Overlooked Implications of the Reconstruction Loss for VAE Disentanglement

Nathan Michlo, **Richard Klein**, **Steven James**
University of the Witwatersrand, Johannesburg, South Africa
nathan.michlo1@students.wits.ac.za, {richard.klein, steven.james}@wits.ac.za

Abstract

Learning disentangled representations with variational autoencoders (VAEs) is often attributed to the regularisation component of the loss. In this work, we highlight the interaction between data and the reconstruction term of the loss as the main contributor to disentanglement in VAEs. We show that standard benchmark datasets have unintended correlations between their subjective ground-truth factors and perceived axes in the data according to typical VAE reconstruction losses. Our work exploits this relationship to provide a theory for what constitutes an adversarial dataset under a given reconstruction loss. We verify this by constructing an example dataset that prevents disentanglement in state-of-the-art frameworks while maintaining human-intuitive ground-truth factors. Finally, we re-enable disentanglement by designing an example reconstruction loss that is once again able to perceive the ground-truth factors. Our findings demonstrate the subjective nature of disentanglement and the importance of considering the interaction between the ground-truth factors, data and notably, the reconstruction loss, which is under-recognised in the literature.

1 Introduction

A fundamental challenge in machine learning is discovering useful representations from high-dimensional data that can be used to solve subsequent tasks effectively. Recently, deep learning approaches have showcased the ability of neural networks to extract meaningful features from high-dimensional inputs for tasks such as classification [Krizhevsky *et al.*, 2012] and reinforcement learning [Mnih *et al.*, 2015]. However, these learned representations are often not semantically meaningful, which can negatively impact interpretability, fairness [Locatello *et al.*, 2019a], and downstream task performance [Locatello *et al.*, 2019b].

Prior work has therefore argued that it is desirable to learn a representation that is *disentangled* [Bengio *et al.*, 2013]. While there is no consensus on what constitutes a disentangled representation, it is generally agreed that it should be factorised so that each latent variable corresponds to a single explanatory variable responsible for generating the data [Burgess *et al.*, 2017]. For example, a single image from a video game may be represented by continuous latent variables governing the x and y positions of the player or enemies, as well as categorical variables governing their clothing or appearance.

A common approach to discovering these representations is variational autoencoders (VAEs) [Kingma and Welling, 2014], which are trained on unlabelled data to learn a lower-dimensional representation capable of reconstructing the input. However, it has been shown that unsupervised methods cannot reliably learn representations without the introduction of supervision or inductive biases [Locatello *et al.*, 2019b]. The recently introduced Ada-GVAE framework partially overcame this problem by using a weakly supervised signal to discover underlying factors [Locatello *et al.*, 2020], but there remains room for improvement.

Interestingly, VAEs do not have an explicit mechanism that encourages the learning of disentangled representations, but it is theorised that this behaviour is related to the regularisation term and the information bottleneck principle [Burgess *et al.*, 2017; Mathieu *et al.*, 2019; Rolinek *et al.*, 2019]. However, despite this hypothesis, there is still no explicit reason for why the representations learnt by these frameworks should align with generative factors in the data. Nonetheless, these frameworks have been shown to produce disentangled representations when trained on synthetically generated data, as measured by appropriate metrics [Eastwood and Williams, 2018; Chen *et al.*, 2018; Zaidi *et al.*, 2020].

In this paper, we aim to understand why VAEs implicitly learn disentangled representations by investigating the interaction between the reconstruction loss of the VAE and the input data. We provide compelling evidence that disentanglement occurs not because of special algorithmic choices or the regularisation term, but because of how VAEs perceive distances between observations in the datasets themselves according to the reconstruction loss, and the fact that these distances accidentally correlate to the chosen ground-truth factors generating the data. In particular, we find that *standardised benchmarks are constructed in such a way that they unintentionally encourage models to learn what appear to be disentangled representations.*

The main, summarised contributions[1] of this paper are:

(i) We introduce the concept of *perceived distance*, in terms

[1] Code is provided at: https://github.com/nmichlo/disent.

of the VAE reconstruction loss, to measure overlap or similarity between dataset pairs. We demonstrate that perceived distances in existing datasets unintentionally correspond to the distances between ground-truth factors, and that VAEs learn these distances, explaining why learnt representations may appear disentangled.

(ii) We provide a technique to visualise the correlation between perceived distances in the data and ground-truth factors generating the data. We use this understanding to provide a theory for what constitutes an adversarial dataset under a given reconstruction loss.

(iii) We reveal the ineffectiveness of state-of-the-art models by using our theory to design a simple, example adversarial dataset with constant perceived distance between elements, over which VAE-based frameworks fail to learn disentangled representations.

(iv) We provide an example solution to the adversarial dataset that modifies the reconstruction loss, and thus perceived distances across the dataset, so that VAE frameworks are again able to capture the ground-truth factors.

(v) We contribute *Disent*, a general PyTorch [Paszke *et al.*, 2017] disentanglement framework, with common models, metrics, and datasets.[2]

2 Background

Assume a dataset $\mathcal{X} = \left\{\boldsymbol{x}^{(0)}, ..., \boldsymbol{x}^{(n)}\right\}$ is a set of independent and identically distributed (i.i.d) observations $\boldsymbol{x} \in \mathbb{R}^{\mathrm{N}}$, generated by some random process involving an unobserved random variable $\boldsymbol{z} \in \mathbb{R}^{\mathrm{D}}$ of lower dimensionality $\mathrm{D} \ll \mathrm{N}$. Additionally, the true *prior distribution* $\boldsymbol{z} \sim p_*(\boldsymbol{z})$ and true *conditional distribution* $\boldsymbol{x} \sim p_*(\boldsymbol{x}|\boldsymbol{z})$ are unknown. Variational autoencoders (VAEs) aim to learn this generative process. Unlike autoencoders (AEs), which consist of an encoder $f_{\boldsymbol{\phi}}(\boldsymbol{x}) = \boldsymbol{z}$ and decoder $g_{\boldsymbol{\theta}}(\boldsymbol{z}) = \hat{\boldsymbol{x}}$ with weights $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$, VAEs instead construct a probabilistic encoder by using the output from the encoder or inference model to parameterise approximate posterior distributions $\boldsymbol{z} \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$. The approximate posterior is then sampled during training to obtain representations $\boldsymbol{z}$, which are then decoded using the generative model to obtain reconstructions $\hat{\boldsymbol{x}} \sim p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$.

A *factorised Gaussian encoder* [Kingma and Welling, 2014] is commonly used. The posterior is modelled using a multivariate Gaussian distribution with diagonal covariance $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{x}),\ \boldsymbol{\sigma}_{\boldsymbol{\phi}}(\boldsymbol{x}))$, and the prior is given by the multivariate normal distribution $p_{\boldsymbol{\theta}}(\boldsymbol{z}) = \mathcal{N}(\mathbf{0},\ \mathbf{I})$, with a mean of $\mathbf{0}$ and diagonal covariance $\mathbf{I}$. To enable backpropagation, the reparameterisation trick in Equation (1) is used to sample from the posterior distribution by offsetting the distribution means by scaled noise values.[3]

$$\boldsymbol{z} = \boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{x}) + \boldsymbol{\sigma}_{\boldsymbol{\phi}}(\boldsymbol{x}) \odot \boldsymbol{\epsilon},\ \text{where}\ \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \qquad (1)$$

VAEs maximise the evidence lower bound (ELBO) by minimising the loss given by Equation (4). VAE-based approaches often make slight modifications to this loss [Higgins *et al.*, 2016; Zhao *et al.*, 2017; Hou *et al.*, 2017; Kumar *et al.*, 2018; Chen *et al.*, 2018; Kim and Mnih, 2018; Locatello *et al.*, 2020], but the terms of these modified loss functions can usually still be grouped into reconstruction and regularisation components, given by Equations 2 and 3 respectively. The regularisation term affects the representations learnt by the encoder, while the reconstruction term improves the outputs from the decoder. These terms usually contradict in practice, with strong regularisation leading to worse reconstructions but often better disentanglement [Higgins *et al.*, 2016; Burgess *et al.*, 2017].

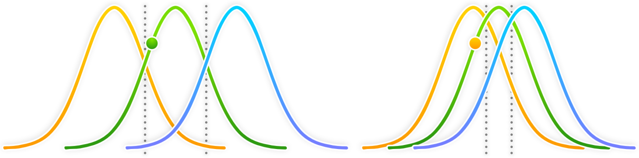


Figure 1: Nearby distributions in the latent space that correspond to different inputs. The VAE reconstructs a sample from the middle distributions. Left: weaker regularisation leads to few sampling mistakes, resembling a lookup table [Mathieu *et al.*, 2019]. Right: strong regularisation leads to more reconstruction mistakes, where samples are attributed to nearby distributions, encouraging reorganisation.

$$\mathcal{L}_{\text{rec}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\right] \qquad (2)$$

$$\mathcal{L}_{\text{reg}}(\boldsymbol{x}) = -D_{\text{KL}}\left(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{z})\right) \qquad (3)$$

$$\mathcal{L}_{\text{VAE}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \mathcal{L}_{\text{rec}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) + \mathcal{L}_{\text{reg}}(\boldsymbol{x}) \qquad (4)$$

2.1 Random Sampling Reorganises VAE Embeddings

Disentanglement in VAEs is generally attributed to the regularisation term in Equation (3); however, we highlight that regularisation only enables the underlying disentanglement mechanism. Disentanglement arises rather as a result of VAEs reorganising the latent space to minimise reconstruction mistakes due to random sampling from the probabilistic encoder during training. Through this mechanism, a VAE will place similar observations according to the reconstruction loss in Equation (2) close together in the latent space [Burgess *et al.*, 2017; Mathieu *et al.*, 2019; Zietlow *et al.*, 2021], as this action minimises sampling errors.

The regularisation term enables this interaction by controlling the overlap between latent distributions corresponding to different inputs. If these distributions overlap sufficiently, the decoder will often attribute a random sample to an incorrect input, see Figure 1. Thus, a mistake will be made during the decoding process, which encourages reorganisation to minimise the reconstruction error.

3 Related Work

The following works are the most applicable to our research, falling under three general categories: (i) explanations for disentanglement, (ii) the role of the reconstruction loss in disentanglement, and (iii) problems with disentanglement.

Firstly, Burgess *et al.*[2017] relate VAEs to the information bottleneck principle. Which explains that random sampling leads to a local minimisation of the reconstruction loss which

[2] Disent framework repository: https://github.com/nmichlo/disent. Code is provided under the MIT license.

[3] The notation $\odot$ represents the element-wise product.

reorganises the latent space so that points close in pixel space are close in the latent space. Mathieu *et al.*[2019] argue that VAEs do not explicitly encourage disentanglement through their design. Rather, they provide the explanation that the diagonal prior typically used in VAEs when combined with random sampling produces a similar effect to PCA. Our work takes inspiration from these ideas to develop the theory of perceived overlap in VAEs, which we use to analyse datasets and improve or hinder disentanglement.

Secondly, inspired by Burgess *et al.*[2017] most modern frameworks offer some way to balance the regularisation and reconstruction components of the loss, the ControlVAE automates this process [Shao *et al.*, 2020]. Hou *et al.*[2017] instead swap out the reconstruction loss of VAEs for a perceptual loss function, which can improve the representations learnt by the model. Zietlow *et al.*[2021] extend the analysis of Mathieu *et al.*[2019]; However, emphasis is placed on constructing adversarial datasets that hinder disentanglement performance using a mild transformation, obtained from trained models which achieve poor disentanglement scores. Our work provides intuition by constructing an example adversarial dataset that targets a specific reconstruction loss, and then remedies this problem by adjusting the loss.

Finally, Locatello *et al.*[2019b] show that useful representations cannot be reliably learnt with unsupervised methods, unless inductive biases are introduced, and Gondal *et al.*[2019] show that representations learnt on synthetic data often do not transfer well to real-world data. Our work investigates the interplay between the reconstruction loss and data as the main bias in VAEs, standard choices accidentally disentangle synthetic data.

4 Existing Disentanglement Datasets

Consider the 3D Shapes dataset [Burgess and Kim, 2018] in Figure 2a, which contains observations of shapes fixed in the centre of the image with progressively changing attributes or factors such as size and colour. If, as humans, we are given unordered observations from a traversal along the size factor of 3D Shapes, it would be easy to order these observations using a perceived increase or decrease in the size of the shape. We might even say that the shapes in the images overlap by different amounts, considering shapes that are closer in size to possess more overlap, and thus also considering them as closer together in terms of distance. This idea naturally extends to VAEs ordering pairs of observations, and so we seek to investigate the correspondence between how these frameworks perceive distances over data points to reorganise the latent space and the ground-truth factors themselves.

4.1 Dataset Ground-truth Distance

Synthetic datasets [Burgess and Kim, 2018; LeCun *et al.*, 2004; Matthey *et al.*, 2017; Reed *et al.*, 2015; Gondal *et al.*, 2019] used for benchmarking disentanglement frameworks are all generated from $\mathrm{F} \in \mathbb{N}^+$ ground-truth factors of variation. Each factor $i \in [\mathrm{F}]$ represents some property about the data that can be varied,[4] and has a dimensionality or size of $f_i > 0$ where $f_i \in \mathbb{N}^+$. The set of all factors used for generating the dataset is written as $\mathcal{Y} = [f_1] \times ... \times [f_\mathrm{F}]$. The full dataset is generated from this set of factors using some ground-truth generative process $\mathcal{X} = \{g_*(\boldsymbol{y}) \mid \boldsymbol{y} \in \mathcal{Y}\}$. Examples of this generative process are given in Figure 2.

With this construction of synthetic datasets, it is fitting to describe the *ground-truth distances* between observations $\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)} \in \mathcal{X}$ using the Manhattan or ℓ_1 distance between their corresponding ground-truth factors $\boldsymbol{y}^{(a)}, \boldsymbol{y}^{(b)} \in \mathcal{Y}$, [5] as in Equation (5). It is important to note that this choice may not be optimal for single factors; rather, ℓ_1 distance naturally aligns with how the datasets are constructed.

$$\mathrm{d}_{\mathrm{gt}}(\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)}) = \|\boldsymbol{y}^{(a)} - \boldsymbol{y}^{(b)}\|_1. \tag{5}$$

4.2 VAE Perceived Distance

With the idea of ground-truth distances between observations, we need a distance measure between observations as perceived by VAE frameworks. We derive the *perceived distance* between dataset elements from the noisy sampling procedure and the chosen reconstruction loss in a VAE framework.

Let $\boldsymbol{z}^{(b)} \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}^{(a)})$ be a (possibly incorrect) sample from the posterior distribution corresponding to some input element $\boldsymbol{x}^{(a)} \in \mathcal{X}$. Since the regularisation term encourages latent distributions to overlap, this sample $\boldsymbol{z}^{(b)}$ may be incorrectly attributed by the decoder to a distribution corresponding to some other element from the dataset $\boldsymbol{x}^{(b)} \in \mathcal{X}$, with reconstruction $\hat{\boldsymbol{x}}^{(b)} \approx \boldsymbol{x}^{(b)}$. As the VAE objective consisting of the regularisation and reconstruction losses is jointly optimised, the decoder becomes better at reconstructing the inputs. In an ideal scenario, the inputs map to outputs ($\hat{\boldsymbol{x}} \to \boldsymbol{x}$), and reconstructions are samples from our dataset: $\hat{\boldsymbol{x}} \in \mathcal{X}$. While this is not the case in practice due to the regularisation term, we derive the perceived distance in Equation (7) from this assumption that $\hat{\boldsymbol{x}} \to \boldsymbol{x}$. This allows us to directly compare the elements $\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)} \in \mathcal{X}$ within a dataset using the reconstruction loss as a distance function:

$$\mathrm{d}_{\mathrm{pcv}}(\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)}) = \lim_{\hat{\boldsymbol{x}} \to \boldsymbol{x}} \mathcal{L}_{\mathrm{rec}}(\boldsymbol{x}^{(a)}, \hat{\boldsymbol{x}}^{(b)}) \tag{6}$$

$$= \mathcal{L}_{\mathrm{rec}}(\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)}). \tag{7}$$

The perceived distance depends on the choice of reconstruction loss, which in literature is usually the pixel-wise Mean Squared Error (MSE) for data that is assumed to be normally distributed. We assume MSE is used throughout the rest of this work, unless specified. Note that analyses are similar for other pixel-wise losses, such as Binary Cross-Entropy (BCE).

4.3 Perceived Distances Correspond to Ground-Truth

In Section 4.1, all the ground-truth factors of a dataset are defined as the set $\mathcal{Y} = [f_1] \times \ldots \times [f_\mathrm{F}]$. In Equation (9), we now define a *factor traversal* $\mathcal{Y}^{(a,i)} \subset \mathcal{Y}$ as the ordered set of all the coordinates along a factor $i \in [\mathrm{F}]$ such that the set passes through a point $\boldsymbol{y}^{(a)} \in \mathcal{Y}$. The number of elements in the traversal is equal to the size of the chosen factor

[4] The bracket notation [F] gives the natural numbers set $\{1, ..., \mathrm{F}\}$.

[5] When indexing $\boldsymbol{y}^{(a)}, \boldsymbol{z}^{(a)}, \boldsymbol{x}^{(a)}$, for convenience a may be an integer or a ground-truth factor $a = \boldsymbol{y}^{(a)}$

$|\mathcal{Y}^{(a,i)}| = f_i$, and each element in the traversal generates the same traversal $\forall \boldsymbol{y}^{(b)} \in \mathcal{Y}^{(a,i)},\ \mathcal{Y}^{(a,i)} = \mathcal{Y}^{(b,i)}$. Figure 2 gives examples of traversals.

$$\begin{aligned} \mathcal{Y}^{(a,i)} &= \ldots \times \left\{ y_{i-1}^{(a)} \right\} \times [f_i] \times \left\{ y_{i+1}^{(a)} \right\} \times \ldots & (8) \\ &= \left\{ (\ldots,\, y_{i-1}^{(a)},\, j,\, y_{i+1}^{(a)},\, \ldots) \mid \forall j \in [f_i] \right\} & (9) \end{aligned}$$

We compute the distance matrix $\tilde{D}^{(a,i)} \in \mathbb{R}^{f_i \times f_i}$, for some distance function d, between pairwise elements along a factor traversal $\mathcal{Y}^{(a,i)}$, written in Equation (10) using matrix notation.

$$\tilde{D}^{(a,i)} = \left(\mathrm{d}(\boldsymbol{x}^{(u)}, \boldsymbol{x}^{(v)}) \right) \in \mathbb{R}^{f_i \times f_i}\ \forall u, v \in \mathcal{Y}^{(a,i)} \quad (10)$$

To examine the ground-truth factors within our datasets, we compute the average distance matrix $D^{(i)} = \mathbb{E}_{a \in \mathcal{Y}}[\tilde{D}^{(a,i)}]$ for each factor $i \in [\mathrm{F}]$. We plot these results in Figure 3 for both the ground-truth distance d_{gt} and perceived distance $\mathrm{d}_{\mathrm{pcv}}$. It is immediately obvious from these plots that the ground-truth distances and the distances perceived by a VAE may accidentally correspond.

Finally, we relate our work to Burgess *et al.*[2017] by outlining a direct approach in our extended paper in the supplementary material for computing relative factor importance using perceived distances. A factor is considered more important if a VAE prefers to learn it before another factor.

4.4 VAEs Learn Perceived Distances

We compute distance matrices over a trained β-VAE at various levels of the network, including the representation layer and reconstructions. At each level of the VAE, the learnt distances all correspond to the original perceived distances already present within the dataset, see Figure 4. Since VAEs reorganise the embedding space according to the perceived distances, and noting results from Section 4.3, VAEs may discover structures that are similar to the underlying ground-truth factors.

Our results in Figures 3 and 4 provide empirical evidence that VAEs mimic the distances already present in the dataset according to the reconstruction loss. To appear disentangled if the goal is factored representations, individual latent units will need to encode portions of the distances that correspond to factors within the data. However, it is known that VAEs with diagonal priors are rotationally invariant [Mathieu *et al.*, 2019], thus the same distances between the means $\boldsymbol{\mu}$ of latent distributions can be learnt for any arbitrary rotation of the latent space. This suggests that VAEs disentangle by accident, since ground-truth factors naturally correspond with distances in the dataset. If these perceived distances were to change such that they do not correspond to the ground-truth distances, VAEs might not be able to learn meaningful representations. This is highlighted by the fact that VAEs are already known to perform poorly on real-world data [Gondal *et al.*, 2019].

5 Adversarial Datasets

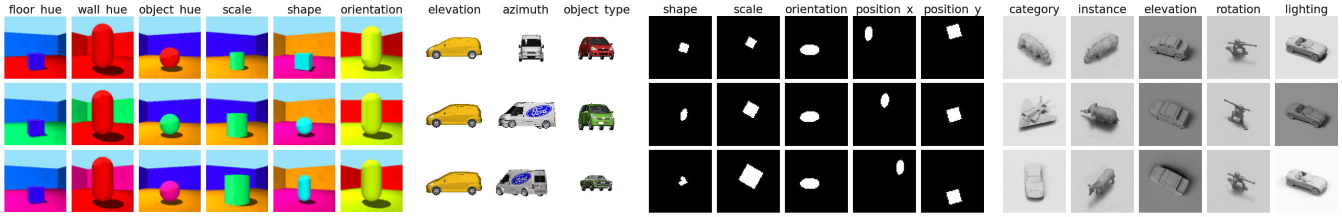
In the previous section, we highlighted the striking similarity between the ground-truth distances and the perceived distances between observations in synthetic ground-truth datasets. This suggests that disentanglement occurs because latent distances accidentally correspond to ground-truth distances, when the latent space is reorganised to minimise reconstruction errors and perceived distances from the data space are captured.

Consider the example of a single chess piece moving across a chess board; there are no smooth transitions between grid points, since the piece is only valid when placed in the middle of squares. We describe such a dataset as having *constant perceived distance*. This property is adversarial in nature as it is impossible for a VAE to order these observations using pixel-wise perceived distance. It is tempting to think that a harder case is if the perceived distances do not correspond to ground-truth distances; however, an (incorrect) ordering can then still be found. Existing datasets such as Cars3D already satisfy this incorrect ordering, which may explain the generally worse disentanglement performance compared to other datasets, see Figure 3.

Formally, we say that a dataset has constant overlap when the pairwise distances over factor traversals are all equal. Let $i \in [\mathrm{F}]$ be a factor and $\boldsymbol{y}^{(a)} \in \mathcal{Y}$ be ground-truth coordinate vector. Then, for all elements over the factor traversal $\forall \boldsymbol{y}^{(b)} \in \mathcal{Y}^{(a,i)} / \left\{ \boldsymbol{y}^{(a)} \right\}$, the corresponding perceived distance is constant such that $\mathrm{d}_{\mathrm{pcv}}(\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)}) = C_f$ with $C_f \in \mathbb{R}$ and $C_f > 0$. Along factor traversals in such a dataset, no distinct ordering of elements can be found when a VAE tries to minimise the sampling error over the reconstruction loss. Going forward, we only consider the case where $\forall f \in [\mathrm{F}],\ C_f = C$ for some $C > 0$.

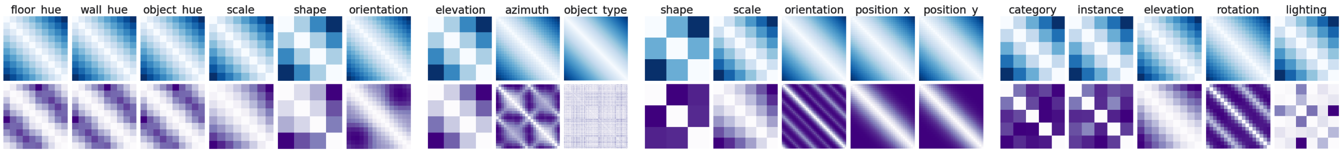
5.1 Example XYSquares Adversarial Dataset

Taking inspiration from the chess piece example, we design a synthetic adversarial dataset called *XYSquares* (See Figure 5)



(a) 3D Shapes (b) Cars3D (c) dSprites (d) NORB

Figure 2: Common existing datasets used to benchmark disentanglement frameworks. These synthetic datasets are generated from ground-truth factors. Columns: represent a traversal along a single factor.



(a) 3D Shapes (b) Cars3D (c) dSprites (d) NORB

Figure 3: Distances in the ground-truth factor space naturally correspond to distances in the data space for current synthetic datasets. Top Row: Average ground-truth distance (ℓ_1) matrices over factor traversals. Bottom Row: Average pixel-wise perceived distance (MSE) matrices over observations from the same factor traversals. Columns: Different ground-truth factors within each dataset.

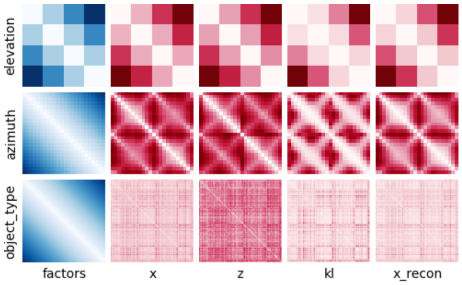


Figure 4: β-VAEs learn similar distances between observations at all levels of the network depending on the reconstruction loss. Rows: Different factors of the Cars3D dataset (Top to bottom: elevation, azimuth, car type), Columns: Distance matrices computed over factor traversals (Left to right: ground-truth distances, perceived distances between observations, ℓ_2 distances over latent distribution means, KL divergences between latent distributions, perceived distances between reconstructions).

that specifically targets VAEs that use a pixel-wise reconstruction loss such as MSE, resulting in constant perceived distances. The dataset consists of three 8×8 pixel squares in a world of size of 64×64. This leaves 8 grid positions along each axis without any pixel-wise overlap. The three squares are each assigned a colour according to R $(1, 0, 0)$, G $(0, 1, 0)$ and B $(0, 0, 1)$ to avoid any channel-wise overlap. With 6 ground-truth factors (three squares moving along two axes), each with 8 possible values, this gives a total dataset size of $8^6 = 262144$ observations. In the rightmost column of Figure 8, we validate that this leads to constant perceived distances between observation pairs in factor traversals.

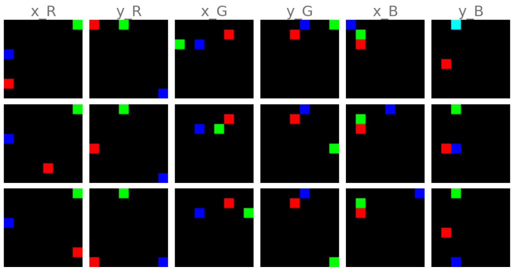


Figure 5: Columns represent ground-truth factor traversals over our adversarial XYSquares dataset. Pixel-wise losses measure constant values along these traversals.

5.2 Experimental Setup

We now investigate the performance of VAEs on our new dataset. In particular, we use the unsupervised β-VAE [Higgins *et al.*, 2016] and the state-of-the-art weakly supervised Ada-GVAE [Locatello *et al.*, 2020]. The β-VAE scales the VAE regularisation term with a coefficient $\beta > 0$, while the Ada-GVAE breaks symmetry and encourages shared latent variables between pairs of observations. This is achieved by averaging together latent distributions between observation pairs that are estimated to remain unchanged when the KL divergence is below some threshold. We note that if the weakly supervised Ada-GVAE performs poorly, then it is highly likely that another unsupervised method will also perform poorly.

We use the same Adam [Kingma and Ba, 2015] optimiser and convolutional neural architecture as Burgess *et al.*[2017]. To evaluate disentangled representations, we use the MIG [Chen *et al.*, 2018] (Mutual Information Gap) and DCI Disentanglement [Eastwood and Williams, 2018] scores. MIG measures the mutual information between the highest and second highest latent units for each factor, and DCI Disentanglement measures how much each latent unit captures a ground-truth factor using a predictive model.

Finally, we perform an extensive hyper-parameter grid search for existing frameworks and datasets before running our own experiments. Hyperparameters include the learning rate, size of the latent dimension, training steps, batch size and β values. See the supplementary material for further details on all experiments conducted throughout the remainder of the paper.

5.3 Example Adversarial Dataset Results

Figure 6 shows that the disentanglement performance over XYSquares is extremely poor compared to existing datasets, even with the state-of-the-art Ada-GVAE. We are concerned only with the maximum score obtained for each model and dataset, as the graph is plotted over the hyper-parameter sweeps. This validates our adversarial dataset hypothesis in Section 5. Not only is the disentanglement performance poor, but much smaller values for β are needed when tuning the regularisation loss. Example latent traversals from a VAE trained over the adversarial dataset are given in Figure 7, results are far from disentangled and do not correspond in any way to the ground-truth factors in Figure 5.

5.4 Example of Varying Levels of Overlap

We have examined the effect of training on existing datasets with significant amounts of overlap, as well as our own adver-

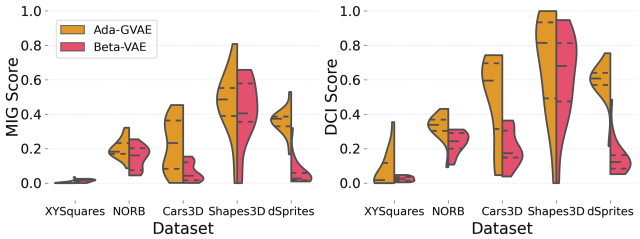


Figure 6: Densities over repeated runs for the attained MIG scores (left) and DCI Disentanglement scores (right) for the weakly-supervised Ada-GVAE (Left half of densities) and β-VAE (Right half of densities). XYSquares hurts the disentanglement performance significantly. Quartiles are marked with horizontal lines. We sweep over β values and latent dimension sizes. See the supplementary material for details.

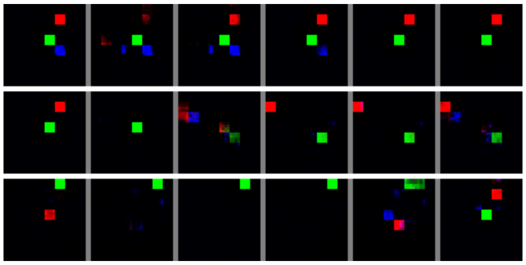


Figure 7: VAEs with pixel-wise losses fail to learn disentangled representations over the XYSquares dataset. Rows show latent traversals over a subset of latent units of a β-VAE. Varying one latent unit does not have an obvious effect or correspond to ground-truth factors.

sarial dataset with constant perceived distances according to pixel-wise losses. However, we have not investigated increasing levels of overlap in datasets, or rather reducing perceived between observations that are also close in ground-truth factor space. To do so, we modify XYSquares by decreasing the spacing between grid points while keeping the number of grid points constant along each factor, ensuring the dataset size remains fixed at $8^6 = 262144$ observations.

The original adversarial dataset, with a spacing of 8, has a constant distance value of $\mathrm{d}_{\text{pcv}}(\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)}) = C_8$. As the spacing s decreases from $8 \rightarrow 1$ over the datasets, the probability increases that any two observations re-sampled along a single factor traversal overlap $p(\mathrm{d}_{\text{pcv}}(\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)}) < C_8)$ and should thus be placed closer together in the latent space. More overlap leads to more unique distance values which in turn allows for easier ordering of data points. We visualise this concept using ground-truth and perceived distance matrices in Figure 8.

We verify our statements through the experimental results in Figure 9, where the β-VAE and Ada-GVAE are trained on these datasets. As the spacing decreases and overlap is introduced, the disentanglement performance improves, since it is easier for a VAE to introduce an ordering over representations. Even for the XYSquares dataset with 1 pixel of overlap between grid points, an ordering of elements along factor traversals can be induced. However, the probability of a VAE encountering these scenarios in the latent space due to random sampling is low, and thus it is still not always easy for the model to learn disentangled representations over such a dataset.

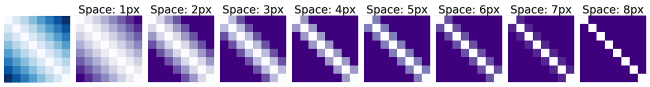


Figure 8: Ground-truth distance matrices (far left) and pixel-wise perceived distance matrices (left to right) over factor traversals. The spacing between grid-points of XYSquares decreases from 8px to 1px, which improves the correlation between perceived distances and ground-truth distances.

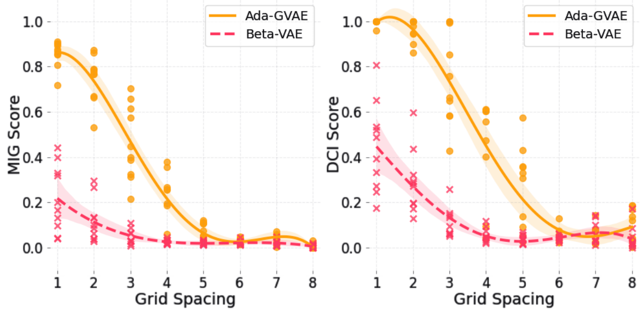


Figure 9: XYSquares spacing vs disentanglment score (MIG – left, DCI Disentanglement – right). Decreasing (left to right) levels of overlap leads to decreased disentanglement performance. Each experiment is repeated 5 times with previously tuned hyper-parameters. See the supplementary material for further details.

6 Example of Introducing Overlap

The previous section focused on increasing overlap by changing the underlying dataset; however, this still does not solve the case for the original XYSquares dataset with constant pixel-wise perceived distance. Throughout this paper, we have provided evidence that VAEs disentangle based on their reconstruction loss, which happens to align with ground-truth factors of variation in common benchmark datasets. This correspondence is not optimal for all tasks and we propose that this leads to the poor disentanglement performance in these settings. Our solution is to choose a loss function that modifies perceived distances such that they also correspond to ground-truth distances.

The new loss function we choose cannot be a pixel-wise approach, as this does not capture the distances due to the spatial nature of the XYSquares dataset. For the sake of simplicity in this example, we convert the existing pixel-wise loss function into a spatially aware loss function by introducing a differentiable augmentation to its inputs. An appropriate augmentation for our dataset is a channel-wise box blur. The problem, however, is that the decoder needs to be able to reconstruct the data, and so purely replacing the pixel-wise loss with the augmented loss may not succeed. Rather, in Equation (11), we append the augmented term to the existing loss and scale it by a constant $\alpha > 0$.

$$\mathcal{L}_{\text{Overlap}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \mathcal{L}_{\text{rec}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) + \alpha\, \mathcal{L}_{\text{rec}}(\text{blur}(\boldsymbol{x}),\, \text{blur}(\hat{\boldsymbol{x}})) \tag{11}$$

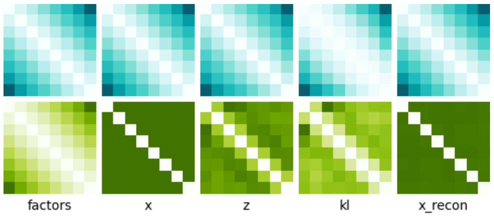


Figure 10: β-VAEs learn similar distances between observations. Top row: box blur augmented MSE. Bottom row: pixel-wise MSE loss. Columns: Distance matrices computed over factor traversals. All factors of XYSquares have the same statistics. This plot is constructed in a similar way to Figure 4.

6.1 Example Augmented Loss Experiments

We choose a channel-wise box blur for our loss function with a radius of 31, or a total kernel size of 63×63. We efficiently implement large filters using the Fast Fourier Transform. The size of the filter ensures that if two observations have active pixels on opposite sides of the images, then overlap will still be introduced between them. We set $\alpha = 63^2$, while this appears large, a box blur kernel is normalised so that the sum of all its values is 1. We accordingly update our perceived distance measure and evaluate the new distances over the XYSquares dataset for each factor in Figure 10 after training and tuning β-VAEs.

Finally, in Figure 11, we compare the performance of the spatially-aware loss function to the original pixel-wise loss. Our new loss significantly improves the disentanglement performance over the adversarial dataset. This is because it allows our models to capture perceived distances between observations that align with the ground-truth factors.

While our choice of loss may not be optimal for disentanglement of these specific x and y factors from our adversarial dataset, disentanglement results are impressive. This is important because it provides the intuition that changing the loss function changes perceived distances and affects the ability of VAE frameworks to learn disentangled representations. We leave learning or identification of optimal reconstruction losses for different datasets, to improve disentanglement, as future work.

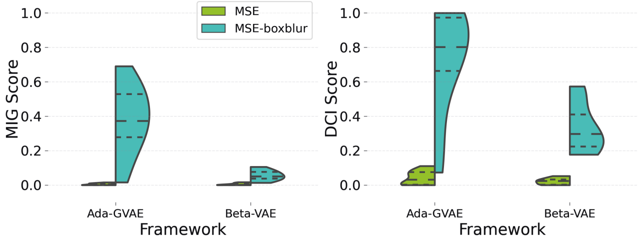


Figure 11: MIG and DCI scores for Ada-GVAE and β-VAE using the MSE loss and our modified loss function. Introducing a spatially aware loss function allows us to capture ground-truth distances between observations and allows the models to disentangle the adversarial XYSquares dataset.

7 Considerations for Disentanglement Research

We highlight the similarity between introducing overlap in Section 6 through the reconstruction loss function and varying levels of overlap in Section 5.4 through modifications to the construction of the dataset itself. Both methods aim to improve disentanglement by changing perceived distances to better correspond to the ground-truth factors, while keeping ground-truth factors fixed.

The problem is that ground-truth factors can indeed change, and this choice, while at the discretion of the researcher, is largely ignored in literature. For example, a researcher may choose RGB, HSV or categorical representations for colours, they may choose binary or continuous encodings for positions, or they may split or merge various factors together.

As our work shows, disentanglement is largely dependent on the chosen reconstruction loss and not special algorithmic choices. Obtaining improved disentanglement results under current VAE disentanglement frameworks will ultimately require supervision from the researcher to adjust perceived distances of the model to the task at hand. This contradicts the current notion that unsupervised and weakly supervised disentanglement methods can automatically uncover these human interpretable ground-truth factors [Higgins *et al.*, 2016].

Ultimately, benchmarking against synthetic datasets with already subjective ground-truth factors will thus always remain problematic. There are infinitely many datasets with infinitely many choices as to what constitutes their ground-truth factors. Accurate disentanglement through future methods may need general world knowledge so that the methods can adapt to the task at hand.

8 Conclusion

In this paper, we demonstrated that there are fundamental characteristics of existing datasets that encourage VAEs to learn disentangled representations. Our work provides a theory for how VAEs perceive distances between pairs of observations in datasets. We used this theory to provide intuition by constructing an adversarial dataset for pixel-wise losses over-which state-of-the-art VAEs fail to learn disentangled representations. Finally, we re-enabled disentanglement over the example adversarial dataset by again adjusting perceived distances, instead through a change of the VAE reconstruction loss to capture the ground-truth factors of the dataset.

Our results highlight issues in current representation learning approaches. We find that the focus on regularisation for disentanglement is misplaced, rather, disentanglement is largely accidental, and careful choice of the reconstruction loss or data is needed to capture the ultimately subjective ground-truth factors. This is impractical in the real world, since perceived distances *cannot* be a prerequisite for true disentanglement. More advanced methods are therefore required that can uncover true meaning within the data.

Acknowledgements

Computations were performed using the High Performance Computing infrastructure provided by the Mathematical Sciences Support unit at the University of the Witwatersrand.

References

[Bengio *et al.*, 2013] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[Burgess and Kim, 2018] Christopher Burgess and Hyunjik Kim. 3D shapes dataset. https://github.com/deepmind/3dshapes-dataset, 2018. Accessed: 2023-01-01.

[Burgess *et al.*, 2017] Christopher Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in β-VAE. In *Workshop on Learning Disentangled Representations at the 31st Conference on Neural Information Processing Systems,*, 2017.

[Chen *et al.*, 2018] Ricky Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, pages 2615–2625, 2018.

[Dittadi *et al.*, 2021] Andrea Dittadi, Frederik Träuble, Francesco Locatello, Manuel Wuthrich, Vaibhav Agrawal, Ole Winther, Stefan Bauer, and Bernhard Schölkopf. On the transfer of disentangled representations in realistic settings. In *International Conference on Learning Representations*, 2021.

[Eastwood and Williams, 2018] Cian Eastwood and Christopher Williams. A framework for the quantitative evaluation of disentangled representations. In *International Conference on Learning Representations*, 2018.

[Gondal *et al.*, 2019] Muhammad Waleed Gondal, Manuel Wuthrich, Djordje Miladinovic, Francesco Locatello, Martin Breidt, Valentin Volchkov, Joel Akpo, Olivier Bachem, Bernhard Schölkopf, and Stefan Bauer. On the transfer of inductive bias from simulation to the real world: a new disentanglement dataset. *Advances in Neural Information Processing Systems*, 32:15740–15751, 2019.

[Higgins *et al.*, 2016] 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. 2016.

[Hou *et al.*, 2017] Xianxu Hou, Linlin Shen, Ke Sun, and Guoping Qiu. Deep feature consistent variational autoencoder. In *2017 IEEE Winter Conference on Applications of Computer Vision*, pages 1133–1141. IEEE, 2017.

[Kim and Mnih, 2018] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International Conference on Machine Learning*, pages 2649–2658. PMLR, 2018.

[Kingma and Ba, 2015] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[Kingma and Welling, 2014] Diederik Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25:1097–1105, 2012.

[Kumar *et al.*, 2018] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations. In *International Conference on Learning Representations*, 2018.

[LeCun *et al.*, 2004] Yann LeCun, Fu Jie Huang, and Leon Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004.

[Locatello *et al.*, 2019a] Francesco Locatello, Gabriele Abbati, Thomas Rainforth, Stefan Bauer, Bernhard Schölkopf, and Olivier Bachem. On the fairness of disentangled representations. In *Advances in Neural Information Processing Systems*, pages 14584–14597, 2019.

[Locatello *et al.*, 2019b] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, pages 4114–4124. PMLR, 2019.

[Locatello *et al.*, 2020] Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *International Conference on Machine Learning*, pages 6348–6359. PMLR, 2020.

[Mathieu *et al.*, 2019] Emile Mathieu, Tom Rainforth, Nana Siddharth, and Yee Whye Teh. Disentangling disentanglement in variational autoencoders. In *International Conference on Machine Learning*, pages 4402–4412. PMLR, 2019.

[Matthey *et al.*, 2017] Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. dSprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset, 2017. Accessed: 2023-01-01.

[Mnih *et al.*, 2015] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[Paszke *et al.*, 2017] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. 2017.

[Reed *et al.*, 2015] Scott Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. Deep visual analogy-making. *Advances in Neural Information Processing Systems*, 28:1252–1260, 2015.

[Rolinek *et al.*, 2019] Michal Rolinek, Dominik Zietlow, and Georg Martius. Variational autoencoders pursue PCA directions (by accident). In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pages 12406–12415, 2019.

[Shao *et al.*, 2020] Huajie Shao, Shuochao Yao, Dachun Sun, Aston Zhang, Shengzhong Liu, Dongxin Liu, Jun Wang, and Tarek Abdelzaher. Controlvae: Controllable variational autoencoder. In *International Conference on Machine Learning*, pages 8655–8664. PMLR, 2020.

[Zaidi *et al.*, 2020] Julian Zaidi, Jonathan Boilard, Ghyslain Gagnon, and Marc-André Carbonneau. Measuring disentanglement: A review of metrics. *arXiv preprint arXiv:2012.09276*, 2020.

[Zhao *et al.*, 2017] Shengjia Zhao, Jiaming Song, and Stefano Ermon. InfoVAE: Information maximizing variational autoencoders. *arXiv preprint arXiv:1706.02262*, 2017.

[Zietlow *et al.*, 2021] Dominik Zietlow, Michal Rolinek, and Georg Martius. Demystifying inductive biases for β-VAE based architectures. *arXiv preprint arXiv:2102.06822*, 2021.

Overlooked Implications of the Reconstruction Loss for VAE Disentanglement Supplementary Material

A Identifying Factor Importance

A factor in a dataset is considered more important if a VAE prefers to learn it before another factor. Burgess *et al.*[2017] identify this order of importance through a slow increase of the information capacity of VAEs during training. We note that simply by looking at the average perceived distance between observations along factor traversals, this ordering can be determined. Factors with a greater average distance will minimise the error in the reconstruction loss due to random sampling the most when learnt first. These factors (or components thereof) will thus generally be preferred.

To compute the average perceived distance along a factor f, we sample a ground-truth coordinate vector $\boldsymbol{y}^{(a)} \in \mathcal{Y}$ and then another random different coordinate vector $\boldsymbol{y}^{(b)} \in \mathcal{Y}^{(a,i)}$ over the traversal for factor f passing through $\boldsymbol{y}^{(a)}$. Note that $\boldsymbol{y}^{(a)} \neq \boldsymbol{y}^{(b)}$. Then, we compute the perceived distance between the corresponding observations $\mathrm{d}_{\text{pcv}}(\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)})$. We repeat this process to compute the expected perceived distance along factor traversals, given by Equation (12).

$$d_i = \mathbb{E}_{a\in\mathcal{Y},\ b\in\mathcal{Y}^{(a,i)},\ a\neq b}\left[\mathrm{d}_{\text{pcv}}(\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)})\right] \tag{12}$$

We determine the factor importance for dSprites as: $d_{\text{x}} \approx 0.058$ and $d_{\text{y}} \approx 0.057$ position, then $d_{\text{scale}} \approx 0.025$, then $d_{\text{shape}} \approx 0.022$, and finally $d_{\text{orientation}} \approx 0.017$. This aligns with the order determined by Burgess *et al.*[2017]. Computing estimates over an entire dataset can be intractable—for our estimates, we sample at least 50000 pairs per factor.

Additionally, we compute the average perceived distance between any random pairs in the datasets (see Equation (13)) and find that the average distance is higher. For dSprites specifically, we have $d_{\text{ran}} \approx 0.075$. This suggests that the ground-truth factors correspond to axes in the data that minimise the reconstruction loss and is further evidence as to why VAEs appear to learn disentangled results.

$$d_{\text{ran}} = \mathbb{E}_{a\in\mathcal{Y},\ b\in\mathcal{Y},\ a\neq b}\left[\mathrm{d}_{\text{pcv}}(\boldsymbol{x}^{(a)}, \boldsymbol{x}^{(b)})\right] \tag{13}$$

A.1 Factor Importance Results

In Appendix A, we relate our work to Burgess *et al.*[2017] by estimating the importance of different factors over the dSprites [Matthey *et al.*, 2017] dataset using the reconstruction loss (MSE) as the perceived distance function between observation pairs.

We compute and list the order of importance of factors from the remaining datasets in Table A1. These importance values are computed as the average perceived distances between 50000 randomly sampled observation pairs taken along random factor traversals. Factors with higher average perceived distances will be prioritised by the model. For comparison, the average distance between any random pair in the dataset is also given. The average distances between pairs along factor traversals are usually less than the random distance, indicating that the ground-truth factors usually correspond to axes in the data that minimise errors.

Dataset	Factor	Mean Dist.	Dist. Std.
Cars3D	**random**	0.0519	0.0188
	azimuth	0.0355	0.0185
	object type	0.0349	0.0176
	elevation	0.0174	0.0100
3D Shapes	**random**	0.2432	0.0918
	wall hue	0.1122	0.0661
	floor hue	0.1086	0.0623
	object hue	0.0416	0.0292
	shape	0.0207	0.0161
	scale	0.0182	0.0153
	orientation	0.0116	0.0079
Small NORB	**random**	0.0535	0.0529
	lighting	0.0531	0.0563
	category	0.0113	0.0066
	rotation	0.0090	0.0071
	instance	0.0068	0.0048
	elevation	0.0034	0.0030
dSprites	**random**	0.0754	0.0289
	position y	0.0584	0.0378
	position x	0.0559	0.0363
	scale	0.0250	0.0148
	shape	0.0214	0.0095
	orientation	0.0172	0.0106
XYSquares	**random**	0.0308	0.0022
	y (R, G, B)	0.0104	0.0000
	x (R, G, B)	0.0104	0.0000

Table A1: Average perceived distances sampled along random factor traversals for different datasets. Components of factors with higher average distances will usually be prioritised by the model.

We visualise the distribution of distances along factor traversals using cumulative frequency plots as in Figure A1. It is interesting to note the distinct shift in structure for the adversarial XYSquares dataset, since distance values are constant depending on the number of differing factors.

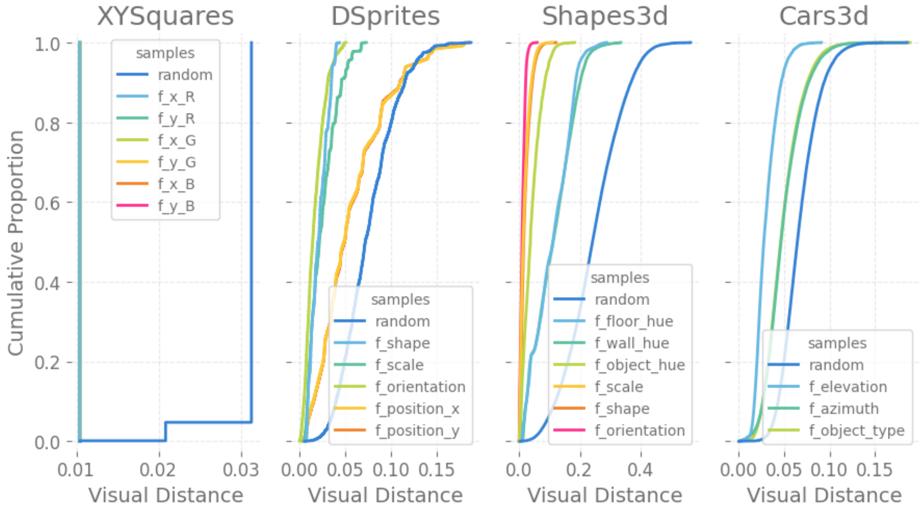


Figure A1: Cumulative proportion of perceived distance values between pairs sampled along factor traversals, compared to perceived distances between random pairs. Factors which are more important for the VAE to learn first to minimise the reconstruction loss have higher average perceived distances (lines shifted further to the right). This corresponds to the experimental results from Burgess *et al.*[2017] which show that as the information capacity of a VAE is increased, it learns factors in order. For dSprites, this is x and y position, followed by scale, then shape, and finally orientation.

B Implementation Details

In this section, we describe our various implementation details of the β-VAE [Higgins *et al.*, 2016] and Ada-GVAE [Locatello *et al.*, 2020] frameworks, as well as the handling and standardisation of the different ground-truth datasets.

B.1 Beta Normalisation

For general consistency across datasets with different numbers of channels and models with different numbers of latent units, we implement beta normalisation as described by Higgins *et al.*[2016].

Instead of taking the sum over the KL divergence in the regularisation term and the sum over elements in the reconstruction term of the VAE loss, we instead compute the means over elements in both terms and adjust the β value accordingly.

B.2 Symmetric KL

The original Ada-GVAE implementation uses the asymmetric KL divergence $D_{\mathrm{KL}}\left(p \,||\, q\right)$ as the distance function between the corresponding latent units of observation pairs. The Ada-GVAE uses this distance measure to estimate which of these latent distributions should be averaged together.

We instead follow the approach of Dittadi *et al.*[2021] and use the symmetric KL divergence to compute these distances between latent units, improving the averaging procedure and computation of the threshold. The symmetric KL divergence is defined in Equation (14).

$$\tilde{D}_{\mathrm{KL}}\left(p,\ q\right) = \frac{1}{2} D_{\mathrm{KL}}\left(p \,||\, q\right) + \frac{1}{2} D_{\mathrm{KL}}\left(q \,||\, p\right) \tag{14}$$

B.3 Sampling Ada-GVAE Pairs

The Ada-GVAE [Locatello *et al.*, 2020] framework introduces weak supervision by sampling pairs of observations such that there are always $k \in [1, \mathrm{F}]$ differing factors between them, where F is the total number of factors generating the dataset. We use the weaker but more realistic case for sampling each pair, where k is sampled uniform randomly from the range $[1, \mathrm{F}]$ as described in the original paper.

B.4 Dataset Standardisation

For improved consistency and training performance, dataset observations are standardised. We first resize the observations to a width and height of 64×64 pixels using bilinear filtering if needed. Then the observations are normalised such that on average each channel of the image has a mean of 0 and a standard deviation of 1. Normalisation constants for each channel are precomputed across the entire dataset and are given in Table A2.

Dataset	**Mean**	**Std**
Cars3D	**R**: 0.897667614997663 **G**: 0.889165802006751 **B**: 0.885147515814868	0.225031955315030 0.239946127898126 0.247921063196844
3D Shapes	**R**: 0.502584966788819 **G**: 0.578759756608967 **B**: 0.603449973185958	0.294081404355556 0.344397908751721 0.366168598152475
Small NORB	0.752091840108860	0.095638790168273
dSprites	0.042494423521890	0.195166458806261
XYSquares	**R**: 0.015625 **G**: 0.015625 **B**: 0.015625	0.124034734589209 0.124034734589209 0.124034734589209

Table A2: Precomputed channel-wise normalisation constants for datasets, assuming values of the input data are in the range $[0, 1]$.

C Experiment Hyper-Parameters

In this section, we give further details on the experiments conducted throughout the paper and their chosen hyper-parameters. For easier comparison with prior work, we use similar hyper-parameters, optimiser and model choices to Higgins *et al.*; Kim and Mnih; Locatello *et al.*[2016; 2018; 2019b].

C.1 Model Architecture

We use similar convolutional encoder and decoder models as Higgins *et al.*[2016]. A full description of the basic VAE architecture is given in Table A3. The Gaussian encoder parameterises the mean and log variance of each latent distribution. The decoder uses the Gaussian derived Mean Squared Error (MSE) as the loss function. The number of input channels the encoder receives and the number of output channels the decoder produces depends on the dataset the model is trained on, this is either 1 or 3 channels.

Encoder		
Input	{1 **or** 3}x64x64	
Conv.	32x4x4	(*stride 2*, ReLU)
Conv.	32x4x4	(*stride 2*, ReLU)
Conv.	64x4x4	(*stride 2*, ReLU)
Conv.	64x4x4	(*stride 2*, ReLU)
Linear	256	(ReLU)
2x Linear	{9 **or** 25}	
Decoder		
Input	{9 **or** 25}	
Linear	256	(ReLU)
Linear	1024	(*reshape 64x4x4*, ReLU)
Upconv.	64x4x4	(*stride 2*, ReLU)
Upconv.	32x4x4	(*stride 2*, ReLU)
Upconv.	32x4x4	(*stride 2*, ReLU)
Upconv.	{1 **or** 3}x4x4	(*stride 2*)

Table A3: VAE encoder and decoder architectures. The model's inputs and outputs change based on the number of channels in the dataset, while the number of latent units the model has depends on the experiment hyper-parameters.

C.2 Optimiser And Batch Size

Models are trained using the Adam [Kingma and Ba, 2015] optimiser with a learning rate of 10^{-3}. A batch size of 256 is used in the case of the β-VAE [Higgins *et al.*, 2016]. Similarly, in the case of the weakly-supervised Ada-GVAE [Locatello *et al.*, 2020], 256 observation pairs are sampled per batch using the strategy from Appendix B.3.

C.3 Experiment Sweeps

Experiment plots and results are all produced from models trained over grid searches of hyper-parameters. Grid search values are given in Table A4. If values are not specified in the hyper-parameter sweep, then default values from the corresponding section of the experiment or supplementary material are used.

C.4 Total Compute

We estimate that approximately ~ 1040 hours of compute across a computing cluster have been used to train the models needed to generate the plots and results presented throughout this paper.

Due to the inherent high variance of unsupervised VAE results, multiple runs using the same hyper-parameters but different random seeds are needed for comparing frameworks [Locatello *et al.*, 2019b]. This susceptibility of unsupervised methods to the starting random seed makes extended comparisons between frameworks prohibitive due to the computational cost.

Experiment	Total	Hyper-Parameters
5.3. Example Adversarial Dataset Results (Figure 6)	$8 \times 2 \times 2 \times 5$ $= 160$ $\times 1$ repeats $= 160$ $\times \sim 4$h ≈ 640h	train steps $= 115200$ beta $(\beta) \in \{0.000316, 0.001, 0.00316,$ $0.01, 0.0316, 0.1, 0.316, 1.0\}$ framework $\in \{\beta$-VAE, Ada-GVAE$\}$ latents $(\mathrm{D}) \in \{9, 25\}$ dataset $\in \{$dSprites, 3D Shapes, Cars3D, Small NORB, XYSquares$\}$
5.4. Example of Varying Levels of Overlap (Figure 9)	$2 \times 2 \times 8$ $= 32$ $\times 5$ repeats $= 160$ $\times \sim 2$h ≈ 320h	train steps $= 57600$ beta $(\beta) \in \{0.001, 0.00316\}$ framework $\in \{\beta$-VAE, Ada-GVAE$\}$ latents $(\mathrm{D}) = 9$ dataset $=$ XYSquares grid spacing $\in \{8, 7, 6, 5, 4, 3, 2, 1\}$
6.1. Example Augmented Loss Experiments (Figure 11)	$2 \times 2 \times 2$ $= 8$ $\times 5$ repeats $= 40$ $\times \sim 2$h ≈ 80h	train steps $= 57600$ beta $(\beta) \in \{0.0001, 0.0316\}$ framework $\in \{\beta$-VAE, Ada-GVAE$\}$ latents $(\mathrm{D}) = 25$ dataset $=$ XYSquares recon. loss $\in \{$MSE, BoxBlurMSE$\}$ box blur radius $= 31$ (63x63 in size) box blur weight $= 63^2 = 3969$

Table A4: Grid search hyper-parameters used for the different experiments throughout this paper.