VAE APPROXIMATION ERROR: ELBO AND EXPONENTIAL FAMILIES

Anonymous authors
Paper under double-blind review

ABSTRACT

The importance of Variational Autoencoders reaches far beyond standalone generative models — the approach is also used for learning latent representations and can be generalized to semi-supervised learning. This requires a thorough analysis of their commonly known shortcomings: posterior collapse and approximation errors. This paper analyzes VAE approximation errors caused by the combination of the ELBO objective and encoder models from conditional exponential families, including, but not limited to, commonly used conditionally independent discrete and continuous models. We characterize subclasses of generative models consistent with these encoder families. We show that the ELBO optimizer is pulled away from the likelihood optimizer towards the consistent subset and study this effect experimentally. Importantly, this subset can not be enlarged, and the respective error cannot be decreased, by considering deeper encoder/decoder networks.

1 INTRODUCTION

Variational autoencoders proposed by Kingma & Welling (2014) strive at learning complex data distributions $p_d(x)$, $x \in \mathcal{X}$ in a generative way. They introduce latent variables $z \in \mathcal{Z}$ and model the joint distribution as $p_\theta(x \mid z)p(z)$, where $p(z)$ is a simple distribution that is usually assumed to be known. The conditional distribution $p_\theta(x \mid z)$, called *decoder*, is modeled in terms of a deep network parametrized by $\theta \in \Theta$. Models defined in this way allow to sample from $p_\theta(x) = \mathbb{E}_{p(z)} p_\theta(x \mid z)$ easily, however at the price that computing the *posterior* $p_\theta(z \mid x) = p_\theta(x \mid z)p(z)/p_\theta(x)$ is usually intractable. To handle this problem, VAE approximates the posterior $p_\theta(z \mid x)$ by an amortized inference *encoder* model $q_\phi(z \mid x)$ parametrized by $\phi \in \Phi$. Given the empirical data distribution $p_d(x)$, the model is learned by maximizing the evidence lower bound (ELBO) of the data log-likelihood $L(\theta) = \mathbb{E}_{p_d} \log p_\theta(x)$. It can be expressed in the following two equivalent forms:

$$L_B(\theta, \phi) = \mathbb{E}_{p_d}\big[\mathbb{E}_{q_\phi} \log p_\theta(x \mid z) - D_{\mathrm{KL}}(q_\phi(z \mid x) \,\|\, p(z))\big] \tag{1a}$$

$$= L(\theta) - \mathbb{E}_{p_d}\big[D_{\mathrm{KL}}(q_\phi(z \mid x) \,\|\, p_\theta(z \mid x))\big]. \tag{1b}$$

The first form allows for stochastic optimization of ELBO while the second form shows that the gap between log-likelihood and ELBO is exactly the mismatch between the encoder and the posterior.

VAEs constitute a powerful deep learning extension of the expectation maximization approach to handle latent variables. They are useful not only as generative models but also, *e.g.*, in semi-supervised learning (Kingma et al., 2014; Mattei & Frellsen, 2019). Furthermore the encoder part constructs an efficient embedding of the data in the latent space, useful in many applications. The outreach of the VAE approach requires therefore a careful empirical and theoretical analysis of the problems and trade offs involved. The most important ones are (i) posterior collapse (He et al., 2019; Lucas et al., 2019; Dai et al., 2018; Dai & Wipf, 2019) and (ii) approximation errors caused by an inappropriate choice of the encoder family.

The VAE approximation error has been studied (*e.g.*, Cremer et al. 2018; Hjelm et al. 2016; Kim et al. 2018) so far mainly empirically. The problem also occurs and is well-recognized in the context of variational inference and variational Bayesian inference, where the target posterior distribution is expected to be complex. It is commonly understood, that the mean field approximation of $p_\theta(z \mid x)$ by $q_\phi(z \mid x)$ in (1b) significantly limits variational Bayesian inference. In contrast, in VAEs, the decoder may adopt to compensate for the chosen encoder family. The effect of this coupling, we

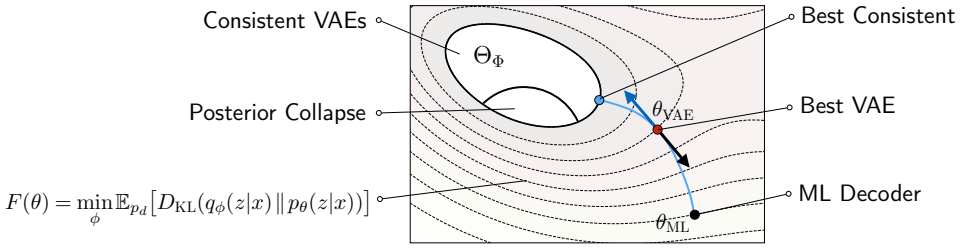


Figure 1: Diagram of the VAE trade-off. The optimal solution θ_{VAE} is "in between" the maximum likelihood solution θ_{ML} and the best solution in the class Θ_Φ of consistent VAEs, where the posterior approximation error function $F(\theta)$ vanishes. We give an explicit characterization of this consistent set. At θ_{VAE} there is a balance between the gradient of $-F$ (blue arrow) and the gradient of the data log-likelihood (black arrow).

believe, is not fully understood. The phenomenon of decoder adopting to the posterior was experimentally observed, *e.g.*, by Cremer et al. (2018, Section 5.4), noting that the approximation error is often dominated by the amortization error. Turner & Sahani (2011, Sec. 1.4) analytically show for linear state space models that simpler variational approximations (such a mean-field) can lead to less bias in parameter estimation than more complicated structured approximations. Similarly, Shu et al. (2018) view the VAE objective as providing a regularization and show that making the amortized inference model smoother, while increasing the amortization gap, leads to a better generalization.

The common (empirical) understanding of the importance of the gap between the approximate and the true posterior has led to many generalizations of standard VAEs, which achieve impressive practical results, notably, tighter bounds using importance weighting (Burda et al., 2016; Nowozin, 2018), encoders employing normalizing flows (Rezende & Mohamed, 2015; Kingma et al., 2016), hierarchical and autoregressive encoders (Vahdat & Kautz, 2020; Sønderby et al., 2016; Ranganath et al., 2016), MRF encoders (Vahdat et al., 2020) and more. While these extensions mitigate the posterior mismatch problem, they often come at a price of a more difficult training and more expensive inference. Furthermore, simpler encoders may be of practical interest. Burda et al. (2016, Appendix C) illustrates that IWAE approximate posteriors are less regular and more spread out. In contrast, factorized encoders provide simple embeddings useful for downstream tasks such as semantic hashing (Chaidaroon & Fang, 2017).

The aim of this paper is to study the approximation error of VAEs and its impact on the learned decoder. We consider a setting that generalizes many common VAEs, in particular popular models where encoder and decoder are conditionally independent Bernoulli or Gaussian distributions: we assume that both decoder and encoder are conditional exponential families. We identify the subclass of generative models where the encoder can model the posterior exactly, referred to as *consistent* VAEs. We give a characterization of consistent VAEs revealing that this set in fact does not depend on the complexity of the involved neural networks. We further show that the ELBO optimizer is pulled towards this set away from the likelihood optimizer. Specializing the characterization to several common VAE models, we show that the respective consistent models turn out to be RBM-like in many cases. We experimentally investigate the detrimental effect in one case and show that a simpler but more consistent VAE can perform better in the other.

2 Problem Statement

We adopt the following notion of approximation error. Consider a generative model class $\mathcal{P}_\Theta = \{p_\theta(x, z) \mid \theta \in \Theta\}$, the encoder class $\mathcal{Q}_\Phi = \{q_\phi(z \,|\, x) \mid \phi \in \Phi\}$ and the data distribution $p_d(x)$. The maximum likelihood generative model is given by $\theta_{\mathrm{ML}} \in \operatorname{argmax}_{\theta\in\Theta} \mathbb{E}_{p_d(x)} \log p_\theta(x)$. For a decoder with parameters θ we define its *approximation error* as the likelihood difference $L(\theta_{\mathrm{ML}}) - L(\theta)$. Respectively, the *VAE approximation error* is defined for a given θ as:

$$L(\theta_{\mathrm{ML}}) - \max_\phi L_B(\theta, \phi) \geq L(\theta_{\mathrm{ML}}) - L(\theta). \tag{2}$$

In order for this error to become zero, two conditions are necessary and sufficient:

- Parameters (θ, ϕ) must be optimal for the ELBO objective.

- ELBO must be *tight* at (θ, ϕ), *i.e.*, $L_B(\theta, \phi) = L(\theta)$.

Assuming that the optimality can be achieved, we study the non-tightness gap $L(\theta) - L_B(\theta, \phi)$. From (1b) it expresses as $\mathbb{E}_{p_d}\big[D_{\mathrm{KL}}(q_\phi(z \,|\, x) \,\|\, p_\theta(z \,|\, x))\big]$. It follows that ELBO is tight at (θ, ϕ) iff $q_\phi(z \,|\, x) \equiv p_\theta(z \,|\, x)$. Hence, we define the *consistent set* $\Theta_\Phi \subseteq \Theta$ as the subset of distributions $p_\theta(x, z)$ whose posteriors are in $\mathcal{Q}_\Phi$, *i.e.*,

$$\Theta_\Phi = \big\{\theta \in \Theta \;\big|\; \exists \phi \in \Phi : q_\phi(z \,|\, x) \equiv p_\theta(z \,|\, x)\big\}. \tag{3}$$

The KL-divergence in the ELBO objective (1b) can vanish only if $\theta \in \Theta_\Phi$. If the likelihood maximizer θ_{ML} is not contained in Θ_Φ, then this KL-divergence pulls the optimizer towards Θ_Φ and away from θ_{ML} as illustrated in Fig. 1.

We characterize the consistent set Θ_Φ, on which the bound is tight, and show that this set is quite narrow and does not depend on the complexity of the encoder and decoder networks beyond simple 1-layer linear mappings of sufficient statistics.

3 Theoretical Analysis

We consider a general class of VAEs, where both encoder and decoder are defined as exponential families. This class includes many common models, in particular Gaussian VAEs and Bernoulli VAEs with conditional independence assumptions, but also more complex ones, *e.g.*, where the encoder is a conditional random field (Vahdat et al., 2020)[1].

Assumption 1 (Exponential family VAE)**.** *Let $\mathcal{X}$ and $\mathcal{Z}$ be sets of observations and latent variables, respectively. We consider VAE models defined by*

$$p_\theta(x \,|\, z) = h(x) \exp\big[\langle \nu(x), f_\theta(z)\rangle - A(f_\theta(z))\big] \tag{4a}$$

$$q_\phi(z \,|\, x) = h'(z) \exp\big[\langle \psi(z), g_\phi(x)\rangle - B(g_\phi(x))\big], \tag{4b}$$

where $\nu\colon \mathcal{X} \to \mathbb{R}^n$ and $\psi\colon \mathcal{Z} \to \mathbb{R}^m$ are fixed sufficient statistics of dimensionality n and m; $f_\theta\colon \mathcal{Z} \to \mathbb{R}^n$ and $g_\phi\colon \mathcal{X} \to \mathbb{R}^m$ are the the decoder, resp., encoder networks, depending on learnable parameters θ, resp. ϕ; $h\colon \mathcal{X} \to \mathbb{R}_+$, $h'\colon \mathcal{Z} \to \mathbb{R}_+$ are strictly positive base measures and A, B denote the respective log-partition functions.

Notice that this assumption imposes no restrictions on the nature of random variables x and z. They can be discrete or continuous, univariate or multivariate. Similarly, it imposes no restrictions on the complexity of the decoder and encoder networks $f_\theta(z)$ and $g_\phi(x)$.

Characterization of the consistent set. In the first step of our analysis, we investigate the conditions under which the approximation error of an exponential family VAE can be made exactly zero. As discussed above, a tight VAE (θ, ϕ) must satisfy $\forall (x, z) \;\; q_\phi(z \,|\, x) = p_\theta(z \,|\, x)$, which leads to the following theorem.

Theorem 1. *The consistent set Θ_Φ of an exponential family VAE is given by decoders of the form*

$$p(x \,|\, z) = h(x) \exp\big[\langle \nu(x), W\psi(z)\rangle + \langle \nu(x), u\rangle - A(z)\big], \tag{5}$$

where W is a $n \times m$ matrix and $u \in \mathbb{R}^n$. Moreover, the corresponding encoders have the form

$$q(z \,|\, x) = h'(z) \exp\big[\big\langle \psi(z), W^T \nu(x)\big\rangle + \langle \psi(z), v\rangle - B(x)\big], \tag{6}$$

where $v \in \mathbb{R}^m$.

This is a direct consequence of a theorem by Arnold & Strauss (1991) (see Appendix A.1 for more details). For a tight VAE, Theorem 1 states that the decoder and encoder take the form (5) and (6) with the interaction between x and z parametrized by a single matrix W instead of the complex neural networks with parameters θ, ϕ. Thus they turn out to be *generalized linear models* (GLMs). The corresponding joint probability distribution is an EF Harmonium (Welling et al., 2005):

$$p(x, z) = h(x) h'(z) \exp\big(\langle \nu(x), W\psi(z)\rangle + \langle \nu(x), u\rangle + \langle \psi(z), v\rangle - A\big). \tag{7}$$

[1]Notice, however, that this class does not include VAEs with advanced encoder families like normalizing flows, hierarchical and autoregressive encoders.

Corollary 1. *The subset Θ_Φ of consistent models can not be enlarged by considering more complex encoder networks $g(x)$, provided that the affine family $W^\mathsf{T}\nu(x)$ can already be represented.*

Corollary 2. *Let the decoder network family be affine in $\psi(z)$,* i.e.*, $f(z) = W\psi(z) + a$ and let the encoder network family $g(x)$ include at least all affine maps $V\nu(x) + b$. Then any global optimum of ELBO attains a zero approximation error.*

VAEs can escape consistency when they degenerate to an invertible flow. In practice, VAE models are almost never tight. It is therefore natural to ask, whether a small VAE posterior mismatch error implies closeness of the optimal decoder to some decoder in the consistent set.

Definition 1. *A VAE (p_θ, q_ϕ) is ε-*tight *for some $\varepsilon > 0$ if $\mathbb{E}_{p_d(x)}[D_{\mathrm{KL}}(q_\phi(z\,|\,x) \,\|\, p_\theta(z\,|\,x))] \leq \varepsilon$.*

It turns out that the the KL divergence has a leak, which allows a VAE to approach tightness while not approaching consistency. In the continuous case an example satisfying ε-tightness with non-linear decoder follows from Dai & Wipf (2019, Theorem 2). They show for a class of Gaussian VAEs with general neural networks f_θ, g_ϕ, that it is possible to build a sequence of network parameters θ_t, ϕ_t with the following properties: i) the target distribution is approximated arbitrary well, ii) the posterior mismatch $D_{\mathrm{KL}}(q_{\phi_t}(z\,|\,x) \,\|\, p_{\theta_t}(z\,|\,x))$ approaches zero and iii) both the encoder and decoder approach deterministic mappings. The VAE thus approaches a flow model (or invertible neural network) between the data manifold and a subspace of the latent space (Dai & Wipf, 2019). Clearly, in a general case the flow must be non-linear. A similar case can be made for discrete variables, see Example A.1.

Non-deterministic nearly-tight VAEs approach consistency. We would however argue that the mode where the decoder and encoder are nearly-deterministic is not a natural VAE solution. By making additional assumptions, excluding such deterministic solutions, and restricting ourselves to the finite space in order to simplify the analysis, we can show that the decoder of an ε-tight VAE does indeed approach a GLM.

Theorem 2. *Let (p_θ, q_ϕ) be an exponential family VAE (Assumption 1) on a discrete space $\mathcal{X} \times \mathcal{Z}$ with strictly positive encoder $q_\phi(z\,|\,x)$ and decoder posterior $p_\theta(z\,|\,x)$, both bounded from below by $\alpha > 0$. If the VAE is ε-tight, then there exists $W \in \mathbb{R}^{n,m}$ such that the decoder can be approximated by a GLM of the form $\tilde{p}(x\,|\,z) = \tilde{h}(x)\exp\big[\langle \nu(x), W\psi(z)\rangle - \tilde{A}(z)\big]$ with the error*

$$\mathbb{E}_{p_d(x)}\Big[\big(\log p_\theta(x\,|\,z) - \log \tilde{p}(x\,|\,z)\big)^2\Big] \leq \tfrac{\varepsilon}{2\alpha^2} + o(\varepsilon) \quad \forall z \in \mathcal{Z}. \tag{8}$$

The proof is given in Appendix A.3.

3.1 Cases Analysis

This subsection gives a detailed analysis of consistent VAE models in several concrete cases of practical interest.

Diagonal Gaussian VAE Let us consider a Gaussian VAE, as commonly applied to image generation (*e.g.*, Dai & Wipf (2019)). Let $\mathcal{X} = \mathbb{R}^n$, $\mathcal{Z} = \mathbb{R}^m$, $p(x\,|\,z) = \mathcal{N}(x\,|\,\mu_d(z), \sigma_d^2 I)$, $q(z\,|\,x) = \mathcal{N}(z\,|\,\mu_e(x), \mathrm{diag}(\sigma_e^2(x)))$, where μ_d, μ_e and σ_e are neural networks and σ_d is a common pixel observation noise parameter. The decoder has minimal sufficient statistics $\nu(x) = x$ and the base measure $h(x) = \mathcal{N}(x\,|\,0, \sigma_d^2 I)$. The encoder has minimal sufficient statistics $\psi(z) = (z, z^2)$, where the square is coordinate-wise. Theorem 1 implies that a tight optimal VAE has the joint model

$$p(x, z) \propto h(x)\exp\big[\big\langle x, Wz + Vz^2 + a\big\rangle + \langle b, z\rangle + \big\langle c, z^2\big\rangle\big] \tag{9}$$

for some matrices W, V and vectors a, b, c. Furthermore, the integral over z must be finite for all x and therefore $x^\mathsf{T}V + c < 0$ must hold for all $x \in \mathbb{R}^n$. This is possible only if $V = 0$ and $c < 0$. The joint distribution is therefore a multivariate Gaussian and the same holds for its marginal $p(x)$. The neural network $\mu_d(z)$ must degenerate to $\mu_d(z) = \sigma_d^2 \cdot (Wz + a)$ and the two neural networks for the encoder to $\sigma_e^2(x) = -1/2c$ and $\mu_e(x) = -(W^\mathsf{T}x + b)/2c$, where divisions are coordinate-wise. VAEs with such simplified, linear Gaussian encoder-decoder pairs, called "linear VAEs" (Lucas et al., 2019) are known to be consistent and to match the probabilistic PCA model (Dai et al., 2018; Lucas et al., 2019). In this context, our Corollary 2 is a generalization of (Lucas et al.,

2019, Lemma 1) showing consistency of linear VAEs, to decoders in any exponential family with natural parameters being a linear mapping of any fixed lifted latent representation $\psi(z)$.

We argue that a joint Gaussian model is too simplistic to generate complex data such as realistic images and that in this case the VAE error is detrimental. In Section 4.2 we experimentally confirm that optimizing ELBO for a general decoder network μ_d causes qualitative and quantitative degradation relative to the ML decoder. Note that if we allowed σ_d to be dependent on z, the resulting joint statistics in $\nu \otimes \psi$ would include terms $x^2 z$, $x^2 z^2$. The joint distribution would not be Gaussian and may be in fact multi-modal (see Anil Bhattacharayya's distribution in Arnold et al. 2001).

Bernoulli-MRF VAE Vahdat et al. (2020) proposed to consider encoders in the Markov Random Field (MRF) family, in particular encoders of the form $q(z|x) \propto \exp(\langle z^1, V(x) z^2\rangle + \langle b^1(x), z^1\rangle + \langle b^2(x), z_2\rangle)$, where z^1, z^2 are two groups of latent variables and interaction weights V, b^1, b^2 are computed by the encoder network. In this case $q(z|x)$ is itself a (conditional) RBM. While evaluating $q(z|x)$ is difficult, MCMC sampling is efficient. We assume binary observations x and a conditionally independent Bernoulli decoder family as above. The decoder thus has sufficient statistics $\nu_e = (x)$ and the encoder has $\psi = (z^1, z^2, z^1 \otimes z^2)$. Introducing homogeneous constant components $x_0 = z_0^1 = z_0^2 = 1$, the family of consistent joint distributions can be compactly described as

$$p(x, z) = \exp\big[\textstyle\sum_{i,j,k} W_{i,j,k}\, x_i\, z_j^1\, z_k^2\big], \tag{10}$$

where the summation in all indices starts from 0 and $-W_{0,0,0}$ is the log-partition function. This joint model is a higher order MRF with the highest order potentials given by cubic monomials.

Standard Bernoulli VAEs are a special case of the Bernoulli-MRF model, obtained when the interaction weights V are zero. The joint distribution of such tight optimal VAEs takes the form

$$p(x, z) = \tfrac{1}{c} \exp(x^\mathsf{T} W z + u^\mathsf{T} x + v^\mathsf{T} z), \tag{11}$$

which is a restricted Boltzmann machine (RBM). Since RBMs are well known for being useful in many applications (dimensionality reduction, collaborative filtering, feature learning, topic modeling), we hypothesize that they can make a good baseline for Bernoulli VAEs and furthermore that the effect of pulling the VAE solution towards an RBM may be benign in case of insufficient data. For example IWAE test likelihood in (Burda et al., 2016) is worse than that of an RBM (Burda et al., 2015) on the Omniglot dataset. Further debiasing of the IWAE may not improve test likelihood (Nowozin, 2018).

Bernoulli VAE for Semantic Hashing One important application of Bernoulli VAEs is the *semantic hashing* problem, initially proposed and modeled with RBMs (Salakhutdinov & Hinton, 2009). The problem is to assign to each document / image a compact binary latent code that can be used for quick retrieval by the nearest neighbor search. We will detail now a more recent VAE model for text documents (Chaidaroon & Fang, 2017; Shen et al., 2018) and show that it can be tight only in a full posterior collapse. We correct the encoder so as to allow a larger consistent set and observe that the resulting consistent joint distribution forms a multinomial-Bernoulli RBM.

Let $x \in \mathbb{N}^K$ be word counts in a document with words from a dictionary of size K. Let $z \in \{0, 1\}^m$ be a binary latent code. Let $l = \sum_k x_k$ denote the document's length. We assume that the document length is independent of the latent topic and its distribution $p(l)$ can be learned separately (*e.g.*, a log-normal distribution is a good fit). The decoder is defined using the multinomial distribution model (words in the document are drawn from the same categorical distribution corresponding to its topic):

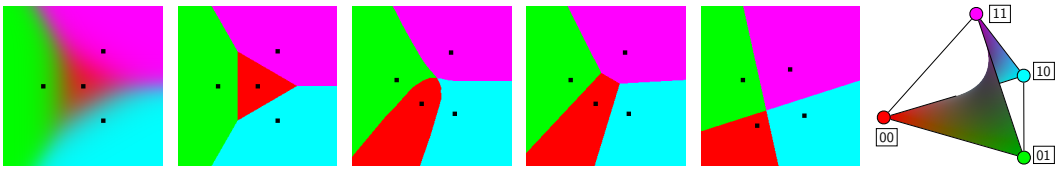
$$p(x, l \,|\, z) = p(l) h(x \,|\, l) \exp(f(z)^\mathsf{T} x - l A(f(z))), \tag{12}$$

where $f(z)$ is a neural network mapping the latent code to the logits of word occurrence probabilities, $A(\eta) = \log \sum_k \exp(\eta_k)$ and $h(x \,|\, l) = [\![\sum_k x_k = l]\!] \big(l! / \prod_k x_k!\big)$ is the base measure[2]. The sufficient statistics are the word counts x. The prior $p(z)$ is assumed uniform Bernoulli.

The encoder is the conditionally independent Bernoulli model, expressed as

$$q(z \,|\, x, l) \propto \exp(g(x)^\mathsf{T} z), \tag{13}$$

[2]Prior work (Chaidaroon & Fang, 2017; Shen et al., 2018) omits the base measure as it has no trainable parameters.



(a) (b) (c) (d) (e) (f)

Figure 2: Artificial example. **(a)** Color-coded posterior distribution $p^*(z \mid x)$. (b-e): Decision maps ($\arg\max_z$) of: **(b)** true posterior $p^*(z \mid x)$, **(c)** factorized encoder $q_\phi(z \mid x)$ after joint learning, **(d)** model posterior $p_\theta(z \mid x)$ after joint learning, **(e)** RBM trained on the same data. Gaussian centers $\mu(z)$ are shown as black dots. **(f)** Probability simplex of distributions over the four binary configurations. The vertices correspond to pure (deterministic) binary states represented by the code and its respective color. The surface shows the manifold of factorized distributions realizable by $q(z \mid x)$. Notice that the two edges $(00, 11)$ and $(01, 10)$ are not in the manifold because they correspond to switching of two bits simultaneously in a correlated way. The factorized approximation cannot model transitions between these states. Hence, when learning VAE, these pairs of states are repulsed in the decision maps (c), (d).

where $g(x)$ is the encoder network. Chaidaroon & Fang (2017) experimented with the encoder and decoder design and recommended using TFIDF features instead of raw counts. First, we note that the inverse document frequency (IDF) is not relevant, since it can be learned by the first linear transform in the encoder. Effectively, the term frequency (TF), given by x/l, is used. This choice is adopted in later works (Shen et al., 2018; Zamani Dadaneh et al., 2020; Ñanculef et al., 2020). It might seem reasonable that the latent code modeling the document topic should not depend on the document length, only on the distribution of words in the document. However, we will argue that this rationale is misleading for stochastic encoders.

We apply Theorem 1 to two groups of variables: observed (x, l) and latent z with $h(x, l) = h(x \mid l)p(l)$, $\nu(x, l) = x$ and $\psi(z) = z$. It follows that the consistent joint family is

$$p(x, l, z) = h(x, l) \exp(x^\mathsf{T} W z + a^\mathsf{T} x + b^\mathsf{T} z + c). \tag{14}$$

This however implies that $g(x) = Wx + b$, *i.e.* the encoder network must be linear in x. Consequently, it cannot match a function of word frequencies x/l (as chosen by design) unless $W = 0$, *i.e.* a completely trivial model with an encoder not depending on x. Such an encoder would imply full posterior collapse. The corresponding consistent set Θ_Φ coincides with the set of collapsed VAEs where the decoder does not depend on the latent variable z in Fig. 1. We conjecture that the inherent inconsistency of this VAE has a detrimental effect on learning.

If instead, we let the encoder network to access word counts x directly, we obtain that $g(x) = Wx + b$ can form a consistent VAE. Inspecting this encoder model in more detail, we see that it builds up topic confidence in proportion to the evidence (total word counts), as the true posterior would. Indeed, the true posterior $p(z \mid x, l)$ satisfies the factorization by Bayes's theorem: $p(z \mid x, l) = p(x \mid z, l)p(z)/p(x \mid l)$. The prior $p(z)$ is constant by design, $p(x \mid l)$ does not vary with z and $p(x \mid z, l)$ factors over all word instances according to (12). In other words, the coupling between x and z in $\log p(z \mid x, l)$ is linear in x.

In Section 4 we study the proposed correction experimentally and show that it enables learning better models under a variety of settings.

4 Experiments

4.1 Artificial Example

To start with, we illustrate our findings on a toy example. We consider a simple Gaussian mixture model for which we can easily generate samples and compute all necessary quantities including the ELBO objective. We define the ground truth model to be $p^*(x, z) = p^*(z)p^*(x \mid z)$, with $z \in \{1 \dots 4\}$, $p^*(z) \equiv 0.25$, $x \in \mathbb{R}^2$, $p^*(x \mid z) = \mathcal{N}(x \mid \mu(z), \sigma^2 I)$, *i.e.* a mixture of four 2D Gaussians. Fig. 2(a) shows the color-coded posterior distribution $p^*(z \mid x)$. We assign a color to each component and represent $p^*(z \mid x)$ for each pixel $x \in \mathbb{R}^2$ by the corresponding mixture of the component colors.

For better interpretability, we illustrate further results by decision maps $\arg\max_z p(z \mid x)$. Fig. 2(b) shows the decision map for $p^*(z \mid x)$.

The aim of the experiment is to learn a VAE and to study the influence of the factorization assumption on the results. We use the decoder architecture as in the ground truth model — a Gaussian distribution $p_\theta(x \mid z) = \mathcal{N}(x \mid \theta z_{oh}, \sigma^2 I)$, where z_{oh} is the one-hot (categorical) representation of z, and θ is a $2{\times}4$ matrix that maps the four latent codes to 2D centers at general locations. Note that the ground truth model is contained in the chosen decoder family. Hence, the ML solution is the ground truth decoder $p^*(x \mid z)$. We restrict the encoder to factor over the binary representation of the code $z_b \in \{0,1\}^2$ and define it as $q_\phi(z_b \mid x) \propto \exp\langle g_\phi(x), z_b\rangle$, where $g_\phi(x)$ is implemented as a feed-forward network with two hidden layers, each with 64 units and ReLU activations.

First, we pre-train our factorized encoder by optimizing ELBO and keeping the ground truth decoder fixed. The next step is to jointly train the encoder and decoder by maximizing ELBO. Since we are interested how the ELBO objective distorts the likelihood solution, we start with the ground truth decoder and the pre-trained encoder from the previous step. The ELBO-optimal decoder has to match not only the training data, but also the inexact, factorizing encoder. The resulting $q_\phi(z \mid x)$ is shown in Fig. 2(c) and the learned model posterior $p_\theta(z \mid x) \propto p(z)p_\theta(x \mid z)$ in Fig. 2(d). Note that they match each other pretty well, but differ substantially from the ground truth posterior shown in Fig. 2(b). The impact of the factorization is clearly visible – one can see two decision boundaries (one for each bit of z_b), which together partition the x-space into four regions, approximating the true posterior. For comparison, Fig. 2(e) shows the posterior of an RBM trained on the same data. It is clearly seen that the ELBO optimizer is pulled away from the likelihood optimizer towards an RBM solution. Notice also the explanation given in Fig. 2(f).

The approximation error (2) in the decoder caused by using the factorized encoder is only 0.03 nats (see Appendix B.1), but the qualitative difference in Fig. 2 appears detrimental. Summarizing, this simple toy example clearly shows the VAE approximation error caused by the combination of ELBO objective and the factorization assumption for the encoder.

4.2 Gaussian VAEs for CelebA Images

The goal and design of this experiment is similar to the previous one. We first define a ground truth decoder which is used to generate training images. Then we pre-train an encoder by ELBO keeping the ground truth decoder fixed. Finally, we train both model parts starting from the ground truth decoder and the pre-trained encoder.

The ground truth generative model is obtained by training a convolutional Generative Adversarial network (GAN) using code of Inkawhich (2017) on the CelebA dataset Liu et al. (2015). We scale and crop all images to $64{\times}64$ pixels. In order to get a stochastic decoder, we equip the GAN generator $x = d(z)$, $z \in \mathbb{R}^{100}$, $x \in \mathbb{R}^{64\times64\times3}$ with *image noise* σ_d. The ground truth generative model is thus defined as $p^*(x,z) = p^*(z)p^*(x \mid z)$, where $p^*(z) = \mathcal{N}(z \mid 0, I)$, $p^*(x \mid z) = \mathcal{N}(x \mid \mu_d(z), \sigma_d^2 I)$, and σ_d^2 is a common noise variance for all pixels and color channels. We chose $\sigma_d = 0.05$ (the color values are normalized to $[-1,1]$). This corresponds to an image noise level, which is just visible, but does not disturb visual perception essentially.

The decoder family of the considered VAE consists of networks with the same architecture as $d(z)$. This ensures that the ground truth decoder is a likelihood maximizer of the VAE model. The encoder is defined as $q_\phi(z \mid x) = \mathcal{N}(x \mid \mu_e(x), \mathrm{diag}(\sigma_e^2(x)))$, where $\mu_e, \sigma_e \in \mathbb{R}^{100}$ are two outputs of a convolutional neural network with an architecture similar to the architecture of the discriminator used for training the GAN (except the output layer), *i.e.*, $q_\phi(z \mid x)$ is a multivariate Gaussian with diagonal covariance matrix whose parameters depend on x.

We pre-train the encoder fully supervised by maximizing its conditional log-likelihood $\mathbb{E}_{p^*(x,z)} \log q_\phi(z \mid x)$ on examples drawn from the ground truth generating model $p^*(x,z)$.[3] The results of pre-training are shown in Fig. 3. Then we jointly learn the encoder and decoder by maximizing ELBO on x-samples drawn from the ground truth model. We start the learning with the ground truth decoder $p^*(x \mid z)$ and the encoder obtained in the previous step. Two variants are considered for this training: (i) keeping the image noise σ_d fixed and (ii) learning it along with other

[3] In contrast to the previous experiment we optimize the "forward" KL-divergence, i.e. the conditional likelihood, instead of the "reverse" KL-divergence used in ELBO for simplicity.

Figure 3: Results for the encoder learned by supervised conditional likelihood. Top row: training samples $\hat{x} \sim p^*(x)$. Second row: the corresponding reconstructions from mean values of z, i.e. $x \sim p^*(x \mid \mu_e(\hat{x}))$. Third row: reconstructions from sampled z, i.e. $\hat{z} \sim \mathcal{N}(\mu_e(\hat{x}), \sigma_e^2(\hat{x}))$ followed by $x \sim p^*(x \mid \hat{z})$.

Figure 4: Visual comparison – images drawn from the original/learned models. Each column corresponds to a particular value of $z \sim \mathcal{N}(0, I)$. Top row: the ground truth model, middle row: learned decoder with fixed σ_d, bottom row: decoder with learned σ_d.

model parameters. We evaluate the results quantitatively by computing the Frechet Inception Distances (FID) between the ground truth model $p^*(x \mid z)$ and the obtained decoders $p_\theta(x \mid z)$ using the code of Seitzer (2020). For this we generate 200k images from each model. The obtained values are given in Tab. 1. Fig. 4 shows images generated by the ground truth model and the two learned models.

To conclude, ELBO optimization harms the decoder considerably as clearly seen both from FID-scores and the generated images. Models with higher ELBO values have worse FID-scores and produce less realistic images.

Table 1: Optimizing the ELBO starting from the ML solution degrades the FID-score. The first row corresponds to the pre-trained encoder for the ground truth decoder, its FID-score therefore compares two image sets, both generated by the ground truth model.

Experiment	ELBO	FID
optimize encoder (conditional likelihood)	-364513.78	0.13
optimize encoder and decoder (ELBO, fixed σ_d)	-5898.94	77.10
optimize encoder and decoder (ELBO, learned σ_d)	9035.69	117.87

4.3 Bernoulli VAE for Text Documents

This experiment compares training of the VAE model for semantic hashing discussed in Section 3.1 with and without our proposed correction on the *20Newsgroups* dataset (Lang & Rennie, 2008). We describe the dataset, preprocessing and optimization details in Appendix B.2.

We compare three encoders: **e1**: linear encoder on word counts (the proposed correction), **e2**: deep (2 hidden layers) encoder using word frequencies and **e3**: a linear encoder on frequencies. The encoders are compared across different numbers of latent Bernoulli variables (bits) and different decoder depths. The decoder depth denotes the number of fully connected hidden ReLU layers (0-2). In both the encoder and decoder we use 512 units in hidden layers. The prior work mainly used linear decoders following the ablation study of Shen et al. (2018). Our experiments also suggest that using deep decoders in combination with longer bit-length leads to a significant overfitting. When the decoder is linear, the posterior distribution is tractable and is linear as well, *i.e.*, the VAE model is equivalent to a Multinomial-Bernoulli RBM. We experimentally verify that a linear decoder indeed works better in this case. However, perhaps more surprisingly, we also find out that it works better even for non-linear decoders.

Table 2 show the achieved training and test Negative ELBO (NELBO) values. We observe across all settings that the simple encoder e1 is consistently better than the more complex encoder e2 which in turn is significantly better than the linear encoder on frequencies e3. We conclude that the use of VAEs with deep encoders based on word frequencies (Chaidaroon & Fang, 2017; Shen et al., 2018; Zamani Dadaneh et al., 2020; Ñanculef et al., 2020) is sub-optimal for this dataset. We do not

expect additional regularization techniques proposed in these works to change this conclusion. We also observe that linear *decoders* generalize better under 32 and 64 bits compared to more complex decoders, which suffer from overfitting. This implies that in these cases the best encoder-decoder combination is linear, *i.e*. the basic RBM model.

Table 2: Training and test NELBO values for Text-VAE with different configurations of bits, decoder and encoder. Bold highlights the best encoder choice and underlined bold values are the best decoder-encoder combinations for each bit-length.

TRAINING									
Bits	dhidden=0			dhidden=1			dhidden=2		
	e1	e2	e3	e1	e2	e3	e1	e2	e3
8	**419**	429	439	**321**	390	415	**<u>325</u>**	370	421
16	**382**	398	419	**201**	329	407	**<u>164</u>**	269	413
32	**337**	358	412	**165**	189	403	**<u>132</u>**	159	411
64	**296**	324	407	**171**	189	398	**<u>134</u>**	149	409

TEST									
Bits	dhidden=0			dhidden=1			dhidden=2		
	e1	e2	e3	e1	e2	e3	e1	e2	e3
8	**423**	429	435	**<u>413</u>**	421	424	**418**	423	427
16	**409**	417	421	**<u>404</u>**	422	420	**410**	416	422
32	**<u>396</u>**	413	416	**399**	413	418	**406**	416	421
64	**<u>392</u>**	411	414	**398**	413	417	**406**	417	422

5 CONCLUSIONS

We have analyzed the approximation error of VAEs in a general setting, when both the decoder and encoder are exponential families. This includes commonly used VAE variants as, *e.g*., Gaussian VAEs and Bernoulli VAEs. We have shown that the subset of generative models consistent with the encoder class is quite restricted: we concisely characterize it as bilinear models on the sufficient statistics of both decoder and encoder, *i.e*., RBM-like models. This connection between tight VAEs and RBMs / EF Harmoniums was not previously known. The consistent subset can not be enlarged by using more complex encoder networks as long as encoder's sufficient statistics remain unchanged. In combination with the ELBO objective, this causes an approximation error — the ELBO optimizer is pulled away from the data likelihood optimizer towards this subset. Moreover, we proved theoretically that close-to-tight VAEs have decoders close to GLMs.

We have shown that the error is detrimental when the consistent subset is too restrictive. On the other hand it may result in a useful regularization when it contains more expressive models. In the cases where a lot of data is available and a high quality generative model is of the primary interest, more expressive encoder families are required in addition to large networks. However, in cases when the learned representations are of interest, models with simple encoders can be used such as in the case of the Text VAE model in our example and their performance can be improved by considering consistency, *i.e*., adjusting the encoder and decoder so that the consistent set is large enough.

We restricted this study to exponential families. While in richer models discussed in the introduction, the approximation error still exists, it is made small by design, and it is less relevant and harder to analyze it theoretically. One possible open direction where such analysis would make sense is to consider fully factorized non-exponential cases, *e.g*., Student-t VAEs (Takahashi et al., 2018) or models satisfying hierarchical or partial factorization (Maalø e et al., 2019).

REFERENCES

B. C. Arnold and D. J. Strauss. Bivariate distributions with conditionals in prescribed exponential families. *Journal of the Royal Statistical Society Series B (Methodological)*, 53(2):365–375, 1991.

Barry C. Arnold, Enrique Castillo, and Jose Maria Sarabia. Conditionally specified distributions: An introduction. *Statistical Science*, 16(3):249 – 274, 2001.

Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Accurate and conservative estimates of MRF log-likelihood using reverse annealing. In *AISTATS*, volume 38, pp. 102–110, 09–12 May 2015.

Yuri Burda, Roger B. Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. In *ICLR*, 2016.

Suthee Chaidaroon and Yi Fang. Variational deep semantic hashing for text documents. In *SIGIR Conference on Research and Development in Information Retrieval*, pp. 75–84, 2017.

Chris Cremer, Xuechen Li, and David Duvenaud. Inference suboptimality in variational autoencoders. In *ICML*, volume 80, pp. 1078–1086, 2018.

Bin Dai and David Wipf. Diagnosing and enhancing VAE models. In *ICLR*, 2019.

Bin Dai, Yu Wang, John Aston, Gang Hua, and David Wipf. Connections with robust PCA and the role of emergent sparsity in variational autoencoder models. *Journal of Machine Learning Research*, 19(41):1–42, 2018.

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

Devon Hjelm, Russ R Salakhutdinov, Kyunghyun Cho, Nebojsa Jojic, Vince Calhoun, and Junyoung Chung. Iterative refinement of the approximate posterior for directed belief networks. In *Advances in Neural Information Processing Systems*, volume 29, pp. 4691–4699, 2016.

Nathan Inkawhich. DCGAN tutorial, 2017. URL https://pytorch.org/tutorials/beginner/dcgan_faces_tutorial.html.

Yoon Kim, Sam Wiseman, Andrew Miller, David Sontag, and Alexander Rush. Semi-amortized variational autoencoders. In *ICML*, volume 80, pp. 2678–2687, 2018.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

Diederik P. Kingma, Danilo J. Rezende, Shakir Mohamed, and Max Welling. Semi-supervised learning with deep generative models. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pp. 3581–3589, Cambridge, MA, USA, 2014. MIT Press.

Diederik P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In *NeurIPS*, volume 29, pp. 4743–4751, 2016.

Ken Lang and Jason Rennie. The 20 newsgroups dataset, 2008. http://qwone.com/~jason/20Newsgroups/.

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

James Lucas, George Tucker, Roger B Grosse, and Mohammad Norouzi. Don't blame the ELBO! a linear VAE perspective on posterior collapse. In *NeurIPS*, volume 32, pp. 9408–9418, 2019.

Lars Maalø e, Marco Fraccaro, Valentin Liévin, and Ole Winther. BIVA: A very deep hierarchy of latent variables for generative modeling. In *NeurIPS*, volume 32, 2019.

Pierre-Alexandre Mattei and Jes Frellsen. MIWAE: Deep generative modelling and imputation of incomplete data sets. In *ICML*, volume 97, pp. 4413–4423, 09–15 Jun 2019.

Ricardo Ñanculef, Francisco Alejandro Mena, Antonio Macaluso, Stefano Lodi, and Claudio Sartori. Self-supervised Bernoulli autoencoders for semi-supervised hashing. *CoRR*, abs/2007.08799, 2020.

Sebastian Nowozin. Debiasing evidence approximations: On importance-weighted autoencoders and jackknife variational inference. In *International Conference on Learning Representations*, 2018.

Rajesh Ranganath, Dustin Tran, and David Blei. Hierarchical variational models. In *ICML*, volume 48, pp. 324–333, 20–22 Jun 2016.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *ICML*, volume 37, pp. 1530–1538, 2015.

Ruslan Salakhutdinov and Geoffrey Hinton. Semantic hashing. *Int. J. Approx. Reasoning*, 50(7): 969–978, July 2009.

Maximilian Seitzer. pytorch-fid: FID Score for PyTorch. https://github.com/mseitzer/pytorch-fid, August 2020. Version 0.1.1.

Dinghan Shen, Qinliang Su, Paidamoyo Chapfuwa, Wenlin Wang, Guoyin Wang, Ricardo Henao, and Lawrence Carin. NASH: Toward end-to-end neural architecture for generative semantic hashing. In *Annual Meeting of the Association for Computational Linguistics*, pp. 2041–2050, jul 2018.

Rui Shu, Hung H Bui, Shengjia Zhao, Mykel J Kochenderfer, and Stefano Ermon. Amortized inference regularization. In *NeurIPS*, volume 31, 2018.

Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. In *Advances in Neural Information Processing Systems*, volume 29, pp. 3738–3746, 2016.

Hiroshi Takahashi, Tomoharu Iwata, Yuki Yamanaka, Masanori Yamada, and Satoshi Yagi. Student-t variational autoencoder for robust density estimation. In *IJCAI*, pp. 2696–2702, 7 2018.

R. E. Turner and M. Sahani. Two problems with variational expectation maximisation for time-series models. In *Bayesian Time series models*, chapter 5, pp. 109–130. 2011.

Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. In *NeurIPS*, volume 33, pp. 19667–19679, 2020.

Arash Vahdat, Evgeny Andriyash, and William Macready. Undirected graphical models as approximate posteriors. In *ICML*, volume 119, pp. 9680–9689, 13–18 Jul 2020.

Max Welling, Michal Rosen-Zvi, and Geoffrey E Hinton. Exponential family harmoniums with an application to information retrieval. In *NeurIPS*, volume 17, 2005.

Mingzhang Yin and Mingyuan Zhou. ARM: Augment-REINFORCE-merge gradient for stochastic binary networks. In *ICLR*, 2019.

Siamak Zamani Dadaneh, Shahin Boluki, Mingzhang Yin, Mingyuan Zhou, and Xiaoning Qian. Pairwise supervised hashing with Bernoulli variational auto-encoder and self-control gradient estimator. In *UAI*, volume 124, pp. 540–549, 2020.

Appendix

A PROOFS

A.1 PROOF OF THEOREM 1

The proof directly follows from the characterization of conditionally specified joint distributions in the exponential family given by Arnold & Strauss (1991), see (Arnold et al., 2001, Theorem 3):

Theorem A.1 (Arnold & Strauss 1991)**.** *Let $x \in \mathcal{X}$ and $z \in \mathcal{Z}$ be random variables with a strictly positive joint distribution such that both conditional distributions are exponential families with densities*

$$p(x \mid z) = h(x) \exp\big[\langle \nu(x), f(z) \rangle - A(z)\big] \tag{15a}$$

$$p(z \mid x) = h'(z) \exp\big[\langle \psi(z), g(x) \rangle - B(x)\big], \tag{15b}$$

where $\nu : \mathcal{X} \to \mathbb{R}^n$ and $\psi : \mathcal{Z} \to \mathbb{R}^m$ are minimal sufficient statistics, $f : \mathcal{Z} \to \mathbb{R}^n$ and $g : \mathcal{X} \to \mathbb{R}^m$ are any mappings, $h(x)$ and $h'(z)$ are base measures, and A and B denote the respective log-partition functions[4]*.*

Then there exists a matrix $W \in \mathbb{M}(n+1, m+1)$, such that the density of the joint distribution can be represented as

$$p(x, z) = h(x) h'(z) \exp \langle \nu_e(x), W \psi_e(z) \rangle \,, \tag{16}$$

where ν_e, ψ_e denote the statistics vectors extended with an additional component 1.

A.2 EXAMPLE OF A DISCRETE VAE APPROACHING A FLOW

Example A.1. In this example we construct a VAE that can be arbitrary close to tight one, but where the decoder network does not approach a linear map. Let $\mathcal{X} = \{-1, 1\}^2$, $\mathcal{Z} = \{-1, 1\}^2$ and let $p(z)$ be uniform. Let the decoder be conditionally independent

$$p(x|z) = \exp\big[\beta \langle x, \pi(z) \rangle - A(\beta \pi(z))\big] \tag{17}$$

where $\pi(z)$ denotes the invertible mapping of $\mathcal{Z}$ to $\mathcal{X}$ given by

$$x_1 = z_1 \tag{18}$$

$$x_2 = z_1 z_2. \tag{19}$$

If the parameter β is sufficiently large, the distribution $p(x \mid z)$ approaches the deterministic distribution $\delta_{x=\pi(z)}$. Its posterior therefore also approaches the deterministic distribution $\delta_{z=\pi^{-1}(x)}$ (notice that $\pi^{-1} = \pi$). Let the encoder be the conditionally independent model $q(z \mid x) = \exp\big[\beta \langle z, \pi^{-1}(x) \rangle - A(\beta \pi^{-1}(x))\big]$. This decoder by design approaches $\delta_{x=\pi(z)}$ as well and thus the VAE (p, q) achieves ε-tightness for sufficiently large β. At the same time the deviation between logarithms of probabilities $\log q(z|x)$ and $\log p(z|x)$ grows with β.

A.3 PROOF OF THEOREM 2

The idea of the proof is to bound the difference between $\log p(z|x)$ and $\log q(z|x)$, which is done by Proposition A.2 and then in the space of log-probabilities to approximate the non-linear mapping $f_e(z)$ by a linear one as detailed in Proposition A.1. Choosing the norms and the approximation appropriately allows to construct a bound, which despite the discreteness assumption about the observation space $\mathcal{X}$ in Theorem 2 does not depend on its cardinality.

For a finite set $X \subset \mathcal{X}$ let $\mathcal{H}$ be the $|X|$-dimensional vector space with the inner product $\langle u, v \rangle_{p_d} = \sum_{x \in X} p_d(x) u(x) v(x)$, assuming that $p_d(x) > 0$ for all $x \in \mathcal{X}$. The respective norm will be denoted as $\| \cdot \|_{p_d}$.

[4]The log-partition function is usually defined as a function of the (natural) parameter. In this context we consider h,h', f, g to be fixed and consider the dependence on x, z only.

Proposition A.1. *Under model Assumption 1, for any finite $X \subseteq \mathcal{X}$ there exists a matrix $W \in \mathbb{R}^{n,m}$ and a GLM*

$$\tilde{p}(x \mid z) = \tilde{h}(x) \exp(\langle \nu(x), W\psi(z) \rangle - \tilde{A}(z)). \tag{20}$$

such that the following inequality holds for all $z \in \mathcal{Z}$:

$$\sum_{x \in X} p_d(x) \big| \log p(x \mid z) - \log \tilde{p}(x \mid z) \big|^2 \leq \sum_{x \in X} p_d(x) \big| \log q(z \mid x) - \log p(z \mid x) \big|^2. \tag{21}$$

That is, if the RHS is small, the decoder network $f(z)$ can be effectively replaced by a linear model $W\psi(z)$.

Proof. For clarity, we will omit the dependence of the decoder and encoder on their parameters θ, resp. ϕ. Throughout the proof we will also assume that a single $z \in \mathcal{Z}$ is fixed.
First, we expand

$$\log q(z|x) = \langle \psi(z), g(x) \rangle - B(x) + \log h'(z); \tag{22a}$$

$$\begin{aligned} \log p(z|x) &= \log p(x|z) + \log p(z) - \log p(x) \\ &= \langle \nu(x), f(z) \rangle - A(z) + \log h(x) + \log p(z) - \log p(x), \end{aligned} \tag{22b}$$

where $A(z) = A(f(z))$ and $B(x) = B(g(x))$. We can therefore represent

$$\log q(z|x) - \log p(z|x) = \langle \psi(z), g(x) \rangle - B(x) + \log p(x) - \log h(x) \tag{23}$$

$$- \Big(\langle \nu(x), f(z) \rangle + \log p(z) - \log h'(z) - A(z) \Big) \tag{24}$$

$$= \langle \psi_e(z), g_e(x) \rangle - \langle \nu_e(x), f_e(z) \rangle \,, \tag{25}$$

where

$$\psi_e(z) = (\psi(z),\ 1); \tag{26}$$

$$\nu_e(x) = (\nu(x),\ 1); \tag{27}$$

$$g_e(x) = (g(x),\ \log p(x) - B(x) - \log h(x)); \tag{28}$$

$$f_e(z) = (f(z),\ \log p(z) - A(z) - \log h'(z)). \tag{29}$$

With this representation we have:

$$\sum_{x \in X} p_d(x) \big| \langle \nu_e(x), f_e(z) \rangle - \langle \psi_e(z), g_e(x) \rangle \big|^2 \tag{30}$$

$$= \sum_{x \in X} p_d(x) \big| \log q(z \mid x) - \log p(z \mid x) \big|^2 =: \Delta^2. \tag{31}$$

Let V be the matrix with rows $\nu_e(x)$ for all $x \in X$. Let G be the matrix with rows $g_e(x)$ for all $x \in X$. We can rewrite the condition (30) in the form

$$\xi = V f_e(z) - G\psi_e(z), \tag{32a}$$

$$\|\xi\|_{p_d}^2 = \Delta^2, \tag{32b}$$

where $\xi \in \mathbb{R}^{|X|}$ is the vector of residuals. Let P be the orthogonal projection onto the range of V in the space $\mathcal{H}$. Multiplying (32a) by P on the left, we obtain

$$PV f_e(z) - PG\psi_e(z) = P\xi. \tag{33}$$

Because $PV = V$ we obtain

$$V f_e(z) - PG\psi_e(z) = P\xi. \tag{34}$$

We therefore can consider the decoder network approximation $\tilde{f}_e(z) = \tilde{W}\psi_e(z)$, where $\tilde{W}$ is the solution to the consistent system of linear equations $V\tilde{W} = PG$ and is independent of z. We can therefore express

$$\|V f_e(z) - V\bar{f}_e(z)\|_{p_d} = \|P\xi\|_{p_d} \leq \|\xi\|_{p_d} = \Delta, \tag{35}$$

where the inequality holds because P is an orthogonal projection in $\mathcal{H}$.

We obtained that the decoder network $f_e(z)$ can be approximated by a linear mapping $\tilde{W}\psi_e(z)$ such that

$$\sum_{x \in X} p_d(x) \big| \langle \nu_e(x), f_e(z) \rangle - \langle \nu_e(x), \tilde{W}\psi_e(z) \rangle \big|^2 \leq \Delta^2. \tag{36}$$

Expressing back

$$\langle \nu_e(x), f_e(z)\rangle = \langle \nu(x), f(z)\rangle + \log p(z) - A(z) - \log h'(z) \tag{37}$$
$$= \log p(x|z) + \log p(z) - \log h(x) - \log h'(z) \tag{38}$$

We obtain that

$$\sum_{x\in X} p_d(x)\big|\log p(x|z) - \log \tilde{p}(x|z)\big|^2 \leq \Delta^2, \tag{39}$$

where

$$\log \tilde{p}(x|z) = \log \tilde{h}(x) + \langle \nu(x), W\psi(z)\rangle - \tilde{A}(z); \tag{40a}$$
$$\log \tilde{h}(x) = \log h(x) + \langle \nu(x), \tilde{W}_{1:n,n+1}\rangle; \tag{40b}$$
$$\tilde{A}(z) = A(z) - \log p(z) + \log h'(z) + \langle \tilde{W}_{n+1,1:n}, \psi(z)\rangle + \tilde{W}_{n+1,n+1}; \tag{40c}$$
$$W = \tilde{W}_{1:n,1:n}. \tag{40d}$$

□

Proposition A.2. *Under model Assumption 1, let $\mathcal{X}$ and $\mathcal{Z}$ be discrete (finite) sets and let $z \in \mathcal{Z}$ be chosen. If $q(z\,|\,x) \geq \alpha$ and $p(z|x) \geq \alpha$ for all $x \in X$, where $X \subseteq \mathcal{X}$ and*

$$\mathbb{E}_{p_d(x)}[D_{\mathrm{KL}}(q(z\,|\,x)\,\|\,p(z\,|\,x))] \leq \varepsilon, \tag{41}$$

then

$$\sum_{x\in X} p_d(x)\big(\log p(z\,|\,x) - \log q(z\,|\,x)\big)^2 \leq \tfrac{\varepsilon}{\alpha^2} + o(\varepsilon). \tag{42}$$

Proof. Let us denote $\varepsilon(x) = D_{\mathrm{KL}}(q(z\,|\,x)\,\|\,p(z\,|\,x))$. Pinsker's inequality assures for each x

$$\sup_{S\subset\mathcal{Z}} |\mathbb{P}_{q(z\,|\,x)}(S) - \mathbb{P}_{p(z\,|\,x)}(S)|^2 \leq \varepsilon(x)/2. \tag{43}$$

Substituting $S = \{z\}$ we obtain

$$|p(z\,|\,x) - q(z\,|\,x)|^2 \leq \varepsilon(x)/2. \tag{44}$$

By taking expectation in $p_d(x)$ on both sides we obtain a variant of Pinsker's inequality:

$$\mathbb{E}_{p_d(x)}|p(z\,|\,x) - q(z\,|\,x)|^2 \leq \mathbb{E}_{p_d(x)}\varepsilon(x)/2 = \tfrac{1}{2}\mathbb{E}_{p_d(x)}[D_{\mathrm{KL}}(q(z\,|\,x)\,\|\,p(z\,|\,x))] \leq \varepsilon/2. \tag{45}$$

Notice that the LHS depends on the given z. Because all summands are non-negative it follows that

$$\forall X' \subseteq \mathcal{X} \quad \sum_{x\in X'} p_d(x)|p(z\,|\,x) - q(z\,|\,x)|^2 \leq \varepsilon/2. \tag{46}$$

This inequality will be used in several places below.

Consider $x \in X$ such that $p(z\,|\,x) > q(z\,|\,x)$. Then

$$|\log p(z\,|\,x) - \log q(z\,|\,x)|^2 = \log^2 \tfrac{p(z\,|\,x)}{q(z\,|\,x)} \tag{47a}$$
$$= \log^2(1 + \tfrac{p(z\,|\,x)-q(z\,|\,x)}{q(z\,|\,x)}) \tag{47b}$$
$$\leq \log^2(1 + \tfrac{p(z\,|\,x)-q(z\,|\,x)}{\alpha}), \tag{47c}$$

where the inequality holds because $\log^2$ is monotonously increasing for arguments greater equal than 1, which is ensured. Let us now consider $x \in X$ such that $p(z\,|\,x) < q(z\,|\,x)$. Then

$$|\log p(z\,|\,x) - \log q(z\,|\,x)|^2 = \log^2 \tfrac{q(z\,|\,x)}{p(z\,|\,x)} \tag{48a}$$
$$= \log^2(1 + \tfrac{q(z\,|\,x)-p(z\,|\,x)}{p(z\,|\,x)}) \tag{48b}$$
$$\leq \log^2(1 + \tfrac{q(z\,|\,x)-p(z\,|\,x)}{\alpha}). \tag{48c}$$

In total, we obtain

$$|\log p(z\,|\,x) - \log q(z\,|\,x)|^2 \leq \log^2(1 + \tfrac{|p(z\,|\,x)-q(z\,|\,x)|}{\alpha}). \tag{49}$$

Let us denote $u(x) = \frac{|p(z\,|\,x)-q(z\,|\,x)|}{\alpha}$ and partition the set X into two parts:

$$X_1 = \{x \in X \mid u(x) < u_0\} \tag{50a}$$
$$X_2 = \{x \in X \mid u(x) \geq u_0\}, \tag{50b}$$

where we chose $u_0 \in [\sqrt{e-1}, 5]$ for reasons to be clarified below. For both parts, *i.e.* $k = 1, 2$, we have

$$\textstyle\sum_{x\in X_k} p_d(x)|\log q(z\,|\,x) - \log p(z\,|\,x)|^2 \leq \sum_{x\in X_k} p_d(x) \log^2(1 + \frac{|p(z\,|\,x)-q(z\,|\,x)|}{\alpha}). \tag{51}$$

For X_1, (51) can be further bounded as

$$\leq \textstyle\sum_{x\in X_1} p_d(x) \log(1 + \frac{|p(z\,|\,x)-q(z\,|\,x)|^2}{\alpha^2}) \tag{52a}$$
$$\leq \log \textstyle\sum_{x\in X_1} p_d(x)\Big(1 + \frac{|p(z\,|\,x)-q(z\,|\,x)|^2}{\alpha^2}\Big) \tag{52b}$$
$$= \log\Big(p_d(X_1) + \tfrac{1}{\alpha^2}\textstyle\sum_{x\in X_1} p_d(x)|p(z\,|\,x) - q(z\,|\,x)|^2\Big) \tag{52c}$$
$$\leq \log\Big(1 + \tfrac{\varepsilon}{2\alpha^2}\Big) = \tfrac{\varepsilon}{2\alpha^2} + o(\varepsilon), \tag{52d}$$

where the first inequality holds for $u_0 \leq 5$, because in this case $\log^2(1+u) \leq \log(1+u^2)$ holds, the second inequality is the Jensen's inequality for $\log$ and the last inequality uses (46) under monotone $\log$.

For X_2 we have the following. Let $V = p_d(X_2) = \sum_{x\in X_2} p_d(x)$. We can express

$$\textstyle\sum_{x\in X_2} p_d(x) \log^2\Big(1 + \frac{|p(z\,|\,x)-q(z\,|\,x)|}{\alpha}\Big) \tag{53a}$$
$$= V\Big(\textstyle\sum_{x\in X_2} p_d(x\,|\,X_2) \log^2\Big(1 + \frac{|p(z\,|\,x)-q(z\,|\,x)|}{\alpha}\Big)\Big) \tag{53b}$$

Using that $u < u^2$ on X_2 and that $\log^2(1+u)$ is monotone for a positive argument, we can bound (53b) as

$$\leq V\Big(\textstyle\sum_{x\in X_2} p_d(x\,|\,X_2) \log^2\Big(1 + \frac{|p(z\,|\,x)-q(z\,|\,x)|^2}{\alpha^2}\Big)\Big). \tag{54}$$

Further, using that $\log^2(1+v)$ is concave on X_2 for $v \geq e-1$, we have

$$\leq V \log^2\Big(\textstyle\sum_{x\in X_2} p_d(x\,|\,X_2)\big(1 + \frac{|p(z\,|\,x)-q(z\,|\,x)|^2}{\alpha^2}\big)\Big) \tag{55a}$$
$$= V \log^2\Big(1 + \textstyle\sum_{x\in X_2} p_d(x\,|\,X_2)\frac{|p(z\,|\,x)-q(z\,|\,x)|^2}{\alpha^2}\Big) \tag{55b}$$
$$\leq V \log^2\Big(1 + \tfrac{\varepsilon}{2\alpha^2 V}\Big), \tag{55c}$$

where in the last step we used (46).

Next we show that V itself is bounded above by $\frac{\varepsilon}{2u_0^2\alpha^2}$. It follows from

$$u_0^2 V = \sum_{x\in X_2} p_d(x)u_0^2 \leq \sum_{x\in X_2} p_d(x)\frac{|p(z\,|\,x)-q(z\,|\,x)|^2}{\alpha^2} \leq \sum_{x\in X} p_d(x)\frac{|p(z\,|\,x)-q(z\,|\,x)|^2}{\alpha^2} \leq \frac{\varepsilon}{2\alpha^2}, \tag{56}$$

where the last inequality is again (46).

Now, letting $r = \frac{\varepsilon}{2\alpha^2 V} \geq u_0^2 \geq 1$ we can bound (55c) as follows

$$V \log^2\Big(1 + \tfrac{\varepsilon}{2\alpha^2 V}\Big) = \tfrac{\varepsilon}{2\alpha^2}\tfrac{1}{r}\log^2\Big(1 + r\Big) \tag{57a}$$
$$\leq \tfrac{\varepsilon}{2\alpha^2}\sup_{r\geq 1}\tfrac{1}{r}\log^2\Big(1 + r\Big) \leq \tfrac{\varepsilon}{2\alpha^2}0.64. \tag{57b}$$

□

Theorem 2 follows by Proposition A.2 and Proposition A.1 with the choice $X = \mathcal{X}$.

B DETAILS OF EXPERIMENTAL SETUP

B.1 ARTIFICIAL EXAMPLE

We give here the achieved likelihood and ELBO values for this experiment. The negative entropy $\sum_x p^*(x) \log p^*(x)$ of the ground truth model, *i.e.*, the best reachable data log-likelihood, is -3.65. The ELBO of the pre-trained VAE is -3.74. During the second step of training, *i.e.*, the joint learning of the encoder and decoder by ELBO maximization, the ELBO value increases from -3.74 to -3.70. The data log-likelihood $\sum_x p^*(x) \log p_\theta(x)$ drops at the same time from -3.65 to -3.68. The approximation error (2) in the decoder caused by using the factorized encoder is only 0.03 nats, but the qualitative difference between the ground truth model and the ELBO optimizer model shown in Fig. 2 appears detrimental.

B.2 BERNOULLI VAE FOR TEXT DOCUMENTS

Dataset In this experiment we used the version of the *20Newsgroups* data set (Lang & Rennie, 2008) denoted as "processed" by the authors. The dataset contains bag-of-words representations of documents and is split into a training set with 11269 documents and a test set with 7505 documents. We keep only the 10000 most frequent words in the training set, which is a common pre-processing (each of the omitted words occurs not more than in 10 documents).

Optimization To train VAE we used the state-of-the-art unbiased gradient estimator ARM (Yin & Zhou, 2019) and Adam optimizer with learning rate 0.001. We did not use the test set for parameter selection. We train for 1000 epochs using 1-sample ARM and then for 500 more epochs using 10 samples for computing each gradient estimate with ARM. We report the lowest negative ELBO (NELBO) values for the training set and test set during all epochs.

Models The decoder model (12) describes words as independent draws from a categorical distribution specified by the neural network $f(z)$. This network respectively has a structure

$$\text{Linear} \to \underbrace{(\text{ReLU} \to \text{Linear})}_{\times \text{dhidden}} \to \text{Logsoftmax}.$$

The input dimension equals to the number of latent bits, the output dimension equals the number of words in the dictionary, 10000. For decoders with dhidden $= 1, 2$ the hidden layers contained 512 units.

The encoder networks **e2**, **e3** take on the input word frequencies $x / \sum_k x_k$, the encoder network **e1** takes on input word counts x. For the deep encoder **e2** we used 2 hidden ReLU layers with 512 units each.