

---

# Disentangling Disentanglement in Variational Autoencoders

---

Emile Mathieu<sup>\*1</sup> Tom Rainforth<sup>\*1</sup> N. Siddharth<sup>\*2</sup> Yee Whye Teh<sup>1</sup>

## Abstract

We develop a generalisation of disentanglement in variational autoencoders (VAEs)—*decomposition* of the latent representation—characterising it as the fulfilment of two factors: a) the latent encodings of the data having an appropriate level of overlap, and b) the aggregate encoding of the data conforming to a desired structure, represented through the prior. Decomposition permits disentanglement, i.e. explicit independence between latents, as a special case, but also allows for a much richer class of properties to be imposed on the learnt representation, such as sparsity, clustering, independent subspaces, or even intricate hierarchical dependency relationships. We show that the  $\beta$ -VAE varies from the standard VAE predominantly in its control of latent overlap and that for the standard choice of an isotropic Gaussian prior, its objective is invariant to rotations of the latent representation. Viewed from the decomposition perspective, breaking this invariance with simple manipulations of the prior can yield better disentanglement with little or no detriment to reconstructions. We further demonstrate how other choices of prior can assist in producing different decompositions and introduce an alternative training objective that allows the control of both decomposition factors in a principled manner.

## 1. Introduction

An oft-stated motivation for learning disentangled representations of data with deep generative models is a desire to achieve interpretability (Bengio et al., 2013; Chen et al., 2017)—particularly the *decomposability* (see §3.2.1 in Lip-ton, 2016) of latent representations to admit intuitive explanations. Most work has focused on capturing purely

*independent* factors of variation (Alemi et al., 2017; Ansari and Soh, 2019; Burgess et al., 2018; Chen et al., 2018; 2017; Eastwood and Williams, 2018; Esmaeili et al., 2019; Higgins et al., 2016; Kim and Mnih, 2018; Xu and Durrett, 2018; Zhao et al., 2017), typically evaluating this using purpose-built, synthetic data (Eastwood and Williams, 2018; Higgins et al., 2016; Kim and Mnih, 2018), whose generative factors are independent by construction.

This conventional view of disentanglement, as recovering independence, has subsequently motivated the development of formal evaluation metrics for independence (Eastwood and Williams, 2018; Kim and Mnih, 2018), which in turn has driven the development of objectives that target these metrics, often by employing regularisers explicitly encouraging independence in the representations (Eastwood and Williams, 2018; Esmaeili et al., 2019; Kim and Mnih, 2018).

We argue that such an approach is not generalisable, and potentially even harmful, to learning interpretable representations for more complicated problems, where such simplistic representations cannot accurately mimic the generation of high dimensional data from low dimensional latent spaces, and more richly structured dependencies are required.

We posit a generalisation of disentanglement in VAEs—*decomposing* their latent representations—that can help avoid such pitfalls. We characterise decomposition in VAEs as the fulfilment of two factors: a) the latent encodings of data having an appropriate level of overlap, and b) the aggregate encoding of data conforming to a desired structure, represented through the prior. We emphasize that neither of these factors is sufficient in isolation: without an appropriate level of overlap, encodings can degrade to a lookup table where the latents convey little information about data, and without the aggregate encoding of data following a desired structure, the encodings do not decompose as desired.

Disentanglement *implicitly* makes a choice of decomposition: that the latent features are independent of one another. We make this *explicit* and exploit it to both provide improvement to disentanglement through judicious choices of structure in the prior, and to introduce a more general framework flexible enough to capture alternate, more complex, notions of decomposition such as sparsity, clustering, hierarchical structuring, or independent subspaces.

<sup>\*</sup>Equal contribution <sup>1</sup>Department of Statistics <sup>2</sup>Department of Engineering, University of Oxford. Correspondence to: Emile Mathieu <emile.mathieu@stats.ox.ac.uk>, Tom Rainforth <rainforth@stats.ox.ac.uk>, N. Siddharth <nsid@robots.ox.ac.uk>.

To connect our framework with existing approaches for encouraging disentanglement, we provide a theoretical analysis of the  $\beta$ -VAE (Alemi et al., 2018; 2017; Higgins et al., 2016), and show that it typically only allows control of latent overlap, the first decomposition factor. We show that it can be interpreted, up to a constant offset, as the standard VAE objective with its prior annealed as  $p_\theta(\mathbf{z})^\beta$  and an additional maximum entropy regularization of the encoder that increases the stochasticity of the encodings. Specialising this result for the typical choice of a Gaussian encoder and isotropic Gaussian prior indicates that the  $\beta$ -VAE, up to a scaling of the latent space, is equivalent to the VAE plus a regulariser encouraging higher encoder variance. Moreover, this objective is invariant to rotations of the learned latent representation, meaning that it does not, on its own, encourage the latent variables to take on meaningful representations any more than an arbitrary rotation of them.

We confirm these results empirically, while further using our decomposition framework to show that simple manipulations to the prior can improve disentanglement, and other decompositions, with little or no detriment to the reconstruction accuracy. Further, motivated by our analysis, we propose an alternative objective that takes into account the distinct needs of the two factors of decomposition, and use it to learn clustered and sparse representations as demonstrations of alternative forms of decomposition. An implementation of our experiments and suggested methods is provided at <http://github.com/iffsid/disentangling-disentanglement>.

## 2. Background and Related Work

### 2.1. Variational Autoencoders

Let  $\mathbf{x}$  be an  $\mathcal{X}$ -valued random variable distributed according to an unknown generative process with density  $p_{\mathcal{D}}(\mathbf{x})$  and from which we have observations,  $X = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ . The aim is to learn a latent-variable model  $p_\theta(\mathbf{x}, \mathbf{z})$  that captures this generative process, comprising of a fixed<sup>1</sup> prior over latents  $p(\mathbf{z})$  and a parametric likelihood  $p_\theta(\mathbf{x}|\mathbf{z})$ . Learning proceeds by minimising a divergence between the true data generating distribution and the model w.r.t  $\theta$ , typically

$$\arg \min_{\theta} \text{KL}(p_{\mathcal{D}}(\mathbf{x}) \parallel p_\theta(\mathbf{x})) = \arg \max_{\theta} \mathbb{E}_{p_{\mathcal{D}}(\mathbf{x})} [\log p_\theta(\mathbf{x})]$$

where  $p_\theta(\mathbf{x}) = \int_{\mathcal{Z}} p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$  is the marginal likelihood, or evidence, of datapoint  $\mathbf{x}$  under the model, approximated by averaging over the observations.

However, estimating  $p_\theta(\mathbf{x})$  (or its gradients) to any sufficient degree of accuracy is typically infeasible. A common strategy to ameliorate this issue involves the introduction of a parametric inference model  $q_\phi(\mathbf{z}|\mathbf{x})$  to construct a varia-

tional evidence lower bound (ELBO) on  $\log p_\theta(\mathbf{x})$  as follows

$$\begin{aligned} \mathcal{L}(\mathbf{x}; \theta, \phi) &\triangleq \log p_\theta(\mathbf{x}) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z}|\mathbf{x})) \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z})). \end{aligned} \quad (1)$$

A variational autoencoder (VAE) (Kingma and Welling, 2014; Rezende et al., 2014) views this objective from the perspective of a deep stochastic autoencoder, taking the inference model  $q_\phi(\mathbf{z}|\mathbf{x})$  to be an encoder and the likelihood model  $p_\theta(\mathbf{x}|\mathbf{z})$  to be a decoder. Here  $\theta$  and  $\phi$  are neural network parameters, and learning happens via stochastic gradient ascent (SGA) using unbiased estimates of  $\nabla_{\theta, \phi} \frac{1}{n} \sum_{i=1}^n \mathcal{L}(\mathbf{x}_i; \theta, \phi)$ . Note that when clear from the context, we denote  $\mathcal{L}(\mathbf{x}; \theta, \phi)$  as simply  $\mathcal{L}(\mathbf{x})$ .

### 2.2. Disentanglement

Disentanglement, as typically employed in literature, refers to independence among features in a representation (Bengio et al., 2013; Eastwood and Williams, 2018; Higgins et al., 2018). Conceptually, however, it has a long history, far longer than we could reasonably do justice here, and is far from specific to VAEs. The idea stems back to traditional methods such as ICA (Hyvärinen and Oja, 2000; Yang and Amari, 1997) and conventional autoencoders (Schmidhuber, 1992), through to a range of modern approaches employing deep learning (Achille and Soatto, 2019; Chen et al., 2016; Cheung et al., 2014; Hjelm et al., 2019; Makhzani et al., 2015; Mathieu et al., 2016; Reed et al., 2014).

Of particular relevance to this work are approaches that explore disentanglement in the context of VAEs (Alemi et al., 2017; Chen et al., 2018; Esmaeili et al., 2019; Higgins et al., 2016; Kim and Mnih, 2018; Siddharth et al., 2017). Here one aims to achieve independence between the dimensions of the aggregate encoding, typically defined as  $q_\phi(\mathbf{z}) \triangleq \mathbb{E}_{p_{\mathcal{D}}(\mathbf{x})} [q(\mathbf{z}|\mathbf{x})] \approx \frac{1}{n} \sum_{i=1}^n q(\mathbf{z}|\mathbf{x}_i)$ . The significance of  $q_\phi(\mathbf{z})$  is that it is the marginal distribution induced on the latents by sampling a datapoint and then using the encoder to sample an encoding given that datapoint. It can thus informally be thought of as the pushforward distribution for “sampling” representations in the latent space.

Within the disentangled VAEs literature, there is also a distinction between unsupervised approaches, and semi-supervised approaches wherein one has access to the true generative factor values for some subset of data (Bouchacourt et al., 2018; Kingma et al., 2014; Siddharth et al., 2017). Our focus, however, is on the unsupervised setting.

Much of the prior work in the field has either implicitly or explicitly presumed a slightly more ambitious definition of disentanglement than considered above: that it is a measure of how well one captures *true* factors of variation (which happen to be independent by construction for synthetic data), rather than just independent factors. After all, if we wish

<sup>1</sup>Learning the prior is possible, but omitted for simplicity.

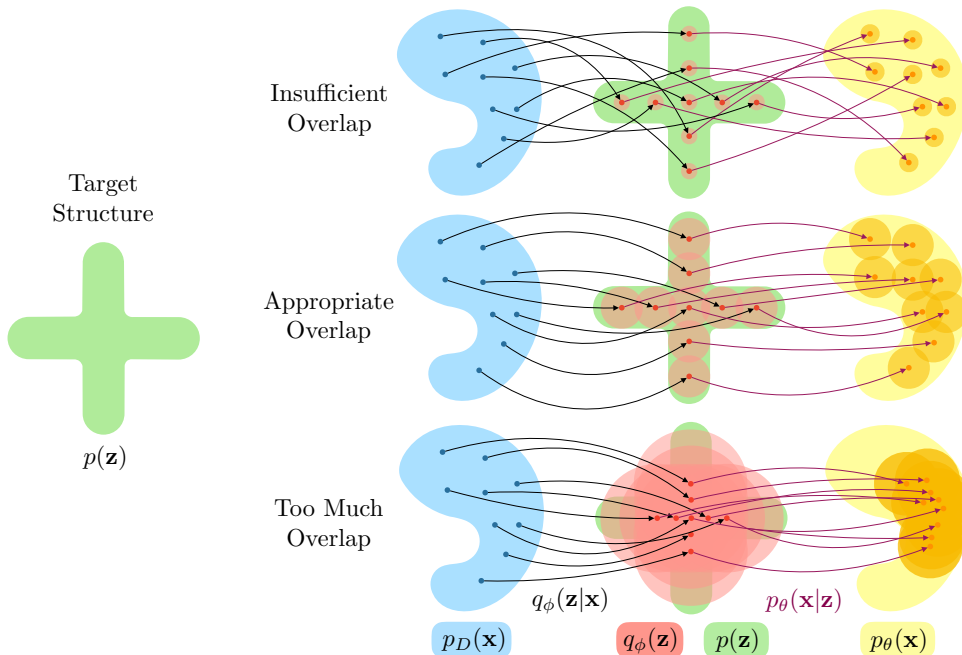


Figure 1. Illustration of decomposition where the desired structure is a cross shape (enforcing *sparsity*), expressed through the prior  $p(\mathbf{z})$  as shown on the left. In the scenario where there is insufficient overlap [top], we observe a lookup table behavior: points that are close in the data space are not close in the latent space and so the latent space loses meaning. In the scenario where there is too much overlap [bottom], the latent variable and observed datapoint convey little information about one another, such that the latent space again loses meaning. Note that if the distributional form of the latent distribution does not match that of the prior, as is the case here, this can also prevent the aggregate encoding matching the prior when the level of overlap is large.

for our learned representations to be interpretable, it is necessary for the latent variables to take on clear-cut meaning.

One such definition is given by Eastwood and Williams (2018), who define it as the extent to which a latent dimension  $d \in D$  in a representation predicts a true generative factor  $k \in K$ , with each latent capturing at most one generative factor. This implicitly assumes  $D \geq K$ , as otherwise the latents are unable to explain all the true generative factors. However, for real data, the association is more likely  $D \ll K$ , with one learning a low-dimensional abstraction of a complex process involving many factors. Consequently, such simplistic representations cannot, by definition, be found for more complex datasets that require more richly structured dependencies to be able to encode the information required to generate higher dimensional data. Moreover, for complex datasets involving a finite set of datapoints, it might not be reasonable to presume that one could capture the elements of the true generative process—the data itself might not contain sufficient information to recover these and even if it does, the computation required to achieve this through model learning is unlikely to be tractable.

The subsequent need for richly structured dependencies between latent dimensions has been reflected in the motivation for a handful of approaches (Bouchacourt et al., 2018; Esmaeili et al., 2019; Johnson et al., 2016; Siddharth

et al., 2017) that explore this through graphical models, although employing mutually-inconsistent, and not generalisable, interpretations of disentanglement. This motivates our development of a decomposition framework as a means of extending beyond the limitations of disentanglement.

### 3. Decomposition: A Generalisation of Disentanglement

The commonly assumed notion of disentanglement is quite restrictive for complex models where the true generative factors are not independent, very large in number, or where it cannot be reasonably assumed that there is a well-defined set of “true” generative factors (as will be the case for many, if not most, real datasets). To this end, we introduce a generalization of disentanglement, *decomposition*, which at a high-level can be thought of as imposing a desired structure on the learned representations. This permits disentanglement as a special case, for which the desired structure is that  $q_\phi(\mathbf{z})$  factors along its dimensions.

We characterise the decomposition of latent spaces in VAEs to be the fulfilment of two factors (as shown in Figure 1):

- An “appropriate” level of overlap in the latent space—ensuring that the range of latent values capable of encod-

ing a particular datapoint is neither too small, nor too large. This is, in general, dictated by the level of stochasticity in the encoder: the noisier the encoding process is, the higher the number of datapoints which can plausibly give rise to a particular encoding.

- b. The aggregate encoding  $q_\phi(\mathbf{z})$  matching the prior  $p(\mathbf{z})$ , where the latter expresses the desired dependency structure between latents.

The overlap factor (a) is perhaps best understood by considering extremes—too little, and the latents effectively become a lookup table; too much, and the data and latents do not convey information about each other. In either case, meaningfulness of the latent encodings is lost. Thus, without the *appropriate* level of overlap—dictated both by noise in the true generative process and dataset size—it is not possible to enforce meaningful structure on the latent space. Though quantitatively formalising overlap in general scenarios is surprisingly challenging (c.f. § 7 and Appendix D), we note for now that when the encoder distribution is unimodal, it is typically well-characterized by the mutual information between the data and the latents  $I(\mathbf{x}; \mathbf{z})$ .

The regularisation factor (b) enforces a congruence between the (aggregate) latent embeddings of data and the dependency structures expressed in the prior. We posit that such structure is best expressed in the prior, as opposed to explicit independence regularisation of the marginal posterior (Chen et al., 2018; Kim and Mnih, 2018), to enable the *generative* model to express the desired decomposition, and to avoid potentially violating self-consistency between the encoder, decoder, and true data generating distributions. The prior also provides a rich and flexible means of expressing desired structure by defining a generative process that encapsulates dependencies between variables, as with a graphical model.

Critically, *neither factor is sufficient in isolation*. An inappropriate level of overlap in the latent space will impede interpretability, irrespective of quality of regularisation, as the latent space need not be meaningful. Conversely, without the pressure to regularise to the prior, the latent space is under no constraint to exhibit the desired structure.

Decomposition is inherently subjective as we must choose the structure of the prior we regularise to depending on how we intend to use our learned model or what kind of features we would like to uncover from the data. This may at first seem unsatisfactory compared to the seemingly objective adjustments often made to the ELBO by disentanglement methods. However, disentanglement *is itself* a subjective choice for the decomposition. We can embrace this subjective nature through judicious choices of the prior distribution; ignoring this imposes unintended assumptions which can have unwanted effects. For example, as we will later show, the rotational invariance of the standard prior  $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, I)$  can actually hinder disentanglement.

## 4. Deconstructing the $\beta$ -VAE

To connect existing approaches to our proposed framework, we now consider, as a case study, the  $\beta$ -VAE (Higgins et al., 2016)—an adaptation of the VAE objective (ELBO) to learn better-disentangled representations. Specifically, it scales the KL term in the standard ELBO by a factor  $\beta > 0$  as

$$\mathcal{L}_\beta(\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \beta \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}) || p(\mathbf{z})). \quad (2)$$

Hoffman et al. (2017) showed that the  $\beta$ -VAE target can be viewed as a standard ELBO with the alternative prior  $r(\mathbf{z}) \propto q_\phi(\mathbf{z})^{(1-\beta)} p(\mathbf{z})^\beta$ , along with terms involving the mutual information and the prior’s normalising constant.

We now introduce an alternate deconstruction as follows

**Theorem 1.** *The  $\beta$ -VAE target  $\mathcal{L}_\beta(\mathbf{x})$  can be interpreted in terms of the standard ELBO,  $\mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi)$ , for an adjusted target  $\pi_{\theta, \beta}(\mathbf{x}, \mathbf{z}) \triangleq p_\theta(\mathbf{x} | \mathbf{z}) f_\beta(\mathbf{z})$  with annealed prior  $f_\beta(\mathbf{z}) \triangleq p(\mathbf{z})^\beta / F_\beta$  as*

$$\mathcal{L}_\beta(\mathbf{x}) = \mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi) + (\beta - 1)H_{q_\phi} + \log F_\beta \quad (3)$$

where  $F_\beta \triangleq \int_{\mathbf{z}} p(\mathbf{z})^\beta d\mathbf{z}$  is constant given  $\beta$ , and  $H_{q_\phi}$  is the entropy of  $q_\phi(\mathbf{z} | \mathbf{x})$ .

*Proof.* All proofs are given in Appendix A. □

Clearly, the second term in (3), enforcing a maximum entropy regulariser on the posterior  $q_\phi(\mathbf{z} | \mathbf{x})$ , allows the value of  $\beta$  to affect the overlap of encodings in the latent space. We thus see that it provides a means of controlling decomposition factor (a). However, it is itself not sufficient to enforce disentanglement. For example, the entropy of  $q_\phi(\mathbf{z} | \mathbf{x})$  is independent of its mean  $\mu_\theta(\mathbf{x})$  and is independent to rotations of  $\mathbf{z}$ , so it is clearly incapable of discouraging certain representations with poor disentanglement. All the same, having the wrong level of regularization can, in turn, lead to an inappropriate level of overlap and undermine the ability to disentangle. Consequently, this term is still important.

Although the precise impact of prior annealing depends on the original form of the prior, the high-level effect is the same—larger values of  $\beta$  cause the effective latent space to collapse towards the modes of the prior. For uni-modal priors, the main effect of annealing is to reduce the scaling of  $\mathbf{z}$ ; indeed this is the only effect for generalized Gaussian distributions. While this would appear not to have any tangible effects, closer inspection suggests otherwise—it ensures that the scaling of the encodings matches that of the prior. Only incorporating the maximum-entropy regularisation will simply cause the scaling of the latent space to increase. The rescaling of the prior now cancels this effect, ensuring the scaling of  $q_\phi(\mathbf{z})$  matches that of  $p(\mathbf{z})$ .

Taken together, this implies that the  $\beta$ -VAE’s ability to encourage disentanglement is predominantly through *direct*

control over the level of overlap. It places no other direct constraint on the latents to disentangle (although in some cases, the annealed prior may inadvertently encourage better disentanglement), but instead helps avoid the pitfalls of inappropriate overlap. Amongst other things, this explains why large  $\beta$  is not universally beneficial for disentanglement, as the level of overlap can be increased too far.

#### 4.1. Special Case – Gaussians

We can gain further insights into the  $\beta$ -VAE in the common use case—assuming a Gaussian prior,  $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, \Sigma)$ , and Gaussian encoder,  $q_\phi(\mathbf{z} | \mathbf{x}) = \mathcal{N}(\mathbf{z}; \mu_\phi(\mathbf{x}), S_\phi(\mathbf{x}))$ . Here it is straightforward to see that annealing simply scales the latent space by  $1/\sqrt{\beta}$ , i.e.  $f_\beta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, \Sigma/\beta)$ . Given this, it is easy to see that a VAE trained with the adjusted target  $\mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi)$ , but appropriately scaling the latent space, will behave identically to one trained with the original target  $\mathcal{L}(\mathbf{x})$ . It will also have an identical ELBO as the expected reconstruction is trivially the same, while the KL between Gaussians is invariant to scaling both equally. More precisely, we have the following result.

**Corollary 1.** *If  $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, \Sigma)$  and  $q_\phi(\mathbf{z} | \mathbf{x}) = \mathcal{N}(\mathbf{z}; \mu_\phi(\mathbf{x}), S_\phi(\mathbf{x}))$ , then,*

$$\mathcal{L}_\beta(\mathbf{x}; \theta, \phi) = \mathcal{L}(\mathbf{x}; \theta', \phi') + \frac{(\beta - 1)}{2} \log |S_{\phi'}(\mathbf{x})| + c \quad (4)$$

where  $\theta'$  and  $\phi'$  represent rescaled networks such that

$$\begin{aligned} p_{\theta'}(\mathbf{x} | \mathbf{z}) &= p_\theta(\mathbf{x} | \mathbf{z}/\sqrt{\beta}), \\ q_{\phi'}(\mathbf{z} | \mathbf{x}) &= \mathcal{N}(\mathbf{z}; \mu_{\phi'}(\mathbf{x}), S_{\phi'}(\mathbf{x})), \\ \mu_{\phi'}(\mathbf{x}) &= \sqrt{\beta} \mu_\phi(\mathbf{x}), \quad S_{\phi'}(\mathbf{x}) = \beta S_\phi(\mathbf{x}), \end{aligned}$$

and  $c \triangleq \frac{D(\beta-1)}{2} \left(1 + \log \frac{2\pi}{\beta}\right) + \log F_\beta$  is a constant, with  $D$  denoting the dimensionality of  $\mathbf{z}$ .

Noting that as  $c$  is irrelevant to the training process, this indicates an equivalence, up to scaling of the latent space, between training with the  $\beta$ -VAE objective and a maximum-entropy regularised version of the standard ELBO

$$\mathcal{L}_{H, \beta}(\mathbf{x}) \triangleq \mathcal{L}(\mathbf{x}) + \frac{(\beta - 1)}{2} \log |S_\phi(\mathbf{x})|, \quad (5)$$

whenever  $p(\mathbf{z})$  and  $q_\phi(\mathbf{z} | \mathbf{x})$  are Gaussian. Note that we implicitly presume suitable adjustment of neural-network hyper-parameters and the stochastic gradient scheme to account for the change of scaling in the optimal networks.

Moreover, the stationary points for the two objectives  $\mathcal{L}_\beta(\mathbf{x}; \theta, \phi)$  and  $\mathcal{L}_{H, \beta}(\mathbf{x}; \theta', \phi')$  are equivalent (c.f. Corollary 2 in Appendix A), indicating that optimising for (5) leads to networks equivalent to those from optimising the  $\beta$ -VAE objective (2), up to scaling the encodings by a factor of

$\sqrt{\beta}$ . Under the isotropic Gaussian prior setting, we further have the following result showing that the  $\beta$ -VAE objective is invariant to rotations of the latent space.

**Theorem 2.** *If  $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, \sigma I)$  and  $q_\phi(\mathbf{z} | \mathbf{x}) = \mathcal{N}(\mathbf{z}; \mu_\phi(\mathbf{x}), S_\phi(\mathbf{x}))$ , then for all rotation matrices  $R$ ,*

$$\mathcal{L}_\beta(\mathbf{x}; \theta, \phi) = \mathcal{L}_\beta(\mathbf{x}; \theta^\dagger(R), \phi^\dagger(R)) \quad (6)$$

where  $\theta^\dagger(R)$  and  $\phi^\dagger(R)$  are transformed networks such that

$$\begin{aligned} p_{\theta^\dagger}(\mathbf{x} | \mathbf{z}) &= p_\theta(\mathbf{x} | R^T \mathbf{z}), \\ q_{\phi^\dagger}(\mathbf{z} | \mathbf{x}) &= \mathcal{N}(\mathbf{z}; R \mu_\phi(\mathbf{x}), R S_\phi(\mathbf{x}) R^T). \end{aligned}$$

This shows that the  $\beta$ -VAE objective does not directly encourage latent variables to take on meaningful representations when using the standard choice of an isotropic Gaussian prior. In fact, on its own, it encourages latent representations which match the true generative factors no more than it encourages *any arbitrary rotation* of these factors, with such rotations capable of exhibiting strong correlations between latents. This view is further supported by our empirical results (see Figure 2), where we did not observe any gains in disentanglement (using the metric from Kim and Mnih (2018)) from increasing  $\beta > 0$  with an isotropic Gaussian prior trained on the *2D Shapes* dataset (Matthey et al., 2017). It may also go some way to explaining the extremely high levels of variation we found in the disentanglement-metric scores between different random seeds at train time.

It should be noted, however, that the value of  $\beta$  can indirectly influence the level of disentanglement when using a mean-field assumption for the encoder distribution (i.e. restricting  $S_\phi(\mathbf{x})$  to be diagonal). As noted by Rolinek et al. (2018); Stühmer et al. (2019), increasing  $\beta$  can reinforce existing inductive biases, wherein mean-field assumptions encourage representations which reduce dependence between the latent dimensions (Turner and Sahani, 2011).

## 5. An Objective for Enforcing Decomposition

Given the characterisation set out above, we now develop an objective that incorporates the effect of both factors (a) and (b). Our analysis of the  $\beta$ -VAE tells us that its objective allows direct control over the level of overlap, i.e. factor (a). To incorporate direct control over the regularisation (b) between the marginal posterior and the prior, we add a divergence term  $\mathbb{D}(q_\phi(\mathbf{z}), p(\mathbf{z}))$ , yielding

$$\begin{aligned} \mathcal{L}_{\alpha, \beta}(\mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x})} [\log p_\theta(\mathbf{x} | \mathbf{z})] \\ &\quad - \beta \text{KL}(q_\phi(\mathbf{z} | \mathbf{x}) \| p(\mathbf{z})) - \alpha \mathbb{D}(q_\phi(\mathbf{z}), p(\mathbf{z})) \end{aligned} \quad (7)$$

allowing control over how much factors (a) and (b) are enforced, through appropriate setting of  $\beta$  and  $\alpha$  respectively.

Note that such an additional term has been previously considered by Kumar et al. (2017), with  $\mathbb{D}(q_\phi(\mathbf{z}), p(\mathbf{z})) =$

$\text{KL}(q_\phi(\mathbf{z}) \parallel p(\mathbf{z}))$ , although for the sake of tractability they rely instead on moment matching using covariances. There have also been a number of approaches that decompose the standard VAE objective in different ways (e.g. Dilokthanakul et al., 2019; Esmaeili et al., 2019; Hoffman and Johnson, 2016) to expose  $\text{KL}(q_\phi(\mathbf{z}) \parallel p(\mathbf{z}))$  as a component, but, as we discuss in Appendix C, this can be difficult to compute correctly in practice, with common approaches leading to highly biased estimates whose practical behaviour is very different than the divergence they are estimating, unless very large batch sizes are used.

Wasserstein Auto-Encoders (Tolstikhin et al., 2018) formulate an objective that includes a general divergence term between the prior and marginal posterior, computed using either maximum mean discrepancy (MMD) or a variational formulation of the Jensen-Shannon divergence (a.k.a GAN loss). However, we find that empirically, choosing the MMD’s kernel and numerically stabilising its U-statistics estimator to be tricky, and designing and learning a GAN to be cumbersome and unstable. Consequently, the problems of choosing an appropriate  $\mathbb{D}(q_\phi(\mathbf{z}), p(\mathbf{z}))$  and generating reliable estimates for this choice are tightly coupled, with a general purpose solution remaining an important open problem; see further discussion in Appendix C.

## 6. Experiments

### 6.1. Prior for Axis-Aligned Disentanglement

We first show how subtle changes to the prior distribution can yield improvements in disentanglement. The standard choice of an isotropic Gaussian has previously been justified by the correct assertion that the latents are independent under the prior (Higgins et al., 2016). However, as explained in § 4.1, the rotational invariance of this prior means that it does not directly encourage axis-aligned representations. Priors that break this rotational invariance should be better suited for learning disentangled representations. We assess this hypothesis by training a  $\beta$ -VAE (i.e. (7) with  $\alpha = 0$ ) on the 2D *Shapes* dataset (Matthey et al., 2017) and evaluating disentanglement using the metric of Kim and Mnih (2018).

Figure 2 demonstrates that notable improvements in disentanglement can be achieved by using non-isotropic priors: for a given reconstruction loss, implicitly fixed by  $\beta$ , non-isotropic Gaussian priors got better disentanglement scores, with further improvement achieved when the prior variance is learnt. With a product of Student-t priors  $p_\nu(\mathbf{z})$  (noting  $p_\nu(\mathbf{z}) \rightarrow \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$  as  $\nu \rightarrow \infty$ ), reducing  $\nu$  only incurred a minor reconstruction penalty, for improved disentanglement. Interestingly, very low values of  $\nu$  caused the disentanglement score to drop again (though still giving higher values than the Gaussian). We speculate that this may be related to the effect of heavy tails on the disentanglement metric itself,

rather than being an objectively worse disentanglement. Another interesting result was that for an isotropic Gaussian prior, as per the original  $\beta$ -VAE setup, no gains at all were achieved in disentanglement by increasing  $\beta$ .

### 6.2. Clustered Prior

We next consider an alternative decomposition one might wish to impose—*clustering* of the latent space. For this, we use the “pinwheels” dataset from (Johnson et al., 2016) and a mixture of four equally-weighted Gaussians as our prior. We then conduct an ablation study to observe the effect of varying  $\alpha$  and  $\beta$  in  $\mathcal{L}_{\alpha,\beta}(\mathbf{x})$  (as per (7)) on the learned representations, taking the divergence to be  $\text{KL}(p(\mathbf{z}) \parallel q_\phi(\mathbf{z}))$  (see Appendix B for details).

We see in Figure 3 that increasing  $\beta$  increases the level of overlap in  $q_\phi(\mathbf{z})$ , as a consequence of increasing the encoder variance for individual datapoints. When  $\beta$  is too large, the encoding of a datapoint loses meaning. Also, as a single datapoint encodes to a Gaussian distribution,  $q_\phi(\mathbf{z}|\mathbf{x})$  is unable to match  $p(\mathbf{z})$  exactly. Because  $q_\phi(\mathbf{z}|\mathbf{x}) \rightarrow q_\phi(\mathbf{z})$  when  $\beta \rightarrow \infty$ , this in turn means that overly large values of  $\beta$  actually cause a mismatch between  $q_\phi(\mathbf{z})$  and  $p(\mathbf{z})$  (see top right of Figure 3). Increasing  $\alpha$ , instead always improved the match between  $q_\phi(\mathbf{z})$  and  $p(\mathbf{z})$ . Here, the finiteness of the dataset and the choice of divergence results in an increase in overlap with increasing  $\alpha$ , but only up to the level required for a non-negligible overlap between the nearby datapoints: large values of  $\alpha$  did not cause the encodings to collapse to a mode.

### 6.3. Prior for Sparsity

Finally, we consider a commonly desired decomposition—sparsity, which stipulates that only a small fraction of available factors are employed. That is, a *sparse representation* (Olshausen and Field, 1996) can be thought of as one where each embedding has a significant proportion of its dimensions *off*, i.e. close to 0. Sparsity has often been considered for feature-learning (Coates and Ng, 2011; Larochelle and Bengio, 2008) and employed in the probabilistic modelling literature (Lee et al., 2007; Ranzato et al., 2007).

Common ways to achieve sparsity are through a specific penalty (e.g.  $l_1$ ) or a careful choice of prior (peaked at 0). Concomitant with our overarching desire to encode requisite structure in the prior, we adopt the latter, constructing a sparse prior as  $p(\mathbf{z}) = \prod_d (1 - \gamma) \mathcal{N}(z_d; 0, 1) + \gamma \mathcal{N}(z_d; 0, \sigma_0^2)$  with  $\sigma_0^2 = 0.05$ . This mixture distribution can be interpreted as a mixture of samples being either *off* or *on*, whose proportion is set by the weight parameter  $\gamma$ . We use this prior to learn a VAE for the *Fashion-MNIST* dataset (Xiao et al., 2017) using the objective  $\mathcal{L}_{\alpha,\beta}(\mathbf{x})$  (as per (7)), taking the divergence to be an MMD with a kernel that only considers difference between the marginal distri-

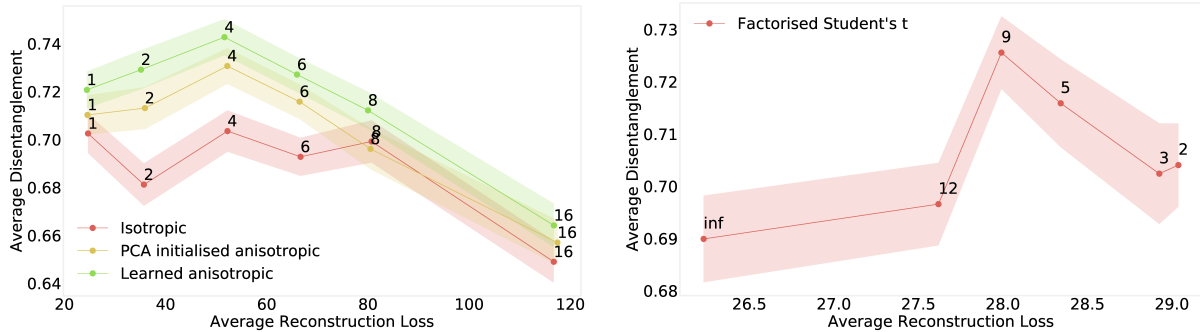


Figure 2. Reconstruction loss vs disentanglement metric of Kim and Mnih (2018). [Left] Using an anisotropic Gaussian with diagonal covariance either learned, or fixed to principal-component values of the dataset. Point labels represent different values of  $\beta$ . [Right] Using  $p_\nu(\mathbf{z}) = \prod_d \text{STUDENT-T}(z_d; \nu)$  for different  $\nu$  with  $\beta = 1$ . Note the different x-axis scaling. Shaded areas represent  $\pm 2$  standard errors for estimated mean disentanglement calculated using 100 separately trained networks. We thus see that the variability on the disentanglement metric is very large, presumably because of stochasticity in whether learned dimensions correspond to true generative factors. The variability in the reconstruction was only negligible and so is not shown. See Appendix B for full experimental details.

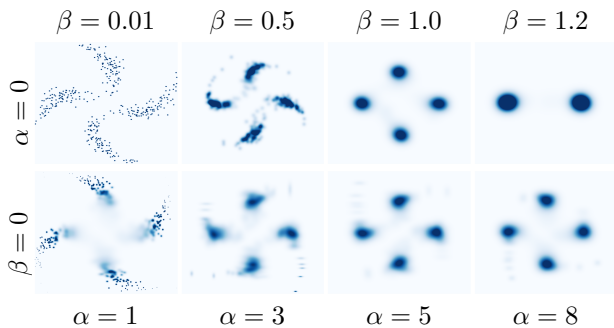


Figure 3. Density of aggregate posterior  $q_\phi(\mathbf{z})$  with different  $\alpha, \beta$  for spirals dataset with a mixture of Gaussian prior.

butions (see Appendix B for details).

We measure a representation’s sparsity using the Hoyer extrinsic metric (Hurley and Rickard, 2008). For  $\mathbf{y} \in \mathbb{R}^d$ ,

$$\text{Hoyer}(\mathbf{y}) = \frac{\sqrt{d} - \|\mathbf{y}\|_1 / \|\mathbf{y}\|_2}{\sqrt{d} - 1} \in [0, 1],$$

yielding 0 for a fully dense vector and 1 for a fully sparse vector. Rather than employing this metric directly to the mean encoding of each datapoint, we first normalise each dimension to have a standard deviation of 1 under its aggregate distribution, i.e. we use  $\bar{z}_d = z_d / \sigma(z_d)$  where  $\sigma(z_d)$  is the standard deviation of dimension  $d$  of the latent encoding taken over the dataset. This normalisation is important as one could achieve a “sparse” representation simply by having different dimensions vary along different length scales (something the  $\beta$ -VAE encourages through its pruning of dimensions (Stühmer et al., 2019)), whereas we desire a representation where different datapoints “activate” different features. We then compute overall sparsity by averaging over the dataset as  $\text{Sparsity} = \frac{1}{n} \sum_i^n \text{Hoyer}(\bar{z}_i)$ . Figure 4

(left) shows that substantial sparsity can be gained by replacing a Gaussian prior ( $\gamma = 0$ ) by a sparse prior ( $\gamma = 0.8$ ). It further shows substantial gains from the inclusion of the aggregate posterior regularization, with  $\alpha = 0$  giving far low sparsity than  $\alpha > 0$ , when using our sparse prior. The use of our sparse prior did not generally harm the reconstruction compared. Large values of  $\alpha$  did slightly worsen the reconstruction, but this drop-off was much slower than increases in  $\beta$  (note that  $\alpha$  is increased to much higher levels than  $\beta$ ). Interestingly, we see that  $\beta$  being either too low or too high also harmed the sparsity.

We explore the qualitative effects of sparsity in Figure 5, using a network trained with  $\alpha = 1000, \beta = 1$ , and  $\gamma = 0.8$ , corresponding to one of the models in Figure 4 (left). The top plot shows the average encoding magnitude for data corresponding to 3 of the 10 classes in the Fashion-MNIST dataset. It clearly shows that the different classes (trousers, dress, and shirt) predominantly encode information along different sets of dimensions, as expected for sparse representations (c.f. Appendix B for plots for all classes). For each of these classes, we explore the latent space along a particular ‘active’ dimension—one with high average encoding magnitude—to observe if they capture meaningful features in the image. We first identify a suitable ‘active’ dimension for a given instance (top row) from the dimension-wise magnitudes of its encoding, by choosing one, say  $d$ , where the magnitude far exceeds  $\sigma_0^2$ . Given encoding value  $z_d$ , we then interpolate along this dimension (keeping all others fixed) in the range  $(z_d, z_d + \text{sign}(z_d))$ ; the sign of  $z_d$  indicating the direction of interpolation. Exploring the latent space in such a manner demonstrates a variety of consistent feature transformations in the image, both within class (a, b, c), and across classes (d), indicating that these sparse dimensions do capture meaningful features in the image.

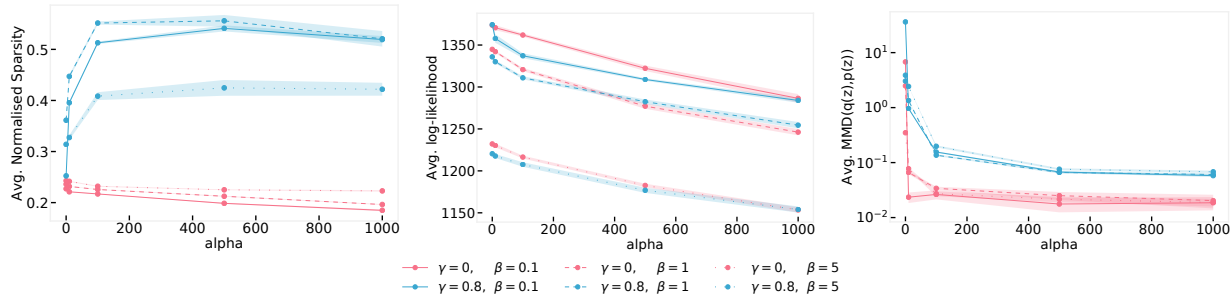


Figure 4. [Left] Sparsity vs regularisation strength  $\alpha$  (c.f. (7), high better). [Center] Average reconstruction log-likelihood  $\mathbb{E}_{p_D(\mathbf{x})}[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})]]$  vs  $\alpha$  (higher better). [Right] Divergence (MMD) vs  $\alpha$  (lower better). Note here that the different values of  $\gamma$  represent regularizations to different distributions, with regularization to a Gaussian (i.e.  $\gamma = 0$ ) much easier to achieve than the sparse prior, hence the lower divergence. Shaded areas represent  $\pm 2$  standard errors in the mean estimate calculated using 8 separately trained networks. See Appendix B for full experimental details.

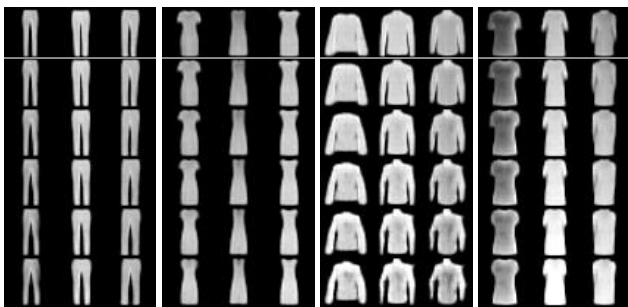
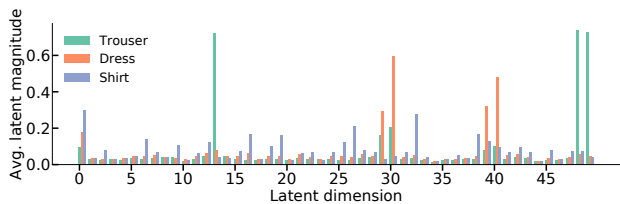


Figure 5. Qualitative evaluation of sparsity. [Top] Average encoding magnitude over data for three example classes in *Fashion-MNIST*. [Bottom] Latent interpolation ( $\downarrow$ ) for different datapoints (top layer) along particular ‘active’ dimensions. (a) Separation between the trouser legs (dim 49). (b) Top/Collar width of dresses (dim 30). (c) Shirt shape (loose/fitted, dim 19). (d) Style of sleeves across different classes—t-shirt, dress, and coat (dim 40).

Concurrent to our work, Tonolini et al. (2019) also considered imposing sparsity in VAEs with a spike-slab prior (such that  $\sigma_0 \rightarrow 0$ ). In contrast to our work, they do not impose a constraint on the aggregate encoder, nor do they evaluate their results with a quantitative sparsity metric that accounts for the varying length scales of different latent dimensions

## 7. Discussion

**Characterising Overlap** Precisely formalising what constitutes the level of overlap in the latent space is surprisingly

subtle. Prior work has typically instead considered controlling the level of compression through the mutual information between data and latents  $I(\mathbf{x}; \mathbf{z})$  (Alemi et al., 2018; 2017; Hoffman and Johnson, 2016; Phuong et al., 2018), with, for example, (Phuong et al., 2018) going on to discuss how controlling the compression can “explicitly encourage useful representations.” Although  $I(\mathbf{x}; \mathbf{z})$  provides a perfectly serviceable characterisation of overlap in a number of cases, the two are not universally equivalent and we argue that it is the latter which is important in achieving useful representations. In particular, if the form of the encoding distribution is not fixed—as when employing normalising flows, for example— $I(\mathbf{x}; \mathbf{z})$  does not necessarily characterise overlap well. We discuss this in greater detail in Appendix D.

However, when the encoder is unimodal with fixed form (in particular the tail behaviour is fixed) and the prior is well-characterised by Euclidean distances, then these factors have a substantially reduced ability to vary for a given  $I(\mathbf{x}; \mathbf{z})$ , which subsequently becomes a good characterisation of the level of overlap. When  $q_\phi(\mathbf{z}|\mathbf{x})$  is Gaussian, controlling the variance of  $q_\phi(\mathbf{z}|\mathbf{x})$  (with a fixed  $q_\phi(\mathbf{z})$ ) should similarly provide an effective means of achieving the desired overlap behaviour. As this is the most common use case, we leave the development of more a general definition of overlap to future work, simply noting that this is an important consideration when using flexible encoder distributions.

**Can VAEs Uncover True Generative Factors?** In concurrently published work, Locatello et al. (2019) question the plausibility of learning unsupervised disentangled representations with meaningful features, based on theoretical analyses showing an equivalence class of generative models, many members of which could be entangled. Though their analysis is sound, we posit a counterargument to their conclusions, based on the *stochastic* nature of the encodings used during training. Namely, that this stochasticity means that they need not give rise to the same ELBO scores (an



important exception is the rotational invariance for isotropic Gaussian priors). Essentially, the encoding noise forces nearby encodings to relate to similar datapoints, while standard choices for the likelihood distribution (e.g. assuming conditional independence) ensure that information is stored in the encodings, not just in the generative network. These restrictions mean that the ELBO prefers smooth representations and, provided the prior is not rotationally invariant, means that there no longer need be a class of different representations with the same ELBO; simpler representations are preferred to more complex ones.

The exact form of the encoding distribution is also important here. For example, imagine we restrict the encoder variance to be isotropic and then use a two dimensional prior where one latent dimension has a much larger variance than the other. It will be possible to store more information in the prior dimension with higher variance (as we can spread points out more relative to the encoder variance). Consequently, that dimension is more likely to correspond to an important factor of the generative process than the other. Of course, this does not imply that this is a true factor of variation in the generative process, but neither is the meaning that can be attributed to each dimension completely arbitrary.

All the same, we agree that an important area for future work is to assess when, and to what extent, one might expect learned representations to mimic the true generative process, and, critically, when it should not. For this reason, we actively avoid including any notion of a true generative process in our definition of decomposition, but note that, analogously to disentanglement, it permits such extension in scenarios where doing so can be shown to be appropriate.

## 8. Conclusions

In this work, we explored and analysed the fundamental characteristics of learning disentangled representations, and showed how these can be generalised to a more general framework of *decomposition* (Lipton, 2016). We characterised the learning of decomposed latent representation with VAEs in terms of the control of two factors: i) overlap in the latent space between encodings of different datapoints, and ii) regularisation of the aggregate encoding distribution to the given prior, which encodes the structure one would wish for the latent space to have.

Connecting prior work on disentanglement to this framework, we analysed the  $\beta$ -VAE objective to show that its contribution to disentangling is primarily through direct control of the level of overlap between encodings of the data, expressed by maximising the entropy of the encoding distribution. In the commonly encountered case of assuming an isotropic Gaussian prior and an independent Gaussian posterior, we showed that control of overlap is the *only* effect of the  $\beta$ -VAE. Motivated by this observation, we

developed an alternate objective for the ELBO that allows control of the two factors of decomposability through an additional regularisation term. We then conducted empirical evaluations using this objective, targeting alternate forms of decompositions such as clustering and sparsity, and observed the effect of varying the extent of regularisation to the prior on the quality of the resulting clustering and sparseness of the learnt embeddings. The results indicate that we were successful in attaining those decompositions.

## Acknowledgements

EM, TR, YWT were supported in part by the European Research Council under the European Union’s Seventh Framework Programme (FP7/2007–2013) / ERC grant agreement no. 617071. TR research leading to these results also received funding from EPSRC under grant EP/P026753/1. EM was also supported by Microsoft Research through its PhD Scholarship Programme. NS was funded by EPSRC/MURI grant EP/N019474/1.

## References

- Alessandro Achille and Stefano Soatto. Emergence of invariance and disentanglement in deep representations. *Journal of Machine Learning Research*, 19(50), 2019.
- Alexander Alemi, Ben Poole, Ian Fischer, Joshua Dillon, Rif A Saurous, and Kevin Murphy. Fixing a broken ELBO. In *International Conference on Machine Learning*, pages 159–168, 2018.
- Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations*, 2017.
- Abdul Fatir Ansari and Harold Soh. Hyperprior induced unsupervised disentanglement of latent representations. In *AAAI Conference on Artificial Intelligence*, 2019.
- Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8):1798–1828, August 2013. ISSN 0162-8828.
- Diane Bouchacourt, Ryota Tomioka, and Sebastian Nowozin. Multi-level variational autoencoder: Learning disentangled representations from grouped observations. In *AAAI Conference on Artificial Intelligence*, 2018.
- Christopher P. Burgess, Irina Higgins, Arka Pal, Loïc Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in  $\beta$ -vae. *CoRR*, abs/1804.03599, 2018.

- Ricky T. Q. Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, 2018.
- Xi Chen, Yan Duan, Rein Houthoofd, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2172–2180, 2016.
- Xi Chen, Diederik P Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel. Variational Lossy Autoencoder. 2017.
- Brian Cheung, Jesse A Livezey, Arjun K Bansal, and Bruno A Olshausen. Discovering hidden factors of variation in deep networks. *arXiv preprint arXiv:1412.6583*, 2014.
- Adam Coates and Andrew Y. Ng. The importance of encoding versus training with sparse coding and vector quantization. In Lise Getoor and Tobias Scheffer, editors, *ICML*, pages 921–928. Omnipress, 2011.
- Nat Dilokthanakul, Nick Pawlowski, and Murray Shanahan. Explicit information placement on latent variables using auxiliary generative modelling task, 2019. URL <https://openreview.net/forum?id=H1l-SjA5t7>.
- Justin Domke and Daniel Sheldon. Importance weighting and variational inference. In *Advances in Neural Information Processing Systems*, pages 4471–4480, 2018.
- Cian Eastwood and Christopher K. I. Williams. A framework for the quantitative evaluation of disentangled representations. In *International Conference on Learning Representations*, 2018.
- Babak Esmaeili, Hao Wu, Sarthak Jain, N Siddharth, Brooks Paige, and Jan-Willem van de Meent. Hierarchical Disentangled Representations. *Artificial Intelligence and Statistics*, 2019.
- Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *Proceedings of the International Conference on Learning Representations*, 2016.
- Irina Higgins, David Amos, David Pfau, Sebastien Racaniere, Loic Matthey, Danilo Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. *arXiv preprint arXiv:1812.02230*, 2018.
- R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*, 2019.
- Matthew D Hoffman and Matthew J Johnson. ELBO surgery: yet another way to carve up the variational evidence lower bound. In *Workshop on Advances in Approximate Bayesian Inference, NIPS*, pages 1–4, 2016.
- Matthew D Hoffman, Carlos Riquelme, and Matthew J Johnson. The  $\beta$ -VAE’s Implicit Prior. In *Workshop on Bayesian Deep Learning, NIPS*, pages 1–5, 2017.
- Niall P. Hurley and Scott T. Rickard. Comparing measures of sparsity. *IEEE Transactions on Information Theory*, 55:4723–4741, 2008.
- Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430, 2000.
- Matthew Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in Neural Information Processing Systems*, pages 2946–2954. 2016.
- Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International Conference on Machine Learning*, 2018.
- Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.
- Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.
- Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, 2014.
- Soheil Kolouri, Phillip E. Pope, Charles E. Martin, and Gustavo K. Rohde. Sliced wasserstein auto-encoders. In *International Conference on Learning Representations*, 2019.
- Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational Inference of Disentangled Latent Concepts from Unlabeled Observations. *arXiv.org*, November 2017.
- Hugo Larochelle and Yoshua Bengio. Classification using discriminative restricted boltzmann machines. In *International Conference on Machine Learning*, pages 536–543, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-205-4.

- Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. In B. Schölkopf, J. C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, pages 801–808. MIT Press, 2007.
- Zachary C Lipton. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*, 2016.
- Francesco Locatello, Stefan Bauer, Mario Lucic, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. *International Conference on Machine Learning*, 2019.
- Chris J Maddison, John Lawson, George Tucker, Nicolas Heess, Mohammad Norouzi, Andriy Mnih, Arnaud Doucet, and Yee Teh. Filtering variational objectives. In *Advances in Neural Information Processing Systems*, pages 6573–6583, 2017.
- Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.
- Michael F Mathieu, Junbo Jake Zhao, Junbo Zhao, Aditya Ramesh, Pablo Sprechmann, and Yann LeCun. Disentangling factors of variation in deep representation using adversarial training. In *Advances in Neural Information Processing Systems*, pages 5040–5048, 2016.
- Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. dsprites: Disentanglement testing sprites dataset. <https://github.com/deepmind/dsprites-dataset/>, 2017.
- B. Olshausen and D. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.
- Kaare Brandt Petersen, Michael Syskind Pedersen, et al. The matrix cookbook. *Technical University of Denmark*, 7(15):510, 2008.
- Mary Phuong, Max Welling, Nate Kushman, Ryota Tomioka, and Sebastian Nowozin. The mutual autoencoder: Controlling information in latent code representations, 2018. URL <https://openreview.net/forum?id=HkbnWqx CZ>.
- Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *International Conference on Learning Representations*, 2016.
- Tom Rainforth, Robert Cornish, Hongseok Yang, Andrew Warrington, and Frank Wood. On nesting monte carlo estimators. In *International Conference on Machine Learning*, pages 4264–4273, 2018a.
- Tom Rainforth, Adam R. Kosiorek, Tuan Anh Le, Chris J. Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. *International Conference on Machine Learning*, 2018b.
- Marc Ranzato, Christopher Poultney, Sumit Chopra, and Yann L. Cun. Efficient learning of sparse representations with an energy-based model. In *Advances in Neural Information Processing Systems*, pages 1137–1144. MIT Press, 2007.
- Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018.
- Scott Reed, Kihyuk Sohn, Yuting Zhang, and Honglak Lee. Learning to disentangle factors of variation with manifold interaction. In *International Conference on Machine Learning*, pages 1431–1439, 2014.
- Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. 2014.
- Michal Rolínek, Dominik Zietlow, and Georg Martius. Variational Autoencoders Pursue PCA Directions (by Accident). *arXiv preprint arXiv:1812.06775*, 2018.
- Jürgen Schmidhuber. Learning factorial codes by predictability minimization. *Neural Computation*, 4(6):863–879, 1992.
- N. Siddharth, T Brooks Paige, Jan-Willem Van de Meent, Alban Desmaison, Noah Goodman, Pushmeet Kohli, Frank Wood, and Philip Torr. Learning disentangled representations with semi-supervised deep generative models. In *Advances in Neural Information Processing Systems*, pages 5925–5935, 2017.
- Jan Stühmer, Richard Turner, and Sebastian Nowozin. ISA-VAE: Independent subspace analysis with variational autoencoders, 2019. URL [https://openreview.net/forum?id=rJ1\\_NhR9K7](https://openreview.net/forum?id=rJ1_NhR9K7).
- Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schölkopf. Wasserstein auto-encoders. In *International Conference on Learning Representations*, 2018.
- Francesco Tonolini, Bjorn Sand Jensen, and Roderick Murray-Smith. Variational sparse coding, 2019. URL <https://openreview.net/forum?id=SkeJ6iR9Km>.
- Richard E. Turner and Maneesh Sahani. Two problems with variational expectation maximisation for time-series models. *D. Barber, T. Cemgil, and S. Chiappa (eds.), Bayesian Time series models, chapter 5*, page 109–130, 2011.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

Jiacheng Xu and Greg Durrett. Spherical Latent Spaces for Stable Variational Autoencoders. In *Conference on Empirical Methods in Natural Language Processing*, 2018.

Howard Hua Yang and Shun-ichi Amari. Adaptive online learning algorithms for blind separation: maximum entropy and minimum mutual information. *Neural computation*, 9(7):1457–1482, 1997.

Shengjia Zhao, Jiaming Song, and Stefano Ermon. Infovae: Information maximizing variational autoencoders. *CoRR*, abs/1706.02262, 2017. URL <http://arxiv.org/abs/1706.02262>.

## A. Proofs for Disentangling the $\beta$ -VAE

**Theorem 1.** *The  $\beta$ -VAE target  $\mathcal{L}_\beta(\mathbf{x})$  can be interpreted in terms of the standard ELBO,  $\mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi)$ , for an adjusted target  $\pi_{\theta, \beta}(\mathbf{x}, \mathbf{z}) \triangleq p_\theta(\mathbf{x} | \mathbf{z})f_\beta(\mathbf{z})$  with annealed prior  $f_\beta(\mathbf{z}) \triangleq p(\mathbf{z})^\beta / F_\beta$  as*

$$\mathcal{L}_\beta(\mathbf{x}) = \mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi) + (\beta - 1)H_{q_\phi} + \log F_\beta \quad (3)$$

where  $F_\beta \triangleq \int_{\mathbf{z}} p(\mathbf{z})^\beta d\mathbf{z}$  is constant given  $\beta$ , and  $H_{q_\phi}$  is the entropy of  $q_\phi(\mathbf{z} | \mathbf{x})$ .

*Proof.* Starting with (2), we have

$$\begin{aligned} \mathcal{L}_\beta(\mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x})} [\log p_\theta(\mathbf{x} | \mathbf{z})] + \beta H_{q_\phi} \\ &\quad + \beta \mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x})} [\log p(\mathbf{z})] \\ &= \mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x})} [\log p_\theta(\mathbf{x} | \mathbf{z})] + (\beta - 1)H_{q_\phi} + H_{q_\phi} \\ &\quad + \mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x})} [\log p(\mathbf{z})^\beta - \log F_\beta] + \log F_\beta \\ &= \mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x})} [\log p_\theta(\mathbf{x} | \mathbf{z})] + (\beta - 1)H_{q_\phi} \\ &\quad - \text{KL}(q_\phi(\mathbf{z} | \mathbf{x}) \| f_\beta(\mathbf{z})) + \log F_\beta \\ &= \mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi) + (\beta - 1)H_{q_\phi} + \log F_\beta \end{aligned}$$

as required.  $\square$

**Corollary 1.** *If  $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, \Sigma)$  and  $q_\phi(\mathbf{z} | \mathbf{x}) = \mathcal{N}(\mathbf{z}; \mu_\phi(\mathbf{x}), S_\phi(\mathbf{x}))$ , then,*

$$\mathcal{L}_\beta(\mathbf{x}; \theta, \phi) = \mathcal{L}(\mathbf{x}; \theta', \phi') + \frac{(\beta - 1)}{2} \log |S_{\phi'}(\mathbf{x})| + c \quad (4)$$

where  $\theta'$  and  $\phi'$  represent rescaled networks such that

$$\begin{aligned} p_{\theta'}(\mathbf{x} | \mathbf{z}) &= p_\theta(\mathbf{x} | \mathbf{z} / \sqrt{\beta}), \\ q_{\phi'}(\mathbf{z} | \mathbf{x}) &= \mathcal{N}(\mathbf{z}; \mu_\phi(\mathbf{x}), S_{\phi'}(\mathbf{x})), \\ \mu_{\phi'}(\mathbf{x}) &= \sqrt{\beta} \mu_\phi(\mathbf{x}), \quad S_{\phi'}(\mathbf{x}) = \beta S_\phi(\mathbf{x}), \end{aligned}$$

and  $c \triangleq \frac{D(\beta-1)}{2} (1 + \log \frac{2\pi}{\beta}) + \log F_\beta$  is a constant, with  $D$  denoting the dimensionality of  $\mathbf{z}$ .

*Proof.* We start by noting that

$$\begin{aligned} \pi_{\theta, \beta}(\mathbf{x}) &= \mathbb{E}_{f_\beta(\mathbf{z})} [p_\theta(\mathbf{x} | \mathbf{z})] = \mathbb{E}_{p(\mathbf{z})} [p_\theta(\mathbf{x} | \mathbf{z} / \sqrt{\beta})] \\ &= \mathbb{E}_{p(\mathbf{z})} [p_{\theta'}(\mathbf{x} | \mathbf{z})] = p_{\theta'}(\mathbf{x}) \end{aligned}$$

Now considering an alternate form of  $\mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi)$  in (3),

$$\begin{aligned} \mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi) &= \log \pi_{\theta, \beta}(\mathbf{x}) - \text{KL}(q_\phi(\mathbf{z} | \mathbf{x}) \| \pi_{\theta, \beta}(\mathbf{z} | \mathbf{x})) \\ &= \log p_{\theta'}(\mathbf{x}) - \mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x})} \left[ \log \left( \frac{q_\phi(\mathbf{z} | \mathbf{x}) p_{\theta'}(\mathbf{x})}{p_\theta(\mathbf{x} | \mathbf{z}) f_\beta(\mathbf{z})} \right) \right] \\ &= \log p_{\theta'}(\mathbf{x}) \\ &\quad - \mathbb{E}_{q_{\phi'}(\mathbf{z} | \mathbf{x})} \left[ \log \left( \frac{q_\phi(\mathbf{z} / \sqrt{\beta} | \mathbf{x}) p_{\theta'}(\mathbf{x})}{p_\theta(\mathbf{x} | \mathbf{z} / \sqrt{\beta}) f_\beta(\mathbf{z} / \sqrt{\beta})} \right) \right]. \quad (8) \end{aligned}$$

We first simplify  $f_\beta(\mathbf{z} / \sqrt{\beta})$  as

$$\begin{aligned} f_\beta(\mathbf{z} / \sqrt{\beta}) &= \frac{1}{\sqrt{2\pi}^{D/2} |\Sigma|^{D/2}} \exp \left( -\frac{1}{2} \mathbf{z}^T \Sigma^{-1} \mathbf{z} \right) \\ &= p(\mathbf{z}) \beta^{(D/2)}. \end{aligned}$$

Further, denoting  $\mathbf{z}_\dagger = \mathbf{z} - \sqrt{\beta} \mu_{\phi'}(\mathbf{x})$ , and  $\mathbf{z}_\ddagger = \mathbf{z}_\dagger / \sqrt{\beta} = \mathbf{z} / \sqrt{\beta} - \mu_{\phi'}(\mathbf{x})$ , we have

$$\begin{aligned} q_{\phi'}(\mathbf{z} | \mathbf{x}) &= \frac{1}{\sqrt{2\pi}^{D/2} |S_\phi(\mathbf{x})|^{D/2}} \exp \left( -\frac{1}{2} \mathbf{z}_\dagger^T S_\phi(\mathbf{x})^{-1} \mathbf{z}_\dagger \right), \\ q_\phi \left( \frac{\mathbf{z}}{\sqrt{\beta}} | \mathbf{x} \right) &= \frac{1}{\sqrt{2\pi}^{D/2} |S_\phi(\mathbf{x})|^{D/2}} \exp \left( -\frac{1}{2} \mathbf{z}_\ddagger^T S_\phi(\mathbf{x})^{-1} \mathbf{z}_\ddagger \right) \\ \text{giving } q_\phi \left( \mathbf{z} / \sqrt{\beta} | \mathbf{x} \right) &= q_{\phi'}(\mathbf{z} | \mathbf{x}) \beta^{(D/2)}. \end{aligned}$$

Plugging these back in to (8) while remembering  $p_\theta(\mathbf{x} | \mathbf{z} / \sqrt{\beta}) = p_{\theta'}(\mathbf{x} | \mathbf{z})$ , we have

$$\begin{aligned} \mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi) &= \log p_{\theta'}(\mathbf{x}) - \mathbb{E}_{q_{\phi'}(\mathbf{z} | \mathbf{x})} \left[ \log \left( \frac{q_{\phi'}(\mathbf{z} | \mathbf{x}) p_{\theta'}(\mathbf{x})}{p_{\theta'}(\mathbf{x} | \mathbf{z}) p(\mathbf{z})} \right) \right] \\ &= \mathcal{L}(\mathbf{x}; \theta, \phi), \end{aligned}$$

showing that the ELBOs for the two setups are the same. For the entropy term, we note that

$$\begin{aligned} H_{q_\phi} &= \frac{D}{2} (1 + \log 2\pi) + \frac{1}{2} \log |S_\phi(\mathbf{x})| \\ &= \frac{D}{2} \left( 1 + \log \frac{2\pi}{\beta} \right) + \frac{1}{2} \log |S_{\phi'}(\mathbf{x})|. \end{aligned}$$

Finally substituting for  $H_{q_\phi}$  and  $\mathcal{L}(\mathbf{x}; \pi_{\theta, \beta}, q_\phi)$  in (3) gives the desired result.  $\square$

**Corollary 2.** *Let  $[\theta', \phi'] = g_\beta([\theta, \phi])$  represent the transformation required to produced the rescaled networks in Corollary 1. If  $0 < |\det \nabla_{\theta, \phi} g([\theta, \phi])| < \infty \forall [\theta, \phi]$ , then*

$$\nabla_{\theta, \phi} \mathcal{L}_\beta(\mathbf{x}; \theta, \phi) = \mathbf{0} \Leftrightarrow \nabla_{\theta', \phi'} \mathcal{L}_{H, \beta}(\mathbf{x}; \theta', \phi') = \mathbf{0}.$$

Thus  $[\theta^*, \phi^*]$  being a stationary point of  $\frac{1}{n} \sum_{i=1}^n \mathcal{L}_\beta(\mathbf{x}_i; \theta, \phi)$  indicates that  $g_\beta([\theta^*, \phi^*])$  is a stationary point of  $\frac{1}{n} \sum_{i=1}^n \mathcal{L}_{H, \beta}(\mathbf{x}_i; \theta', \phi')$  and vice-versa.

*Proof.* Starting from Corollary 1 we have that

$$\begin{aligned} \nabla_{\theta, \phi} \mathcal{L}_\beta(\mathbf{x}; \theta, \phi) &= \nabla_{\theta, \phi} \mathcal{L}_{H, \beta}(\mathbf{x}; \theta', \phi') \\ &= (\nabla_{\theta, \phi} g_\beta([\theta, \phi])) \nabla_{\theta', \phi'} \mathcal{L}_{H, \beta}(\mathbf{x}; \theta', \phi'), \end{aligned}$$

so  $\nabla_{\theta', \phi'} \mathcal{L}_{H, \beta}(\mathbf{x}; \theta', \phi') = \mathbf{0} \implies \nabla_{\theta, \phi} \mathcal{L}_\beta(\mathbf{x}; \theta, \phi) = \mathbf{0}$  given our assumption that  $|\det \nabla_{\theta, \phi} g([\theta, \phi])| < \infty \forall [\theta, \phi]$ . Further, as  $0 < |\det \nabla_{\theta, \phi} g([\theta, \phi])| \forall [\theta, \phi]$ ,  $(\nabla_{\theta, \phi} g_\beta([\theta, \phi]))^{-1}$  exists and has a finite determinant, so  $\nabla_{\theta, \phi} \mathcal{L}_\beta(\mathbf{x}; \theta, \phi) = \mathbf{0}$  also implies  $\nabla_{\theta', \phi'} \mathcal{L}_{H, \beta}(\mathbf{x}; \theta', \phi') = \mathbf{0}$ .  $\square$

**Theorem 2.** If  $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; 0, \sigma I)$  and  $q_\phi(\mathbf{z} | \mathbf{x}) = \mathcal{N}(\mathbf{z}; \mu_\phi(\mathbf{x}), S_\phi(\mathbf{x}))$ , then for all rotation matrices  $R$ ,

$$\mathcal{L}_\beta(\mathbf{x}; \theta, \phi) = \mathcal{L}_\beta(\mathbf{x}; \theta^\dagger(R), \phi^\dagger(R)) \quad (6)$$

where  $\theta^\dagger(R)$  and  $\phi^\dagger(R)$  are transformed networks such that

$$\begin{aligned} p_{\theta^\dagger}(\mathbf{x} | \mathbf{z}) &= p_\theta(\mathbf{x} | R^T \mathbf{z}), \\ q_{\phi^\dagger}(\mathbf{z} | \mathbf{x}) &= \mathcal{N}(\mathbf{z}; R\mu_\phi(\mathbf{x}), RS_\phi(\mathbf{x})R^T). \end{aligned}$$

*Proof.* If  $\mathbf{z} \sim q_\phi(\mathbf{z} | \mathbf{x})$  and  $\mathbf{y} = R\mathbf{z}$  then, by Petersen et al. (§8.1.4 2008)), we have

$$\mathbf{y} \sim \mathcal{N}(\mathbf{y}; R\mu_\phi(\mathbf{x}), RS_\phi(\mathbf{x})R^T).$$

Consequently, the changes made by the transformed networks cancel to give the same reconstruction error as

$$\begin{aligned} \mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x})}[\log p_\theta(\mathbf{x} | \mathbf{z})] &= \mathbb{E}_{q_{\phi^\dagger}(\mathbf{z} | \mathbf{x})}[\log p_\theta(\mathbf{x} | R^T \mathbf{z})] \\ &= \mathbb{E}_{q_{\phi^\dagger}(\mathbf{z} | \mathbf{x})}[\log p_{\theta^\dagger}(\mathbf{x} | \mathbf{z})]. \end{aligned}$$

Furthermore, the KL divergence between  $q_\phi(\mathbf{z} | \mathbf{x})$  and  $p_\theta(\mathbf{z})$  is invariant to rotation, because of the rotational symmetry of the latter, such that  $\text{KL}(q_\phi(\mathbf{z} | \mathbf{x}) \| p(\mathbf{z})) = \text{KL}(q_{\phi^\dagger}(\mathbf{z} | \mathbf{x}) \| p(\mathbf{z}))$ . The result now follows from noting that the two terms of the  $\beta$ -VAE are equal under rotation.  $\square$

## B. Experimental Details

**Disentanglement - 2d-shapes:** The experiments from Section 6 on the impact of the prior in terms disentanglement are conducted on the **2D Shapes** (Matthey et al., 2017) dataset, comprising of 737,280 binary 64 x 64 images of 2D shapes with ground truth factors [number of values]: shape [3], scale [6], orientation [40], x-position [32], y-position [32]. We use a convolutional neural network for the encoder and a deconvolutional neural network for the decoder, whose architectures are described in Table 1a. We use [0, 1] normalised data as targets for the mean of a Bernoulli distribution and negative cross-entropy for  $\log p(\mathbf{x} | \mathbf{z})$ . We rely on the Adam optimiser (Kingma and Ba, 2015; Reddi et al., 2018) with learning rate  $1e^{-4}$ ,  $\beta_1 = 0.9$ , and  $\beta_2 = 0.999$ , to optimise the  $\beta$ -VAE objective from (3).

For  $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \text{diag}(\sigma))$ , experiments were run with a batch size of 64 and for 20 epochs. For  $p(\mathbf{z}) = \prod_d \text{STUDENT-T}(z_d; \nu)$ , experiments were run with a batch size of 256 and for 40 epochs. In Figure 2, the *PCA initialised anisotropic* prior is initialised so that its standard deviations are set to be the first  $D$  singular values of the data. These are then mapped through a softmax function to ensure that the  $\beta$  regularisation coefficient is not implicitly scaled compared to the isotropic case. For the *learned anisotropic* priors, standard deviations are first initialised as just described, and then learned along with the model through a log-variance parametrisation.

| Encoder                      | Decoder                        |
|------------------------------|--------------------------------|
| Input 64 x 64 binary image   | Input $\in \mathbb{R}^{10}$    |
| 4x4 conv. 32 stride 2 & ReLU | FC. 128 ReLU                   |
| 4x4 conv. 32 stride 2 & ReLU | FC. 4x4 x 64 ReLU              |
| 4x4 conv. 64 stride 2 & ReLU | 4x4 upconv. 64 stride 2 & ReLU |
| 4x4 conv. 64 stride 2 & ReLU | 4x4 upconv. 64 stride 2 & ReLU |
| FC. 128                      | 4x4 upconv. 32 stride 2 & ReLU |
| FC. 2x10                     | 4x4 upconv. 1. stride 2        |

(a) 2D-shapes dataset.

| Encoder                  | Decoder                  |
|--------------------------|--------------------------|
| Input $\in \mathbb{R}^2$ | Input $\in \mathbb{R}^2$ |
| FC. 100. & ReLU          | FC. 100 & ReLU           |
| FC. 2x2                  | FC. 2x2                  |

(b) Pinwheel dataset.

| Encoder  |
|--|
| Input 32 x 32 x 1 channel image                      |
| 4x4 conv. 32 stride 2 & BatchNorm2d & LeakyReLU(.2)  |
| 4x4 conv. 64 stride 2 & BatchNorm2d & LeakyReLU(.2)  |
| 4x4 conv. 128 stride 2 & BatchNorm2d & LeakyReLU(.2) |
| 4x4 conv. 50, 4x4 conv. 50                           |
| Decoder  |
| Input $\in \mathbb{R}^{50}$                          |
| 4x4 upconv. 128 stride 1 pad 0 & BatchNorm2d & ReLU  |
| 4x4 upconv. 64 stride 2 pad 1 & BatchNorm2d & ReLU   |
| 4x4 upconv. 32 stride 2 pad 1 & BatchNorm2d & ReLU   |
| 4x4 upconv. 1 stride 2 pad 1                         |

(c) Fashion-MNIST dataset.

Table 1. Encoder and decoder architectures.

We rely on the metric presented in §4 and Appendix B of Kim and Mnih (2018) as a measure of axis-alignment of the latent encodings with respect to the true (known) generative factors. Confidence intervals in Figure 2 were computed via the assumption of normally distributed samples with unknown mean and variance, with 100 runs of each model.

**Clustering - Pinwheel** We generated spiral cluster data<sup>2</sup>, with  $n = 400$  observations, clustered in 4 spirals, with radial and tangential standard deviations respectively of 0.1 and 0.30, and a rate of 0.25. We use fully-connected neural networks for both the encoder and decoder, whose architectures are described in Table 1b. We minimise the objective from (7), with  $\mathbb{D}$  chosen to be the inclusive KL and  $q_\phi(\mathbf{z})$  approximated by the aggregate encoding of the full dataset:

$$\begin{aligned} \mathbb{D}(q_\phi(\mathbf{z}), p(\mathbf{z})) &= \text{KL}(p(\mathbf{z}) \| q_\phi(\mathbf{z})) \\ &= \mathbb{E}_{p(\mathbf{z})}[\log(p(\mathbf{z})) - \log(\mathbb{E}_{p_{\mathcal{D}}(\mathbf{x})}[q_\phi(\mathbf{z} | \mathbf{x})])] \end{aligned}$$

<sup>2</sup><http://hips.seas.harvard.edu/content/synthetic-pinwheel-data-matlab>.

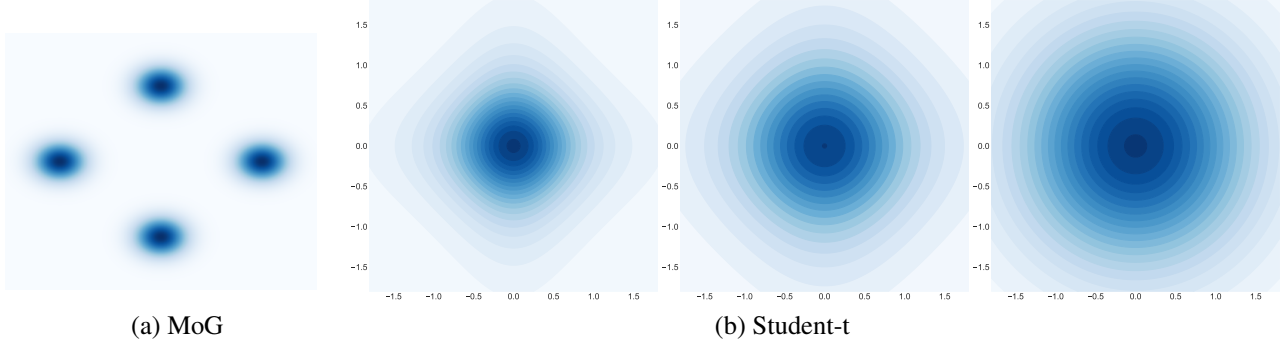


Figure 6. (a) PDF of Gaussian mixture model prior  $p(\mathbf{z})$ , as per (9). (b) PDF for a 2-dimensional factored Student-t distributions  $p_\nu$  with degree of freedom  $\nu = \{3, 5, 100\}$  (left to right). Note that  $p_\nu(\mathbf{z}) \rightarrow \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$  as  $\nu \rightarrow \infty$ .

$$\approx \sum_{j=1}^B \left( \log p(\mathbf{z}_j) - \log \left( \sum_{i=1}^n q_\phi(\mathbf{z}_j | \mathbf{x}_i) \right) \right)$$

with  $\mathbf{z}_j \sim p(\mathbf{z})$ . A Gaussian likelihood is used for the encoder. We trained the model for 500 epochs using the Adam optimiser (Kingma and Ba, 2015; Reddi et al., 2018), with  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$  and a learning rate of  $1e^{-3}$ . The batch size is set to  $B = n$ .

The Gaussian mixture prior (c.f. Figure 6(a)) is defined as

$$\begin{aligned} p(\mathbf{z}) &= \sum_{c=1}^C \pi^c \mathcal{N}(\mathbf{z} | \boldsymbol{\mu}^c, \boldsymbol{\Sigma}^c) \\ &= \sum_{c=1}^C \pi^c \prod_{d=1}^D \mathcal{N}(z_d | \mu_d^c, \sigma_d^c) \end{aligned} \quad (9)$$

with  $D = 2$ ,  $C = 4$ ,  $\boldsymbol{\Sigma}^c = 0.03I_D$ ,  $\pi^c = 1/C$ , and  $\mu_d^c \in \{0, 1\}$ .

**Sparsity - Fashion-MNIST** The experiments from Section 6 on the latent representation’s sparsity are conducted on the **Fashion-MNIST** (Xiao et al., 2017) dataset, comprising of 70,000 greyscale images resized to  $32 \times 32$ .

To enforce sparsity, we relied on a prior defined as a factored univariate mixture of a standard and low variance normal distributions:

$$p(z) = \prod_d (1 - \gamma) \mathcal{N}(z_d; 0, 1) + \gamma \mathcal{N}(z_d; 0, \sigma_0^2)$$

with  $\sigma_0^2 = 0.05$ . The weight,  $\gamma$ , of the low-variance component indicates how likely samples are to come from that component, hence to be *off*.

We minimised the objective from (7), with  $\mathbb{D}(q_\phi(\mathbf{z}), p(\mathbf{z}))$  taken to be a dimension-wise MMD with a sum of *Cauchy* kernels on each dimension. Equivalently, we can think of this as calculating a single MMD using the single kernel

$$k(\mathbf{x}, \mathbf{y}) = \sum_{d=1}^D \sum_{\ell}^L \frac{\sigma_\ell}{\sigma_{\ell-1} + (x_d - y_d)^2}. \quad (10)$$

where  $\sigma_\ell \in \{0.2, 0.4, 1, 2, 4, 10\}$  are a set of length scales.

This dimension-wise kernel only enforces a congruence between the marginal distributions of  $\mathbf{x}$  and  $\mathbf{y}$  and so, strictly speaking, its MMD does not constitute a valid divergence metric in the sense that we can have  $\mathbb{D}(q_\phi(\mathbf{z}), p(\mathbf{z})) = 0$  when  $q_\phi(\mathbf{z})$  and  $p(\mathbf{z})$  are not identical distributions: it only requires their marginals to match to get zero divergence.

The reasons we chose this approach are twofold. Firstly, we found that conventional kernels based on the Euclidean distance between encodings produced gradients with insurmountably high variances, meaning that effectively minimizing the divergence to get  $q_\phi(\mathbf{z})$  and  $p(\mathbf{z})$  to match was not possible, even for very large batch sizes and  $\alpha \rightarrow \infty$ .

Secondly, though just matching the marginal distributions is not sufficient to ensure sparsity—as one could have some points with all dimensions close to the origin and some with all dimensions far away—a combination of the need to achieve good reconstructions and noise in the encoder process should prevent this from occurring. In short, provided the noise from the encoder is properly regulated, there is little information that can be stored in latent dimensions near the origin because of the high level of overlap forced in this region. Therefore, for a datapoint to be effectively encoded, it must have at least some of its latent dimensions outside of this region. Coupled with the need for most of the latent values to be near the origin to match the marginal distributions, this, in turn, enforces a sparse representation. Consequently, the loss in sparsity performance relative to using a hypothetical kernel that is both universal and has stable gradient estimates should only be relatively small, as is borne out in our empirical results. This may, however, be why we see a slight drop in sparsity performance for very large values of  $\alpha$ .

We use a convolutional neural network for the encoder and a deconvolutional neural network for the decoder, whose architectures come from the DCGAN model (Radford et al., 2016) and are described in Table 1c. We use  $[0, 1]$  normalised data as targets for the mean of a Laplace distribution

with fixed scaling of 0.1. We rely on the Adam optimiser with learning rate  $5e^{-4}$ ,  $\beta_1 = 0.5$ , and  $\beta_2 = 0.999$ . The model is then trained (on the training set) for 80 epochs with a batch-size of 500.

As an extrinsic measure of *sparsity*, we use the *Hoyer* metric (Hurley and Rickard, 2008), defined for  $\mathbf{y} \in \mathbb{R}^d$  by

$$\text{Hoyer}(\mathbf{y}) = \frac{\sqrt{d} - \|\mathbf{y}\|_1 / \|\mathbf{y}\|_2}{\sqrt{d} - 1} \in [0, 1],$$

yielding 0 for a fully dense vector and 1 for a fully sparse vector. We additionally normalise each dimension to have a standard deviation of 1 under its aggregate distribution, i.e. we use  $\bar{z}_d = z_d / \sigma(z_d)$  where  $\sigma(z_d)$  is the standard deviation of dimension  $d$  of the latent encoding taken over the dataset. Overall sparsity is computed by averaging over the dataset as  $\text{Sparsity} = 1/n \sum_i \text{Hoyer}(\bar{z}_i)$ .

As discussed in the main text, we use a trained model with  $\alpha = 1000$ ,  $\beta = 1$ , and  $\gamma = 0.8$  to perform a qualitative analysis of sparsity using the *Fashion-MNIST* dataset. Figure 7 shows the per-class average embedding magnitude for this model, a subset of which was shown in the main text. As can be seen clearly, the different classes utilise predominantly different subsets of dimensions to encode the image data, as one might expect for sparse representations.

### C. Posterior regularisation

The aggregate posterior regulariser  $\mathbb{D}(q(\mathbf{z}), p(\mathbf{z}))$  is a little more subtle to analyse than the entropy regulariser as it involves both the choice of divergence and potential difficulties in estimating that divergence. One possible choice is the exclusive Kullback-Leibler divergence  $\text{KL}(q(\mathbf{z}) \parallel p(\mathbf{z}))$ , as previously used (without additional entropy regularisation) by (Dilokthanakul et al., 2019; Esmaeili et al., 2019), but also implicitly by (Chen et al., 2018), through the use of a total correlation (TC) term. We now highlight a shortfall with this choice of divergence due to difficulties in its empirical estimation.

In short, the approaches used to estimate the  $\text{H}[q(\mathbf{z})]$  (noting that  $\text{KL}(q(\mathbf{z}) \parallel p(\mathbf{z})) = -\text{H}[q(\mathbf{z})] - \mathbb{E}_{q(\mathbf{z})}[\log p(\mathbf{z})]$ , where the latter term can be estimated reliably by a simple Monte Carlo estimate) can exhibit very large biases unless very large batch sizes are used, resulting in quite different effects from what was intended. In fact, our results suggest they will exhibit behaviour similar to the  $\beta$ -VAE if the batch size is too small. These biases arise from the effects of nesting estimators (Rainforth et al., 2018a), where the variance in the nested (inner) estimator for  $q(\mathbf{z})$  induces a bias in the overall estimator. Specifically, for any random variable  $\hat{Z}$ ,

$$\mathbb{E}[\log(\hat{Z})] = \log(\mathbb{E}[\hat{Z}]) - \frac{\text{Var}[\hat{Z}]}{2Z^2} + O(\varepsilon)$$

where  $O(\varepsilon)$  represents higher-order moments that get dominated asymptotically if  $\hat{Z}$  is a Monte-Carlo estimator (see Proposition 1c in Maddison et al. (2017), Theorem 1 in Rainforth et al. (2018b), or Theorem 3 in Domke and Sheldon (2018)). In this setting,  $\hat{Z} = \hat{q}(\mathbf{z})$  is the estimate used for  $q(\mathbf{z})$ . We thus see that if the variance of  $\hat{q}(\mathbf{z})$  is large, this will induce a significant bias in our KL estimator.

To make things precise, we consider the estimator used for  $\text{H}[q(\mathbf{z})]$  in Chen et al. (2018); Dilokthanakul et al. (2019); Esmaeili et al. (2019)

$$\text{H}[q(\mathbf{z})] \approx \hat{\text{H}} \triangleq -\frac{1}{B} \sum_{b=1}^B \log \hat{q}(\mathbf{z}_b), \text{ where} \quad (11a)$$

$$\hat{q}(\mathbf{z}_b) = \frac{q_\phi(\mathbf{z}_b | \mathbf{x}_b)}{n} + \frac{n-1}{n(B-1)} \sum_{b' \neq b} q_\phi(\mathbf{z}_b | \mathbf{x}_{b'}), \quad (11b)$$

$\mathbf{z}_b \sim q_\phi(\mathbf{z} | \mathbf{x}_b)$ , and  $\{\mathbf{x}_1, \dots, \mathbf{x}_B\}$  is the mini-batch of data used for the current iteration for dataset size  $n$ . Esmaeili et al. (2019) correctly show that  $\mathbb{E}[\hat{q}(\mathbf{z}_b)] = \tilde{q}(\mathbf{z}_b)$ , with the first term of (11b) comprising an exact term in  $\tilde{q}(\mathbf{z}_b)$  and the second term of (11b) being an unbiased Monte-Carlo estimate for the remaining terms in  $\tilde{q}(\mathbf{z}_b)$ .

To examine the practical behaviour of this estimator when  $B \ll n$ , we first note that the second term of (11b) is, in practice, usually very small and dominated by the first term. This is borne out empirically in our own experiments, and also noted in Kim and Mnih (2018). To see why this is the case, consider that given encodings of two independent data points, it is highly unlikely that the two encoding distributions will have any notable overlap (e.g. for a Gaussian encoder, the means will most likely be very many standard deviations apart), presuming a sensible latent space is being learned. Consequently, even though this second term is unbiased and may have an expectation comparable or even larger than the first, it is heavily skewed—it is usually negligible, but occasionally large in the rare instances where there is substantial overlap between encodings.

Let the second term of (11b) be  $T_2$  and the event that this is significant be  $E_S$ , such that  $\mathbb{E}[T_2 | \neg E_S] \approx 0$ . As explained above,  $\mathbb{P}(E_S) \ll 1$  typically. We now have

$$\begin{aligned} \mathbb{E}[\hat{\text{H}}] &= \mathbb{P}(E_S) \mathbb{E}[\hat{\text{H}} | E_S] + (1 - \mathbb{P}(E_S)) \mathbb{E}[\hat{\text{H}} | \neg E_S] \\ &= \mathbb{P}(E_S) \mathbb{E}[\hat{\text{H}} | E_S] + (1 - \mathbb{P}(E_S)) \\ &\quad \cdot (\log n - \frac{1}{B} \sum_{b=1}^B \mathbb{E}[\log q_\phi(\mathbf{z}_b | \mathbf{x}_b) | \neg E_S] - \mathbb{E}[T_2 | \neg E_S]) \\ &= \mathbb{P}(E_S) \mathbb{E}[\hat{\text{H}} | E_S] + (1 - \mathbb{P}(E_S)) \\ &\quad \cdot (\log n - \mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1) | \neg E_S] - \mathbb{E}[T_2 | \neg E_S]) \\ &\approx \mathbb{P}(E_S) \mathbb{E}[\hat{\text{H}} | E_S] \\ &\quad + (1 - \mathbb{P}(E_S)) (\log n - \mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1)]) \end{aligned}$$



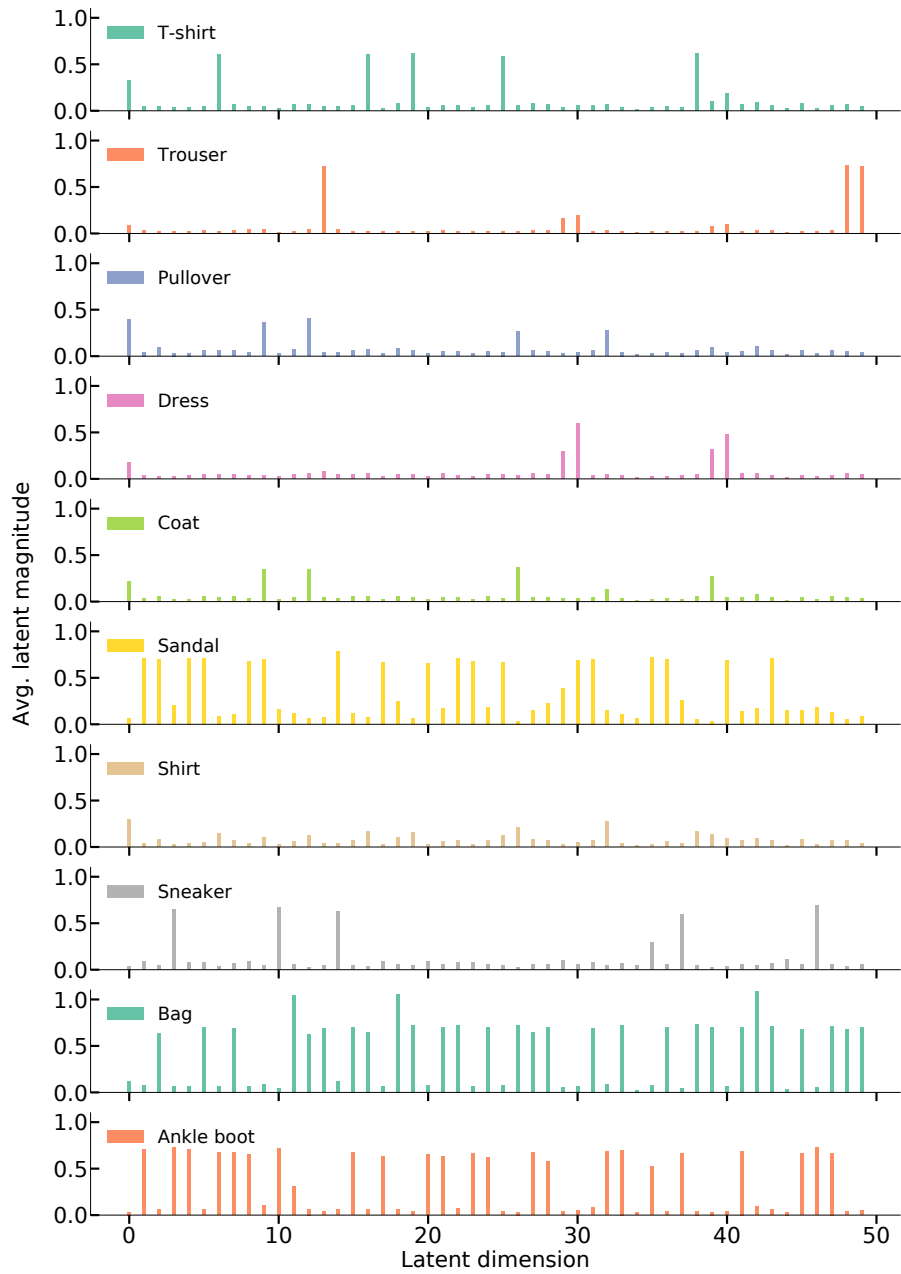


Figure 7. Average encoding magnitude over data for each classes in *Fashion-MNIST*.

where the approximation relies firstly on our previous assumption that  $\mathbb{E}[T_2 | \neg E_S] \approx 0$  and also that  $\mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1) | \neg E_S] \approx \mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1)]$ . This second assumption will also generally hold in practice, firstly because the occurrence of  $E_S$  is dominated by whether two similar datapoints are drawn (rather than by the value of  $\mathbf{x}_1$ ) and secondly because  $\mathbb{P}(E_S) \ll 1$  implies that

$$\begin{aligned} & \mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1)] \\ &= (1 - \mathbb{P}(E_S)) \mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1) | \neg E_S] \\ & \quad + \mathbb{P}(E_S) \mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1) | E_S] \\ & \approx \mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1) | \neg E_S]. \end{aligned}$$

Characterising  $\mathbb{E}[\hat{H} | E_S]$  precisely is a little more challenging, but it can safely be assumed to be smaller than  $\mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1)]$ , which is approximately what would result from all the  $\mathbf{x}'_b$  being the same as  $\mathbf{x}_b$ . We thus see that even when the event  $E_S$  does occur, the resulting estimates will still, at most, be on a comparable scale to when it does not. Consequently, whenever  $E_S$  is rare, the  $(1 - \mathbb{P}(E_S)) \mathbb{E}[\hat{H} | \neg E_S]$  term will dominate and we thus have

$$\begin{aligned} \mathbb{E}[\hat{H}] & \approx \log n - \mathbb{E}[\log q_\phi(\mathbf{z}_1 | \mathbf{x}_1)] \\ & = \log n + \mathbb{E}_{p(\mathbf{x})}[\mathbb{H}[q_\phi(\mathbf{z} | \mathbf{x})]]. \end{aligned}$$

We now see that the estimator mimics the  $\beta$ -VAE regularisation up to a constant factor  $\log n$ , as adding the  $\mathbb{E}_{q(\mathbf{z})}[\log p(\mathbf{z})]$  back in gives

$$\begin{aligned} & -\mathbb{E}[\hat{H}] - \mathbb{E}_{q(\mathbf{z})}[\log p(\mathbf{z})] \\ & \approx \mathbb{E}_{p(\mathbf{x})}[\text{KL}(q_\phi(\mathbf{z} | \mathbf{x}) \| p(\mathbf{z}))] - \log n. \end{aligned}$$

We should thus expect to empirically see training with this estimator as a regulariser to behave similarly to the  $\beta$ -VAE with the same regularisation term whenever  $B \ll n$ . Note that the  $\log n$  constant factor will not impact the gradients, but does mean that it is possible, even likely, that negative estimates for  $\hat{\text{KL}}$  will be generated, even though we know the true value is positive.

Overcoming the problem can, at least to a certain degree, be overcome by using very large batch sizes  $B$ , at an inevitable computational and memory cost. However, the problem is potentially exacerbated in higher dimensional latent spaces and larger datasets, for which one would typically expect the typical overlap of datapoints to decrease.

### C.1. Other Divergences

As discussed in the main paper,  $\text{KL}(q(\mathbf{z}) \| p(\mathbf{z}))$  is far from the only aggregate posterior regulariser one might use. Though we do not analyse them formally, we expect many alternative divergence-estimator pairs to suffer from similar

issues. For example, using Monte Carlo estimators with the inclusive Kullback-Leibler divergence  $\text{KL}(p(\mathbf{z}) \| q(\mathbf{z}))$  or the sliced Wasserstein distance (Kolouri et al., 2019) both result in nested expectations analogously to  $\text{KL}(q(\mathbf{z}) \| p(\mathbf{z}))$ , and are therefore likely to similarly induce substantial bias without using large batch sizes.

Interestingly, however, MMD and generative adversarial network (GAN) regularisers of the form discussed in (Tolstikhin et al., 2018) do not result in nested expectations and therefore are necessarily not prone to the same issues: they produce unbiased estimates of their respective objectives. Though we experienced practical issues in successfully implementing both of these—we found the signal-to-noise-ratio of the MMD gradient estimates to be very low, particularly in high dimensions, while we experienced training instabilities for the GAN regulariser—their apparent theoretical advantages may indicate that they are preferable approaches, particularly if these issues can be alleviated. The GAN-based approach to estimating the total correlation introduced by Kim and Mnih (2018) similarly allows a nested expectation to be avoided, at the cost of converting a conventional optimization into a minimax problem.

Given the failings of the available existing approaches, we believe that further investigation into divergence-estimator pairs for  $\mathbb{D}(q(\mathbf{z}), p(\mathbf{z}))$  in VAEs is an important topic for future work that extends well beyond the context of this paper, or even the general aim of achieving decomposition. In particular, the need for congruence between the posterior (encoder), likelihood (decoder), and marginal likelihood (data distribution) for a generative model, means that ensuring  $q(\mathbf{z})$  is close to  $p(\mathbf{z})$  is a generally important endeavour for training VAEs. For example, mismatch between  $q(\mathbf{z})$  and  $p(\mathbf{z})$  will cause samples drawn from the learned generative model to mismatch the true data-generating distribution, regardless of the fidelity of our encoder and decoder.

## D. Characterising Overlap

Reiterating the argument from the main text, although the mutual information  $I(\mathbf{x}; \mathbf{z})$  between data and latents provides a perfectly serviceable characterisation of overlap in a number of cases, the two are not universally equivalent and we argue that it is overlap which is important in achieving useful representations. In particular, if the form of the encoding distribution is not fixed—as when employing normalising flows, for example— $I(\mathbf{x}; \mathbf{z})$  does not necessarily characterise overlap well.

Consider, for example, an encoding distribution that is a mixture between the prior and a uniform distribution on a tiny  $\epsilon$ -ball around the mean encoding  $\mu_\phi(\mathbf{x})$ , i.e.  $q_\phi(\mathbf{z} | \mathbf{x}) = \lambda \cdot \text{Uniform}(\|\mu_\phi(\mathbf{x}) - \mathbf{z}\|_2 < \epsilon) + (1 - \lambda) \cdot p(\mathbf{z})$ . If the encoder and decoder are sufficiently flexible to learn

arbitrary representations, one now could arrive at *any* value for mutual information simply by an appropriate choice of  $\lambda$ . However, enforcing structuring of the latent space will be effectively impossible due to the lack of any pressure (other than a potentially small amount from internal regularization in the encoder network itself) for similar encodings to correspond to similar datapoints; the overlap between any two encodings is the same unless they are within  $\epsilon$  of each other.

While this example is a bit contrived, it highlights a key feature of overlap that  $I(\mathbf{x}; \mathbf{z})$  fails to capture:  $I(\mathbf{x}; \mathbf{z})$  does not distinguish between large overlap with a small number of other datapoints and small overlap with a large number of other datapoints. This distinction is important because we are particularly interested in *how many* other datapoints one datapoint's encoding overlaps with when imposing structure—the example setup fails because each datapoint has the same level of overlap with all the other datapoints.

Another feature that  $I(\mathbf{x}; \mathbf{z})$  can fail to account for is a notion of *locality* in the latent space. Imagine a scenario where the encoding distributions are extremely multimodal with similar sized modes spread throughout the latent space, such as  $q(\mathbf{z}|\mathbf{x}) = \sum_{i=1}^{1000} \mathcal{N}(\mathbf{z}; \mu_\phi(\mathbf{x}) + m_i, \sigma I)$  for some constant scalar  $\sigma$ , and vectors  $m_i$ . Again we can achieve almost any value for  $I(\mathbf{x}; \mathbf{z})$  by adjusting  $\sigma$ , but it is difficult to impose meaningful structure regardless as each datapoint can be encoded to many different regions of the latent space.