 2016). Their standard implementation consists of two coupled components: an encoder that maps an observation x to a distribution over latent variables z, and a decoder that maps latent variables back to the observation space. This encoder–decoder structure is now so pervasive that it is often regarded as an intrinsic feature of the VAE architecture.

From the perspective of the generative model, however, the role of the encoder is not obvious. A latent-variable generative model is fully specified by a prior $p(z)$ and a conditional likelihood $p_\theta(x \mid z)$; its marginal likelihood $p_\theta(x) = \int p_\theta(x \mid z)p(z), dz$ does not require an encoder. This raises a basic architectural question: *If the encoder is not required by the generative model, why does the VAE architecture shall contain one?*

Two narratives have been offered to explain the encoder's presence. The first is historical: VAEs can be viewed as a *probabilistic extension of deterministic autoencoders*, inheriting the encoder–decoder structure from their non-probabilistic counterparts (Mathieu et al., 2019; Dai & Wipf, 2019). While genealogically accurate, architectural inheritance does not constitute a principled justification, as a VAE encoder trained end-to-end is qualitatively different from a deterministic bottleneck. The second narrative is Bayesian: in latent-variable models the posterior $p_\theta(z \mid x)$ is typically intractable,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and the encoder provides an *amortized variational approximation* (Kingma & Welling, 2014; Blei et al., 2017). This is rigorous but limited in scope: it applies when posterior inference is itself a goal, which is not the case in pure generative modeling. In that setting z is integrated out in the marginal likelihood, and the encoder plays no obvious role in the optimized loss (Kingma & Welling, 2014; Doersch, 2016).

In this paper we offer a third, foundational perspective: *the encoder acts as an optimal variance-reduction mechanism that emerges from a minimum-variance principle*. Starting from a small set of weak assumptions about the generative process and the statistical efficiency of Monte Carlo estimation, we apply a classical result taken from importance-sampling (IS) theory (Kahn & Marshall, 1953) to justify an input-dependent proposal for efficient training under finite-budget Monte Carlo estimation. This justification holds whenever marginal likelihoods are estimated, regardless of whether posterior inference is a downstream goal or not.

Our contributions are as follows:

1. We derive the encoder–decoder structure of VAEs from a minimum-variance principle for finite-budget importance-sampling estimation of the marginal likelihood. In particular, we show that the variance-minimizing proposal is input-dependent and that its amortized implementation naturally induces the VAE encoder (Section 4).
2. We show that the ELBO is aligned with the same minimum-variance principle, establishing a unified statistical foundation for both the VAE architecture and its training objective (Section 5).
3. We provide empirical evidence supporting the theoretical predictions on a benchmark with controlled latent structure. Encoder-based models achieve lower importance-sampling estimator variance and higher log-likelihood than encoder-free baselines, with the difference increasing with the dimensionality of the latent space (Section 6).

2 Problem Setting

2.1 Setting and Assumptions

Consider a dataset with N samples $\mathcal{D} = \{x_i \in \mathbb{R}^d \,|\, i = 1, ..., N\}$ generated from an unknown distribution $p_x^*(x)$. We wish to approximate the data-generating process with a parametric model, whose parameters can be learned from mini-batches of the data.

In many generative tasks, it is reasonable to assume that the data are characterized by an underlying structure that can be captured by latent variables. As such:

Assumption 2.1 *The data generating process is governed by a latent random variable $z \in \mathbb{R}^l$, i.e., z is not directly observed.*

This translates to the generative model:

$$\begin{cases} z \sim p^*(z) \\ x \sim p^*(x \,|\, z) \end{cases} \tag{1}$$

where $p^*(z)$ is the prior distribution over the latent variable z and $p^*(x \,|\, z)$ is the likelihood of the observed data given the latent variable. Figure 1 illustrates the generative process for an image x as described by Equation equation 1:

The true prior and conditional distributions are not known *a priori*. We therefore approximate them by considering flexible parametric families $p_\theta(z)$ and $p_\theta(x \,|\, z)$ respectively, which may be implemented using universal function approximators. We assume:

Assumption 2.2 *The chosen model families $p_\theta(z)$ and $p_\theta(x \,|\, z)$ are sufficiently expressive to approximate the true data-generating distributions $p^*(z)$ and $p^*(x \,|\, z)$ respectively.*

The parameterized generative process is given by

$$\begin{cases} z \sim p_\theta(z), \\ x \sim p_\theta(x \,|\, z), \end{cases} \tag{2}$$

with marginal likelihood:

$$p_\theta(x) = \int p_\theta(x \mid z)\, p_\theta(z)\, dz. \tag{3}$$

Parameter estimation can be achieved through maximum log-likelihood estimation:

$$\theta^* = \arg\max_\theta \mathbb{E}_{x \sim p_x^*(x)}[\log p_\theta(x)] = \arg\max_\theta \mathbb{E}_{x \sim p_x^*(x)}\left[\log \int p_\theta(x \mid z)\, p_\theta(z)\, dz\right]. \tag{4}$$

When the posterior $p_\theta(z \mid x)$ is tractable, the optimization problem in equation 4 can be solved via classical algorithms such as Expectation-Maximization (EM). When it is not the case, direct maximization becomes infeasible and approximate methods are required.

Under the i.i.d. assumption for x, the expectation is approximated by the sample mean:

$$\theta^* \approx \arg\max_\theta \sum_{i=1}^{N} \log p_\theta(x_i) = \arg\max_\theta \sum_{i=1}^{N} \log \int p_\theta(x_i \mid z)\, p_\theta(z)\, dz. \tag{5}$$

Note that the latent variable z is integrated out, and the encoder does not appear in the optimization problem. In principle, one could optimize equation 5 using only the prior $p_\theta(z)$ as a sampling distribution (Kingma & Welling, 2014; Doersch, 2016). However, as we discuss in Section 4.1, this approach is characterized by high variance in the estimate of $p_\theta(x)$, which hinders optimization even for small latent spaces.

Lowering the estimator variance is therefore desirable, as it provides more stable gradient estimates and, in turn, more efficient optimization (Ross, 2006; Roux et al., 2012; Faghri et al., 2020; Ghosh et al., 2023; Manenti & Alippi, 2025). One general strategy for reducing variance is Importance Sampling (IS) (Kahn, 1949), which consists of drawing samples from a more suitable proposal distribution. By analyzing the proposal distribution that minimizes the variance of the resulting estimator, we show that the encoder becomes a necessary component of VAEs. In particular, this claim emerges naturally from the following:

Desideratum 2.3 (Minimum-Variance) *The distribution used to sample latent variables for estimating the marginal likelihood equation 3 should minimize the variance of the resulting Monte Carlo estimator.*

Desideratum 2.3 provides the selection criterion for proposal distributions from which latent variables are sampled.

3 Related Work

Encoder justification in the literature. The most common justification for the encoder in the VAE literature is *amortized approximate inference*. In the original VAE formulation (Kingma & Welling, 2014) and in closely related stochastic backpropagation approaches (Rezende et al., 2014),

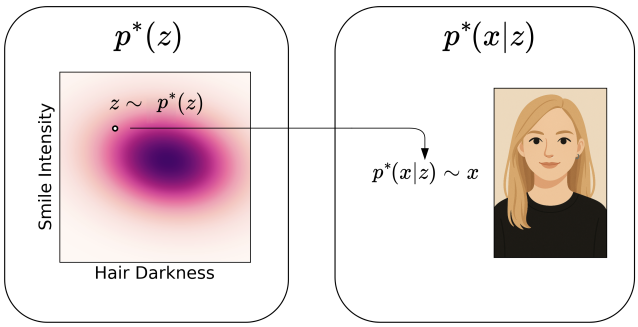


Figure 1: The generative process of an image using a latent variable z as in equation 1.

a neural network is introduced to parameterize a tractable distribution $q_\phi(z \mid x)$ that approximates the generally intractable posterior $p_\theta(z \mid x)$. A recognition model replaces per-observation variational optimization with a shared inference map learned across the dataset, yielding scalable amortized inference (Kingma & Welling, 2014; Blei et al., 2017; Kingma & Welling, 2019). From this viewpoint, the encoder is primarily a computational mechanism that makes variational learning practical in high-dimensional settings.

Another line of work analyzes the encoder through *inference suboptimality* and the *amortization gap*, namely the gap between the best variational parameters attainable by a shared inference network and those obtainable by direct per-instance optimization (Cremer et al., 2018). This perspective motivates richer or partially optimized inference procedures, including semi-amortized and iterative amortized inference (Kim et al., 2018; Marino et al., 2018), as well as regularization and theory for when amortization can generalize or even match factorized variational inference (Shu et al., 2018; Margossian & Blei, 2024). These works explain why encoders are often effective in practice, but they still treat the encoder mainly as a computationally advantageous approximation mechanism.

A different motivation comes from *representation learning* and *compression*. Since VAEs instantiate a stochastic encoder–decoder bottleneck, they are often interpreted as probabilistic extensions of deterministic autoencoders (Mathieu et al., 2019; Dai & Wipf, 2019) that learn compressed latent representations useful for reconstruction and downstream structure discovery (Doersch, 2016; Higgins et al., 2017). Related information-theoretic analyses emphasize that the encoder regulates how much information about x is retained in z, thereby shaping the learned representation and the trade-off between reconstruction, latent usage, and regularization (Alemi et al., 2017; Zhao et al., 2017). These perspectives clarify why encoder–decoder architectures are useful for representation learning; however, they do not explain why an encoder should be necessary for efficient likelihood-based training of latent-variable generative models.

Estimator variance and optimization in VAEs. Several works have made an explicit connection between the variance of different estimators and the downstream optimization objectives. For example, Domke & Sheldon (2018) generalized the ELBO loss function by considering a generic random variable R such that $\mathbb{E}[R] = p(x)$ and linking its variance to the looseness of the IW-ELBO loss (Burda et al., 2016). Similar results were obtained in Maddison et al. (2017b). Finke & Thiery (2019) and Thin et al. (2021) then further analyzed IS-based bounds and their properties. Other works also demonstrated that increasing the tightness on the log likelihood bound can *reduce* the signal-to-noise ratio (SNR) of inference-network gradients, harming optimization (Rainforth et al., 2018). Along similar lines, low-variance training algorithms have been developed to better train latent variable models (Mnih & Rezende, 2016; Tucker et al., 2019; Bauer & Mnih, 2021).

Our contribution in context. Overall, prior work has made explicit connections among estimator variance, bound tightness, and the practical efficiency of training loss functions for learning amortized inference. To the best of our knowledge, however, existing literature motivates the encoder primarily as an amortized inference mechanism or as a representation-learning tool. Moreover, variance appears mainly through bound looseness or in the gradient signal-to-noise ratio. We are not aware of prior work *deriving the encoder as a direct consequence of a minimum-variance principle, nor showing that the ELBO loss function can be justified from that same minimum-variance principle*.

4 The Encoder from a Minimum-Variance Principle

4.1 The Inefficiency of Prior-Based Sampling

Consider estimating the marginal likelihood $p_\theta(x)$ for a fixed observation x through Monte Carlo by drawing M i.i.d. samples $z_1, \ldots, z_M \sim p_\theta(z)$:

$$\hat{p}_\theta(x) = \frac{1}{M} \sum_{j=1}^{M} p_\theta(x \mid z_j). \tag{6}$$

Estimates given by equation 6 can be used to optimize a Monte Carlo surrogate of the marginal-likelihood objective, as described in Algorithm 1. The variance of this estimator, $\mathbb{V}_{z \sim p_\theta(z)}[p_\theta(x \mid z)]$,

Algorithm 1 Prior-sampling Monte Carlo optimization of the marginal likelihood.

Input: dataset $\mathcal{D}$, number of samples M, learning rate η
Initialize θ
for x_i in $\mathcal{D}$ **do**
  for $j = 1, \ldots, M$ **do**
    $z_j \sim p_\theta(z)$ ▷ with gradients handled by an appropriate estimator
  end for
  $\hat{p}_\theta(x_i) \leftarrow \frac{1}{M} \sum_{j=1}^{M} p_\theta(x_i \,|\, z_j)$
  $\theta \leftarrow \theta + \eta \, \nabla_\theta \log \hat{p}_\theta(x_i)$
end for

is generally too large for naive Monte Carlo optimization to be effective. This is due to the fact that, even for low-dimensional latent spaces, the prior distribution assigns negligible mass to regions of the z-space compatible with a specific observation x: the vast majority of prior samples yield near-zero likelihood, and the estimator's variance grows with the dimensionality of the latent space. While consistent and unbiased in the limit $M \to \infty$, for practical values of M the high variance often hinders training.

4.2 The Encoder as a Solution

The preceding discussion shows that both the latent-variable model and the maximum-likelihood criterion are well defined without introducing an encoder. However, optimizing the marginal likelihood by sampling latent variables directly from the prior can lead to high-variance estimators. A standard way to address this issue is importance sampling.

For any proposal distribution $q_x(z)$ whose support contains the support of $p_\theta(x, z)$, the marginal likelihood in Equation equation 3 can be written as

$$p_\theta(x) = \int \frac{p_\theta(x, z)}{q_x(z)} q_x(z) \, dz = \mathbb{E}_{z \sim q_x(z)} \left[\frac{p_\theta(x, z)}{q_x(z)} \right], \tag{7}$$

where the proposal is allowed to depend on the observation x.

The variance of the corresponding single-sample importance-sampling estimator is

$$\mathbb{V}_{z \sim q_x(z)} \left[\frac{p_\theta(x, z)}{q_x(z)} \right] = \mathbb{E}_{z \sim q_x(z)} \left[\left(\frac{p_\theta(x, z)}{q_x(z)} \right)^2 \right] - p_\theta(x)^2 = \int \frac{p_\theta(x, z)^2}{q_x(z)} \, dz - p_\theta(x)^2. \tag{8}$$

Equation equation 8 shows that, for a fixed generative model and observation x, the computational and statistical efficiencies of the marginal-likelihood estimator are determined by the choice of proposal distribution $q_x(z)$. Desideratum 2.3 can therefore be formalized as an optimization problem over admissible proposal distributions: among all admissible densities, we aim to determine the proposal distribution that minimizes the variance of the importance-sampling estimator. The following claim summarizes the role of the encoder under this principle.

Claim 4.1 *The parametric encoder $q_\phi(z \,|\, x)$ is the natural amortized approximation to the variance-minimizing proposal distribution.*

We show that the claim holds by specializing results from importance-sampling theory (Kahn & Marshall, 1953):

Theorem 4.2 (Minimum-variance proposal for IS) *Let $\bar{f} = \int f(z) \, \pi(z) \, dz$, and let $q(z)$ be any proposal distribution such that $q(z) > 0$ whenever $f(z)\pi(z) \neq 0$. Then the proposal distribution minimizing the variance of the importance-sampling estimator*

$$\hat{f} = \frac{f(z)\pi(z)}{q(z)}, \qquad z \sim q(z),$$

is given by

$$q^*(z) = \arg\min_{q(z)} \left\{ \mathbb{V}_{z \sim q(z)} \left[\frac{f(z)\pi(z)}{q(z)} \right] \right\} \propto |f(z)| \, \pi(z).$$

If $f(z) \geq 0$ for all z, then

$$q^*(z) \propto f(z)\,\pi(z).$$

When the theorem is specialized to our setting, the following corollary holds:

Corollary 4.3 (Posterior as optimal proposal) *Let x be fixed and let $p_\theta(x, z)$ be a non-negative integrable function of z with $p_\theta(x) = \int p_\theta(x, z)\,dz > 0$. Among all densities q_x satisfying $\int q_x(z)\,dz = 1$ and $\operatorname{supp}\big(p_\theta(x, \cdot)\big) \subseteq \operatorname{supp}(q_x)$, the variance of the importance-sampling estimator in Equation equation 8 is minimized by*

$$q_x^*(z) = p_\theta(z \,|\, x) = \frac{p_\theta(x, z)}{p_\theta(x)}. \tag{9}$$

The proof follows from Theorem 4.2 by identifying $f(z)\pi(z)$ with $p_\theta(x, z)$ in the minimum-variance importance-sampling result. Since $p_\theta(x, z) \geq 0$, the optimal proposal satisfies $q_x^*(z) \propto p_\theta(x, z)$. Normalizing over z gives $q_x^*(z) = p_\theta(x, z)/p_\theta(x) = p_\theta(z \,|\, x)$ by Bayes' rule.

Corollary 4.3 enables Claim 4.1: the variance-minimizing proposal is the posterior, and an encoder is an input-dependent parametric approximation to this proposal. This has three consequences for VAE design.

1. **Input dependence is necessary.** The optimal proposal $p_\theta(z \,|\, x)$ depends on x. Therefore, any input-agnostic proposal, such as the prior $p_\theta(z)$, is suboptimal whenever $p_\theta(z \,|\, x) \neq p_\theta(z)$, i.e., whenever the latent variable carries information about the observation.
2. **The encoder is the natural variance-reduction mechanism.** The variance-minimizing proposal is the posterior $p_\theta(z \,|\, x)$, which is generally unavailable. Approximating it with a parametric family $q_\phi(z \,|\, x)$ – e.g., implemented as a neural network mapping x to distributional parameters – yields the standard VAE encoder. The encoder is therefore the natural parametric realization of the variance-optimal sampling strategy, and its introduction is justified in any setting where the marginal likelihood is estimated by Monte Carlo methods.
3. **Joint training is required.** The optimal proposal depends on the current generative model, since $p_\theta(z \,|\, x) \propto p_\theta(x \,|\, z)p_\theta(z)$. Consequently, the encoder must track the decoder and prior during training, while the decoder benefits from the lower-variance estimates induced by the encoder. This coupling motivates joint optimization of encoder and decoder parameters.

In the next section, we show that the standard ELBO loss is aligned with the minimum-variance principle.

5 The ELBO in Relation to the Encoder and Minimum Variance

Claim 4.1 identifies, for fixed x and θ, the posterior $p_\theta(z \,|\, x)$ as the optimal proposal for ordinary importance sampling. We now clarify how this relates to the standard VAE training loss. By Jensen's inequality:

$$\log p_\theta(x) = \log \mathbb{E}_{q_\phi(z \,|\, x)}\left[\frac{p_\theta(x, z)}{q_\phi(z \,|\, x)}\right] \;\geq\; \mathbb{E}_{q_\phi(z \,|\, x)}\left[\log \frac{p_\theta(x, z)}{q_\phi(z \,|\, x)}\right] \;=:\; \mathcal{L}(\theta, \phi;\, x). \tag{10}$$

where $\mathcal{L}(\theta, \phi; x)$ is the traditional Evidence Lower Bound (ELBO). Using the factorization $p_\theta(x, z) = p_\theta(x \,|\, z)\, p_\theta(z)$, the ELBO can be written as

$$\mathcal{L}(\theta, \phi; x) = \mathbb{E}_{q_\phi(z \,|\, x)}[\log p_\theta(x \,|\, z)] - D_{\mathrm{KL}}(q_\phi(z \,|\, x) \,\|\, p_\theta(z))\,. \tag{11}$$

This is the ELBO loss that is usually optimized in practice, with the first term being the reconstruction loss and the second term assessing the discrepancy between q_ϕ and the prior distribution through the KL divergence. Alternatively, adding and subtracting $\log p_\theta(z \,|\, x)$ in equation 10 yields

$$\mathcal{L}(\theta, \phi; x) = \log p_\theta(x) - D_{\mathrm{KL}}(q_\phi(z \,|\, x) \,\|\, p_\theta(z \,|\, x))\,. \tag{12}$$

Equation equation 12 shows that, for fixed x and θ, maximizing the ELBO loss with respect to ϕ requires, as a necessary condition, minimizing $D_{\mathrm{KL}}(q_\phi(z\,|\,x)\,\|\,p_\theta(z\,|\,x))$. Hence the ELBO is maximized only if

$$q_\phi(z\,|\,x) = p_\theta(z\,|\,x) \quad \text{a.e.}$$

This is exactly the same distribution identified in Corollary 4.3 as the minimum-variance proposal for estimating $p_\theta(x)$ with importance sampling.

This result does not imply that the ELBO and the minimum-variance criterion are equivalent objectives. In fact, the variance in equation 8 can be rewritten as

$$\mathbb{V}_{z\sim q_x(z)}\left[\frac{p_\theta(x,z)}{q_x(z)}\right] = \int \frac{p_\theta(x,z)^2}{q_x(z)}dz - p_\theta(x)^2 = p_\theta(x)^2\chi^2(p_\theta(z\,|\,x)\|q_x(z)). \tag{13}$$

For fixed x and θ, variance minimization for the ordinary IS estimator is associated with a χ^2-type divergence between the posterior $p_\theta(z\,|\,x)$ and the proposal $q_\phi(z\,|\,x)$, whereas ELBO maximization requires the minimization of the KL divergence $D_{\mathrm{KL}}(q_\phi(z\,|\,x)\,\|\,p_\theta(z\,|\,x))$. Thus, the two criteria differ away from the optimum, even though they point to the same optimal proposal.

The above analysis allows us to provide a useful interpretation of the ELBO: *although it is not itself the minimum-variance objective, it is optimized by the same proposal distribution that minimizes the variance of the ordinary IS estimator*, namely the posterior $p_\theta(z\,|\,x)$. From this perspective, training an encoder $q_\phi(z\,|\,x)$ with the ELBO can be viewed as learning an amortized approximation to the variance-optimal proposal while maximizing a bound on the marginal log-likelihood.

6 Experiments

In this section, we empirically validate the theoretical claims derived above. The goal is to show that when the latent space becomes increasingly large, an input-dependent proposal yields more stable likelihood estimates than an unconditional prior-based proposal. To this end, we introduce a synthetic dataset with a controlled latent structure and compare: (i) an *encoder-based* architecture implementing a VAE model, and (ii) an *encoder-free* architecture relying solely on a learnable prior. Both models share the same decoder and loss, in order to isolate the role of the proposal distribution.

The experiments show that the encoder-based architecture consistently achieves lower relative variance of the importance weights and higher marginal log-likelihood estimates than the encoder-free baseline. Moreover, the gap increases with the dimensionality of the latent space: as the number of latent variables grows, the encoder-free proposal becomes increasingly unstable, whereas the encoder-based proposal remains in a comparatively low-variance regime. These results support the interpretation developed in Section 4: the encoder acts as an amortized, input-dependent approximation to the variance-optimal proposal.

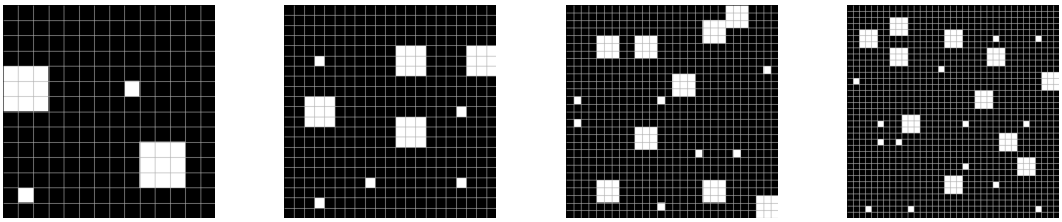
6.1 The Grid-Squares Dataset

We generate a dataset of spatially arranged squares, referred to as the `Grid-Squares` dataset. Each image is composed of I^2 independent 7×7 sub-images arranged on a regular grid, yielding a full image of size $7I \times 7I$. By varying I, we directly control the dimensionality and combinatorial complexity of the latent space.

Each 7×7 sub-image is generated from *two discrete latent variables*: a *size* variable $z_s \in \{1, 3\}$ and a *position* variable $z_p \in \{(x, y)\,|\,x, y \in \{1, 3, 5\}\}$. Pixels in each sub-image are activated according to a Chebyshev-radius rule:

$$\max(|i - z_{px}|, |j - z_{py}|) < z_s. \tag{14}$$

This construction yields a dataset with a known ground-truth generative process and factorized discrete latents. For each value of I, we generate 2000 training images, 500 validation images, and 500 test images. The size of the latent space scales exponentially with the number of sub-images $|\mathcal{Z}_I| = (2 \cdot 9)^{I^2} = 18^{I^2}$. Figure 2 shows representative samples for increasing values of I.



(a) $I = 2$ (b) $I = 3$ (c) $I = 4$ (d) $I = 5$

Figure 2: Samples from the `Grid-Squares` dataset for increasing grid sizes I. Grid lines are shown for visualization purposes only.

6.2 Model Architectures

We evaluate two distinct architectures for the proposal distribution: an encoder-based architecture, which performs amortized inference conditioned on the input image, and an encoder-free one, which acts as an unconditional generative baseline. Both models utilize a shared decoder architecture and employ Gumbel-Softmax relaxation (Jang et al., 2017; Maddison et al., 2017a; Huijben et al., 2022) to enable differentiable sampling of discrete latents. Additional details in Appendix A.

Encoder-based The first architecture implements an input-dependent proposal $q_\phi(z \mid x)$. An encoder network maps x to logits defining categorical distributions over the discrete latent variables associated with each sub-region of the image.

Encoder-free The second architecture models the prior distribution $p_\theta(z)$ without conditioning on the input data x. It consists of learnable prior distributions, which represent the logits of the categorical distributions for size and position.

Shared Decoder In both architectures, sampled latent codes are decoded independently into 7×7 patches through a shared neural network. The final layer employs a sigmoid activation, allowing the outputs to be interpreted as Bernoulli parameters over pixels. The reconstructed patches are then arranged according to the underlying grid structure, yielding the full $7I \times 7I$ image.

6.3 Variance and Performance

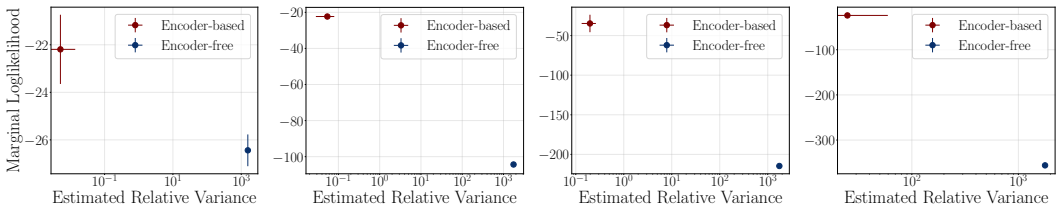
We compare the architectures across different numbers of sub-images $I^2 \in \{4, 9, 16, 25\}$.

For each trained model and each test image x, we estimate the log-likelihood as in Burda et al. (2016) using $M = 2048$ samples. This estimator is a tighter lower bound on $\log p_\theta(x)$ than the single-sample ELBO and consistent. Using the same samples, we also estimate its empirical variance.

For models with high-dimensional latent spaces, the encoder-free method often failed to converge, resulting in an estimated likelihood far from the true likelihood, and a misleadingly small variance estimate. We therefore report the relative variance of the importance weights, normalized by the likelihood estimate, with lower values indicating more stable likelihood estimation.

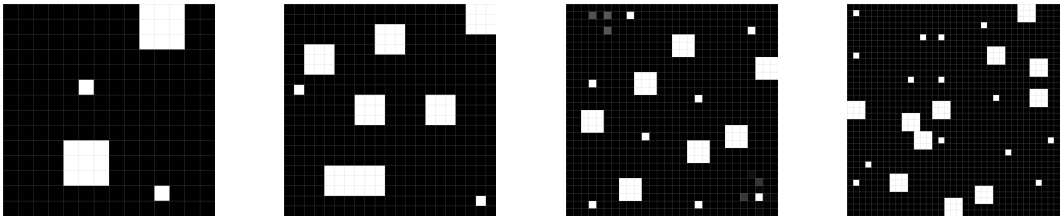
Figure 3 compares the encoder-based and encoder-free models in terms of estimated relative variance and marginal log-likelihood. Across grid sizes, the encoder-based proposal achieves higher likelihood estimates while maintaining lower relative variance. The gap becomes larger as I increases: the encoder-free model, becomes increasingly unstable as the latent space grows, whereas the encoder-based model remains in a lower variance regime.

These quantitative results support the minimum-variance interpretation developed in Section 4: conditioning the proposal on the observation x provides a more efficient estimator of the marginal likelihood than prior-based sampling. The qualitative samples show the same trend. As shown in Figures 4 and 5, the encoder-based model produces coherent samples for all values of I, while the encoder-free model visibly deteriorates for $I \geq 3$.



(a) $I = 2$ (b) $I = 3$ (c) $I = 4$ (d) $I = 5$

Figure 3: Estimated relative variance versus marginal log-likelihood for the encoder-based and encoder-free proposal distributions across increasing grid sizes I.



(a) $I = 2$ (b) $I = 3$ (c) $I = 4$ (d) $I = 5$

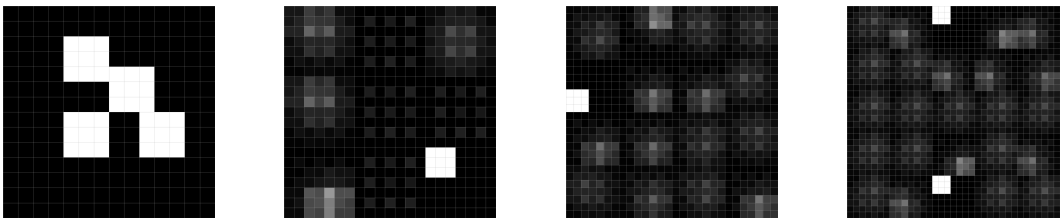
Figure 4: Generated samples from the encoder-based model for increasing grid sizes I.

7 Conclusions

This work revisits the role of the encoder in Variational Autoencoders from the perspective of statistical efficiency. While the latent-variable maximum-likelihood objective is well defined without an encoder, direct estimation of the marginal likelihood by sampling from the prior can lead to high-variance Monte Carlo estimates. By applying results from importance-sampling theory, we show that the encoder is needed to obtain the variance-minimizing proposal distribution. We also demonstrate that the ELBO is aligned with the same minimum-variance principle: the standard VAE loss can be interpreted as training an amortized approximation to the variance-optimal proposal while simultaneously optimizing a tractable lower bound on the marginal log-likelihood.

Finally, experiments on the controlled `Grid-Squares` dataset support the theoretical analysis. Compared with an encoder-free baseline using an unconditional proposal, the encoder-based model achieves lower relative importance-weight variance, higher likelihood estimates, and more coherent generated samples. The advantage becomes more pronounced as the latent space grows, consistent with the fact that prior-based sampling becomes increasingly inefficient in high-dimensional spaces.

Overall, these results suggest that the encoder–decoder structure of VAEs arises from a minimum-variance principle for marginal-likelihood estimation. This perspective places the encoder and the ELBO under a common statistical-efficiency argument, and may inform the analysis of richer variational families, alternative losses, and low-variance training schemes.



(a) $I = 2$ (b) $I = 3$ (c) $I = 4$ (d) $I = 5$

Figure 5: Generated samples from the encoder-free model for increasing grid sizes I.

AI USE STATEMENT

In this work, we used generative AI tools for designing or providing feedback on research methodology or experiments, assisting with translation and re-writing, helping create or edit software code, brainstorming, editing the paper to improve readability, identifying relevant literature, and proposing a title or keywords.

We have not used generative AI tools for generating synthetic datasets, helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in writing proofs, proposing or refining hypotheses, implementing methods, cleaning or reformatting datasets, supporting qualitative or thematic data analysis, interpreting results, formulating questions for surveys or interviews, creating or modifying scientific figures or images, suggesting experimental parameters, creating artifacts, drafting parts of the paper, transcribing recordings of research material, summarizing or analyzing existing literature, discovering research topics or identifying gaps, sourcing or searching for information, formatting references, or suggesting a structure for the paper.

We have reviewed all AI-assisted work, and we take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations*, 2017.

Matthias Bauer and Andriy Mnih. Generalized doubly reparameterized gradient estimators. In *International Conference on Machine Learning*, pp. 738–747. PMLR, 2021.

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

Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. In *4th International Conference on Learning Representations (ICLR)*, 2016.

Chris Cremer, Xuechen Li, and David Duvenaud. Inference suboptimality in variational autoencoders. In *International conference on machine learning*, pp. 1078–1086. PMLR, 2018.

Bin Dai and David Wipf. Diagnosing and enhancing vae models. In *International Conference on Learning Representations*, 2019.

Carl Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*, 2016.

Justin Domke and Daniel R Sheldon. Importance weighting and variational inference. *Advances in neural information processing systems*, 31, 2018.

Fartash Faghri, David Duvenaud, David J Fleet, and Jimmy Ba. A study of gradient variance in deep learning. *arXiv preprint arXiv:2007.04532*, 2020.

Axel Finke and Alexandre H Thiery. On importance-weighted autoencoders. *arXiv preprint arXiv:1907.10477*, 2019.

Arna Ghosh, Yuhan Helena Liu, Guillaume Lajoie, Konrad Kording, and Blake Aaron Richards. How gradient estimator variance and bias impact learning in neural networks. In *The Eleventh International Conference on Learning Representations*, 2023.

Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. Array programming with numpy. *Nature*, 585(7825):357–362, 2020.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy2fzU9gl.

Iris AM Huijben, Wouter Kool, Max B Paulus, and Ruud JG Van Sloun. A review of the gumbel-max trick and its extensions for discrete stochasticity in machine learning. *IEEE transactions on pattern analysis and machine intelligence*, 45(2):1353–1371, 2022.

John D Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9 (03):90–95, 2007.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017.

Herman Kahn. *Stochastic (Monte Carlo) Attenuation Analysis: June 14, 1949. R- 163*. R (Rand Corporation). Rand Corporation, 1949. URL `https://books.google.ch/books?id=W0q5AAAAIAAJ`.

Herman Kahn and Andy W Marshall. Methods of reducing sample size in monte carlo computations. *Journal of the Operations Research Society of America*, 1(5):263–278, 1953.

Yoon Kim, Sam Wiseman, Andrew Miller, David Sontag, and Alexander Rush. Semi-amortized variational autoencoders. In *International Conference on Machine Learning*, pp. 2678–2687. PMLR, 2018.

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

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations*, 2014.

C Maddison, A Mnih, and Y Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *Proceedings of the international conference on learning Representations*. International Conference on Learning Representations, 2017a.

Chris J Maddison, John Lawson, George Tucker, Nicolas Heess, Mohammad Norouzi, Andriy Mnih, Arnaud Doucet, and Yee Teh. Filtering variational objectives. *Advances in Neural Information Processing Systems*, 30, 2017b.

Alessandro Manenti and Cesare Alippi. Beyond softmax: A natural parameterization for categorical random variables. *arXiv preprint arXiv:2509.24728*, 2025.

Charles C Margossian and David M Blei. Amortized variational inference: when and why? In *Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence*, pp. 2434–2449, 2024.

Joe Marino, Yisong Yue, and Stephan Mandt. Iterative amortized inference. In *International Conference on Machine Learning*, pp. 3403–3412. PMLR, 2018.

Emile Mathieu, Tom Rainforth, Nana Siddharth, and Yee Whye Teh. Disentangling disentanglement in variational autoencoders. In *International conference on machine learning*, pp. 4402–4412. PMLR, 2019.

Andriy Mnih and Danilo Rezende. Variational inference for monte carlo objectives. In *International Conference on Machine Learning*, pp. 2188–2196. PMLR, 2016.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Tom Rainforth, Adam Kosiorek, Tuan Anh Le, Chris Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. In *International Conference on Machine Learning*, pp. 4277–4285. PMLR, 2018.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pp. 1278–1286. PMLR, 2014.

Sheldon M Ross. *Simulation*. Academic Press, 2006.

Nicolas Roux, Mark Schmidt, and Francis Bach. A stochastic gradient method with an exponential convergence _rate for finite training sets. *Advances in neural information processing systems*, 25, 2012.

Rui Shu, Hung H Bui, Shengjia Zhao, Mykel J Kochenderfer, and Stefano Ermon. Amortized inference regularization. *Advances in Neural Information Processing Systems*, 31, 2018.

Achille Thin, Nikita Kotelevskii, Arnaud Doucet, Alain Durmus, Eric Moulines, and Maxim Panov. Monte carlo variational auto-encoders. In *International Conference on Machine Learning*, pp. 10247–10257. PMLR, 2021.

George Tucker, Dieterich Lawson, Shixiang Gu, and Chris J. Maddison. Doubly reparameterized gradient estimators for monte carlo objectives. In *International Conference on Learning Representations*, 2019.

Shengjia Zhao, Jiaming Song, and Stefano Ermon. Infovae: Information maximizing variational autoencoders. *arXiv preprint arXiv:1706.02262*, 2017.

APPENDIX

A ADDITIONAL EXPERIMENTAL DETAILS

A.1 ARCHITECTURE DETAILS

Each image is represented as a $7I \times 7I$ grid of local regions, where each region corresponds to a 7×7 patch in the final image. Each grid cell is assigned two discrete latent variables: a size variable z_s with $K_s = 2$ categories and a position variable z_p with $K_p = 9$ categories.

Encoder-based proposal. The encoder-based proposal uses a convolutional network to map the input image to categorical logits for the latent variables in each grid cell. The input image is first reshaped to include a single channel and is passed through three convolutional layers with 3×3 kernels, padding of one, and ReLU activations. All convolutional layers use 32 channels.

The resulting feature tensor is flattened and passed to two linear heads. One head outputs logits for the size variables,

$$\ell_s(x) \in \mathbb{R}^{I^2 \times K_s},$$

and the other outputs logits for the position variables,

$$\ell_p(x) \in \mathbb{R}^{I^2 \times K_p}.$$

Thus, the encoder produces one categorical distribution for each latent type and each grid cell.

Encoder-free proposal. The encoder-free model instead uses learnable categorical logits that do not depend on the input image. It contains two trainable logit vectors,

$$\ell_s \in \mathbb{R}^{K_s}, \qquad \ell_p \in \mathbb{R}^{K_p},$$

which are broadcast across the batch dimension and across the I^2 grid cells. Consequently, all cells share the same marginal distributions over size and position.

Discrete latent sampling. For both proposal models, samples are drawn using a straight-through Gumbel-Softmax estimator. Given logits ℓ and temperature τ, a relaxed Gumbel-Softmax sample is first produced. The forward pass then uses the corresponding hard one-hot vector, while the backward pass retains gradients through the relaxed sample. This gives discrete latent assignments during decoding while preserving differentiability with respect to the proposal logits.

For a sampled one-hot vector $\tilde{z}$, the probability assigned to the sampled category is computed by taking the inner product between $\tilde{z}$ and the softmax probabilities induced by the corresponding logits. This is used to evaluate the proposal probability of the sampled latent configuration.

Decoder. The decoder is shared by both models. For each grid cell, the sampled one-hot size and position vectors are concatenated,

$$\tilde{z}_j = [\tilde{z}_{s,j}, \tilde{z}_{p,j}] \in \mathbb{R}^{K_s+K_p}.$$

This vector is passed through a two-layer multilayer perceptron with hidden width 64, ReLU activation, and a sigmoid output layer. The output has dimension 49 and is reshaped into a 7×7 patch:

$$\hat{x}_j \in [0,1]^{7\times 7}.$$

The sigmoid output is interpreted as Bernoulli pixel probabilities.

The decoded patches are finally rearranged according to the fixed $I \times I$ grid structure to form an image of size $7I \times 7I$.

Generation. For unconditional generation, latent variables are sampled from the corresponding categorical distributions and then passed through the shared decoder. In the encoder-free model these distributions are the learned priors. In the encoder-based model, generation without conditioning is performed by sampling random simplex vectors and converting them to one-hot latent assignments before decoding.

A.2 Training Details

For both proposal models and all datasets, we use the Adam optimizer with weight decay 10^{-4}. The learning rate is selected by tuning over

$$\{10^{-4}, 3\times 10^{-4}, 10^{-3}, 3\times 10^{-3}, 10^{-2}\}.$$

Each model is trained for 250 epochs with a batch size of 32.

The Gumbel-Softmax temperature is annealed linearly during training. We initialize the temperature at $\tau_{\text{init}} = 1$ and linearly anneal it to $\tau_{\text{final}} = 0.01$ by the end of training.

B Computational Resources and Open Source Software

The paper's experiments were run on a workstation with AMD EPYC 7513 processors and NVIDIA RTX A5000 GPUs.

The developed code relies on PyTorch (Paszke et al., 2019) and the following additional open-source libraries: NumPy (Harris et al., 2020) and Matplotlib (Hunter, 2007).