Doubly Robust Conditional VAE via Decoder Calibration: An Implicit KL Annealing Approach

Anonymous authors
Paper under double-blind review

Abstract

While many variants of Variational Autoencoders proposed, a unified understanding remains unclear. In particular, σ-VAEs utilize a scaled identity matrix $\sigma^2 I$ in the decoder variance, while β-VAEs introduce a hyperparameter β to reweight negative ELBO loss. However, existing learning theories on the global optimal VAEs yield limited practical insight toward their empirical success. In addition, previous work showed the mathematical equivalence of the variance scalar σ and the hyperparameter β in the loss landscape, but σ as a model parameter fundamentally differs from β as a hyperparameter. This paper presents a comprehensive analysis of σ-CVAE, revealing its expressiveness and limitations due to suboptimal variational inference. Focusing on the conditional variants, we propose Calibrated Robust σ-CVAE, a doubly robust algorithm that ensures reliable σ estimation while effectively preventing posterior collapse. Our approach, leveraging functional neural decomposition and KL annealing techniques, provides a unified framework to understand both σ-VAEs and β-VAEs regarding parameter optimality and training dynamics. Empirical results demonstrate the superior performance of our method across various conditional density estimation tasks, highlighting its significance for accurate and reliable probabilistic modeling.

1 Introduction

Conditional distributions play an essential role in characterizing the dependence of a response or data $\boldsymbol{y} \in \mathbb{R}^q$ on given covariates or labels $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^p$. Canonical methods, such as regression or density estimators, face challenges when the data generating distribution $p_{gt}(\boldsymbol{y}|\boldsymbol{x})$ is complex and high-dimensional. Deep latent generative models based on amortized variational inference are widely used as a scalable approach to model complex distributions and scale to large datasets. In particular, (Sohn et al., 2015), derived from Variational Autoencoders (VAEs) (Kingma & Welling, 2014; Rezende et al., 2014), introduced Conditional VAEs (CVAEs) to learn conditional distributions.

1.1 Gaussian σ-Conditional VAE

Gaussian σ-CVAE models the marginal distribution $p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x})$ in a parametric form, utilizing a latent variable $\boldsymbol{z}$. It incorporates a Gaussian decoder $p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{z}) = N(\mu_\theta(\boldsymbol{x},\boldsymbol{z}), \sigma^2 I_q)$, where σ is a learnable shared scale parameter (Kingma et al., 2016; Dai & Wipf, 2018) and a data-independent prior $p(\boldsymbol{z}|\boldsymbol{x}) = N(0, I_d)$ where the latent variable z is sampled from(Doersch, 2021).

$$p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x}) = \int N(\boldsymbol{y}|\mu_\theta(\boldsymbol{x},\boldsymbol{z}), \sigma^2 I_q) N(\boldsymbol{z}|0, I_d) d\boldsymbol{z}. \tag{1}$$

It also includes a Gaussian encoder $q_\phi(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x}) = N(\mu_\phi(\boldsymbol{y},\boldsymbol{x}), \Sigma_\phi(\boldsymbol{y},\boldsymbol{x}))$ as an approximate posterior of $p_{\theta,\sigma}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})$, such that the logarithm of $p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x})$ is replaced by a tractable evidence lower bound (ELBO),

$$\text{ELBO} = \mathbb{E}_{q_\phi}[\log p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{z})] - \text{KL}[q_\phi(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})||p(\boldsymbol{z})]. \tag{2}$$

Both the decoder and the encoder being Gaussian distributions simplify the sampling procedure of $q_\phi(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})$ and the computation of the KL divergence in ELBO, allowing it to be scalable to large dataset via the

reparameterization trick (Kingma & Welling, 2014; Rezende et al., 2014). The conditional covariates $\boldsymbol{x}$ influence the mean of the decoder $\mu_\theta(\boldsymbol{x}, \boldsymbol{z})$ assuming a data-independent prior $p(\boldsymbol{z})$ without loss of generality (Zheng et al., 2022).

1.2 Learning theories of CVAE

The most intuitive objective function $\mathcal{L}(\theta, \sigma, \phi)$ of CVAEs is the expected negative ELBO with respect to the ground truth data-generating measure μ_{gt} over the model parameters $\{\theta, \sigma, \phi\}$:

$$\mathcal{L}(\theta, \sigma, \phi) := \int -1 \times \text{ELBO}\ \mu_{gt}(d\boldsymbol{y}d\boldsymbol{x}). \tag{3}$$

The double inequalities in Eq.4 illustrate the optimality of model parameters, highlighting the main objectives of CVAE: variational inference and generative modeling.

$$\mathcal{L}(\theta, \sigma, \phi) \geq \int -\log p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x})\mu_{gt}(d\boldsymbol{y}d\boldsymbol{x}) \geq \int -\log p_{gt}(\boldsymbol{y}|\boldsymbol{x})\mu_{gt}(d\boldsymbol{y}d\boldsymbol{x}). \tag{4}$$

The tightness of the first inequality determines the quality of variational inference, controlled by the distribution family and the parameter ϕ. The gap in the first inequality, termed *inference gap* Cremer et al. (2018), reflects how closely the variational posterior q_ϕ approximates the true posterior. Significant research has focused on enhancing variational inference (Ranganath et al., 2016; Kingma et al., 2016; Cremer et al., 2018; Burda et al., 2016; Nowozin, 2018; Huang et al., 2019).

The tightness of the second inequality determines the quality of probabilistic modeling. We refer to it as *approximation gap*, or parsimony gap following Mattei & Frellsen (2018). θ, σ determines how closely the model $p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x})$ marginally approximates the ground-truth distribution $p_{gt}(\boldsymbol{y}|\boldsymbol{x})$ in terms of the conditional entropy of the data. Often, a tighter second inequality of Eq.4 is established on *unrealistic* theoretical assumptions or on an additional hierarchical structure. For example, a probalistic PCA setup assumes a linear dependence between latent variable and response (Lucas et al., 2019b; Dai et al., 2020; Sicks et al., 2021; Wang & Ziyin, 2022; Dang et al., 2023), which may not be generalized to complex datasets; A global optimal result is established asymptotically by assuming $\sigma \to 0$; Hierarchical Bayesian analysis indicated that assuming an inverse gamma prior distribution on variance σ could expand a Gaussian decoder into a student-t decoder (Takahashi et al., 2018; Stirn & Knowles, 2020).

The existing theoretical works in understanding these inequalities are often *separated*, perhaps due to a lack of comprehensive understanding in all detailed aspects of VAE learning theories.

1.3 Loss equivalence to β-CVAE

Given the same parameter θ, ϕ, Gaussian σ-CVAE has the same objective function as β-CVAE, up to a multiplying constant (Lucas et al., 2019b; Rybkin et al., 2021). As shown below,

$$\mathbb{E}_{q_\phi}[\|\mu_\theta(\boldsymbol{x}, \boldsymbol{z}) - \boldsymbol{y}\|^2]/2\sigma^2 + \text{KL}[q_\phi||p(\boldsymbol{z})] \propto \mathbb{E}_{q_\phi}[\|\mu_\theta(\boldsymbol{x}, \boldsymbol{z}) - \boldsymbol{y}\|^2] + \beta\text{KL}[q_\phi||p(\boldsymbol{z})], \tag{5}$$

the negative ELBO of a Gaussian σ-CVAE on the left side is proportional to the objective function of β-CVAE, assuming a fixed unit variance. Therefore, an optimal σ is believed to be the best β (Lucas et al., 2019a; Rybkin et al., 2021)

The equivalence reveals the subtlety of the hyperparameter β. From a statistics point of view, when the decoder is taken from a location-scale distribution family, scaling the KL divergence between approximate posterior and prior is nothing but scaling its unit diagonal variance. Thus, one should avoid the *explicit* usage of β while assuming a fixed unit variance for accurate and reliable probabilistic modeling. First, a fixed unit variance in Gaussian decoders limits the expressive power of the marginal distribution $p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x})$, causing a potentially larger approximation gap. Secondly, the β-scaled objective lacks interpretability, since it can no longer be seen as an approximate log-likelihood. Lastly, tuning the hyperparameter β is more computationally expensive than learning the parameter σ.

However, β itself still plays an essential role in the robust estimation of the VAE model. To ameliorate posterior collapse or the KL vanishing problem, KL annealing methods introduce the hyperparameter β (Higgins et al., 2016; Chen et al., 2018; Rezende & Viola, 2018) and tune it with a predefined monotonic or cyclical annealing schedule (Raiko et al., 2007; Bowman et al., 2016; Fu et al., 2019).

In summary, the organization and contributions of this paper are as follows.

(1) **A zero approximation gap is generally achievable without optimal variational inference**. In Section 2, we establish a non-asymptotic approximation theorem of continuous Gaussian decoders for arbitrary complex conditional densities. In Lemma 2.1, we point out the identifiability issue of σ to recover the ground truth distribution and one possible way to bypass it is to consider a block neural decomposition (Sobol, 2001). In Theorem 2.4, we prove that σ-CVAE can approximate the arbitrary complex ground truth conditional density $p_{gt}(\boldsymbol{y}|\boldsymbol{x})$ more generally, challenged by suboptimal variational inference and non-identifiability of decoder variance.

(2) **KL annealing is a form of decoder variance calibration**. In Section 3.1, we further analyze the dynamics of σ in a dual-step optimization algorithm, showing that the biased gradient of σ is actually a result of suboptimal variational inference. Considering the equivalence of β and σ, we show in Section 3.2 that the KL annealing techniques against posterior collapse can be seen as a form of decoder variance calibration. This duality of β and σ highlights the fact that an extensive KL annealing scheme could be redundant in practice, and a doubly robust model can be obtained by calibrating σ directly. As an example, we propose Calibrated Robust σ-CVAE in Section 3.3, a simple calibrated Condtional VAE variant which calibrates the parameter σ that can efficiently explore the loss landscape of θ, ϕ to prevent posterior collapse and provide robust variance estimation.

(3) **σ-Calibration is doubly robust, providing both reliable variance estimation and prevention of posterior collapse.** In Section 4.1, we empirically validate that suboptimal variational inference can be the main source of numerical instability in the estimation of σ. More importantly, we confirm the double robustness of our algorithm, showing that it not only provides decoder variance estimation but also fine-tunes suboptimal encoders. Compared to existing KL annealing methods, we validate the superiority and effectiveness of σ calibration. Starting from Section 4.2, we compared the performance of Calibrated Robust σ-CVAE in various conditional density estimation tasks, showing its superiority over various conditional learning methods.

1.4 Related Work

A comprehensive section of Related Work can be found in Appendix A. Table 1 provide a short overview of our setting and contribution compared to existing studies.

Table 1: An high-level comparison with existing VAE theories and applications

Reference	Approximation Assumption	Optimal VI	Expressive Decoder	Doubly Robust
Dai & Wipf (2018)	Simple Riemann Manifold	✓	✓	×
Lucas et al. (2019b)	Decoder's Mean Linearity	✓	×	N.A.
Takahashi et al. (2018)	Inverse-Gamma Prior	×	✓	✓
This paper	Block Functional ANOVA	×	✓	✓

2 How expressive σ-CVAEs can be?

2.1 Data generating distribution and VAE generative model

In this paper, we aim to understand why σ-CVAE is powerful to parameterize a wide range of data-generating distribution. We begin our analysis by defining conditional distributions as a measurable function G.

Lemma 2.1 (**Gaussian noise outsourcing** (Agrawal & Domke, 2021)). *Suppose that $\boldsymbol{x}$, $\boldsymbol{y}$ are random vectors taking values in the standard Borel space $\mathcal{X}$,$\mathcal{Y}$ with μ_X denoting the probability measure on $\mathcal{X}$, and that $P_{\boldsymbol{x},\boldsymbol{y}} : \mathcal{X} \times \mathcal{Y} \rightsquigarrow \mathbb{R}$ is a probability kernel of interests, then for any $m > 1$, and a standard Gaussian random vector $\boldsymbol{z} \in \mathbb{R}^m$ with measure μ_Z that is independent of $\boldsymbol{x}$, there exists a Borel measurable function $G : \mathcal{X} \times \mathbb{R}^m \to \mathcal{Y}$ such that*

$$(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x}, G(\boldsymbol{x}, \boldsymbol{z})) \quad a.s. \tag{6}$$

Lemma 2.1 states that the randomness of any complex data distribution of $\boldsymbol{y}$ can be outsourced to a Gaussian random vector $\boldsymbol{z}$ that is independent of $\boldsymbol{x}$. When $\boldsymbol{x}$ is deterministic or given, such G is the nested function of the quantile function of $\boldsymbol{y}$ given $\boldsymbol{x}$ and a Gaussian cumulative distribution function. Note that the dimension m is not necessarily the same as the latent dimension d of CVAE. Hence, for any latent $m \in \mathbb{N}^+$, there exists a Borel measurable map $\{G^*_m : \mathcal{X} \times \mathbb{R}^m \to \mathcal{Y}\}$ corresponding to the data generating measure μ_{gt}.

Based on Eq,1, we can characterize the Gaussian σ-CVAE model of $\boldsymbol{y}$ given $\boldsymbol{x}$ as a form of G, i.e.,

$$\boldsymbol{y} := G_{\theta,\sigma}(\boldsymbol{x}, (\boldsymbol{z}_1, \boldsymbol{z}_2)) = \mu_\theta(\boldsymbol{x}, \boldsymbol{z}_1) + \sigma \boldsymbol{z}_2. \tag{7}$$

The decomposition in Eq.7 highlights the inherent restrictions of noise outsourcing in the σ-CVAE model. These restrictions include two key aspects: 1) the enforced independence between the d-dimensional latent variable $\boldsymbol{z}_1$ and q-dimensional decoder's unscaled noise $\boldsymbol{z}_2$, restricting their interactions; 2) an additive relationship between the mean and scaled variance, which limits complex interactions and constrains the flexibility of data-dependent variances.

The concept of noise outsourcing reformulates the comparison of distributions into a comparison of variable-transforming measurable maps, eliminating the need to compute log-probability and KL divergence. Such comparisons offer a new perspective on existing approximation theories. For instance, by extending and simplifying the findings of Dai & Wipf (2018) in Proposition B.2, we show that an asymptotic assumption on $\sigma \to 0$ eliminates the two aspects mentioned above, enabling the recovery of the ground-truth distribution $G^*_d(\boldsymbol{x}, \boldsymbol{z})$ through an arbitrarily complex network $\mu_\theta(\boldsymbol{x}, \boldsymbol{z}_1)$. For another example, when the Linear VAE assumes μ_θ as a linear combination of $\boldsymbol{x}$ and $\boldsymbol{z}$, the corresponding $G_{\sigma,\theta}$ is reduced to a fully linear function.

2.2 Non-asymptotic approximation of the σ-CVAE

In this paper, we demonstrate the key differences between the restrictions in Eq.7 and the data-generating map in Eq.6, allowing us to analyze the expressiveness of the σ-CVAE model. A zero approximation gap is essentially the equality conditions between Eq.6 and Eq.7. Specifically, we explore a less-explored yet intuitive approach known as block neural decomposition Sobol (2001); Märtens & Yau (2020).

Definition 2.2 (**Block neural decomposition** (Sobol, 2001)). Suppose that given any response dimension q, there exist $m > q$, such that $G^*_m : \mathcal{X} \times \mathbb{R}^m \to \mathcal{Y}$ equals almost surely to a neural network f_η that can be arbitrarily complex, then a block decomposition of G^*_m on input dimension $\mathcal{X} \times \mathbb{R}^{m-q} \times \mathbb{R}^q$ is

$$\begin{aligned} f_\eta(\boldsymbol{x}, \boldsymbol{z}_1, \boldsymbol{z}_2) =& f^\eta_0 + f^\eta_{\boldsymbol{x}}(\boldsymbol{x}) + f^\eta_{\boldsymbol{z}_1}(\boldsymbol{z}_1) + f^\eta_{\boldsymbol{z}_2}(\boldsymbol{z}_2) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_1}(\boldsymbol{x}, \boldsymbol{z}_1) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_2}(\boldsymbol{x}, \boldsymbol{z}_2) \\ &+ f^\eta_{\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{z}_1, \boldsymbol{z}_2) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{x}, \boldsymbol{z}_1, \boldsymbol{z}_2). \end{aligned} \tag{8}$$

We note that this decomposition is a blockwise expansion (Sobol, 2001) of measurable functions, making nonparametric mixture model in Märtens & Yau (2020) a special case. However, this decomposition is not identifiable and unique. We will now present a unique decomposition as stated below.

Assumption 2.3 (**Block ANOVA representation** (Sobol, 2001))**.** Let μ_x, μ_{z_1}, μ_{z_2} denote the Borel measure of $\boldsymbol{x}$, $\boldsymbol{z}_1$, $\boldsymbol{z}_2$, respectively, and let $\mu_{\mathcal{I}}$ be the measure or product measure of the index subset of $\{\boldsymbol{x}, \boldsymbol{z}_1, \boldsymbol{z}_2\}$. The block ANOVA representation assumes that the functional f in Eq.8 satisfies the following integral constraints, $\int f_{\mathcal{I}}(y_{\mathcal{I}})\mu_{\mathcal{I}}(dy_i) = 0, \quad \forall i \in \mathcal{I}$, for every index subset, $\mathcal{I} \subseteq \{\boldsymbol{x}, \boldsymbol{z}_1, \boldsymbol{z}_2\}$.

If there are no covariates $\boldsymbol{x}$, the decomposition consists of $f_0^\eta + f_{\boldsymbol{z}_1}^\eta(\boldsymbol{z}_1) + f_{\boldsymbol{z}_2}^\eta(\boldsymbol{z}_2) + f_{\boldsymbol{z}_1\boldsymbol{z}_2}^\eta(\boldsymbol{z}_1, \boldsymbol{z}_2)$. This assumption leads to the following integral constraints:

$$\int f_{\boldsymbol{z}_1}^\eta(\boldsymbol{z}_1)\mu(\boldsymbol{z}_1) = 0, \quad \int f_{\boldsymbol{z}_2}^\eta(\boldsymbol{z}_2)\mu(\boldsymbol{z}_2) = 0, \int f_{\boldsymbol{z}_1,\boldsymbol{z}_2}^\eta(\boldsymbol{z}_1, \boldsymbol{z}_2)\mu(\boldsymbol{z}_i) = 0, \quad \forall \boldsymbol{z}_{j\neq i}. \tag{9}$$

By imposing constraints on the measurable function $G_m^* = f_\eta(\boldsymbol{x}, \boldsymbol{z}_1, \boldsymbol{z}_2)$, we obtain a unique decomposition of G_m^* into a sum of orthogonal functional bases. Such decomposition of G_m^* is directly related to σ-CVAE in $G_{\theta,\sigma}(X, (Z_1, Z_2))$. By analyzing their equality conditions, we obtain a non-asymptotic approximation results without the *asymptotic* assumption of $\sigma \to 0$

Theorem 2.4 (**Non-asymptotic approximation of σ-CVAE**)**.** *Under Assumptions 2.3, for some $m > q$, if the block neural decomposition of ground-truth map G_m^* satisfy the following conditions almost everywhere up to zero measure of $\boldsymbol{z}_2$: 1) $f_{\boldsymbol{z}_2}(\boldsymbol{z}_2) = \sigma^* \boldsymbol{z}_2$ and 2) $f_{\boldsymbol{x}\boldsymbol{z}_2}^\eta(\boldsymbol{z}_1, \boldsymbol{z}_2) + f_{\boldsymbol{x}\boldsymbol{z}_2}^\eta(\boldsymbol{x}, \boldsymbol{z}_2) + f_{\boldsymbol{x}\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{x}, \boldsymbol{z}_1, \boldsymbol{z}_2) = C$ for some $\sigma^* > 0$, and constant C, then there exist a σ-CVAE model with decoder parameterized by $\theta(\eta)$, and decoder variance scalar σ such that*

$$G_{\theta,\sigma}(\boldsymbol{x}, (\boldsymbol{z}_1, \boldsymbol{z}_2)) = G_m^*(\boldsymbol{x}, \boldsymbol{z}) \quad a.s. \tag{10}$$

The proof is deferred to Appendix D. The proof is nothing more than pattern matching under the unique decomposition of G_m^*. Theorem 2.4 provides deeper insight into the optimality of decoder parameters. First, it shows that there exist one or more optimal parameters $\{\theta^*, \sigma^*\}$ such that the parametric density $p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x})$ can recover the ground-truth density, but they are not identifiable without additional assumptions. Second, it shows that a zero approximation gap is more generally available and does not require an asymptotic condition taking $\sigma \to 0$. To this extent, theorem 2.4 can be seen as a weaker form of proposition B.2. It does not assume that G_m^* is approximated by a finite Gaussian mixture, which restricts the expressive power of σ-CVAE model Mattei & Frellsen (2018). In addition, it allow non-linear relationship of x and z_1 that goes beyong assuming G_m^* is a probabilistic PCA model in linear VAEs (Lucas et al., 2019b; Dai et al., 2020; Sicks et al., 2021; Wang & Ziyin, 2022; Dang et al., 2023), which cannot extend to the complex datasets used in Section 4.1.

Theorem 2.4 also aligns with the empirical evidence of suboptimality in variational inference presented in Cremer et al. (2018). When q_{ϕ^*} does not perfect recover the true posterior, Theorem 2.4 reassures that there exists a generative distribution $p_{\theta^*,\sigma^*}(\boldsymbol{y}|\boldsymbol{x})$ with global optimal θ^*, σ^* that well approximate the ground truth data generating distribution, demonstrating its generative capabilities.

3 What if variational inference is not optimal?

Given the non-identifiability issue in σ and the existence of suboptimal encoders, it is of great importance that accurate estimates of data-generating distribution are obtained through numerically robust optimization. Traditional wisdom in likelihood-based models has been optimizing the loss by updating σ jointly with $\{\theta, \phi\}$ in Dai et al. (2021); Dai & Wipf (2018) or by updating σ in the dual-step coordinate descent in Rybkin et al. (2021).

In the analysis below, we show how suboptimal variational inference that is not guaranteed in approximation theorem leads to biased parameter estimation. Apart from the unbounded likelihood for having a learned σ, and model non-identifiability, we demonstrate another source of numerical instability in optimizing σ that is caused by *suboptimal* variational inference. As evidenced by the experiments in Section 4.1, we propose novel algorithmic improvements to calibrate the decoder variance.

3.1 The necessity of σ-calibration: bias in the gradient

Given finite N i.i.d observations $D = \{\boldsymbol{y}_i, \boldsymbol{x}_i\}_{i=1}^N$ from $p_{gt}(\boldsymbol{x}, \boldsymbol{y})$, the objective of CVAEs is given by

$$L(\theta, \phi, \sigma) = \frac{1}{N} \sum_{i=1}^{N} \left[\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{y}_i, \boldsymbol{x}_i)}[-\log p_{\theta,\sigma}(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{z}))] + \sum_{i=1}^{N} \mathrm{KL}[q_\phi(\boldsymbol{z}|\boldsymbol{y}_i, \boldsymbol{x}_i)||p(\boldsymbol{z})] \right]. \tag{11}$$

The gradient of $L(\theta, \phi, \sigma)$ with respect to σ is a biased *approximation* of the true gradient of the negative log-likelihood function with respect to σ. To see this, recalling Fisher's identity,

$$\nabla_\sigma \, -\log p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x}) = \mathbb{E}_{p_{\theta,\sigma}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})}[-\nabla_\sigma \log p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})]. \tag{12}$$

However, the possibly biased gradient of σ in the CVAE loss function is given by

$$\nabla_\sigma L(\theta, \phi, \sigma) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{y}_i, \boldsymbol{x}_i)}[-\nabla_\sigma \log p_{\theta,\sigma}(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{z})]. \tag{13}$$

This similar argument can be extended to the dual-step coordinate descent method Rybkin et al. (2021) for robust likelihood estimation, where the optimal σ_t^* is obtained analytically by the maximum likelihood principle with optimal step size. Given $\{\theta_t, \phi_t\}$ at time t,

$$\sigma_t^* = \arg\min_\sigma L(\theta_t, \phi_t, \sigma) = \arg\min_\sigma \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{q_{\phi_t}(\boldsymbol{z}|\boldsymbol{y}_i, \boldsymbol{x}_i)} \left[-\log p_{\theta_t,\sigma}(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{z}) \right], \tag{14}$$

where ideally the expectation should be taken over intractable posterior.

This analysis uncovers an estimation issue in Gaussian σ-VAE models, that is, the biased estimation of variance σ in the presence of suboptimal variational inference. This bias is often overlooked in the literature due to unrealistic simplifications. Such simplifications can arise, for instance, in asymptotic analysis (Dai & Wipf, 2018; Zheng et al., 2022) or in a linear VAE setting (Lucas et al., 2019b; Dai et al., 2020; Wang & Ziyin, 2022). We argue that these simplifications not only limit the true expressive power of VAE models, as stated in Theorem 2.4, but also underestimate the practical challenges in robust parameter estimation.

3.2 Posterior collapse: a compelling evidence for suboptimal encoders

It is analytically infeasible to establish a rigorous metric that quantifies the discrepancy between the approximate posterior and the true intractable posterior $p_{\theta,\sigma}(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x})$. Thus, a rigorous theory of calibration would require additional assumptions. For example, $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})$ can be assumed as a corrupted model of intractable $p_{\theta,\sigma}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})$ by defining a $\epsilon-$corruption model (Acharya et al., 2022) to characterize the discrepancy between the posterior and the approximate variational distribution at convergence. Then one can derive some results about its breakdown point. In such a theory, the geometric median of the gradient of σ could be used as a calibration to approximate its true gradient, but it is computationally expensive when extending to a full diagonal covariance Σ that might hurt its scalability.

In this paper, we argue that actions should be taken if there is clear and compelling evidence of a poor encoder q_ϕ. Specifically, we consider a well-known example of the evidence, termed *posterior collapse* (Lucas et al., 2019a; He et al., 2019; Razavi et al., 2019) or *KL vanishing* (Bowman et al., 2016; Fu et al., 2019). It refers to the problem that the approximate posterior $q_\phi(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x})$ collapses to the standard Gaussian $N(0, I_d)$, resulting in vanishing KL divergence in the training loss function.

$$\frac{1}{N} \sum_{i=1}^{N} \mathrm{KL}[q_\phi(\boldsymbol{z}|\boldsymbol{y}_i, \boldsymbol{x}_i)||p(\boldsymbol{z})] \approx 0. \tag{15}$$

Obviously, the encoder q_ϕ that approximates the true posterior will hardly become the prior. This is because, under Bayes' rule, i.e. $p_{\theta,\sigma}(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x}) \propto p(\boldsymbol{z}) p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})$, $p_{\theta,\sigma}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})$ is never designed to be independent of

$\boldsymbol{z}$ or equivalently, the mean $\mu_\theta(\boldsymbol{x}, \boldsymbol{z})$ in the decoder is not a function of $\boldsymbol{z}$. If the VAE generative model ignores a latent variable, it collapses to nonlinear regression, significantly reducing the expressive power. As a result, posterior collapse often coincides with poor ELBO in complex datasets due to lack of fit.

The standard solution for KL vanishing is KL annealing (Raiko et al. (2007); Bowman et al. (2016); Fu et al. (2019)) using β-VAEs. The idea is that the loss landscape w.r.t. θ, ϕ is highly non-convex when σ is fixed. Thus, tuning the hyperparameter β, which reweights the KL divergence in the loss function, can prevent posterior collapse, resulting in a relatively large KL divergence in Eq.15 and a lower training ELBO loss. For example, the schedule of β in Fu et al. (2019) is

$$\beta_t = 2r/M * 1_{\{0 \leq r < M/2\}}(r) + 1 * 1_{\{M/2 \leq r < M\}}(r), \tag{16}$$

where $r = \text{mod}(t, M)$. The β_t is periodically annealed from 0 to 1 in first half of a M-iteration cycle and stayed at 1 throughout the rest of the cycle. Obviously, the statistical insight of the anneal schedule is obscure in principle. In fact, we can show that these KL annealing techniques are computationally expensive. See Appendix F for a theoretical discussion and experimental evidence.

Considering the mathematical equivalence between σ and β in Eq.5, KL annealing can be seen as the calibration of scaled variance σ. Taking Eq.16 as an example, the loss-equivalent decoder variance σ_t is annealed as follows.

$$\sigma_t^2 = M/4r * 1_{\{0 \leq r < M/2\}}(r) + 1/2 * 1_{\{M/2 \leq r < M\}}(r), \tag{17}$$

where $r = \text{mod}(t, M)$. Thus, preventing posterior collapse can be easily achieved without introducing β, because we can always calibrate σ to change the loss landscape of θ, ϕ.

3.3 σ-calibration: an implicit KL annealing

In this paper, we propose Calibrated Robust CVAE, a simple doubly robust framework incorporating calibrations of σ using the block coordinate gradient descent method.

A general pseudo-code can be found in Algorithm 1. At a high level, this block coordinate gradient descent algorithm consists of three steps. The first step is the optimization of θ, ϕ as a block at a given fixed σ, where K iterations (stochastic) gradient descent is implemented. We refer hyperparameter K as *inner steps*. The second step is the maximum likelihood estimate of σ, obtaining the optimal σ in closed form based on the current θ, ϕ, as described in Rybkin et al. (2021). The third and most important step is called *σ-calibration step* at convergence, where the σ will be calibrated if compelling evidence for calibration is found. In this step, it rejects any suboptimal encoder q_ϕ that possibly leads to an unreliable generating model. Specifically, if the KL divergence term in loss is less than the hyper-parameter *calibration tolerance* C, the calibrated σ^{cal} is given by

$$\sigma_t^{\text{cal}} := |1 + \varphi| \times \sigma_t, \tag{18}$$

where we sample $\varphi \sim N(0, S)$ with variance S starts at 1 increasing by 1 after each calibration. Figure 2(a) illustrates how σ is calibrated throughout the training process.

Rationale behind calibration-at-convergence: The design of the calibration is inspired from the existing KL annealing techniques, but we have three key considerations: 1) calibration must be bidirectional, similar to KL annealing. The variance S, which governs the calibration magnitude, should increase when stronger calibration is required. 2) calibration should be adaptive, as there is no bias in the estimation of σ when an optimal ϕ is obtained. 3) calibration should be lazy, given the iterative nature of the training process.

Our σ-CVAE calibration method offers three key advantages over a β-VAE with KL annealing. First, our objective sticks to the original ELBO loss and avoids explicit usage of β, making it more suitable for reliable probabilistic modeling. Second, our calibration process is straightforward and results-driven, effectively calibrating σ until the desired goal is achieved. A predefined annealing schedule is no longer needed. Third, our calibration is highly interpretable. We utilize the metric of posterior collapse is used to trigger calibration and our hyperparameter C is the permissible deviation from a zero KL divergence.

Algorithm 1: general σ calibration for σ-CVAE

Input: data $\{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^N$, deterministic neural networks $f_{\mu_y}, f_{\mu_z}, f_{S_y}$ with initialized parameter θ_0, ϕ_0, Initialization σ_0, inner number of iteration K of updating θ, ϕ, Calibration tolerance C
Output: θ, ϕ, σ
while is training **do**
  for $i = 1$ **to** K **do**
    Read batch $\{\boldsymbol{x}_j, \boldsymbol{y}_j\}_{j=1}^B$ from data
    Sample $\boldsymbol{z}_j \leftarrow q_{\phi_t}(\boldsymbol{z}|\boldsymbol{y}_j, \boldsymbol{x}_j)$ for $j = 1, ..., B$
    Compute batch loss $L(\theta_t, \phi_t, \sigma_t)$
    Compute gradient $\nabla_\theta L; \nabla_\phi L$
    $\theta_t \leftarrow \theta_t - \alpha \nabla_\theta L$
    $\phi_t \leftarrow \phi_t - \alpha \nabla_\phi L$
  end for
  Sample $\boldsymbol{z}_j \leftarrow q_{\phi_t}(\boldsymbol{z}|\boldsymbol{y}_j, \boldsymbol{x}_j)$ for $j = 1, ..., N$;
  Update Optimal $\sigma_t^2 \leftarrow \frac{1}{N} \sum_{i=1}^N \|\boldsymbol{y}_i - \mu_{\theta_t}(\boldsymbol{x}_i, \boldsymbol{z}_i)\|^2 / q$
  if Convergence **then**
    if $\sum \mathrm{KL}[q_\phi(\boldsymbol{z}|\boldsymbol{y}_i, \boldsymbol{x}_i)||p(\boldsymbol{z})] < C$ **then**
      Calibrate $\sigma_t \leftarrow \sigma_t^{cal}$ as Eq.18
    end if
  else
    Break
  end if
end while

4 Experiments

In this section, we provide more practical insights in the calibration of σ-CVAE through extensive numerical experiments given a finite training sample. We use the Adam (Kingma, 2014) stochastic gradient descent algorithm in training neural networks. The general learning rate is 0.005 and the convergence threshold is 0.001 in the average loss change. Other important details of each of the following experiments are attached in the Appendix E. In addition, we also validate our method in the MNIST (Deng, 2012) and CelebA (Liu et al., 2018) datasets for conditional image generation and reconstruction in Appendix H & I.

4.1 Two-moon dataset

Let $\boldsymbol{x} \in \{-1, 1\}$ be the binary input and let $\boldsymbol{y} \in \mathbb{R}^2$ be generated as follows,

$$\boldsymbol{y} = \begin{cases} \left(\cos(\alpha) + \frac{1}{2} + \epsilon_1, \sin(\alpha) - \frac{1}{6} + \epsilon_2\right), & \text{if } \boldsymbol{x} = -1, \\ \left(\cos(\alpha) - \frac{1}{2} + \epsilon_3, -\sin(\alpha) + \frac{1}{6} + \epsilon_4\right), & \text{if } \boldsymbol{x} = 1, \end{cases}$$

where $\alpha \sim \mathrm{Uniform}[0, \pi]$ and $\epsilon_i \sim N\left(0, \tau^2\right), \forall i = 1, 2, 3, 4$. Note that $\alpha, \epsilon_1, \ldots, \epsilon_4$ are mutually independent. We simulated three datasets of size $n = 5,000$ with $2,500$ for each class using $\tau = 0.1$, which is refered later as **the true** σ. Note that this synthetic dataset does not follow the pPCA model described in Lucas et al. (2019a) and cannot be recovered by a linear CVAE. A well-behaved encoder, in this case, is expected to yield a non-zero KL divergence.

A learned variance is crucial for VAE generative power. In Figure 1, we validate that vanilla σ-CVAE with small K, including $K = 1$ as suggested in Rybkin et al. (2021), may fail to recover the truth. This observation agrees with Dai et al. (2021), which notes that learning σ can have unintended consequences. In our case, a different size of the inner step K would result in completely different generative model at convergence, given the same training dataset.

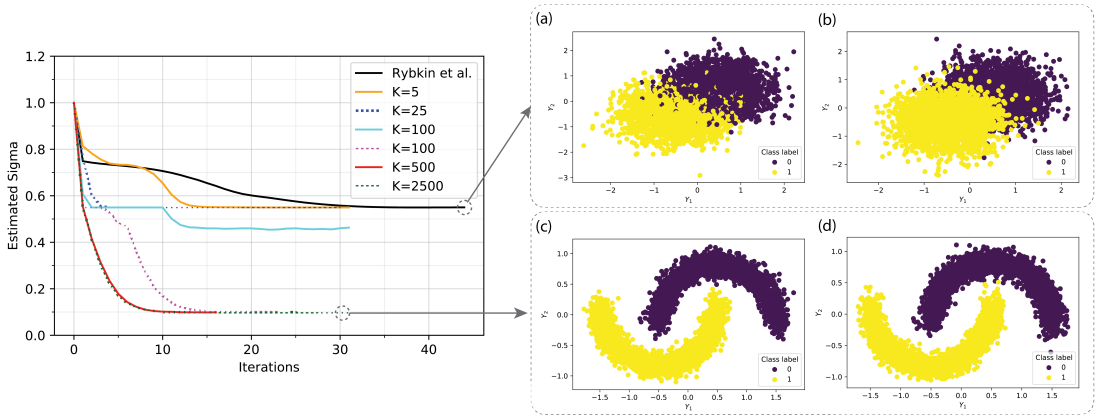
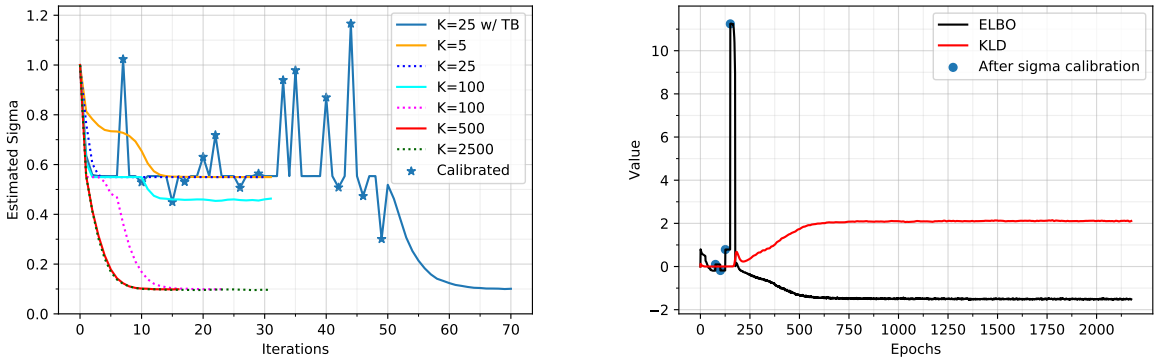


Figure 1: (a),(c) reconstructed training Twomoon dataset; (b),(d) generated samples. Vanilla σ-CVAEs may not recover the true σ, reconstruct training data, and learn the ground-truth distribution. The true σ is 0.1.

The contradictory result is well predicted by Theorem 2.4. Although σ-CVAEs are not guaranteed to achieve an optimal variance inference, $G_{\theta,\sigma}$ in Eq.7 can have amazing expressive capacity in estimating the ground truth map G in Eq.6, particularly when the non-identifiable parameter, variance σ, is accurately estimated.

Better encoder leads to better variance estimation. In addition, we see in Figure 1 (left) that increasing the size of the inner step K leads to a better estimation of σ and a better learned generative model. This is precisely what we emphasize in Section 3.1. The gradient bias of σ can be reduced by obtaining a more informative ϕ through a longer K inner steps. As a result, the dynamics of σ is less likely to be trapped in local optima.

Of course, smaller K is always preferred, and excessively large K should be avoided to ensure computational efficiency. Unfortunately, K is a hyperparameter and an appropriate setup of K to balance the robustness of variance estimation with computational efficiency is generically unknown.



(a) Robust σ estimation with small K

(b) σ-calibration as an KL annealing

Figure 2: (a) Robust estimate of σ is obtained by calibration with a small K (K=25 w/ TB). The calibrated σ are marked as $*$; (b) The KL divergence in ELBO remains positive after several calibrations of σ. The calibration tolerance $C = 0.05$. The true σ is 0.1.

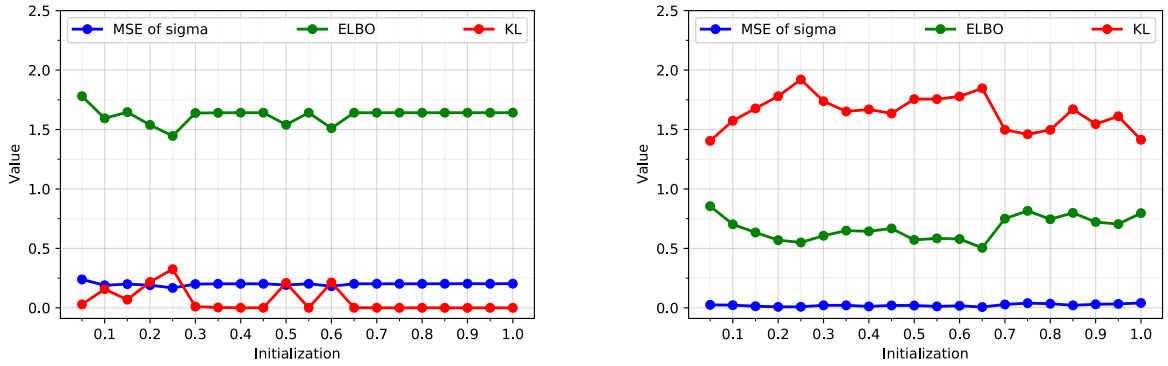
σ-calibration is a fast implicit KL annealing that prevents posterior collapse. In contrast, the results of our proposed calibrated robust σ-CVAE, using a small $K = 25$, is shown in Figure 2. Our proposed calibrated robust σ-CVAE, successfully estimates the decoder variance in Figure 2(a), as well as recover the ground truth distribution with a low ELBO in Figure 2(b).

In addition, variance calibration on σ is as effective as KL annealing techniques in preventing posterior collapse without the need for a reweighting hyperparameter β. This is evident in Figure 2(b). After the

last calibration of σ, we observe a spike in ELBO training loss, increasing from below 1 to above 11. This indicates that the change in σ successfully anneals the loss landscape, preventing the model from converging to a suboptimal q_ϕ obtained before calibration. The KL divergence remains positive after the last σ calibration, showing that the annealed loss landscape leads to a better q_ϕ that is free from posterior collapse.

Thus, the calibration of σ works exactly as KL annealing, but in a fast and implicit manner. Note that the hyperparameters of our calibration are the calibration tolerance C and the inner step K, so a predefined annealing schedule system (Raiko et al., 2007; Bowman et al., 2016; Fu et al., 2019) is no longer needed. Also, we do not assume σ to be fixed at 1, so we optimize the lower bound of likelihood rather than a reweighed loss. Compared with vanilla CVAE that recovers the true σ using large $K = 500, 2500$, our algorithm only needs $30\%, 6\%$ of total epochs to converge, respectively.

Compared to Monotonic Bowman et al. (2016) or Cylical annealing Fu et al. (2019), our algorithm can be $50 - 70\%$ faster to converge. A detailed comparison of wall time savings is in the Appendix.F



(a) without calibration

(b) with calibration

Figure 3: CVAE model using a learned σ (a) without calibration and (b) without our proposed calibration on twomoon training datasets. It prevents a vanishing KL divergence due to posterior collapse and significantly improves the decoder variance estimates and the ELBO at convergence. The ground-truth $\sigma = 0.1$. The calibration tolerance $C = 0.05$. Inner step $K = 25$. Each point is averaged over 20 repetition experiments.

σ-calibration brings doubly robust estimation We compared the results of CVAEs that iteratively update σ with various *initializations* of σ ranging from 0.05 to 1. Shown in Figure 3(a), the CVAE training without calibration ends up with a very high training loss (ELBO), and a large mean squared error (MSE) of the learned σ to the ground-truth. and a vanishing KL divergence (KL) of prior and encoder. In alignment with Dai & Wipf (2018); Dang et al. (2023), we found that a smaller initialization of σ have a higher chance of avoiding posterior collapse with a non-zero KL divergence. However, small initialization are limited in preventing posterior collapse and it still may fail to recover the ground-truth distribution. Results of a fixed-σ CVAE can be found in the ablation study in Appendix G.

Incorporating our σ-calibration, the σ-CVAE significantly outperforms on the metrics, shown in Figure 3 (b). It has a smaller MSE, a nonzero KL, and a smaller ELBO, which is consistent among all the initialization of σ.

Compared to Monotonic Bowman et al. (2016) or Cylical annealing Fu et al. (2019), our algorithm can adjust the hyper-paramter C requiring for better encoders, therefore resulting in significantly less biased estimation of decoder (fine-tuning of C requried). See experimental details in in the Appendix.F

4.2 Comparison with nonparametric conditional density estimator

Here, we compare the performance calibrated σ-CVAE with the Wasserstein generative conditional sampling method (WGCS) Liu et al. (2021), along with three conventional nonparametric conditional density

Table 2: Mean squared error of the estimated conditional mean (MEAN), the estimated standard deviation (SD) and the corresponding simulation standard errors (in parentheses) on test dataset. The numbers in bold indicate the best results among the same row. The evaluation is repeated 10 times. Smaller number means a better estimation of conditional mean and variance.

		CVAE (ours)	WGCS	NNKCDE	CKDE	FlexCode
M_1	Mean	**0.174** (0.004)	1.10(0.05)	2.49(0.01)	3.30(0.02)	2.30(0.01)
	SD	**0.185**(0.005)	0.24(0.04)	0.43(0.01)	0.59(0.01)	1.06(0.08)
M_2	Mean	**0.421**(0.005)	3.71(0.23)	6.09(0.07)	66.76(2.06)	10.20(0.33)
	SD	**2.071**(0.019)	3.52(0.17)	9.33(0.23)	18.87(0.59)	11.08(0.34)
M_3	Mean	**0.106**(0.001)	0.32(0.03)	0.11(0.01)	1.55(0.03)	0.12(0.04)
	SD	0.421(0.001)	**0.10**(0.01)	0.36(0.01)	0.51(0.01)	0.33(0.01)

estimators, including the nearest-neighbor kernel conditional density estimator (NNKCDE) Dalmasso et al. (2020), the conditional kernel density estimator (CKDE) Hall et al. (2004), and a basis expansion method FlexCode (Izbicki & Lee (2017)) in the tasks of estimating the conditional mean and conditional standard deviation.

Three different models M_1, M_2, M_3 simulated datasets are used. Details can be found in Appendix E. We evaluated the learned conditional distribution in terms of the estimated conditional mean $\hat{\mathbb{E}}[\boldsymbol{y}|\boldsymbol{x}_k]$ and the estimated conditional standard deviation $\hat{\mathrm{SD}}[\boldsymbol{y}|\boldsymbol{x}_k]$ and our method produces consistent results and outperforms the other model, as it has the smallest MSEs for estimating the conditional mean and conditional standard deviation in most cases shown in Table 2.

4.3 Comparison with Conditional Flow Based model in benchmark datasets

In this section, we compare the performance of calibrated σ-CVAE with Bayesian conditional normalizing flowsTrippe & Turner (2018) in Table 3 on 6 UCI datasets in terms of the test mean log-likelihood in nats. We must emphasize that σ-CVAEs is not an exact likelihood-based method and their optimization is on the lower bound of the likelihood without global optimal variantional inference implied by Theorem 2.4, so our method is very competitive with other methods.

Table 3: Mean log-likelihood of test data ± variance (in nats), compared with Bayesian normalizing flows Trippe & Turner (2018), mixture density networks, and neural networks with latent inputs on six small UCI benchmark datasets. A higher logarithmic likelihood implies a better generalization of the test set. The random train-test split is 75% to 25% Each experiment is repeated 5 times.

Dataset	boston	concrete	energy	power	wine-red	yacht
N	506	1030	768	9568	1599	308
p/q	13/1	8/1	8/2	4/1	11/1	6/1
MDN-2	-2.65 ± 0.03	-3.23 ± 0.03	-1.60 ± 0.04	-2.73 ± 0.01	-0.91 ± 0.04	-2.70 ± 0.05
MDN-5	-2.73 ± 0.04	-3.28 ± 0.03	-1.63 ± 0.06	-2.70 ± 0.01	$\mathbf{+1.43 \pm 0.07}$	-2.54 ± 0.10
MDN-20	-2.74 ± 0.03	-3.27 ± 0.02	-1.48 ± 0.04	-2.68 ± 0.01	$+1.21 \pm 0.06$	-2.76 ± 0.07
LV-15	-2.64 ± 0.05	-3.06 ± 0.03	-0.74 ± 0.03	-2.81 ± 0.01	-0.98 ± 0.02	-1.01 ± 0.04
LV-5	-2.56 ± 0.05	-3.08 ± 0.02	-0.79 ± 0.02	-2.82 ± 0.01	-0.96 ± 0.01	-1.15 ± 0.05
NF-2	-2.40 ± 0.06	-3.03 ± 0.05	-0.44 ± 0.04	-2.73 ± 0.01	-0.87 ± 0.02	-0.30 ± 0.04
NF-5	-2.37 ± 0.04	-2.97 ± 0.03	-0.67 ± 0.15	$\mathbf{-2.68 \pm 0.01}$	-0.76 ± 0.10	$\mathbf{-0.21 \pm 0.09}$
HMC	$\mathbf{-2.27 \pm 0.03}$	$\mathbf{-2.72 \pm 0.02}$	-0.93 ± 0.01	-2.70 ± 0.00	-0.91 ± 0.02	-1.62 ± 0.02
Dropout	-2.46 ± 0.25	-3.04 ± 0.09	-1.99 ± 0.09	-2.89 ± 0.01	-0.93 ± 0.06	-1.55 ± 0.12
MF	-2.62 ± 0.06	-3.00 ± 0.03	-0.57 ± 0.04	-2.79 ± 0.01	-0.97 ± 0.01	-1.00 ± 0.10
CVAE(ours)	-3.17 ± 0.07	-3.48 ± 0.04	$\mathbf{+0.22 \pm 0.10}$	-3.29 ± 0.03	-1.48 ± 0.06	-0.34 ± 0.02

5 Discussion

Although the KL divergence in Eq.15 is a direct measurement of the difference between the Gaussian encoder and the prior, the criteria of posterior collapse remain flexible. For example, Lucas et al. (2019b) defined the dimension-wise posterior collapse by $\mathrm{P}(\mathrm{KL}[q_\phi(\boldsymbol{z}_i|\boldsymbol{y},\boldsymbol{x})||p(\boldsymbol{z}_i)] \leq \epsilon) > 1-\delta$ for each latent dimension index i in a probably approximately correct manner. Similarly, Dai et al. (2021) defined implicit posterior collapse from observing a large maximum mean discrepancy (Makhzani et al., 2016) between the aggregated posterior $\frac{1}{N}\sum_{i=1}^{N} q_\phi(\boldsymbol{z}|\boldsymbol{y}_i,\boldsymbol{x}_i)$ and $p(\boldsymbol{z})$. This is because marginalizing a well-behaved encoder would be similar to the prior, that is, $\int q_\phi(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})\mu_{gt}(d\boldsymbol{y}d\boldsymbol{x}) \approx \int p_{\theta,\sigma}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})\mu_{gt}(d\boldsymbol{y}d\boldsymbol{x}) \approx p(\boldsymbol{z})$.

Our calibrated dual-step coordinate descent algorithm depends on two hyperparameters, namely C, K. While it eliminates the need for an expensive and ad hoc annealing schedule, these hyperparameters are crucial and must be carefully tuned. These values may vary depending on the specific application. We recommend using a small tolerance C and a moderate K inner steps for the sake of computational efficiency as a general rule of thumb. Using an excessively large tolerance C will not only trigger more calibrations of σ but also lead to potential non-convergence of the algorithm.

The calibration step can be viewed as a warm restart if we calibrate σ at convergence back to its initialized value. That being said, we note that a cold restart of our algorithm remains effective for an extremely poor initialization of θ, ϕ, which we have occasionally encountered. We used default initialization of Pytorch modules, while Xavier's (Glorot & Bengio, 2010) is also a possible choice. An in-depth investigation of initialization strategies would be a valuable follow-up of this work.

If reliable probabilistic modeling and robust estimation are not the top priority, it is not common to incorporate both β and σ simultaneously, especially in image generation tasks, including crispy face generation (Vahdat & Kautz, 2020) and abnormality detection (Loizillon et al., 2024). Based on results from Appendix H & I, σ-calibration itself does not significantly improve the image generation results. It should be noted that the processes for generating images are sometimes unclear, and it is possible that the mean $\mu_\theta(\boldsymbol{x},\boldsymbol{z})$ without sampling additive decoder variance is used as the generated sample. It is neither aligned with the assumptions of a Gaussian decoder nor does it represent an accurate generative process.

Of course, the setup of our analysis and algorithm is subject to several conventional assumptions. These assumptions can be relaxed in several aspects including 1) generalizing our analysis to accommodate a heterogeneous, diagonal, or even full covariance matrix Σ, which offers better approximation properties for more complex datasets, 2) incorporating additional robust estimation techniques, which is essential for addressing challenges such as imbalanced data, unlabeled data, etc., and 3) incorporating with state-of-the-art image variants for better conditional image generations.

In these scenarios, the connection between decoder variance Σ and β is reflected in the leading eigenvalues of Σ, and more importantly, designing and justifying a calibration technique requires additional considerations to ensure that the calibration is not only empirically effective but also theoretically sound. We also highlight that in hierarchical Bayesian analysis (Hoffman & Blei, 2015; Agrawal & Domke, 2021; Margossian & Blei, 2023), θ can be seen as the global latent parameter, and the prior distribution of the local latent variable $\boldsymbol{z}$ could depend on θ, which would also complicate the analysis.

To the best of our knowledge, this work is the first comprehensive study on the variance parameter σ in VAE models. We prove the existence of the bias in the gradient of σ, and the need for calibration in theory and empirically to provide a robust and accurate estimation on complex datasets. It shows that calibrating σ can prevent posterior collapse by avoiding bad local optimal encoders through an annealed loss landscape, such that explicitly introducing a hyperparameter β is more than redundant. The calibration of σ in turn provides a robust estimate of σ itself due to a less biased gradient through better encoders.

In conclusion, our analysis provides deeper statistical insights into existing VAE learning theories on the approximation gap and the loss equivalence of β and σ. More importantly, our proposed algorithm can be practically inspiring for various VAE applications that require reliable probabilistic modeling, while a fixed σ or explicit β is still commonly used by convention.

References

Anish Acharya, Abolfazl Hashemi, Prateek Jain, Sujay Sanghavi, Inderjit S. Dhillon, and Ufuk Topcu. Robust training in high dimensions via block coordinate geometric median descent. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*. PMLR, 2022.

Abhinav Agrawal and Justin Domke. Amortized variational inference for simple hierarchical models. *Advances in Neural Information Processing Systems*, 34:21388–21399, 2021.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. Generating sentences from a continuous space. In *CoNLL*, pp. 10–21. ACL, 2016. ISBN 978-1-945626-19-7.

Yuri Burda, Roger B. Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. In *International Conference on Learning Representations*, 2016.

Lluis Castrejon, Nicolas Ballas, and Aaron Courville. Improved conditional vrnns for video prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7608–7617, 2019.

Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. *Advances in Neural Information Processing Systems*, 2018.

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*, 2018.

Bin Dai, Ziyu Wang, and David Wipf. The usual suspects? reassessing blame for vae posterior collapse. In *International conference on machine learning*, pp. 2313–2322. PMLR, 2020.

Bin Dai, Li Wenliang, and David Wipf. On the value of infinite gradients in variational autoencoder models. *Advances in Neural Information Processing Systems*, 34:7180–7192, 2021.

Niccolò Dalmasso, Taylor Pospisil, Ann B Lee, Rafael Izbicki, Peter E Freeman, and Alex I Malz. Conditional density estimation tools in python and r with applications to photometric redshifts and likelihood-free cosmological inference. *Astronomy and Computing*, 30:100362, 2020.

Hien Dang, Tho Tran Huu, Tan Minh Nguyen, and Nhat Ho. Beyond vanilla variational autoencoders: Detecting posterior collapse in conditional and hierarchical variational autoencoders. In *The Twelfth International Conference on Learning Representations*, 2023.

Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012. doi: 10.1109/MSP.2012.2211477.

Nicki S. Detlefsen, Martin Jørgensen, and Søren Hauberg. Reliable training and estimation of variance networks, 2019.

Carl Doersch. Tutorial on variational autoencoders, 2021.

Hao Fu, Chunyuan Li, Xiaodong Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. Cyclical annealing schedule: A simple approach to mitigating kl vanishing. *arXiv preprint arXiv:1903.10145*, 2019.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research. PMLR, 2010.

Peter Hall, Jeff Racine, and Qi Li. Cross-validation and the estimation of conditional probability densities. *Journal of the American Statistical Association*, 99(468):1015–1026, 2004.

Junxian He, Daniel Spokoyny, Graham Neubig, and Taylor Berg-Kirkpatrick. Lagging inference networks and posterior collapse in variational autoencoders. In *International Conference on Learning Representations*, 2019.

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*, 2016.

Matthew D Hoffman and David M Blei. Structured stochastic variational inference. In *Artificial Intelligence and Statistics*, pp. 361–369, 2015.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

Xianxu Hou, Ke Sun, Linlin Shen, and Guoping Qiu. Improving variational autoencoder with deep feature consistent and generative adversarial training. *Neurocomputing*, 341:183–194, 2019.

Chin-Wei Huang, Kris Sankaran, Eeshan Dhekane, Alexandre Lacoste, and Aaron C. Courville. Hierarchical importance weighted autoencoders. In *International Conference on Machine Learning*, Proceedings of Machine Learning Research, pp. 2869–2878. PMLR, 2019.

Rafael Izbicki and Ann B Lee. Converting high-dimensional regression to high-dimensional conditional density estimation. *Electronic Journal of Statistics*, 11:2800–2831, 2017.

Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International conference on artificial intelligence and statistics*, pp. 2207–2217. PMLR, 2020.

Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

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

Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29, 2016.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Bruce G Lindsay. Mixture models: theory, geometry, and applications. Ims, 1995.

Shiao Liu, Xingyu Zhou, Yuling Jiao, and Jian Huang. Wasserstein generative learning of conditional distribution, 2021.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision*, 2015.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018):11, 2018.

Sophie Loizillon, Yannick Jacob, Maire Aurélien, Didier Dormont, Olivier Colliot, Ninon Burgos, and AP-PRIMAGE Study Group. Detecting brain anomalies in clinical routine with the β-VAE: Feasibility study on age-related white matter hyperintensities. In *Medical Imaging with Deep Learning*, 2024.

James Lucas, George Tucker, Roger Grosse, and Mohammad Norouzi. Understanding posterior collapse in generative latent variable models, 2019a.

James Lucas, George Tucker, Roger B Grosse, and Mohammad Norouzi. Don't blame the elbo! a linear vae perspective on posterior collapse. In *Advances in Neural Information Processing Systems*, volume 32, 2019b.

Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders, 2016.

Charles C. Margossian and David M. Blei. Amortized variational inference: When and why?, 2023.

Kaspar Märtens and Christopher Yau. Neural decomposition: Functional anova with variational autoencoders. In *International Conference on Artificial Intelligence and Statistics*, pp. 2917–2927. PMLR, 2020.

Pierre-Alexandre Mattei and Jes Frellsen. Leveraging the exact likelihood of deep latent variable models. *Advances in Neural Information Processing Systems*, 31, 2018.

Sebastian Nowozin. Debiasing evidence approximations: On importance-weighted autoencoders and jackknife variational inference. In *International conference on learning representations*, 2018.

Tapani Raiko, Harri Valpola, Markus Harva, and Juha Karhunen. Building blocks for variational bayesian learning of latent variable models. *Journal of Machine Learning Research*, 8(1), 2007.

Rajesh Ranganath, Dustin Tran, and David Blei. Hierarchical variational models. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 324–333. PMLR, 2016.

Ali Razavi, Aaron van den Oord, Ben Poole, and Oriol Vinyals. Preventing posterior collapse with delta-vaes. In *International Conference on Learning Representations*, 2019.

Danilo Jimenez Rezende and Fabio Viola. Taming vaes. *arXiv preprint arXiv:1810.00597*, 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.

Oleh Rybkin, Kostas Daniilidis, and Sergey Levine. Simple and effective vae training with calibrated decoders. In *International Conference on Machine Learning*, pp. 9179–9189. PMLR, 2021.

Robert Sicks, Ralf Korn, and Stefanie Schwaar. A generalised linear model framework for β-variational autoencoders based on exponential dispersion families. *Journal of Machine Learning Research*, 22(233): 1–41, 2021.

Nicki Skafte, Martin Jørgensen, and Søren Hauberg. Reliable training and estimation of variance networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Ilya M Sobol. Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and computers in simulation*, 55(1-3):271–280, 2001.

IM Soboĺ. Sensitivity estimates for nonlinear mathematical models. *Math. Model. Comput. Exp.*, 1:407, 1993.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems*, 2015.

Andrew Stirn and David A Knowles. Variational variance: Simple, reliable, calibrated heteroscedastic noise variance parameterization. *arXiv preprint arXiv:2006.04910*, 2020.

Hiroshi Takahashi, Tomoharu Iwata, Yuki Yamanaka, Masanori Yamada, and Satoshi Yagi. Student-t variational autoencoder for robust density estimation. In *IJCAI*, pp. 2696–2702, 2018.

Brian L Trippe and Richard E Turner. Conditional density estimation with bayesian normalising flows, 2018.

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

Yixin Wang, David Blei, and John P Cunningham. Posterior collapse and latent variable non-identifiability. In *Advances in Neural Information Processing Systems*, 2021.

Zihao Wang and Liu Ziyin. Posterior collapse of a linear latent variable model. *Advances in Neural Information Processing Systems*, 35:37537–37548, 2022.

Yuki Yamasaki, Chiaki Doi, Shiori Kitagawa, Hiroyuki Seki, and Hiroshi Shigeno. Data generation with filtered β-vae for the preoperative prediction of adverse events. *IEEE Access*, 11:48667–48676, 2023.

Yijia Zheng, Tong He, Yixuan Qiu, and David P Wipf. Learning manifold dimensions with conditional variational autoencoders. *Advances in Neural Information Processing Systems*, 35:34709–34721, 2022.

A Related Works

Expressive decoder and approximation gap. We considers Gaussian decoders extending from σ-VAE, wherein the variance is a scaled identity matrix $\Sigma = \sigma^2 I_d$. While many recent applications assume a fixed variance, e.g.,$\sigma^2 = 1$ (Castrejon et al., 2019; Yamasaki et al., 2023; Loizillon et al., 2024), critics have highlighted that this assumption leads to limited generative power and a positive approximation gap. If σ is not fixed, one line of research establishes the tightness of the approximation gap in the asymptotic regime $\sigma \to 0$ (Dai & Wipf, 2018; Zheng et al., 2022). The assumptions regarding $\sigma \to 0$ are problematic. Mattei & Frellsen (2018) shows that this can lead to an unbounded likelihood and demonstrate a zero approximation gap between Gaussian VAE and a submodel of nonparametric mixture models (Lindsay, 1995). However, their results do not extend to conditional variants. To mitigate likelihood unboundedness, Takahashi et al. (2018); Skafte et al. (2019); Stirn & Knowles (2020) employ a conjugate gamma prior on the inverse variance $1/\sigma^2$, transforming the decoder $p_\theta(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})$ into a student-t distribution, though there is no rigorous discussion on the approximation gap. Another independent line of research shows that the generative model under linear VAE assumptions corresponds to probabilistic Principal Component Analysis (pPCA) models (Lucas et al., 2019b; Wang & Ziyin, 2022), which also demonstrates a zero approximation gap. Linear VAE theories are rigorous but do not fully account for the success of VAEs. In this paper, we analyze the approximation gap of σ-CVAE for a non-fixed σ, highlighting its expressiveness as a powerful generative model.

Non-identifiability and robust decoder estimation. Flexible covariance parameterization in the Gaussian decoder faces several challenges. The first challenge is the likelihood blow-up problem, where maximum likelihood estimation is ill-defined as the likelihood function $\log p_\theta(\boldsymbol{y}|\boldsymbol{x})$ becomes infinite or unbounded when the covariance approaches zero. For instance, the multilayer perceptrons (MLP) parameterization of the decoder (Kingma & Welling, 2014) results in unbounded likelihood functions for continuous responses (Mattei & Frellsen, 2018). Another challenge is model non-identifiability, where different parameters of the model can yield the same likelihood or ELBO, leading to multiple (local) optima. If μ or σ is modeled a function of z, the likelihood function $p_{\theta,\sigma}(y|x)$ becomes non-identifiable due to latent rotation transformations (Khemakhem et al., 2020). Consequently, the common choice of the covariance matrix for the Gaussian decoder is a scaled identity matrix, which balances model complexity and numerical stability. Also, a dual-step optimization (Detlefsen et al., 2019; Rybkin et al., 2021) is favored over joint optimization of the mean and covariance of the decoder (Dai & Wipf, 2018; Takahashi et al., 2018) to enhance numerical stability. Nonetheless, this paper highlights that challenges remain in achieving robust decoder estimation.

Posterior collapse and KL annealing. Posterior collapse is a prevalent issue indicating suboptimal variational inference of VAEs, where the learned encoders has a near-zero KL divergence from the prior, often due to inadequate approximation of the true posterior distribution (He et al., 2019; Lucas et al., 2019b) or, less frequently, from perfect alignment with the posterior in a degraded latent variable model Wang et al. (2021). To address this, traditional KL annealing methods consider re-weighting the loss function with a tuned hyperparameter β (Higgins et al., 2016; Chen et al., 2018; Rezende & Viola, 2018), or with a predefined monotonic or cyclical annealing schedule that involves cross-validation and intensive computation (Raiko et al., 2007; Bowman et al., 2016; Fu et al., 2019). We argue that calibrating σ on σ-CVAE is equivalent to KL annealing due to the loss equivalence between σ and β, noted in Lucas et al. (2019a); Dai et al. (2021); Rybkin et al. (2021). Furthermore, we propose an implicit σ-calibration approach that dynamically adjusts based on clear evidence of posterior collapse, leading to improved encoder estimations and reduced computational costs.

B Discussion on Theorem 2.4

The approximation theories of this work is also motivated by explaining the expressiveness of the parameterized density $p_{\theta,\sigma}(y|x)$ when σ is not infinitesimal. Here we show a simple extended result from Dai & Wipf (2018).

Assumption B.1 (**Response continuity**). Conditional on any $\boldsymbol{x} \in \mathcal{X}$, there exists a ground truth probability measure of μ_{gt} such that its Radon-Nikodym derivative with respect to the standard Lebesgue measure

is nonzero almost everywhere in $\mathcal{Y}$ given $\forall \boldsymbol{x} \in \mathcal{X}$. Simply put, $p_{gt}(\boldsymbol{y}|\boldsymbol{x})$ exists uniquely and is non-zero almost everywhere up to a null set.

Proposition B.2 (Global optimal σ-CVAE). *Under Assumption B.1, if the latent dimension d is larger than or equal to the dimension of the response q, then for any $\sigma > 0$, there exists a σ-CVAE with encoder network parameterized by θ_σ, and decoder network parameterized by ϕ_σ such that,*

$$\lim_{\sigma \to 0} \mathbf{KL}[p_{\theta_\sigma}(y|x) \| p_{gt}(y|x)] = 0, \lim_{\sigma \to 0} \mathbf{KL}[q_{\phi_\sigma}(z|y,x) \| p_{\theta_\sigma}(z|y,x)] = 0. \tag{19}$$

Inspired by Dai & Wipf (2018), the proof argument involves two steps; first, we show the convergence of the parameterized density to the distribution of the ground truth measure; then we show that the ELBO is asymptotically tight as $\sigma \to 0$ as the KL divergence between the approximate posterior and the ground truth posterior vanishes. In particular, we do not assume that $\boldsymbol{y}$ follows a low-dimensional simple Riemann manifold with dimension less than q, to avoid additional definitions such as active dimension (Zheng et al., 2022). The detailed proof of Proposition B.2 in Appendix

The proposition B.2 states that σ-CVAE asymptotically approximates $p_{gt}(\boldsymbol{y}|\boldsymbol{x})$ and the intractable posterior $p_\theta(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x})$ as $\sigma \to 0$ simultaneously. This proposition establishes the asymptotic expressiveness of the parameterized density $p_{\theta,\sigma}(y|x)$ as $\sigma \to 0$. The global optimality is evident from the fact that the lower bound of Eq.4 is reached. However, this proposition is limited because it does not explain the practical success of experiments in which a nonzero σ is learned. Indeed, Proposition B.2 cannot answer the reason why a *non-zero* σ is learned. It is not clear whether this is due to the limit capacity of the encoder/decoder network or the identifiability of VAEs.

C Proof of proposition B.2

Following the line of proof of theorem 2 in Dai & Wipf (2018), we first consider the case when latent dimension d equals response dimension q

Step 1: Show that $\mathrm{KL}[p_{\theta^*_\sigma}(\boldsymbol{y}|\boldsymbol{x})||p_{gt}(\boldsymbol{y}|\boldsymbol{x})] \to 0$ as $\sigma \to 0$.

Define the function $F : \mathbb{R}^q \to [0,1]^q$ given x as follows,

$$F_x(\boldsymbol{y}) = [F_1(y_1), F_2(y_2; y_1)...., F_q(y_q; y_1, ..., y_{q-1})]^T, \tag{20}$$

$$F_i(y_i; y_1, ..., y_{i-1}) = \int_{y_i^{'}=-\infty}^{y_i} p_{gt}(y_i^{'}|x, y_1, ..., y_{i-1}) dy_i^{'}, \tag{21}$$

where $p_{gt}(y_i^{'}|x, y_1, ..., y_{i-1})$ is the distribution of the i-th dimension of $Y|X$ condition on the first $i-1$ dimensions.

By definition we have $dF_x(\boldsymbol{y}) = p_{gt}(\boldsymbol{y}|\boldsymbol{x})d\boldsymbol{y}$. And since p_{gt} is nonzero everywhere, the conditional distribution function F_x is invertible. Denote its inverse by F_x^{-1}.

Similarly, we can define another differential and invertible function $T : \mathbb{R}^d \to [0,1]^d$

$$T(\boldsymbol{z}) = [\Phi(z_1), \Phi(z_1), ...\Phi(z_d)]^T, \tag{22}$$

where $\Phi()$ is the cumulative density function of the standard Gaussian.

Since $d = q$, we consider possibly non-identifiable decorder parameter $\theta^* \in \{\theta | \mu_\theta(x, z) = F_x^{-1} \circ T(\boldsymbol{z}))\}$, then the corresponding density

$$\begin{aligned} p_\theta(\boldsymbol{y}|\boldsymbol{x}) &= \int_{\mathbb{R}^d} \mathcal{N}(\boldsymbol{y}|\mu_\theta(\boldsymbol{x}, \boldsymbol{z}), \sigma I_d) p(\boldsymbol{z}) d\boldsymbol{z} \\ &= \int_{[0,1]^q} \mathcal{N}(\boldsymbol{y}|F_{\boldsymbol{x}}^{-1}(u), \sigma I_q) du \\ &= \int_{\mathbb{R}^q} \mathcal{N}(\boldsymbol{y}|\boldsymbol{y}', \sigma I_q) p_{gt}(\boldsymbol{y}'|\boldsymbol{x}) d\boldsymbol{y}' \to \int_{\mathbb{R}^q} \delta(\boldsymbol{y} - \boldsymbol{y}') p_{gt}(\boldsymbol{y}'|\boldsymbol{x}) d\boldsymbol{y}' = p_{gt}(\boldsymbol{y}|\boldsymbol{x}), \end{aligned} \tag{23}$$

which uses the fact that p.d.f of $N(\boldsymbol{y}|\boldsymbol{y}', \sigma I_q) \to \delta(\boldsymbol{y}, \boldsymbol{y}')$ as $\sigma \to 0$.

It follows immediately that $\mathrm{KL}[p_\theta(\boldsymbol{y}|\boldsymbol{x})||p_{gt}(\boldsymbol{y}|\boldsymbol{x})] \to 0$.

Step 2: Show $\forall \theta^*_\sigma$, $\exists \phi^*_\sigma$ s.t. $\frac{q_{\phi^*_\sigma}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})}{p_{\theta^*_\sigma}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})} \to const.$

let weights of encoder network ϕ are such that

$$\begin{aligned} \mu_\phi(\boldsymbol{x}, \boldsymbol{y}) &:= T^{-1} \circ F_{\boldsymbol{x}}(\boldsymbol{y}), \\ \Sigma_\phi(\boldsymbol{x}, \boldsymbol{y}) &:= \sigma(S_{\theta,\phi}(\boldsymbol{x}, \boldsymbol{y})^\top S_{\theta,\phi}(\boldsymbol{x}, \boldsymbol{y}))^{-1}, \end{aligned} \tag{24}$$

where $d \times q$ Jacobian matrix $S_{\theta,\phi}(\boldsymbol{x}, \boldsymbol{y}) := \nabla_{\boldsymbol{z}} \mu_\theta(\boldsymbol{x}, \boldsymbol{z})|_{\boldsymbol{z}=\mu_\phi(\boldsymbol{x},\boldsymbol{y})}$.

Denote $\mu_\phi(\boldsymbol{x}, \boldsymbol{y}) = \mu_\phi, \Sigma_\phi(\boldsymbol{x}, \boldsymbol{y}) = \sigma\tilde{\Sigma}_\phi$,

$$p_\theta(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x}) = \frac{p_\theta(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})p(\boldsymbol{z})}{p_\theta(\boldsymbol{y}|\boldsymbol{x})} = \frac{\mathcal{N}(\boldsymbol{y}; \mu_\theta(\boldsymbol{x}, \boldsymbol{z}), \sigma I_m)\mathcal{N}(\boldsymbol{z}; 0, I)}{p_\theta(\boldsymbol{y}|\boldsymbol{x})} \tag{25}$$

let $\boldsymbol{z}' = (\boldsymbol{z} - \mu_\phi)/\sqrt{\sigma}$ and $q'_\phi(\boldsymbol{z}'|\boldsymbol{y}, \boldsymbol{x})$, $p'_\theta(\boldsymbol{z}'|\boldsymbol{y}, \boldsymbol{x})$ be the transformed pdf, then we have

$$\begin{aligned} &\frac{q'_\phi(\boldsymbol{z}'|\boldsymbol{y}, \boldsymbol{x})}{p'_\theta(\boldsymbol{z}'|\boldsymbol{y}, \boldsymbol{x})} = \frac{\mathcal{N}(\sqrt{\sigma}\boldsymbol{z}' + \mu_\phi; \mu_\phi, \sigma\tilde{\Sigma}_\phi)p_\theta(\boldsymbol{y}|\boldsymbol{x})}{\mathcal{N}(\boldsymbol{y}; \mu_\theta(\boldsymbol{x}, \sqrt{\sigma}\boldsymbol{z}' + \mu_\phi), \sigma I_m)\mathcal{N}(\sqrt{\sigma}\boldsymbol{z}' + \mu_\phi; 0, I)} \\ &= C(\boldsymbol{x}, \boldsymbol{y}) \exp\{-\frac{\boldsymbol{z}'^\top \tilde{\Sigma}_\phi^{-1} \boldsymbol{z}'}{2} + \frac{\|\mu_\phi + \sqrt{\sigma}\boldsymbol{z}'\|_2^2}{2} + \frac{\|\boldsymbol{y} - \mu_\theta\left(\boldsymbol{x}, \mu_\phi + \sqrt{\sigma}\boldsymbol{z}'\right)\|_2^2}{2\sigma}\} \\ &= C(\boldsymbol{x}, \boldsymbol{y}) \exp\{-\frac{\boldsymbol{z}'^\top \tilde{\Sigma}_\phi^{-1} \boldsymbol{z}'}{2} + \frac{\|\mu_\phi + \sqrt{\sigma}\boldsymbol{z}'\|_2^2}{2} + \frac{\|\mu_\theta(\boldsymbol{x}, \mu_\phi) - \mu_\theta(\boldsymbol{x}, \mu_\phi + \sqrt{\sigma}\boldsymbol{z}')\|_2^2}{2\sigma}\} \\ &= C(\boldsymbol{x}, \boldsymbol{y}) \exp\{-\frac{\boldsymbol{z}'^\top \tilde{\Sigma}_\phi^{-1} \boldsymbol{z}'}{2} + \frac{\|\mu_\phi + \sqrt{\sigma}\boldsymbol{z}'\|_2^2}{2} + \frac{\|\nabla_{\boldsymbol{z}} \mu_\theta(\boldsymbol{x}, \mu_\phi)\sqrt{\sigma}\boldsymbol{z}'\|_2^2}{2\sigma}\} \text{(Taylor Expansion)} \\ &= C(\boldsymbol{x}, \boldsymbol{y}) \exp\{-\frac{\boldsymbol{z}'^\top \tilde{\Sigma}_\phi^{-1} \boldsymbol{z}'}{2} + \frac{\|\mu_\phi + \sqrt{\sigma}\boldsymbol{z}'\|_2^2}{2} + \frac{\boldsymbol{z}'^\top \nabla_{\boldsymbol{z}} \mu_\theta(\boldsymbol{x}, \mu_\phi)^\top \nabla_{\boldsymbol{z}} \mu_\theta(\boldsymbol{x}, \mu_\phi)\boldsymbol{z}')}{2}\} \\ &= C(\boldsymbol{x}, \boldsymbol{y}) \exp\{\frac{\|\mu_\phi + \sqrt{\sigma}\boldsymbol{z}'\|_2^2}{2}\} \to C(\boldsymbol{x}, \boldsymbol{y}) \exp\{\frac{\|\mu_\phi\|_2^2}{2}\} = 1. \end{aligned} \tag{26}$$

Therefore, without the transformation of $\boldsymbol{z}$, $\lim_{\sigma \to 0} \mathrm{KL}[q_{\phi^*_\sigma}(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x})||p_{\theta^*_\sigma}(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x})] = 0$

For the case where the latent dimension d is larger than the response dimension q, we use the first q latent dimensions to build a projection between $\boldsymbol{z}_{1:q}$ and $\boldsymbol{y}|\boldsymbol{x}$ without the remaining $d - q$ latent dimensions. To be specific, let $\mu_\theta(\boldsymbol{x}, \boldsymbol{z}) := F_x^{-1} \circ T(\boldsymbol{z}_{1:q})$, and using the same derivation of Eq.23, we get

$$\mathrm{KL}[p_\theta(\boldsymbol{y}|\boldsymbol{x})||p_{gt}(\boldsymbol{y}|\boldsymbol{x})] \to 0.$$

Now, we define the mean function of the encoder where $\mu_\phi(\boldsymbol{y}|\boldsymbol{x})_{1:q} = T^{-1} \circ F_{\boldsymbol{x}}(y)$ and $\mu_\phi(\boldsymbol{y}|\boldsymbol{x})_{q+1:d} = 0$ and the variance function of the encoder as follows,

$$\Sigma_\phi(\boldsymbol{x}, \boldsymbol{y}) = \sigma[(S_{\theta,\phi}(\boldsymbol{x}, \boldsymbol{y}), \boldsymbol{n}_{q+1}, ..., \boldsymbol{n}_d)^T (S_{\theta,\phi}(\boldsymbol{x}, \boldsymbol{y}), \boldsymbol{n}_{q+1}, ..., \boldsymbol{n}_d)]^{-1},$$

where $\boldsymbol{n}^d_{i=q+1}$ are a set of q-dimensional column vectors, e.g. orthonormal basis of null$(S_{\theta,\phi})$, such that

$$\begin{aligned} S^T_{\theta,\phi} \boldsymbol{n}_i &= 0, \\ \boldsymbol{n}_i^T \boldsymbol{n}_j &= \mathbf{1}_{i=j}. \end{aligned} \tag{27}$$

The set of $\boldsymbol{n}^d_{i=q+1}$ always exists due to the fact that $S_{\theta,\phi}$ is the $d \times q$ Jacobian matrix with null space at least $d - q$.

This implies,

$$\Sigma_\phi(\boldsymbol{x},\boldsymbol{y}) = \begin{bmatrix} \sigma(S_{\theta,\phi}(\boldsymbol{x},\boldsymbol{y})^\top S_{\theta,\phi}(\boldsymbol{x},\boldsymbol{y}))^{-1} & \mathbf{0} \\ \mathbf{0} & I_{d-q} \end{bmatrix}. \tag{28}$$

Thus, The first q dimensions of $q_\phi(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})$ can exactly match the first q dimensions of true posterior as shown in Eq.26. The remaining $d-q$ dimensions follow a standardized Gaussian distribution that matches the posterior of $p_{\theta^*_\sigma}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})$ due to the fact that these dimensions are not used in likelihood involving $\mu_\theta(\boldsymbol{x},\boldsymbol{z}) := F_x^{-1}\circ T(\boldsymbol{z}_{1:q})$ as it remains to follow the prior distribution. As a result, $\lim_{\sigma\to 0}\mathrm{KL}[q_{\phi^*_\sigma}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})||p_{\theta^*_\sigma}(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{x})] = 0$.

D Proof of theorem 2.4

We consider G^*_m as follows,

$$G^*_m = f^\eta_0 + f^\eta_{\boldsymbol{x}}(\boldsymbol{x}) + f^\eta_{\boldsymbol{z}_1}(\boldsymbol{z}_1) + f^\eta_{\boldsymbol{z}_2}(\boldsymbol{z}_2) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_1}(\boldsymbol{x},\boldsymbol{z}_1) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_2}(\boldsymbol{x},\boldsymbol{z}_2) + f^\eta_{\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{z}_1,\boldsymbol{z}_2) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2) \tag{29}$$

Step 1: we prove, under assumption 2.3, that there exists a unique decomposition of G^*_m in Eq.**??** up to a rotation of coordinates within the block. Specifically, given G^*_m, there is a unique form for $f^\eta_0, f^\eta_{\boldsymbol{x}}(\boldsymbol{x}), ..., f^\eta_{\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{z}_1,\boldsymbol{z}_2)$, and $f^\eta_{\boldsymbol{x}\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)$.

Following the line of proof of Theorem 1 Soboĺ (1993), with an abuse of notation μ, we can see that $f^\eta_0 = \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{x}d\boldsymbol{z}_1d\boldsymbol{z}_2) = 1$, as G^*_m is a proper probability map.

Then, the integration of $G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)$ over any two block dimensions, namely $\int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{z}_1d\boldsymbol{z}_2)$, admits the following,

$$\begin{aligned}\int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{z}_1d\boldsymbol{z}_2) =& f^\eta_0 + f^\eta_{\boldsymbol{x}}(\boldsymbol{x}) + \cancelto{0}{\int f^\eta_{\boldsymbol{z}_1}(\boldsymbol{z}_1)\mu(d\boldsymbol{z}_1) + f^\eta_{\boldsymbol{z}_2}(\boldsymbol{z}_2)\mu(d\boldsymbol{z}_2)} \\ &+ \cancelto{0}{\int f^\eta_{\boldsymbol{x}\boldsymbol{z}_1}(\boldsymbol{x},\boldsymbol{z}_1)\mu(d\boldsymbol{z}_1) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_2}(\boldsymbol{x},\boldsymbol{z}_2)\mu(d\boldsymbol{z}_2)} \\ &+ \cancelto{0}{\int f^\eta_{\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{z}_1d\boldsymbol{z}_2) + \int f^\eta_{\boldsymbol{x}\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{z}_1d\boldsymbol{z}_2)},\end{aligned} \tag{30}$$

where the cancellation to zeros are direct results of assumption 2.3.

Therefore,

$$\begin{aligned} f^\eta_{\boldsymbol{x}}(\boldsymbol{x}) &:= \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{z}_1d\boldsymbol{z}_2) - 1, \\ f^\eta_{\boldsymbol{z}_1}(\boldsymbol{z}_1) &:= \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{x}d\boldsymbol{z}_2) - 1, \\ f^\eta_{\boldsymbol{z}_2}(\boldsymbol{z}_2) &:= \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{x}d\boldsymbol{z}_1) - 1. \end{aligned} \tag{31}$$

Using the same argument, we can see

$$\begin{aligned} f^\eta_{\boldsymbol{x}\boldsymbol{z}_1}(\boldsymbol{x},\boldsymbol{z}_1) &:= \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{z}_2) - \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{x}d\boldsymbol{z}_2) - \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{z}_1d\boldsymbol{z}_2) + 1, \\ f^\eta_{\boldsymbol{x}\boldsymbol{z}_2}(\boldsymbol{x},\boldsymbol{z}_2) &:= \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{z}_1) - \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{x}d\boldsymbol{z}_1) - \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{z}_1d\boldsymbol{z}_2) + 1, \\ f^\eta_{\boldsymbol{x}\boldsymbol{z}_2}(\boldsymbol{z}_1,\boldsymbol{z}_2) &:= \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{x}) - \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{x}d\boldsymbol{z}_2) - \int G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2)\mu(d\boldsymbol{x}d\boldsymbol{z}_1) + 1, \end{aligned} \tag{32}$$

and lastly the residual

$$f^\eta_{\boldsymbol{x}\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2) := G^*_m(\boldsymbol{x},\boldsymbol{z}_1,\boldsymbol{z}_2) - (f^\eta_{\boldsymbol{x}}(\boldsymbol{x}) + f^\eta_{\boldsymbol{z}_1}(\boldsymbol{z}_1) + f^\eta_{\boldsymbol{z}_2}(\boldsymbol{z}_2) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_1}(\boldsymbol{x},\boldsymbol{z}_1) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_2}(\boldsymbol{x},\boldsymbol{z}_2) + f^\eta_{\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{z}_1,\boldsymbol{z}_2) + 1). \tag{33}$$

This shows that all f^η are unique given G_m^*

Step 2: We prove that with additional assumptions including $f^\eta_{\boldsymbol{z}_2}(\boldsymbol{z}_2) = \sigma^* \boldsymbol{z}_2$, σ-CVAE recovers the ground truth G_m^*.

By comparison of block coordinates, $\mu_\sigma(\boldsymbol{x}, \boldsymbol{z}_1) + \sigma \boldsymbol{z}_2 = G_m^*$ is almost surely true if the following holds for all $\boldsymbol{x}, \boldsymbol{z}_1, \boldsymbol{z}_2$ (up to a zero measure of $\boldsymbol{z}_2$).

$$\begin{aligned} \mu_\theta(\boldsymbol{x}, \boldsymbol{z}_1) &= f^\eta_{\boldsymbol{x}}(\boldsymbol{x}) + f^\eta_{\boldsymbol{z}_1}(\boldsymbol{z}_1) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_1}(\boldsymbol{x}, \boldsymbol{z}_1) + C_1 \\ \sigma \boldsymbol{z}_2 &= f^\eta_{\boldsymbol{z}_2}(\boldsymbol{z}_2) \\ f^\eta_{\boldsymbol{x}\boldsymbol{z}_2}(\boldsymbol{z}_1, \boldsymbol{z}_2) + f^\eta_{\boldsymbol{x}\boldsymbol{z}_2}(\boldsymbol{x}, \boldsymbol{z}_2) + f_{\boldsymbol{x}\boldsymbol{z}_1\boldsymbol{z}_2}(\boldsymbol{x}, \boldsymbol{z}_1, \boldsymbol{z}_2) &= C \end{aligned} \tag{34}$$

Where $C_1 := 1 + C$, an important detail is that in alignment with the identifiability assumption 2.3, $f^\eta_{\boldsymbol{z}_2}(\boldsymbol{z}_2)$ cannot have any additive constant. Specifically, $\int f^\eta_{\boldsymbol{z}_2}(\boldsymbol{z}_2)\mu(\boldsymbol{z}_2) = \int \sigma \boldsymbol{z}_2 \mu(\boldsymbol{z}_2) = 0$. Assuming the universal approximation theorem of $\mu_\theta(\boldsymbol{x}, \boldsymbol{z})$, the results are immediately implied by considering the additional assumption.

The conditions and proof universal approximation property can be found in Hornik et al. (1989)

E Experiment details and Algorithm

We implemented the proposed method using the Pytorch 1.8.2 +cu111 with Python 3.7 on a Ubuntu internal cluster with multiple Nvidia GPUs including A10,A30, A100, A100-40GB, A100-80GB, V100. We are not aware of which GPU is used in the experiments due to the task distribution service.

In Section 4.1, we used fully connected 4-layer neural networks with a hyperbolic tangent activation function for the encoding and decoding network. The latent dimension is set to 2 and the width of the hidden layer is $[16, 8, 4, 2]$ and $[2, 4, 16, 4]$, respectively. σ initialized at 1. The batch size is equal to the sample size of the training data. We generate 5000 data points with $2,500$ for each class with a latent variable generated from a standard normal distribution.

In Section 4.2, we used 3 simulations datasets from the following:

M_1 is a non-linear model with additive Gaussian noise:

$\boldsymbol{y} = \boldsymbol{x}_1^2 + \exp(\boldsymbol{x}_2 + \boldsymbol{x}_3/3) + \sin(\boldsymbol{x}_4 + \boldsymbol{x}_5) + \varepsilon$, where $\varepsilon \sim N(0, 1)$.

M_2 is A nonlinear model with multiplicative Non-Gaussian noise:

$\boldsymbol{y} = \left(5 + \boldsymbol{x}_1^2/3 + \boldsymbol{x}_2^2 + \boldsymbol{x}_3^2 + \boldsymbol{x}_4 + \boldsymbol{x}_5\right) * \exp(0.5 \times \varepsilon)$,where, $\varepsilon \sim 0.5N(-2, 1) + 0.5N(2, 1)$.

M_3. A Gaussian Mixture Model:

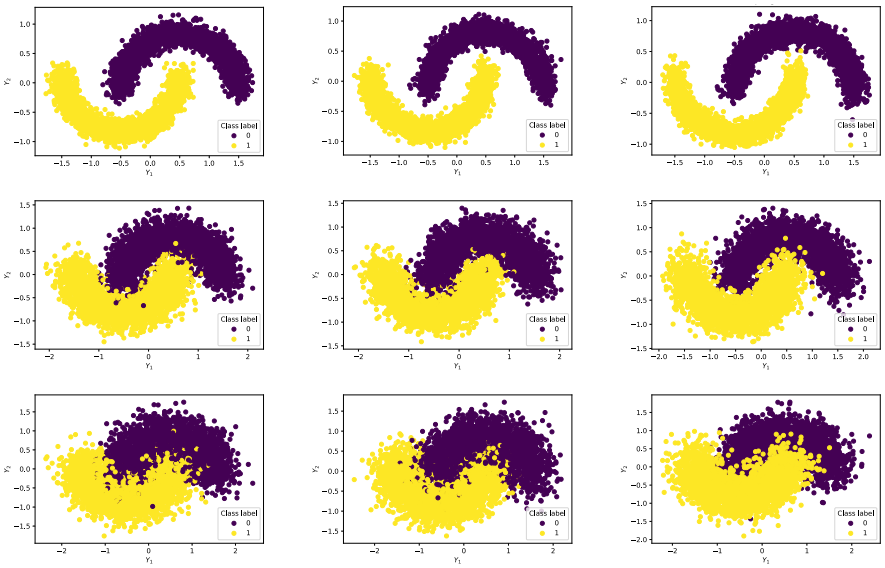
$$\boldsymbol{y} \sim \begin{cases} N(-1 - \boldsymbol{x}_1 - 0.5\boldsymbol{x}_2, 0.5^2), & \text{if } U = 0, \\ N(1 + \boldsymbol{x}_1 + 0.5\boldsymbol{x}_2, 1^2), & \text{if } U = 1, \end{cases} \text{ where } U \sim \text{Binomial}(1, 0.7).$$

The predictor dimension is 5 for M_1 and M_2, and 2 for M_3. The response $\boldsymbol{y}$ is univariate. The sample size $N = 5000$, the test sample size is $k = 2000$. The mean squared error of the mean is $\frac{1}{k}\sum_{i=1}^k \left(\hat{\mathbb{E}}[\boldsymbol{y}|\boldsymbol{x}_k] - \mathbb{E}[\boldsymbol{y}|\boldsymbol{x}_k]\right)^2$. The mean squared error of the standard deviation is $\frac{1}{k}\sum_{i=1}^k \left(\hat{\mathrm{SD}}[\boldsymbol{y}|\boldsymbol{x}_k] - \mathrm{SD}[\boldsymbol{y}|\boldsymbol{x}_k]\right)^2$.

We used fully connected 5-layer neural networks with a hyperbolic tangent activation function for the encoding and decoding network. The latent dimension is set to 5. The width of the hidden layer of the network is $[32, 16, 8, 4]$, and $[8, 32, 16, 4]$. For the WGCS method, the conditional generator G is parameterized using a fully connected neural network. The discriminator D is parameterized using a fully connected two-layer neural network. The noise vector is $\eta \sim N(0, 1)$. For the NNKCDE method, the tuning parameters are chosen by cross-validation. The bandwidth of CKDE is determined based on the standard formula $h = 1.06\sigma n^{-1/(2t+d)}$, where σ is a measure of spread, t is the order of the kernel and d is the dimension of X. The FlexCode basis expansion-based method uses the Fourier basis. The maximum number of bases is set to 40 and the actual number of bases is selected using cross-validation. WGCS generates $10,000$ Monte

Carlo samples to estimate the conditional distribution of each test x_k to calculate the condition mean and conditional standard deviation, while our method uses only 500 samples. For other methods, the estimates are calculated by numerical integration.

F More discussion for experiments in Section 4.1



(a) (b) (c)

Figure 4: Calibrated σ-CVAE on two moon dataset: (a) the training data of size 5000, (b) estimated training data, and (c) sample drawn from calibrated σ-CVAE. $\tau = 0.1, 0.2$ and 0.3 from top to bottom. Best viewed in color.

Additional experiment of learned σ-CVAE using Two moon dataset with $\tau = 0.1$. The ground truth σ recovers τ, which is 0.1.

Calibration in reducing the inner steps. In Table 4, we report the average wall time for convergence for different K epoch for updating θ, ϕ, measured by python function time.perf_counter(). We report the largest wall time out of 5 repetitions of successful experiments with estimated σ, initialized at 1, converged to true value 0.1. Tolerance $C = 0.05$. Same criterion for convergence used in each K. NA represents no correct estimate the true value σ within 3% relative error. We find that success rate of experiments for uncalibrated σ-CVAEs increases with K and due to calibration in the sufficiently large K, our proposed algorithm might be slower than a uncalibrated one.

Table 4: Comparison of wall time for convergence

	K=25	50	100	250	500
Uncalibrated	NA	NA	62.15	76.10	177.81
Calibrated	10.55	12.69	25.00	33.72	269.78

Calibration as warm restart. Nonetheless, the calibration step in our framework is beneficial for the estimation of σ-CVAE even if *optimal* σ is learned. As shown by Lucas et al. (2019b), posterior collapse in q_ϕ occurs even if optimal σ is learned, possibly due to local optima in q_ϕ. Therefore, when calibrating σ, the training dynamics of the parameter ϕ enjoys the rapid change of σ, perhaps due to a smoother loss landscape with larger σ Dai et al. (2021), take advantage to escape the local optima and converge to a possibly better local optima at the time when σ returns to optimally small again. Even if we calibrate the σ back to initialization, the optimization that was done before this calibration would be beneficial as a warm restart. To this extent, our calibration step is an inexpensive and implicit KL annealing scheme without a predefined

monotonic or cyclic schedule Skafte et al. (2019); Stirn & Knowles (2020). Therefore, we recommend that this calibration step should be used even if a sufficiently large K is used. Please note that warm restart is not guarantee the convergence of algorithm in the non-convex ELBO optimization, throughout the experiments, we observe cold restart is necessary for instances with very bad initialization.

Comparison with existing KL annealing methods. The explicit KL annealing is undoubtedly easy and simple to implement. However, it is less efficient and often requires more computational power in terms of preventing posterior collapse. For example, if σ is assumed to be fixed, an explicit KL annealing schedule like Eq.16 is essentially a series of deterministic σ calibrations in Eq.17, due to σ-β equivalence. Therefore, the annealing process of loss landscape is predefined and not adaptive to the performance of encoders. Similarly, when σ is a learned parameter, an explicit KL annealing schedule confounds with it in shaping the loss landscape, similarly to how it would by a single "effective" σ_e. For example, when $\beta = 0.25$ adding to LHS of Eq.5, the "effective" σ_e is half of current σ value, i.e. $\sigma_e = 0.5 * \sigma$. The confounding between β and σ will not vanish unless β is 1. In extreme cases, if the predefined $\beta_t = 1/(2\sigma_t^2)$, the effective σ_e would be fixed at 1. This means that there would be no KL annealing to the loss landscape at all because there would be no change in the effective σ_e.

Such confounding not only complicates the annealing process of loss landscape determined by the effective σ_e, but also hinders the accurate and efficient estimation of σ under maximum likelihood principle.

To provide more solid evidence, we trained Gaussian σ-CVAE with a learned σ using a monotonic or a cyclical annealing schedule tuning β linearly from 0 to 1. For cyclical annealing, we use a cycle period $M = 10$ of 5 maximum cycles . For monotonic anneal, we consider a single monotonic annealing with same length $M * 5 = 50$. The β becomes a fixed value of 1 after the maximum 50 iteration reached untill the algorithm converges. The hyper-parameters of KL annealing are chosen reasonably for a fair comparison.

We report the MSE of learned σ, KLD at convergence under the same setting of Figure 3 average of 20 repetitions. Training Loss is not reported because of the training loss inconsistency that is caused by using different β.

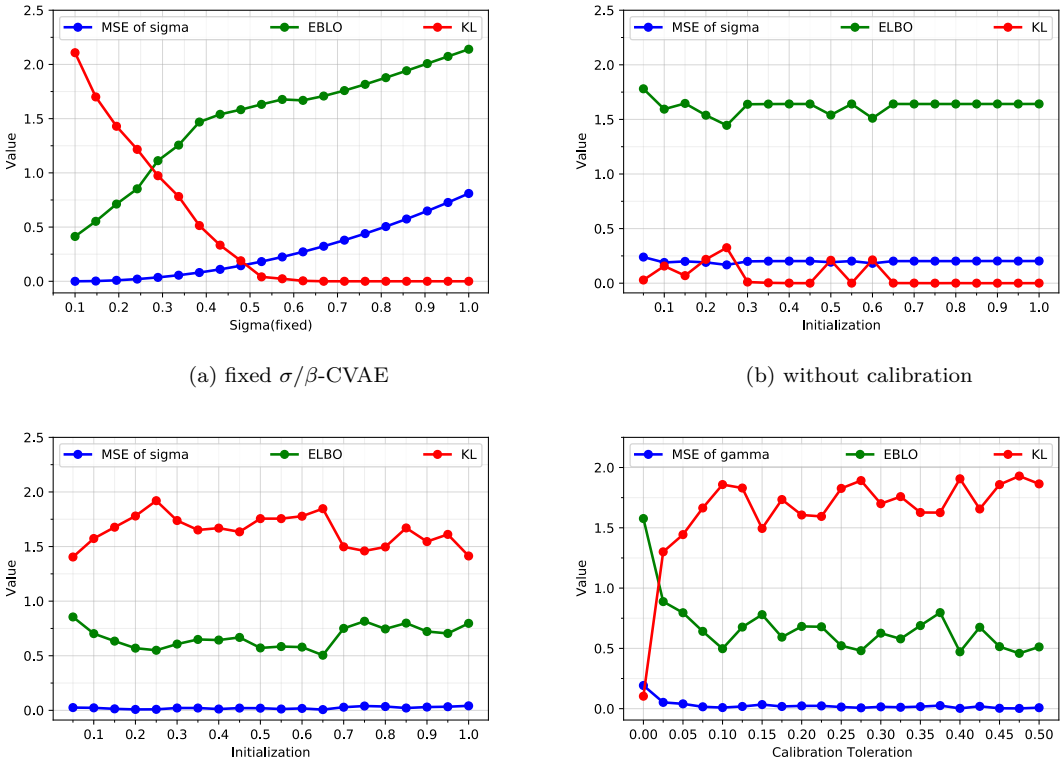
σ Initialized	0.2		0.4		0.6		0.8		1	
	MSE	KLD	MSE	KLD	MSE	KLD	MSE	KLD	MSE	KLD
Monotonic	1.62e-02	1.81	6.09e-03	2.10	5.65e-03	2.19	3.41e-04	2.12	6.06e-04	2.17
Cyclical	1.76e-02	1.77	3.24e-02	1.71	1.92e-02	1.84	2.17e-02	2.01	4.58e-02	1.45
Ours C=0.05	2.41e-02	1.79	2.71e-02	1.64	6.01e-03	1.92	5.38e-02	1.39	2.69e-02	1.70
Ours C=1.96	1.26e-05	2.09	5.77e-05	2.09	1.22e-05	2.08	1.73e-05	2.10	1.76e-05	2.09
Vanilla	0.16	4.17e-01	0.20	1.40e-03	0.17	3.16e-01	0.20	2.20e-04	0.20	5.39e-04

Table 5: Comparison with KL annealing method. Monotonic refers to monotonic annealing (Bowman et al., 2016), and cyclical refers cylical annealing. (Fu et al., 2019). C is Calibration tolerance. We use $K = 25$.

We also report the maximum wall time till convergence under the same setting of Table 4 using $K = 25, 50, 100$ recorded by python function *time.perf_counter()* out of 5 successfuly repetitions.

	K=25	K=50	K=100
Monotonic	24.85	36.26	58.38
Cyclical	10.76	23.01	45.59
Calibrated C=0.05	10.55	12.69	25.00
Vanilla	NA	NA	62.15

Table 6: Comparison with KL annealing method. Monotonic refers to monotonic annealing (Bowman et al., 2016), and cyclical refers cylical annealing (Fu et al., 2019). C is Calibration tolerance



(a) fixed σ/β-CVAE

(b) without calibration

(c) with calibration

(d) ablation

Figure 5: Experiments on twomoon dataset with ground truth $\sigma = 0.1$. Results of the CVAE models are reported using (a) a fixed σ, equivalent to β-CVAE with a predefined β (b) a learned σ without calibration with tolerance 0.05, (c) a learned sigma with calibration, and (d) a learned sigma with difference toleration for calibration which is intialized at 1. K is set to 25. Each data point is averaged over 20 repetitions. MSE of sigma is calculated by the squared error between the learned/fixed sigma and the ground truth.

G Ablation experiments on the effect of calibration

In this section, more experiments are performed to highlight the importance of calibration using the two-moon dataset in Section 4.1, a nontrivial dataset simulated from a nonlinear generating function that causes CVAE models to have reproducible posterior collapse problem. Throughout this section, we use $\sigma = 0.1$ to generate our training dataset once and repeat our experiment 20 times. We use small $K = 25$.

To reiterate the importance of learning σ in CVAEs models, we report the results of those using fixed σ ranging from 0.1 to 1 in Figure 5(a). CVAEs are well specified if the predefined fixed σ is equal and close to the ground truth value. The performance of CVAEs degenerates rapidly as the difference between the fixed σ and the ground truth increases. After the predefined σ is larger than a certain point, the posterior collapse in CVAEs is exhibited by vanishing KL divergence between the approximate posterior and prior. Therefore, we should not use CVAEs with a predefined variance scalar σ when the ground truth σ in the data is unapproachable.

To emphasize the numerical instability in the training dynamics of σ and the posterior collapse of the learned model, we report the results of CVAEs that iteratively update σ without calibration in Figure 5(b) with *various initializations of σ ranging from 0.05 to 1.* The learned CVAE model ends up with a very high ELBO, fails to recover the ground truth σ, and the KL divergence between the approximate posterior and prior vanishes in most cases regardless of the initialization of σ. In Figure 1, we can see that σ is trapped around 0.5 if no calibration is provided. It is known that the non-covex landscape of the ELBO function

w.r.t. θ, ϕ leads to many local optima, but our experiment specifically reveals the difficulty in estimating σ in a dual-step algorithm.

To provide more empirical evidence of our proposed method, we report the results of our calibrated method in Figure 5(c) for different initializations of σ ranging from 0.05 to 1. The tolerance that triggers our proposed calibration step is set to 0.05. The results showed the consistency of our proposed method in accurately estimating the truth of the ground σ, preventing posterior collapse, and thus obtaining locally optimal CVAE models.

To ablate the calibration effect, we report the results of our proposed method with various tolerance hyperparameters C ranging from 0.00 to 0.5 in Figure 5(d). We initialized σ at 1. In the case that tolerance is 0, our proposed calibration step will never be triggered due to the non-negativity of KL divergence. We observe that a small tolerance of 0.05 is empirically effective in ameliorating posterior collapse and robust estimation of σ. Note that the tolerance setting is case-to-case and may not be generalized to other datasets, and a prodigious tolerance, which goes beyond the possible value of KL divergence, leads to non-convergence in the algorithm. Therefore, we recommend a relatively small tolerance hyperparameter that triggers fewer calibrations and offers less extra computational burden.

H Image reconstruction: MNIST

MNIST dataset LeCun et al. (1998) contains $60,000$ gray images with size 28×28. Dividing each image into two parts $\boldsymbol{x}, \boldsymbol{y}$, we treat one part of the image as predictor $\boldsymbol{x}$ and the rest of the image as response $\boldsymbol{y}$. In Appendix H, we use fully connected 3-layer neural networks with a ReLU activation function for the encoding and decoding network. The latent dimension is set to 5. The dimension of X is 196, 392, and 588, and the corresponding dimension of Y is 588,392, and 196. Two fully connected hidden layers are 256 and 128 for the encoder, and we reverse the width in the decoders. The batch size is 100. We update and calibrate σ per 100 iterations with a posterior collapse tolerance set to 0.001. In Figure 6, we consider three situations in which 1/4, 1/2, and 3/4 of the image are observed, and the goal is to learn the distribution of the rest of the image. We chose 10 images from the test dataset to evaluate our method.

We see that the generated images are similar to the truth with reasonable variations, and the variations in reconstruction decrease as the larger part of the image is observed.

I Image generation: CelebA

The CelebA dataset Liu et al. (2015) contains more than $200,000$ colored celebrity face images with 40 facial attribute annotations. In this section, we used the architect of the CVAE models from Hou et al. (2019) and embedded the attribute label of a vector in the images channels. If a label in one dimension is 1, we embed it as an image channel of size 96×96 that takes a value of 1 on every pixel. The width of the hidden layers is set to $[32, 64, 128, 256, 512]$ and the convolution layer with a kernel size of 3 , a stride size of 2, and a padding size of 1. We used LeakyReLU activation functions at a negative slope of 0.02 followed by a batch normalization layer. The number of latent dimensions is set to 32. The batch size is 16. we update and calibrate each gamma per 1809 iterations, which is a factor of total training sample size and posterior collapse tolerance set to 0.05. We apply the proposed method to the generation of human face images with binary label features as a predictor. Since some of the attributes are highly correlated, we validate our proposed framework using the following attributes *Male, Young, Eyeglasses, Bald, Mustache, Smiling* to be the predictor $\boldsymbol{x}$, and the response $\boldsymbol{y}$ here is the center-chopped scaled images with a size of 96×96. We show the six types of true images and the generated images in each row of Figure 7. The attribute labels for these three types are shown in Table 7. Our generated images preserve the face attributes as input with moderate clarity.

Limitations Although our simple calibration σ-CVAE method does not provide a state-of-the-art image generation framework, we hypothesize that the conditional distribution of the face image may not satisfy the assumption 2.3. Expanding the analysis in consideration of modern variants/techniques that are specific for (conditional) imagine generation is important subsequent work. Further efforts are needed to make it

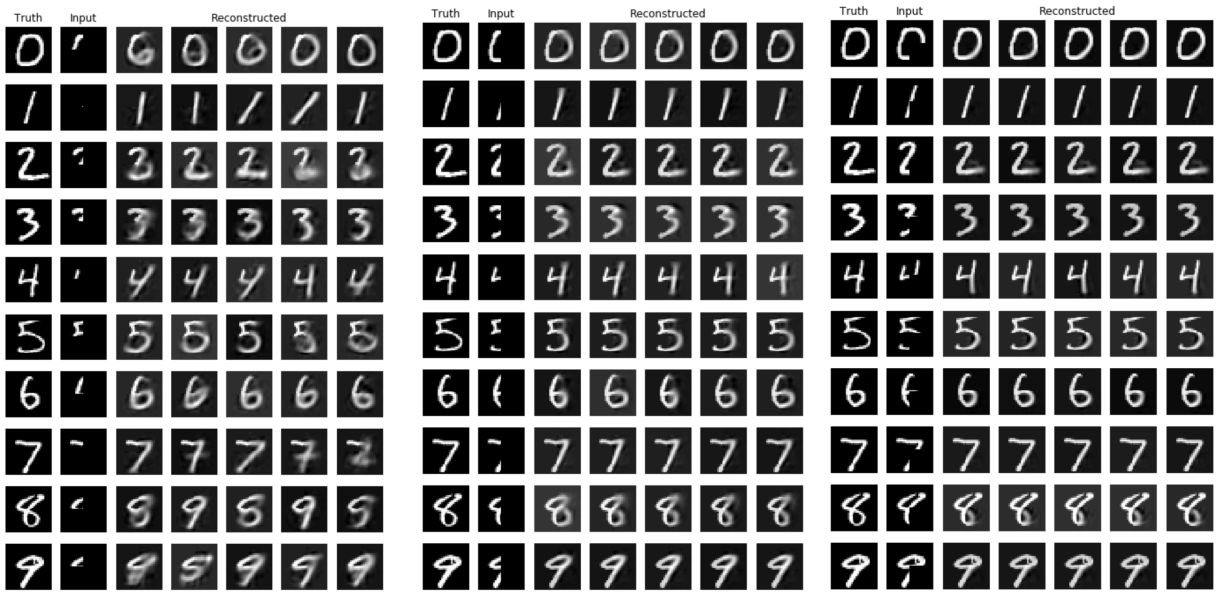


Figure 6: Reconstructing images in the MNIST test dataset. The left column of each panel contains the true images from the test dataset that are not used for training. The second left column contains the given part of the image that are used as predictor (i.e. upper left 1/4 in (a), left half 1/2 in (b), and upper left half 3/4 in (c)), and the rest 5 columns are the reconstructed images.

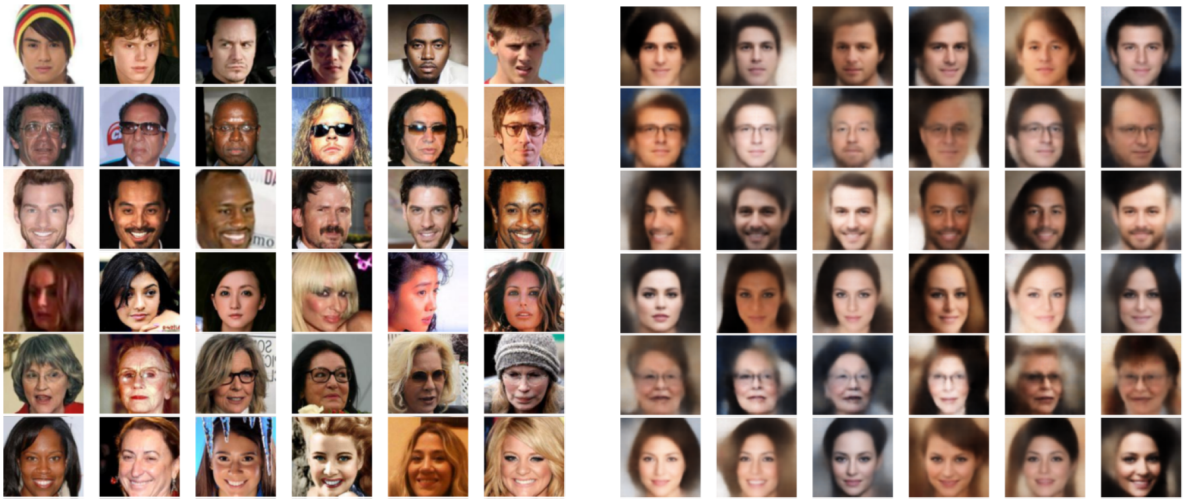


Figure 7: (a) The true images in CelebA. Images of the same row have the same attributes. (b) Generated (not reconstructed) images with the same attributes. Each row corresponds to a specific type of face identical to the same row of (a)

a state-of-the-art (C)VAE variants for conditional image generator. One common challenge in conditional image generation is meaningful quantitative measures to generated samples. Metrics such as test likelihood or MSE failed in measuring the clarity of generated face images, and the ones like FID or Inception Score add extra mathematical assumptions that are difficult to validate.

Table 7: Attributes for six types of face images in Figure

	Male	Young	Eyeglasses	Bald	Mustache	Smile
type 1	+1	+1	-1	-1	-1	-1
type 2	+1	-1	+1	-1	-1	-1
type 3	+1	+1	-1	-1	+1	+1
type 4	-1	+1	-1	-1	-1	-1
type 5	-1	-1	+1	-1	-1	-1
type 6	-1	+1	-1	-1	-1	+1