EMPIRICAL OBSERVATIONS PERTAINING TO LEARNED PRIORS FOR DEEP LATENT VARIABLE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

There exist many forms of deep latent variable models, such as the variational autoencoder and adversarial autoencoder. Regardless of the specific class of model, there exists an implicit consensus that the latent distribution should be regularized towards the prior, even in the case where the prior distribution is learned. Upon investigating the effect of latent regularization on image generation our results indicate that in the case when a sufficiently expressive prior is learned, latent regularization is not necessary and may in fact be harmful insofar as image quality is concerned. We additionally investigate the benefit of learned priors on two common problems in computer vision: latent variable disentanglement, and diversity in image-to-image translation.

1 INTRODUCTION

In the machine learning subfield of deep latent variable models such as variational autoencoders (VAEs) Kingma & Welling (2014) and adversarial autoencoders (AAEs) Makhzani et al. (2016) have attracted a significant amount of research interest Bowman et al. (2016); Kingma et al. (2016); Huang et al. (2017); Rezende & Viola (2018); Dai & Wipf (2019); Tolstikhin et al. (2018). Despite this, in their standard form they are still largely outperformed in terms of synthesized image quality by other deep generative models such as generative adversarial networks (GANs) Goodfellow et al. (2014); Karras et al. (2018a;b), autoregressive models Van den Oord et al. (2016) and flow-based models Dinh et al. (2015; 2017); Kingma & Dhariwal (2018). Even so, latent variable models maintain a number of properties that make them an attractive alternative, such as stable and efficient training as well as efficient synthesis.

In virtually all research done using such models, some form of regularization is imposed on the encoder distribution in order to push it towards the prior distribution. For VAEs, this regularization exists in the form of a KL divergence between approximate posterior and prior, while AAEs force the marginal posterior to match the prior using an adversarial loss. This is of course necessary if the prior is fixed as is often the case, however we argue that when the prior is learnable it is possible to achieve a tight fit between marginal posterior and prior without regularization, and that regularization in this case may actually have a negative impact on sample quality. Experimental results demonstrating this are given in section 4. We also discuss how better disentanglement of the latent variable is achieved as a consequence of removing latent regularization, with experimental results provided in section 5. Finally, we show that removing latent regularization can improve sample diversity in image-to-image translation tasks; results are provided in section 6.

2 BACKGROUND AND PRELIMINARIES

2.1 DEEP LATENT VARIABLE MODELS

Let $\mathbf{x}$ be a random vector residing in space $\mathcal{X}$ with
$\mathbf{x} \sim p(\mathbf{x})$. Additionally, let $\mathbf{X} = \{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(n)}\}$ be a set of i.i.d. observations drawn from the data distribution $p(\mathbf{x})$. When training likelihood-based generative models, we are interested in maximizing the log-likelihood

$$\log p(\mathbf{X}) = \sum_{i=1}^{n} \log p(\mathbf{x}^{(i)}) \tag{1}$$

Latent variable models introduce a latent vector $\mathbf{z}$ residing in latent space $\mathcal{Z}$ which together with $\mathbf{x}$ forms the joint distribution $p(\mathbf{x}, \mathbf{z})$. The likelihood of an individual datapoint is then given by

$$p(\mathbf{x}^{(i)}) = \int_{\mathcal{Z}} p(\mathbf{x}^{(i)}|\mathbf{z})p(\mathbf{z})d\mathbf{z} \tag{2}$$

In the case when the parameterization of $p(\mathbf{x}^{(i)}|\mathbf{z})$ is complex, such as when it is parameterized by a neural network, Eq. 2 becomes intractable.

Variational Autoencoders

Variational autoencoders circumvent the issue of intractability by noting that the log-likelihood of an individual datapoint can be expressed as:

$$\begin{aligned} &\log p(\mathbf{x}^{(i)}) - D_{KL}[q(\mathbf{z}|\mathbf{x}^{(i)})||p(\mathbf{z}|\mathbf{x}^{(i)})] \\ =\; &\mathbb{E}_{q(\mathbf{z}|\mathbf{x}^{(i)})}[\log p(\mathbf{x}^{(i)}|\mathbf{z})] - D_{KL}[q(\mathbf{z}|\mathbf{x}^{(i)})||p(\mathbf{z})] \end{aligned} \tag{3}$$

where the right hand side of Eq. 3 is commonly referred to as the *Variational Lower Bound*, which provides a tractable lower bound for the data likelihood that can be optimized directly.

However, variational autoencoders suffer from a number of problems that degrade the quality of samples. For example, when the posteriors of a set of data points overlap at some point in $\mathcal{Z}$, then the optimal decoding at that point will be a weighted mean of the corresponding data points Rezende & Viola (2018). This contributes to the well known problem of blurry samples. The standard method of encoding the posterior distribution as a diagonal Gaussian is commonly blamed for exarcerbating this issue; the true posterior $p(\mathbf{z}|\mathbf{x}^{(i)})$ will almost certainly never be a diagonal Gaussian, therefore the model is in a sense forced to fit square pegs into round holes, resulting in either posterior overlap or "holes" that have high probability under the prior but are never seen by the decoder Zhao et al. (2017). Works such as IAF Kingma et al. (2016) propose to increase the flexibility of the approximate posterior by turning it into a *normalizing flow* (discussed in a later section), allowing a tighter fit between true and approximate posteriors. In Chen et al. (2017), the authors propose modelling the prior of a variational autoencoder using an autoregressive flow, and demonstrate that it is equivalent to modelling the posterior using IAF along the encoder path, while being deeper along the decoder path.

It is also a well known issue that VAEs tend to not make full use of the latent code, as the objective becomes trapped in a local minima in which the posterior is close to the prior. Such a state occurs early on when the signal from the latent code is weak, resulting in a weak reconstruction term $\log p(\mathbf{x}^{(i)}|\mathbf{z})$ that is easily outweighed by the KL divergence term in the objective. This exacerbates the problems caused by overlapping posteriors, resulting in even blurrier samples, and additionally results in lower sample diversity. This issue is particularly pernicious in the conditional setting if care is not taken, as it is easily possible for the model to entirely ignore the latent code when it is conditioned on a relevant context, resulting in a deterministic mapping.

Many methods for encouraging use of the latent code have been proposed. For instance, annealing the KL divergence term from 0 to full strength Bowman et al. (2016); Huang et al. (2018b) allows the model to largely ignore the KL divergence term at the beginning of training, such that the autoencoder is able to learn a good reconstruction without being constrained by the prior. By the time the KL divergence term is annealed to full strength, the autoencoder is expected to have made good use of the latents, preventing the model from regressing to a state in which the posterior is close the prior. "Free bits", introduced in Kingma et al. (2016), places a limit on the information in nats per latent subset that can contribute to the KL divergence term, ensuring that each subset can contribute at least λ nats of information without penalty. In the context of conditional variational autoencoders, Zhu et al. (2017) proposed to add a latent reconstruction term to the objective to encourage the model to make full use of the latent code.

At their core, all of these techniques are intended as a means of alleviating over-regularization imposed by the KL divergence term in the objective.

This term is of course necessary if the prior takes on a fixed form (e.g. a standard normal distribution). Indeed, all other issues aside, forcing the aggregate posterior distribution towards a pre-determined prior may in and of itself over-regularize the model. In the case that the prior is learned, however, it is possible to eliminate regularization entirely by removing the KL divergence term from the objective, hence obviating the need for any aforementioned techniques.

Generative Moment Matching Networks Li et al. (2015)

Generative Moment Matching Networks (GMMNs) Li et al. (2015) are a very relevant work in which the authors propose using an unregularized autoencoder as a generative model. This is the only work to our knowledge that proposes a latent variable model without any latent regularization. In their proposal they use maximum mean discrepancy (MMD) in order to match the statistics of the latent distribution of the autoencoder and a random variable $\mathbf{z_T} = h(\mathbf{z_0})$, where h is parameterized by a neural network and $\mathbf{z_0} \sim \mathcal{U}(-1, 1)$. In their implementation of MMD they use the Gaussian kernel $k(\mathbf{x}, \mathbf{x}') = \exp(-\frac{1}{2\sigma^2}||\mathbf{x} - \mathbf{x}'||_2^2)$ with bandwidth parameter σ^2 which, when the kernel trick is applied, allows matching between all possible moments of the two distributions. The most obvious drawback with this method is that it requires careful tuning of the bandwidth parameter σ^2, as it plays a crucial role in determining the accuracy of the model. Additionally, training with large batch sizes can be problematic as the complexity of the objective calculation scales quadratically. At the same time, accuracy of training with MMD has been shown to degrade rapidly as the batch size becomes smaller Li et al. (2017).

Adversarial Autoencoders Makhzani et al. (2016)

Another model, closely related to GMMN+AE, is the Adversarial Autoencoder (AAE) Makhzani et al. (2016); Tolstikhin et al. (2018). Rather than matching distributions using MMD, the authors propose utilizing an adversarial objective to regularize the distribution of the autoencoder latent space towards $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Regularization of the autoencoder in this way may introduce excess noise, the reasoning being that GAN discriminators are known to constantly shift their probability mass around during training in response to the generator, and so the decoder of an AAE is forced to deal with noisy latent codes. Additionally, if the data distribution lies on a manifold that is of lower dimensionality than the latent space and if deterministic encoders are used the model will necessarily produce a degenerate latent distribution, in which case the “holes” problem discussed earlier manifests.

Therefore we propose an alternative: train an unregularized autoencoder and separately learn a generator which attempts to generate samples from the same distribution as the autoencoder's latent space, i.e. a GAN is used to learn the prior. We refer to this modification of AAE as Prior-AAE. Both the AAE and Prior-AAE models inherit the drawbacks of adversarial training in general, e.g. mode collapse, unstable training, and lack of convergence Salimans et al. (2016), although many techniques have emerged in recent years to mitigate these issues Metz et al. (2017); Arjovsky et al. (2017); Mescheder (2018).

2.2 Normalizing flows

Normalizing flows are a class of model that allows tractable and exact latent variable inference and log-likelihood evaluation by taking advantage of the change of variables formula for an invertible function $f : \mathcal{X} \to \mathcal{Z}$

$$p_{\mathbf{x}}(\mathbf{x}) = p_{\mathbf{z}}(f(\mathbf{x})) \left| \det(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T}) \right| \tag{4}$$

Choosing a simple distribution for $p_{\mathbf{z}}(\mathbf{z})$ makes evaluation of $p_{\mathbf{z}}(f(\mathbf{x}))$ tractable, and also makes sampling straightforward. In general, computing $\det(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T})$ takes $O(n^3)$ time, where n is the dimensionality of $\mathbf{x}$. We can improve on this however by carefully choosing the form of the bijection $f(\mathbf{x})$. Autoregressive flows Kingma et al. (2016); Papamakarios et al. (2017); Huang et al. (2018a) constrain $f(\mathbf{x})$ to be autoregressive, such that the Jacobian $\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}^T}$ is triangular. Since the determinant of a triangular matrix is the product of its diagonal entries, computation of the Jacobian determinant becomes $O(n)$. For autoregressive flows with affine transformations, $\mathbf{z}$ is given by $\mathbf{z}_i = \mathbf{x}_i s(\mathbf{x}_{1:i-1}) + t(\mathbf{x}_{1:i-1})$ where s and t are neural networks. Greater model capacity can be achieved by composing multiple autoregressive bijections and permuting the order of latent dimensions in-between each, giving $f = f_1 \circ w_1 \circ f_2 \circ w_2 \circ ... \circ f_{T-1} \circ w_{T-1} \circ f_T$ where $w_t(\mathbf{x}) = \mathbf{W_t}\mathbf{x}$ and $\mathbf{W_t}$ is some permutation matrix. Following from this we define the set of random variables

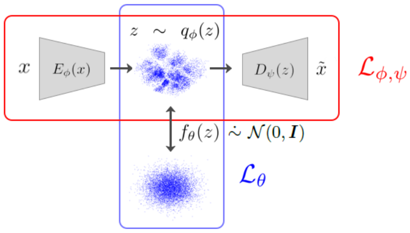

x
$E_\phi(x)$
$z \sim q_\phi(z)$
$D_\psi(z)$
$\tilde{x}$
$\mathcal{L}_{\phi,\psi}$
$f_\theta(z) \sim \mathcal{N}(0, \boldsymbol{I})$
$\mathcal{L}_\theta$

Figure 1: Schematic of our proposal. Please see Eq. 6 and Eq. 8 for definitions of $\mathcal{L}_{\phi,\psi}$ and $\mathcal{L}_\theta$.

$\{\mathbf{z_t} \in \mathcal{Z}_t\}_{t=0}^T$ where $\mathbf{z_{t-1}} = f_t(\mathbf{z_t})$, $\mathbf{z_T} = \mathbf{x}$ and $\mathbf{z_0} = f(\mathbf{x})$. It is worth noting that, as autoregressive flows are universal approximators Huang et al. (2017), they can be used to model arbitrarily complex distributions.

By transforming the posterior samples of a VAE based on an autoregressive flow, the capacity of the posterior can be increased in a way that admits tractable posterior density evaluation, and in doing so it is easy to see that the issue of posterior overlap can be mitigated. However, due to the highly non-convex nature of the objective, it is still easily possible for the model to get trapped in local minima in which the posterior is close to the prior, preventing the model from making full use of the latent code.

3 Regularization experiment approach

In the description of our approach we use the following notation: The encoder is denoted by $E_\phi(\mathbf{x})$, the decoder is denoted by $D_\psi(\mathbf{z})$, and the latent code is denoted by
$\mathbf{z} \sim q_\phi(\mathbf{z})$.

In order to empirically investigate the effect of latent regularization on image quality, we consider two kinds of models. For the first model, we consider VAEs with prior learned via normalizing flow as proposed in Chen et al. (2017), with the KL divergence term weighted by a hyperparameter β. Rather than the autoregressive flow used in Chen et al. (2017), we use a flow that splits the vector in half at each step as in Dinh et al. (2017), as this allows efficient computation of the bijection in both directions, facilitating both efficient inference and efficient sampling. We note again that this gives rise to a set of latents $\{\mathbf{z_t} \in \mathcal{Z}_t\}_{t=0}^T$, where $\mathbf{z_0} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\mathbf{z_T}$ should have distribution as close as possible to $q_\phi(\mathbf{z})$. The bijection is given by $f_\theta : \mathcal{Z}_T \to \mathcal{Z}_0$ with $\mathbf{z_0} = f_\theta(\mathbf{z_T})$.

To evaluate the effect of β we train the model in the following way. Firstly we assume that the encoder E_ϕ encodes the parameters of the posterior $q_\phi(\mathbf{z}|\mathbf{x})$, and that the decoder D_ψ encodes the parameters of $p_\psi(\mathbf{x}|\mathbf{z})$. We then split the objective into two separate functions:

$$\begin{aligned}\mathcal{L}_{\phi,\psi}(\phi,\psi;\mathbf{x}^{(i)}) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x}^{(i)})}[&\log p_\psi(\mathbf{x}^{(i)}|\mathbf{z}) \\ &+ \log p_\theta(\mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x}^{(i)})]\end{aligned} \tag{5}$$

$$\mathcal{L}_\theta(\theta;\mathbf{x}^{(i)}) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x}^{(i)})}[\log p_\theta(\mathbf{z})] \tag{6}$$

The encoder and decoder terms can be dropped from $\mathcal{L}_\theta$ since they are independent of θ. Using the gradients $\nabla_{\phi,\psi}\mathcal{L}_{\phi,\psi}(\phi,\psi;\mathbf{x}^{(i)})$ and $\nabla_\theta\mathcal{L}_\theta(\theta;\mathbf{x}^{(i)})$ concurrently in a gradient ascent step is then equivalent to performing gradient ascent on the original objective. Introducing β into the model, we can rewrite $\mathcal{L}_{\phi,\psi}$ as

$$\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x}^{(i)})}[\log p_\psi(\mathbf{x}^{(i)}|\mathbf{z}) + \beta(\log p_\theta(\mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x}^{(i)}))] \tag{7}$$

For VAEs with a fixed prior (e.g. a standard normal distribution), as β becomes larger we achieve a more structured latent space at the expense of reconstruction quality Matthey et al. (2017), and vice versa as β becomes smaller. When the parameters of the prior $p_\theta(\mathbf{z})$ are learnable, however, decreasing β does not necessarily sacrifice latent structure, as any additional incurred divergence between the aggregate posterior $q_\phi(\mathbf{z})$ and the prior $p_\theta(\mathbf{z})$ can be mitigated by adjusting the parameters of $p_\theta(\mathbf{z})$. When $\beta = 0$ we are left with

$$\mathcal{L}_{\phi,\psi}(\phi, \psi; \mathbf{x}^{(i)}) = \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x}^{(i)})}[\log p_\psi(\mathbf{x}^{(i)}|\mathbf{z})] \tag{8}$$

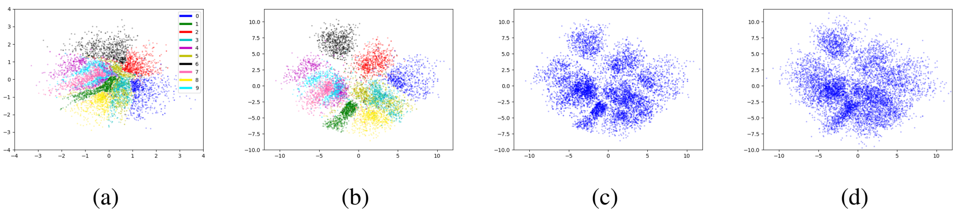
which is the objective of an unregularized autoencoder. It is worth pointing out that at this point, if L2 loss is used for the decoder, we are in fact minimizing the 2-Wasserstein distance between the model and data distributions. This comes from the proof in Tolstikhin et al. (2018) showing the equivalence between the optimal transport objective and $\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}[c(\mathbf{x}, \mathbf{z})]]$ under the constraint that $q_\phi(z) = p_\theta(z)$, where c is a cost function. In their case the constraint is relaxed and replaced by a regularizer that pushes $q_\phi(z)$ towards $p_\theta(z)$. In our case, rather than regularizing $q_\phi(z)$, we are instead learning $p_\theta(z)$.

The second model that we consider is an AAE with learnable prior. This is equivalent to an AAE with fixed prior with the addition that the adversarial loss that regularizes $q_\phi(z)$ is additionally used to learn the prior parameters. We can apply a hyperparameter β to control the strength of the adversarial loss applied to $q_\phi(z)$ in the same way as we did for the first model. As above, if L2 loss is used for the decoder this model minimizes the 2-Wasserstein distance. A disadvantage of this model is that it has no way to infer the latent variable $\mathbf{z_0}$ from an arbitrary data sample; its generative components can freely generate samples of $\mathbf{z_T}$ but the process is not invertible. The ability to infer $\mathbf{z_0}$ can be beneficial for certain tasks. For example, if the dataset is known to have been generated by a set of independent factors of variation, then it is possible that these factors will be recovered along the axes of $\mathcal{Z}_0$ in an unsupervised manner, an outcome generally referred to as disentanglement. Additionally, interpolation in $\mathcal{Z}_0$ is likely to yield more probable images than interpolation in $\mathcal{Z}_T$ as the distribution of $\mathbf{z_T}$ is likely to be highly chaotic and multi-modal.

4 Experiments on values of β

We first train our proposed model on MNIST with a 2-dimensional latent code in order to demonstrate visually the learned latent distributions. This is shown in Figure 2; while the autoencoder has learned a latent distribution that is complex and multi-modal, the samples from the learned prior are a close match.

Our hypothesis is that lowering β towards zero will enable more accurate reconstruction, while learning the parameters of the prior will be sufficient to mitigate the divergence between $q_\phi(z)$ and $p_\theta(z)$. If the hypothesis is correct, decreasing β can only ever be beneficial for sample quality, therefore we experiment with various values of β between 0 and 1 to confirm that this is the case. We carried out experiments on the MNIST, Fashion-MNIST and CIFAR-10 datasets. We chose to adopt the Fréchet Inception Distance (FID) Heusel et al. (2017) to measure image quality, a common measure used in GAN evaluation, for our quantitative comparisons. FID scores are given by the Fréchet distance between layer activations of the Inception v3 network Szegedy et al. (2015),



(a) (b) (c) (d)

Figure 2: Distribution of the latents of a flow prior with $\beta = 0$ after training on MNIST with $\dim(\mathcal{Z}) = 2$. (a) Distribution of $f_\theta(E_\phi(\mathbf{x}))$ for each $\mathbf{x}$ in the test set. Datapoints are colored according to class. (b) Distribution of $E_\phi(\mathbf{x})$ for each $\mathbf{x}$ in the test set. (c) Same as (b) but without class coloring. (d) Distribution of $f_\theta^{-1}(\mathbf{z_0})$ where $\mathbf{z_0} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. It can be seen that while the autoencoder learns a complex latent distribution with classes well separated, the normalizing flow is able to learn a close match.

with lower scores indicating greater similarity between two image sets.

Consistent with the hypothesis, FID scores typically decrease as β is decreased for all models as can be seen in Figure 3. Interpolations in latent space are shown in Figure **??** and 4 in order to demonstrate that the model has learned a smooth manifold and is not memorizing training images. We use spherical linear interpolation as suggested by White (2016). Nearest neighbour samples from both the training and test sets are shown for the MNIST samples in order to demonstrate that the model often generates images that are closer to the test set than the training set.

4.1 Training details

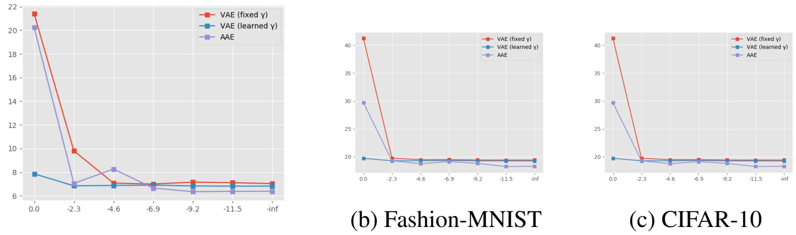
In all of our experiments involving either VAE or FP-VAE we linearly anneal the KL divergence term from 0 to γ over the first 50k iterations in order to encourage use of the latent code. For fair comparison, the autoencoder architecture was the same for all models; we implemented the network architectures used for MNIST and CIFAR-10 described in Chen et al. (2016); Lucic et al. (2018). We used a latent dimensionality of 8 for MNIST and 64 for CIFAR-10. Further details can be found in the supplementary.

5 Experimental results on CelebA

In order to demonstrate results on a higher dimensional dataset and explore the issue of disentanglement, we trained the proposed model as well as a standard VAE on the CelebA dataset Liu et al. (2015). Autoencoders typically perform very poorly on high dimensional images when using L_2 loss, as the reconstructions are often very blurry. As pointed out in Hou et al. (2017), we can use the VGG19 network Simonyan & Zisserman (2015) to improve the appearance of reconstructions, making them sharper and more realistic. We therefore apply VGG19 perceptual loss Johnson et al. (2016) using the `relu_1_1`, `relu_2_1`, `relu_3_1` and `relu_4_1` layers of the VGG19 network. We also trained PG-GAN Karras et al. (2018a) with the same latent dimensionality, which was set to 128. We are of course not attempting to be competitive with state-of-the-art GAN architectures, but are merely showing quantitative and qualitative results for comparison with what is possible with state-of-the-art methods. Images in the training set were cropped to a 128x128 region of the face. Samples are shown in Figure 6, and FID scores are shown in Table 1. Architectural and training details are provided in the supplementary.

5.1 Disentanglement

The term "disentanglement" can cover a broad range of definitions, but a generalized high-level notion is that the model should capture individual factors of variation within linear subspaces of $\mathcal{Z}$. We hypothesize that when the underlying factors of variation of the dataset are not independent, our model will learn a more disentangled representation in the distribution of $\mathbf{z_T}$ than in the distribution of $\mathbf{z_0}$. As posited by Karras et al. (2018b), the decoder is likely to pressure $\mathbf{z_T}$ to take on a disentangled form, since intuitively this should make accurate reconstruction easier as opposed to trying



(b) Fashion-MNIST

(c) CIFAR-10

(a) MNIST

Figure 3: Experimental results using different values of β for different models. X-axis is β in log-scale. Y-axis is FID score.

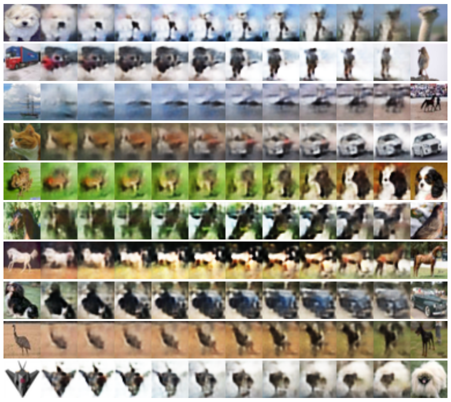


Figure 4: Interpolation between samples from the CIFAR-10 test set for a flow prior with fixed γ and $\beta = 0$. Leftmost and rightmost columns contain real images from the test set before encoding, middle columns contain interpolations between them.

to unwarp a highly entangled representation. In contrast, $\mathbf{z_0}$ is forced to take on a fixed distribution that is highly unlikely to align linearly with the factors of variation. This line of reasoning is equally applicable to standard VAEs that impose a fixed form prior.

The CelebA dataset contains 40 binary attributes that we can consider as such factors of variation, and thus we can use these attributes to calculate an approximate measure of disentanglement. If the model is successful at disentanglement, then for each attribute it should be easy to find a linear hyperplane that partitions the latent encodings into two sets, with each side of the hyperplane corresponding to one of the two possible values of the given attribute. We therefore consider the linear separability score proposed in Karras et al. (2018b). In their work they first train a deep network classifier on the training images that predicts image attributes, and then train a linear SVM classifier that predicts the classifier network's output given the latent variable. After this they calculate the conditional entropy $\mathrm{H}(\mathbf{Y}|\mathbf{X})$ where $\mathbf{Y}$ represents the labels predicted by the deep network classifier, and $\mathbf{X}$ represents the labels predicted by the SVM. It can be seen that lower conditional entropy will correspond to better linear separation, since the SVM will have higher prediction accuracy and thus observing $\mathbf{Y}$ will give less information. By following their procedure exactly, we can quantitatively measure disentanglement purely as a function of the generative process of the model.

We evaluated disentanglement of $\mathbf{z_0}$ and $\mathbf{z_T}$ in our model using the above procedure, and additionally evaluated disentanglement of $\mathbf{z}$ in a standard VAE. Results are reported in Table 1. As expected, $\mathbf{z_T}$ has a more disentangled representation than $\mathbf{z_0}$ and, interestingly, also achieves a more disentangled representation than a standard VAE. In contrast with common methods for inducing disentanglement in VAEs that sacrifice reconstruction quality for improved disentanglement

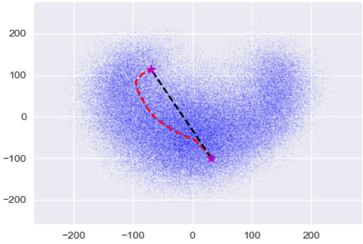
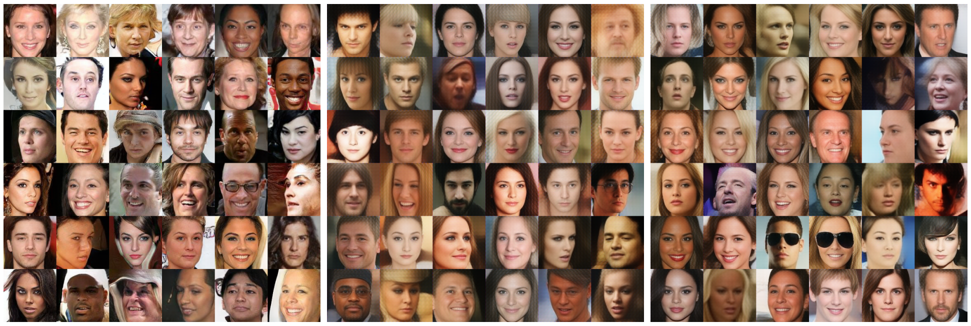


Figure 5: PCA of $E_\phi(x)$ for each x in the CelebA training set. The black line shows the path taken when interpolating between two points in $\mathcal{Z}_T$, while the red line shows the path taken when interpolating between the same two points in $\mathcal{Z}_0$. The black line takes the most direct route, while the red line follows a path of higher average density.



(a) PG-GAN (b) VAE (c) Proposed

Figure 6: Samples from each model after training on the CelebA dataset. Samples were generated randomly and were not cherry-picked. Visually, we can see that the general problem of blurriness faced by VAEs with L_2 loss has been replaced by an increased amount of VGG19 artifacts, while the proposed model samples contain fewer artifacts.

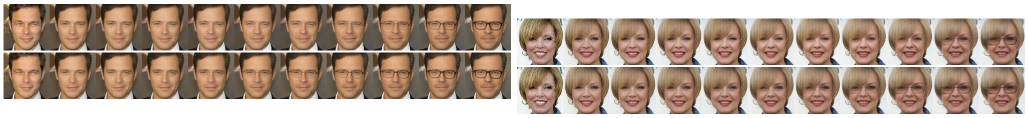


Figure 7: Adding glasses to a face. Top row of each set: interpolation in $\mathcal{Z}_0$. Bottom row of each set: interpolation in $\mathcal{Z}_T$. Leftmost column: the original image before encoding. The top row shows a more abrupt change towards the end, while the bottom row is closer to a constant rate of change.

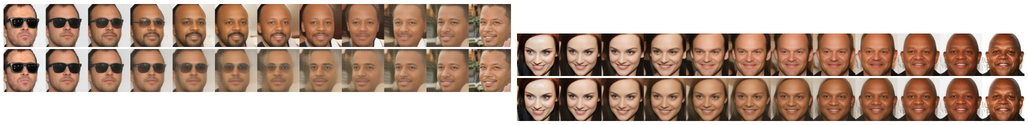


Figure 8: Interpolation between two random samples from the training set. Top row of each set: interpolation in $\mathcal{Z}_0$. Bottom row of each set: interpolation in $\mathcal{Z}_T$. Leftmost and rightmost columns: the original images before encoding. The top row shows more realistic images as a result of the interpolation path having higher average density.

	Separability	FID
VAE	2.14	41.4
β-VAE		2.09
Proposed ($\mathbf{z_0}$)	2.82	33.1
Proposed ($\mathbf{z_T}$)	1.67	

Table 1: Linear separability and FID scores after training on CelebA.

(e.g. β-VAE Matthey et al. (2017)), the proposed model achieves both improved disentanglement and improved reconstruction quality.

5.2 INTERPOLATION

When interpolating between points in latent space the motivation is often to achieve some semantic mixture between two images, or to change some semantic feature of an image such as putting glasses on a person's face. Here we discuss potential differences between interpolating in $\mathcal{Z}_0$ or in $\mathcal{Z}_T$. As a point of clarification, when we say we are interpolating between two images $\mathbf{x}^{(a)}$ and $\mathbf{x}^{(b)}$ in $\mathcal{Z}_T$ we are calculating $D_\psi(\text{lerp}(E_\phi(\mathbf{x}^{(a)}), E_\phi(\mathbf{x}^{(b)}); t))$, and when we say we are interpolating in $\mathcal{Z}_0$ we are calculating $D_\psi(f_\theta^{-1}(\text{slerp}(f_\theta(E_\phi(\mathbf{x}^{(a)})), f_\theta(E_\phi(\mathbf{x}^{(b)})); t)))$, where t varies between 0 and 1.

As demonstrated in the previous section, the warping of $\mathbf{z_T}$ via f_θ causes the representation to become highly entangled. Therefore interpolation in $\mathcal{Z}_0$ is not appropriate when attempting to change semantic features, as moving along the path of interpolation is likely to change many different semantics with varying rates of change at each point on the path. Even if interpolation manages to coincide with the direction of change of a single semantic feature, the rate of change along the path will vary considerably if there exists class imbalance of the feature; this is due to the fixed distribution of $\mathbf{z_0}$. As an example, consider the case of a uniform distribution: the relative amount of space occupied by points with a particular semantic feature would be proportional to the class probability of the feature. What this means in practice is that if we were to generate a sequence of images by interpolating in $\mathcal{Z}_0$ along the direction of a semantic feature with heavy class imbalance, there would be very little change for most of the sequence followed by an abrupt change at the end. We posit that interpolating in $\mathcal{Z}_T$ is more likely to achieve the ideal scenario of having a constant rate of change, as it is not constrained by a fixed distribution. To demonstrate this, we first calculate the mean of $E_\phi(\mathbf{x})$ for all data points labelled as having glasses, and the mean for all data points without, and substract the latter from the former. We can then add this vector to the encoding of an image without glasses, and interpolate between the original and new encodings. Results are shown in Figure 7.

Interpolating between two different images, on the other hand, is best done in $\mathcal{Z}_0$, as the interpolation is more likely to occur along a path of higher average density, resulting in more probable images. A visualization of the difference in paths taken can be seen in Figure 5. If the two images are very different semantically, then interpolation in $\mathcal{Z}_T$ can generate highly improbable images, see Figure 8.

6 DIVERSITY IN IMAGE-TO-IMAGE TRANSLATION

Achieving high sample diversity in multi-modal image-to-image translation tasks is often an explicit goal Zhu et al. (2017); Huang et al. (2018c). When using conditional VAEs for image-to-image translation tasks, the decoder is often able to learn a fairly accurate reconstruction based on the conditioned image alone, and so may ignore the latent code entirely if the KL divergence weight is too strong. Even when methods such as annealing are used, the model may still only make limited use of the latent code, severely affecting the diversity of generated samples. In order to quantitatively test whether our proposed method is able improve diversity, we experiment with the Variational U-net model proposed in Esser et al. (2018). In their work, they attempt to learn the distribution over images of people conditioned on their pose. We modified their implementation such that the prior distribution is learned via normalizing flow, and dropped the KL divergence term in the objective.

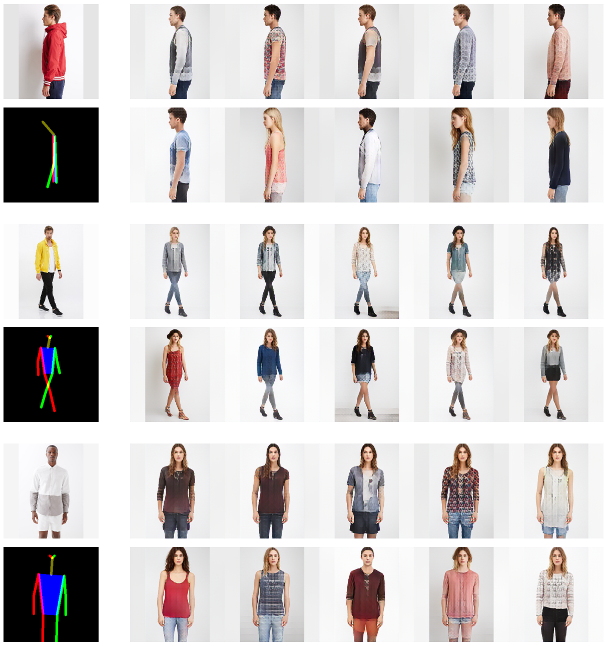


Figure 9: Conditional samples using Variational U-net. Top row: original. Bottom row: proposed. Leftmost column contains the original image and the pose being conditioned on.

	IS	LPIPS
VUNET+VAE	2.63	0.184
VUNET+Proposed	2.70	0.236

Table 2: Inception and LPIPS scores (higher is better) on the DeepFashion dataset after training Variational U-net.

Their model conditions the prior distribution on the given pose such that their objective becomes

$$\log p(\mathbf{x}^{(i)}|\mathbf{y}^{(i)}, \mathbf{z}) - D_{KL}[q(\mathbf{z}|\mathbf{x}^{(i)}, \mathbf{y}^{(i)})||p(\mathbf{z}|\mathbf{y}^{(i)})] \tag{9}$$

where $\mathbf{y}$ is the pose and $\mathbf{x}$ is the real image. For compatibility we therefore learn the mean of $\mathbf{z_0}$ conditioned on the pose and additionally condition each flow transformation f_t on the pose. To measure diversity, we compute the LPIPS distance Zhang et al. (2018) between randomly sampled pairs which were generated by conditioning on the images in the test set. We additionally calculated the Inception score Salimans et al. (2016) of the samples to ensure image quality was not affected. Results comparing our modification with the original implementation after training on the DeepFashion dataset are reported in Table 2. We also show samples in Figure 9.

7 Conclusion

We have proposed eliminating the KL divergence term from the objective in a VAE while simultaneously learning the prior via a normalizing flow, and demonstrated empirically that this results in better sample quality as measured by FID scores, better disentanglement as measured by linear separability, and greater conditional sample diversity as measured by LPIPS. Our results indicate that fixed-form priors over-regularize the model, and should be eschewed in favour of learned priors. Furthermore, regularization of the encoder using the prior, even when the prior is learned, is typically not beneficial. It is our hope that these results may be used to better inform the application of the Variational Autoencoder and its derivatives.

REFERENCES

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. In *International Conference on Machine Learning (ICML)*, 2017.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. Generating sentences from a continuous space. In *The SIGNLL Conference on Computational Natural Language Learning (CoNLL)*, 2016.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

Xi Chen, Diederik P. Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel. Variational lossy autoencoder. In *International Conference on Learning Representations (ICLR)*, 2017.

Bin Dai and David Wipf. Diagnosing and enhancing VAE models. In *International Conference on Learning Representations (ICLR)*, 2019.

Laurent Dinh, David Krueger, and Yoshua Bengio. NICE: non-linear independent components estimation. In *International Conference on Learning Representations (ICLR)*, 2015.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *International Conference on Learning Representations (ICLR)*, 2017.

Patrick Esser, Ekaterina Sutter, and Björn Ommer. A variational U-net for conditional appearance and shape generation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

Xianxu Hou, LinLin Shen, Ke Sun, and Guoping Qiu. Deep feature consistent variational autoencoder. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017.

Chin-Wei Huang, Ahmed Touati, Laurent Dinh, Michal Drozdzal, Mohammad Havaei, Laurent Charlin, and Aaron C. Courville. Learnable explicit density for continuous latent space and variational inference. In *International Conference on Machine Learning (ICML) Workshops*, 2017.

Chin-Wei Huang, David Krueger, Alexandre Lacoste, and Aaron C. Courville. Neural autoregressive flows. In *International Conference on Machine Learning (ICML)*, 2018a.

Chin-Wei Huang, Shawn Tan, Alexandre Lacoste, and Aaron C. Courville. Improving explorability in variational inference with annealed variational objectives. In *Advances in Neural Information Processing Systems (NIPS)*, 2018b.

Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *European Conference on Computer Vision (ECCV)*, 2018c.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, 2016.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations (ICLR)*, 2018a.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *ArXiv:1812.04948*, 2018b.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

Diederik P. Kingma, Tim Salimans, and Max Welling. Improving variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.

Chun-Liang Li, Wei-Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. MMD GAN: towards deeper understanding of moment matching network. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

Yujia Li, Kevin Swersky, and Richard S. Zemel. Generative moment matching networks. In *International Conference on Machine Learning (ICML)*, 2015.

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

Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs Created Equal? A Large-Scale Study. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.

Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian J. Goodfellow. Adversarial autoencoders. In *International Conference on Learning Representations (ICLR)*, 2016.

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 (ICLR)*, 2017.

Lars M. Mescheder. On the convergence properties of GAN training. *ArXiv:1801.04406*, 2018.

Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2017.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked Autoregressive Flow for Density Estimation. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

Danilo Jimenez Rezende and Fabio Viola. Taming VAEs. *ArXiv:1810.00597*, 2018.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein Auto-Encoders. In *International Conference on Learning Representations (ICLR)*, 2018.

Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

Tom White. Sampling generative networks: Notes on a few effective techniques. *ArXiv:1609.04468*, 2016.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Shengjia Zhao, Jiaming Song, and Stefano Ermon. Towards deeper understanding of variational autoencoding models. *ArXiv:1702.08658*, 2017.

Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A. Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

A Appendix

You may include other additional sections here.