# Why do Variational Autoencoders *Really* Promote Disentanglement?

**Pratik Bhowal** [1] **Achint Soni** [2] **Sirisha Rambhatla** [3]

## Abstract

Despite not being designed for this purpose, the use of variational autoencoders (VAEs) has proven remarkably effective for disentangled representation learning (DRL). Recent research attributes this success to certain characteristics of the loss function that prevent latent space rotation, or hypothesize about the orthogonality properties of the decoder by drawing parallels with principal component analysis (PCA). This hypothesis, however, has only been tested experimentally for linear VAEs, and the theoretical justification still remains an open problem. Moreover, since real-world VAEs are often inherently non-linear due to the use of neural architectures, understanding DRL capabilities of real-world VAEs remains a critical task. Our work takes a step towards understanding disentanglement in real-world VAEs to theoretically establish how the orthogonality properties of the decoder promotes disentanglement in practical applications. Complementary to our theoretical contributions, our experimental results corroborate our analysis. Code is available at https://github.com/criticalml-uw/Disentanglement-in-VAE.

## 1. Introduction

Learning human interpretable concepts in generative modeling is crucial for their reliable and controllable application in real-world scenarios (Voynov & Babenko, 2020; Härkönen et al., 2020). A promising avenue in this realm is *Disentangled Representation Learning* (DRL), which aims to uncover the hidden factors of variation in the observed data. A widely accepted definition of DRL posits that each latent variable should encode a single generative factor of the data, as depicted in Fig. 1(Higgins et al., 2018)[1] This characteristic makes Disentangled Representations (DRs) particularly useful for learning interpretable latent spaces (Shen et al., 2022; Nie et al., 2023; Han et al., 2022). Furthermore, this emphasis on interpretability makes DRs a key tool for latent space manipulation, which is essential not only in fields like computer vision (Stammer et al., 2022; Wang et al., 2023; He et al., 2023; Li et al., 2022; Ruan et al., 2022), but also in text based media generation (Wu et al., 2023).

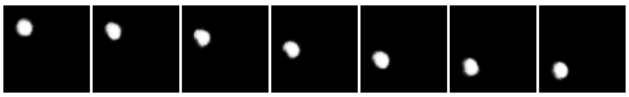


*Figure 1.* Varying a single latent variable, keeping all the other latent variables the same in the dSprites dataset changes only the vertical position of the object in the images while keeping the rest of its attributes constant demonstrating disentanglement.

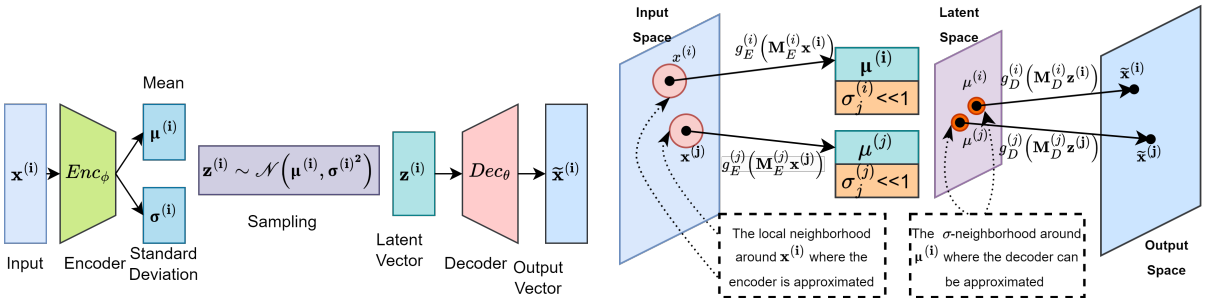
Variational Autoencoders (VAEs) are widely used to learn DRs, owing to their probabilistic encoder-decoder structure and a latent space well-suited for generative modeling. Many benchmark DRL architectures, like $\beta$-VAE (Higgins et al., 2016), DIP-VAE (Kumar et al., 2017), and $\beta$-TCVAE (Chen et al., 2018b), are rooted in the VAE framework. These architectures generally outperform GAN-based methods, such as InfoGAN (Chen et al., 2016) and DR-GAN (Tran et al., 2017) particularly in terms of stability and quality of generation. Furthermore, recent works in promoting disentanglement in diffusion models also utilize this VAE-like probabilistic encoder-decoder model (Zhang et al., 2022; Yang et al., 2023), underscoring the need to understand the mechanisms by which DRL is promoted by VAEs.

While VAEs have found success for DRL, somewhat surprisingly they were not designed for this representation learning task. Consequently, explaining the DRL properties of VAEs has been a focal point in recent research (Burgess

[1]NVIDIA, India [2]Department of Computer Science, University of Waterloo, Ontario, Canada [3]Department of Management Science and Engineering, University of Waterloo, Ontario, Canada. Correspondence to: Pratik Bhowal <pratikbhowal1999@gmail.com>, Sirisha Rambhatla <sirisha.rambhatla@uwaterloo.ca>.

*Proceedings of the $41^{st}$ International Conference on Machine Learning*, Vienna, Austria. PMLR 235, 2024. 

[1]Some works (Locatello et al., 2019; 2020), propose that DRL techniques use inherent biases present in data and the disentanglement metrics to disentangle as more than one set of generative factors can generate statistically identical data.




(a) Variational Autoencoder (VAE) Architecture (b) Local Approximations

*Figure 2.* Panel (a) illustrates a typical VAE setting. On the right, Panel (b) shows the *local* approximation-based VAE used in our analysis.

et al., 2018; Chen et al., 2018b; Rolinek et al., 2019). As shown in Fig. 2(a), a VAE comprises of a probabilistic encoder, $Enc_\phi : \mathbb{X} \to \mathbb{Z}$ and a decoder $Dec_\theta : \mathbb{Z} \to \mathbb{X}$. Here, $\mathbb{X} \in \mathbb{R}^n$ and $\mathbb{Z} \in \mathbb{R}^d$ represent the data space and latent space, respectively. In a setting with a dataset $\{\boldsymbol{x}^{(i)}\}_{i=1}^N$ of N elements where $\boldsymbol{x}^{(i)} \in \mathbb{X}$, employing a fixed Gaussian prior $p(\boldsymbol{z}^{(i)})$ over $\mathbb{Z}$ such that $\boldsymbol{z}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and the reconstructed data points, $\tilde{\boldsymbol{x}}^{(i)}$, obtained using the decoder $Dec_\theta(\boldsymbol{z}^{(i)})$, Kingma & Welling (2013) introduced marginalized log-likelihood as the idealized loss function, which needs to be maximized for training VAEs:

$$\sum_{i=1}^{N} \log(p(\boldsymbol{x}^{(i)})) \tag{1}$$

It is interesting to note that, the rotational symmetry inherent in this formulation (due to the Gaussian prior) is in fact detrimental for DRL. This is because disentangled latent spaces require precise alignment, which can be disrupted by any rotational invariance. The log-likelihood loss, however, is not tractable and is approximated by the evidence lower bound (ELBO) loss function, defined as follows:

$$\begin{aligned} L := & \underbrace{\sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \mathbb{E}_{\boldsymbol{z}^{(i)} \sim q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})}[\log(p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)}))]}_{L_{MLE}} \\ & \underbrace{- \beta \sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} D_{\mathrm{KL}}(q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})||p(\boldsymbol{z}^{(i)}))}_{L_{KL}}, \end{aligned} \tag{2}$$

where the first term is log-likelihood loss $L_{MLE}$ (acts as the reconstruction loss and is approximated by the squared error loss, $L_{rec}$, in most cases), and the second term is KL divergence loss $L_{KL}$, which calculates the similarity between the diagonal posterior probability generated by the encoder as $\boldsymbol{z}^{(i)} \sim Enc_\phi(\boldsymbol{x}^{(i)}) \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)}) = \mathcal{N}(\mu_\phi(\boldsymbol{x}^{(i)}), diag(\sigma^2_\phi(\boldsymbol{x}^{(i)})))$ and the symmetric Gaussian prior probability, $p(\boldsymbol{z}^{(i)})$; $\phi$ are neural network parameters.

Since $q_\phi(\cdot)$ is not rotationally invariant the ELBO loss function is not symmetric. Using this, Rolinek et al. (2019) demonstrates that optimizing the stochastic component of the squared error reconstruction loss ($L_{rec}$) in ELBO can promote local orthogonality in the decoder. The authors further show that this induces PCA-like behavior in the decoder, which, along with the diagonal posterior, aids the VAE in learning DRL. However, Rolinek et al. (2019)'s argument is not entirely sufficient to equate linear VAEs with PCA, as noted by Zietlow et al. (2021). Following this insight, Zietlow et al. (2021) approximate the action of VAE as being *locally linear*, providing a more clear association between PCA and linear VAEs. Our key insight is that while the local linearity assumption of Zietlow et al. (2021) and linearization of Rolinek et al. (2019) have provided a way to understand why VAEs work for DRL, these are restrictive since real-world generation critically relies on non-linearity.

With regard to these previous analyses, our inquiry centers on several pivotal questions that guide the focus of our research: beyond the diagonal posterior characteristic of VAEs, what other mechanisms actively contribute to disentanglement in DRL? Are the assumptions of local linearity, as discussed in Zietlow et al. (2021), and the linearization of the loss function, as in Rolinek et al. (2019), truly adequate for capturing the complexity of DRL? In what ways does the orthogonality within the local decoder's matrix, $\boldsymbol{M}_D^{(i)}$, facilitate the disentanglement process?

In summary, in this paper we investigate VAE to explain disentanglement and our contributions are as follows:

- **Reassessing local linearity in VAEs:** We show that local linearity assumption and the linearization of the stochastic component of the reconstruction loss ($L_{rec}$) are not adequate for learning DRs in practice,
- **Introducing local non-linearity in the VAE decoder:** We present a novel approach by modeling the VAE decoder action as a composition of non-linear and linear functions ($\boldsymbol{M}_D^{(i)}$). Our analysis reveals that this modeling promotes orthogonality in $\boldsymbol{M}_D^{(i)}$, a perspective not previously explored.

- **Linking orthogonality and disentanglement:** We provide theoretical and empirical evidence to confirm the role of orthogonality in $\boldsymbol{M}_D^{(i)}$ for disentanglement.

## 2. Related Works

### 2.1. Disentangled representation learning

The need for interpretibility, transparency, and causality in generative modeling and deep learning models at large have motivated recent works to argue for the need for human-like learning (Lake et al., 2017), representations and understanding of the world (Bengio et al., 2013) and causal inference (Peters et al., 2017). DRL (Gonzalez-Garcia et al., 2018; Jha et al., 2018; Achille & Soatto, 2018; Hu et al., 2018) try to mimic this type of representation learning while also focusing on interpretable latent space.

This interpretability causes DRL to be used in a number of domains including adversarial training (Szabó et al., 2017; Mathieu et al., 2016), synergy task specific sparse predictors (Lachapelle et al., 2023), novel graph neural network framework to learn the causal and bias substructure (Fan et al., 2022), temporally disentangled representation learning (Yao et al., 2022), constrained latent variables (Engel et al., 2017; Bojanowski et al., 2017), and machine learning applications including fairness DRL (Creager et al., 2019; Song et al., 2019) interpretability of machine learning models (Adel et al., 2018; Bengio et al., 2013; Higgins et al., 2016), downstream tasks (Locatello et al., 2019; Gao et al., 2019) and computer vision tasks (Shu et al., 2017; Liao et al., 2020). Computer Vision applications include DRL for one-shot talking head synthesis (Wang et al., 2023), zero-shot segmentation (He et al., 2023), disentanglement of geometric information for cross-view geo-localization (Zhang et al., 2023), point-of-interest recommendation (Qin et al., 2023), synergy with enhanced semantic alignment for image-based 3D model retrieval (Nie et al., 2023), unsupervised domain adaptation (Xie et al., 2022), person identification (Jia et al., 2022; Li et al., 2022), facial expression recognition (Ruan et al., 2022) and medical imaging (Han et al., 2022; Xie et al., 2022).

### 2.2. Variational autoencoder architectures for disentanglement

VAE (Kingma & Welling, 2013), a cornerstone architecture of DRL has been modified to $\beta$-VAE (Higgins et al., 2016), by introducing a hyperparameter, $\beta$ to tradeoff between reconstruction and regularization as shown in (2). To improve the architecture, while Factor-VAE (Kim & Mnih, 2018) and $\beta$-TC-VAE (Chen et al., 2018a) introduced statistical independence in the latent space, (Jeong & Song, 2019) decoupled jointly modeling the continuous and discrete factors of data. Recent developments include learning a controllable generative model in guided-VAE (Ding et al., 2020), sequential variational autoencoder under self-supervision in sequential data in S3VAE (Zhu et al., 2020) and multi-VAE (Xu et al., 2021) a VAE-based multi-view clustering framework which uses disentangled representations. Subsequent works used factorized priors conditionally dependent on auxiliary variables in (Khemakhem et al., 2020; Mita et al., 2021), commutative lie group in (Zhu et al., 2021), and sparse temporal prior in (Klindt et al., 2020).

### 2.3. Inner workings of VAE-based disentanglement architectures

The success of VAE-based architectures inspired researchers to understand its underlying principle. Whereas (Burgess et al., 2018) used the information bottleneck principle, (Kumar & Poole, 2020) studied the regularization effect of the variational family on the local geometry of the decoding model to explain $\beta$-VAEs. Following this (Rolinek et al., 2019) showed the local orthogonality of the decoder matrix, and (Zietlow et al., 2021) showed that the local alignment of the latent space in a VAE is similar to that of PCA. However, these works are based on the linearization of the VAE decoder around a point which does not explain practical scenarios. In our work, we introduce local non-linearity. Further, these works did not establish why the the decoder's local orthogonality promotes disentanglement but rather provided only experimental evidence. In our work, we answer this question.

## 3. From Local Linearity to Introducing Non-linearity

This section models the VAE locally as a composition of linear transformations (represented by matrices) and non-linear transformations (represented by non-linear functions). We demonstrate that minimizing the stochastic component of the reconstruction loss leads to orthogonality among the columns of the matrix representing the linear part of the local VAE decoder. Further, we show why orthogonality is key to ensuring disentanglement. Finally, we explain how the latent variables are selected for each generative factor.

### 3.1. The Problem Formulation

We start by defining the data points as $\{\boldsymbol{x}^{(i)}\}_{i=1}^N \in \mathbb{X}$ and the latent variable $\boldsymbol{z}^{(i)}{}_{i=1}^N \in \mathbb{Z}$ such that $\mathbb{X} \in \mathbb{R}^n$ and $\mathbb{Z} \in \mathbb{R}^d$. The encoder function, denoted as $Enc_\phi(\boldsymbol{x}^{(i)})$, is modeled as a Gaussian distribution: $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)}) = \mathcal{N}(\mu_\phi(\boldsymbol{x}^{(i)}), \text{diag}(\sigma_\phi^2(\boldsymbol{x}^{(i)})))$. In this model, the latent space points $\boldsymbol{z}^{(i)}$ are drawn from the distribution $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$. Through the reparametrization

trick, we represent $\boldsymbol{z}^{(i)}$ as $\mu_\phi(\boldsymbol{x}^{(i)}) + \boldsymbol{\epsilon} \cdot diag(\sigma_\phi(\boldsymbol{x}^{(i)}))$, where $\boldsymbol{\epsilon}$ is sampled from $\mathcal{N}(\mathbf{0}, \mathbf{I})$.

The reconstructed data points, $\tilde{\boldsymbol{x}}^{(i)}$, are obtained using the decoder $Dec_\theta(\boldsymbol{z}^{(i)})$. For simplicity in notation, we denote $\mu_\phi(\boldsymbol{x}^{(i)})$ as $\boldsymbol{\mu}^{(i)} \in \mathbb{R}^d$, $diag(\sigma_\phi(\boldsymbol{x}^{(i)}))$ as $\boldsymbol{\sigma}^{(i)} \in \mathbb{R}^d$, and $\boldsymbol{\epsilon} \cdot \text{diag}(\sigma_\phi(\boldsymbol{x}^{(i)}))$ as $\boldsymbol{\epsilon}^{(i)} \in \mathbb{R}^d$, with $\boldsymbol{\epsilon}^{(i)}$ following a Gaussian distribution $\mathcal{N}(\mathbf{0}, {\boldsymbol{\sigma}^{(i)}}^2)$, which means that

$$\boldsymbol{z}^{(i)} = \boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)},$$

Lastly, we will use the notation $\mathbb{E}_i$ to denote expectations over the index $i$.

From 2, the total loss function is a combination of the KL-Divergence loss ($L_{KL}$) and the reconstruction loss ($L_{MLE}$). It is expressed as follows:

$$L := \sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \left[ L_{MLE}^{(i)} - \beta L_{KL}^{(i)} \right] \tag{3}$$

Since we model both $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ and $p(\boldsymbol{z}^{(i)}))$ to be Gaussian distributions, the KL-Divergence Loss, is given by (Detailed proof in Proposition 2 of Appendix A.2 ):

$$L_{KL}^{(i)} := \frac{1}{2} \sum_j ({\mu_j^{(i)}}^2 + {\sigma_j^{(i)}}^2 - \log({\sigma_j^{(i)}}^2) - 1)$$

Again, assuming $p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)})$ to be a Gaussian distribution, we define it as

$$p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)}) = \mathcal{N}(Dec_\theta(\boldsymbol{z}^{(i)}), \Sigma_\theta),$$

where $\Sigma_\theta = \text{diag}(\sigma_\theta^2(\boldsymbol{z}^{(i)}))$ and $\sigma_\theta^2(\boldsymbol{z}^{(i)})$ is the variance of the decoder distribution for every $\boldsymbol{z}^{(i)}$. The log-likelihood $L_{MLE}^{(i)}$ can then be written as follows: (Detailed proof in Proposition 3 of Appendix A.2 ):

$$L_{MLE}^{(i)} = -\frac{\log(2\pi)}{2} - \frac{\log(|\Sigma_\theta|)}{2} - \mathbb{E}_{\tilde{\boldsymbol{x}}^{(i)}} \left[ \frac{||\boldsymbol{x}^{(i)} - \tilde{\boldsymbol{x}}||^2 \Sigma_\theta^{-1}}{2} \right],$$

where $\tilde{\boldsymbol{x}}^{(i)} = Dec_\theta(\boldsymbol{z}^{(i)})$. Following most previous works on VAE before us, (Rolinek et al., 2019; Zietlow et al., 2021) we approximate $L_{MLE}^{(i)}$ using squared error loss ($L_{rec}^{(i)}$) as follows:

$$L_{rec}^{(i)} := -\mathbb{E}_{\tilde{\boldsymbol{x}}^{(i)}} \left[ ||\tilde{\boldsymbol{x}}^{(i)} - \boldsymbol{x}^{(i)}||^2 \right]$$

Hence, we can write 3 as follows:

$$L := \sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \Big[ -\mathbb{E}_{\tilde{\boldsymbol{x}}^{(i)}} \left[ ||\tilde{\boldsymbol{x}}^{(i)} - \boldsymbol{x}^{(i)}||^2 \right] - \frac{\beta}{2} \sum_j ({\mu_j^{(i)}}^2 + {\sigma_j^{(i)}}^2 + \log({\sigma_j^{(i)}}^2) + 1) \Big] \tag{4}$$

Since the objective is to maximize Equation 4, we eliminate the negative signs from this equation to facilitate minimization. Consequently, the final loss function that we utilize can be stated as follows:

$$\begin{aligned} L &:= \sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \left[ L_{rec}^{(i)} + \beta L_{KL}^{(i)} \right] \text{ where} \\ L_{rec}^{(i)} &= \mathbb{E}_{\tilde{\boldsymbol{x}}^{(i)}} \left[ ||\tilde{\boldsymbol{x}}^{(i)} - \boldsymbol{x}^{(i)}||^2 \right] \text{ and} \\ L_{KL}^{(i)} &= {\mu_j^{(i)}}^2 + {\sigma_j^{(i)}}^2 - \log({\sigma_j^{(i)}}^2) + 1 \end{aligned} \tag{5}$$

By substituting the value of $\boldsymbol{z}^{(i)}$ as $\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}$ with $\boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(\mathbf{0}, {\boldsymbol{\sigma}^{(i)}}^2)$, the reconstruction loss becomes:

$$L_{rec}^{(i)} := \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}} \left[ ||Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - \boldsymbol{x}^{(i)}||^2 \right] \tag{6}$$

Since this paper aims to show that minimizing the stochastic part of the reconstruction loss while fixing the deterministic part and the KL-divergence loss promotes orthogonality and consequently disentanglement, we decompose the reconstruction loss into a stochastic and a deterministic part using Prop. 1.

**Proposition 1.** *Given* $L_{rec}^{(i)} := \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[||Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - \boldsymbol{x}^{(i)}||^2]$, *and assuming that the stochastic estimate,* $Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)})$ *is unbiased around* $Dec_\theta(\boldsymbol{\mu}^{(i)})$, $L_{rec}^{(i)}$ *can be decomposed into deterministic and stochastic parts:*

$$\begin{aligned} L_{rec}^{(i)} &= L_{rec}^{\mu^{(i)}} + L_{rec}^{stoch^{(i)}}, \text{ where,} \\ L_{rec}^{stoch^{(i)}} &:= \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}} ||Dec_\theta(Enc_\phi(\boldsymbol{x}^{(i)})) - Dec_\theta(\boldsymbol{\mu}^{(i)})||^2, \\ L_{rec}^{\mu^{(i)}} &:= \left[ ||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||^2 \right] \end{aligned} \tag{7}$$

To simplify the losses, we define the *polarized regime* as proposed by Rolinek et al. (2019) in Def. 3.1, and assume that the VAE operates in this *polarized regime*.

**Definition 3.1.** *A Variational Autoencoder (VAE), having an encoder* $Enc_\phi$ *and a decoder* $Dec_\theta$, *is said to be operating in a polarized regime if the latent variables can be divided into a set of active* ($\mathbb{V}_a$) *and passive* ($\mathbb{V}_p$) *variables. Here:*

- *For passive variables* $j \in \mathbb{V}_p$, $\mu_j^2(\boldsymbol{x}^{(i)}) \ll 1$ *and* $\sigma_j^2(\boldsymbol{x}^{(i)}) \approx 1$, *while for active variables* $j \in \mathbb{V}_a$, *and* $\sigma_j^2(\boldsymbol{x}^{(i)}) \ll 1$.
- *The decoder ignores the passive latent components, i.e.,* $\frac{\partial Dec_\theta(\boldsymbol{z}^{(i)})}{\partial \boldsymbol{z}_j^{(i)}} = 0$ *for all* $j \in \mathbb{V}_p$.

This definition categorizes latent variables into two groups: passive latent variables, which provide minimal information about the data point, and active latent variables, which convey significant information about the data point.

In our further analysis, we make a crucial approximation regarding the local behaviour of the encoder and decoder in

the VAE. We model the local decoder and the deterministic parts of the local encoder as non-linear functions. These can be succinctly represented as $Dec_\theta(\boldsymbol{z}^{(i)}) = g_D^{(i)}(\boldsymbol{M}_D^{(i)}\boldsymbol{z}^{(i)})$ and $\boldsymbol{\mu}^{(i)} = g_E^{(i)}(\boldsymbol{M}_E^{(i)}\boldsymbol{x}^{(i)})$. Here, $\boldsymbol{M}_E^{(i)}$ and $\boldsymbol{M}_D^{(i)}$ denote the linear transformations, while $g_E^{(i)}$ and $g_D^{(i)}$ represent the respective nonlinearities in a local neighborhood around $\boldsymbol{x}^{(i)}$ and in the $\sigma$-neighborhood of $\boldsymbol{\mu}^{(i)}$. This approximation is visually contrasted in Fig. 2, where the standard VAE architecture is depicted on the left, and our locally approximated VAE model is shown on the right.

The approximation hinges on all practical VAE operating in a polarized regime, where the variance of active latent variables are significantly small (${\sigma_j^{(i)}}^2 \ll 1$) and having finite valued local linear decoders, $\boldsymbol{M}_D^{(i)}$, $\boldsymbol{M}_D^{(i)}\boldsymbol{\epsilon}^{(i)} \ll 1$. Assuming $g_D(\cdot)$ to be the local decoder, we can approximate the decoder $Dec_\theta(\cdot)$ around $\boldsymbol{\mu}^{(i)}$ using Taylor series expansion as follows; proof in Prop. 4 App. A.2.

$$\text{Dec}_\theta(\boldsymbol{z}^{(i)}) = g_D^{(i)}(\boldsymbol{M}_D^{(i)}\boldsymbol{z}^{(i)}) \approx \text{Dec}_\theta(\boldsymbol{\mu}^{(i)}) + f_D^{(i)}(\boldsymbol{M}_D^{(i)}\boldsymbol{\epsilon}^{(i)}) \tag{8}$$

This expression, while an approximation, is grounded in the framework of polarized regimes and the practical behaviour of VAEs under these conditions. For the remainder of the paper, we will refer to $f_D^{(i)}$ simply as $f_D$ and $\boldsymbol{M}_D^{(i)}$ as $\boldsymbol{M}_D$.

Next, $L_{rec}^{stoch^{(i)}}$ is further simplified according to Lem. 1.

**Lemma 1.** *With the approximation of the decoder being locally non-linear such that it can be expressed as $g_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})$, $L_{rec}^{stoch^{(i)}}$ can be expressed as follows:*

$$L_{rec}^{stoch^{(i)}} = \sum_{j=1}^{n} \big\{ var[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] + f_D^2(\mathbf{0}) + f_D(\mathbf{0}) f_D''(\mathbf{0}) var[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] \big\} \tag{9}$$

**Lemma 2.** *Given the local decoder matrix $\boldsymbol{M}_D = \boldsymbol{U}_D\boldsymbol{\Sigma}_D\boldsymbol{V}_D^\top$, local encoder matrix $\boldsymbol{M}_E = \boldsymbol{U}_E\boldsymbol{\Sigma}_E\boldsymbol{V}_E^\top$, local decoder non-linearity $g_D$, local encoder non-linearity $g_E$ , the minimization of $L_{rec}^{\mu^{(i)}}$ depends either only on $\boldsymbol{V}_E$ or only on $\boldsymbol{U}_D$ and $f_D$, i.e., fixing $L_{rec}^{\mu^{(i)}}$ fixes $\boldsymbol{V}_E$, $\boldsymbol{U}_D$ and $f_D$.*

We use Def. 3.1, and Lem. 2 to simplify $L_{KL}$ in Lem. 3

**Lemma 3.** *Fixing the deterministic part of the reconstruction loss ($L_{rec}^{\mu^{(i)}}$) and assuming the VAE is operating in a polarized regime, $L_{KL}$ can be expressed as:*

$$L_{KL} = \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \sum_{j\in\mathbb{V}_a} -\log({\sigma_j^{(i)}}^2) = \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{KL}^{(i)}$$

We define the parameter $C_{KL}$ to investigate its effect on the minimized stochastic reconstruction loss $L_{rec}^{stoch^*}$. While minimizing $L_{rec}^{stoch}$, we maintain $L_{KL}$ as $L_{KL} = C_{KL}$.

### 3.2. Minimizing the stochastic part of the reconstruction loss promotes orthogonal columns in $\boldsymbol{M}_D$

We propose that minimizing $\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{rec}^{stoch^{(i)}}$ while fixing $\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{rec}^{\mu^{(i)}}$ and $\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{KL}^{(i)}$ promotes the columns of $\boldsymbol{M}_D$ to be orthogonal. According to Lemma 2, fixing $\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{rec}^{\mu^{(i)}}$ fixes $\boldsymbol{V}_E$, $\boldsymbol{U}_D$, and $f_D$. Consequently, the minimization process can only be carried out by adjusting $\boldsymbol{V}_D$ and $\boldsymbol{\Sigma}_D$. Thus, based on (9), Lemma 1, 2, and 3, and $C_{KL}$, we formulate the optimization problem as presented in (10) and via the following result.

**Theorem 1.** *Given independent data samples $\mathbf{x}^{(i)}$, if we fix the $\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{KL}^{(i)}$ for a constant $C_{KL}^{(i)}$, and $\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{rec}^{\mu^{(i)}}$, then the minimization of the VAE loss $L$ in (5) reduces to the minimization of the stochastic reconstruction loss $L_{rec}^{stoch^{(i)}}$:*

$$\min_{\sigma_j^{(i)}>0, \boldsymbol{V}_D,} \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log L_{rec}^{stoch^{(i)}} \quad s.t. \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{KL}^{(i)} = C_{KL}. \tag{10}$$

*Then, the following hold for the local minima:*

*(a) Every local minimum is a global minimum.*
*(b) In every global minimum, the columns of every $\boldsymbol{M}_D$ are orthogonal.*

*Further, the variance of a latent variable is inversely proportional to the norm of the corresponding column in the linear part of the local decoder:*

$$(c) \qquad {\sigma_j^{(i)}}^2 \propto \frac{1}{||\boldsymbol{c}_j||^2} \ \ \forall i$$

*where $\boldsymbol{c}_j$ is the $j$-th column of $\boldsymbol{M}_D$.*

Proof given in A.4. Note that, even though the columns of the linear part of the decoder $\boldsymbol{M}_D$ are orthogonal, in general, columns of $f_D(\boldsymbol{M}_D)$ are not.

### 3.3. The principal axes of the latent space aligns with the standard basis

The latent variables, and the SVD decomposition of the linear component of the decoder – $\boldsymbol{M}_D$, and the diagonal covariance of the posterior probability of the VAE aligns the principal axes (or curves) of the latent space with the standard basis $\boldsymbol{e}_i$-s of the latent space. We consider the following lemma, whose proof is detailed in A.5.

**Lemma 4.** *Given $\boldsymbol{M}_D = \boldsymbol{U}_D\boldsymbol{\Sigma}_D\boldsymbol{V}_D^\top$, such that the columns of $\boldsymbol{M}_D$ are orthogonal, and $\boldsymbol{M}_D$ has unique non-zero singular values, the following hold: (a) $\boldsymbol{U}_D$ is an orthogonal matrix, (b) the diagonal elements of $\boldsymbol{\Sigma}_D$ are the norms of the columns of $\boldsymbol{M}_D$, and (c) $\boldsymbol{V}_D = \boldsymbol{I}$.*

The generative factors of data are closely linked to the principal axes or curves, which are characterized by maximum variance (experimental evidence in A.8.6).

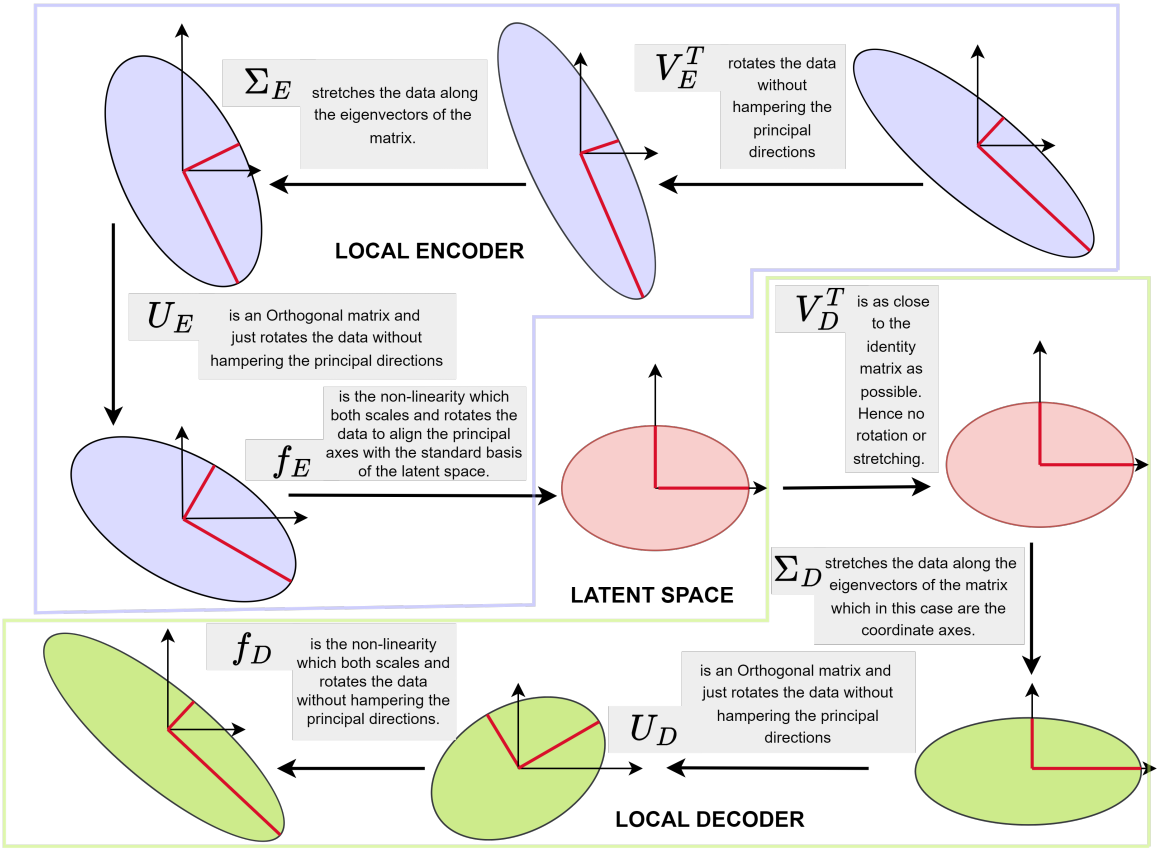



*Figure 3.* The figure illustrates the non-linear and SVD decomposition components of both the encoder and the decoder for a 2D case. The latent space distribution is a Gaussian, with axes of variation of the input data aligned with the standard basis vectors $\boldsymbol{e}_i$ of the latent space. It is noted that $\boldsymbol{V_D^\top}$ equals the identity matrix $\boldsymbol{I}$, indicating that $\boldsymbol{\Sigma}_D$ is responsible for scaling the data along the stand basis vectors.For illustrative clarity, the representation is simplified to a 2D perspective, showing principal axes as straight lines instead of curves, to better convey the transformations enacted by the encoder and decoder on the data.

Techniques like PCA and Hierarchical Non-Linear PCA (h-NLPCA) (Scholz & Vigário, 2002) utilize these axes for data reconstruction by maximizing latent space variance. Similarly, symmetric NLPCA (s-NLPCA) (Kramer, 1991), which uses autoencoder architectures[2], focuses on retaining components with low correlation and high variance.

In the case of VAE, as illustrated in Fig. 3, the latent space distribution would be very close to the Gaussian distribution (due to Gaussian prior) with a diagonal covariance. Therefore, the principal axes of the latent space are aligned with the standard basis vectors $\boldsymbol{e}_i$-s. Hence, we need $\boldsymbol{V_D^\top} = \boldsymbol{I}$, so that the distribution would not be rotated and the axes are preserved.

Moreover, from here we can see that as noted by (Träuble et al., 2021), if we observe data that is correlated, there can be two latent factors that change simultaneously and it would be difficult to identify them (and disentangle them). Hence, in our analyses the decoder works to retain them for successful data reconstruction; see Fig. 3.

[2]Both s-NLPCA and h-NLPCA employ autoencoder architectures for generating compressed representations.

### 3.4. How does orthogonality influence disentanglement?

We now explore why orthogonality is instrumental in promoting disentanglement with the following lemma (proof in A.6). First, we show that given a fixed $L_{rec}^{stoch^{(i)}}$, orthogonality in $\boldsymbol{M}_D$ promotes a lower $L_{KL}$.

**Lemma 5.** *Given a $L_{rec}^{stoch^{(i)}}$, orthogonality in the linear component of the Decoder's function transformation, $\boldsymbol{M}_D$, promotes a lower $L_{KL}$.*

Next, we see how lower $L_{KL}$ and in turn orthogonality of $\boldsymbol{M}_D$ affects the latent space of the VAE. First, we establish that the samples closer in data space are also closer in the latent space too. Further, a lower KL divergence loss brings these samples closer.

**Theorem 2.** *For a VAE, given $\boldsymbol{z}^{(i)} \sim Enc_\phi(\boldsymbol{x}^{(i)})$ and $\boldsymbol{z}^{(k)} \sim Enc_\phi(\boldsymbol{x}^{(k)})$, where, $\boldsymbol{x}^{(k)}$ are the $k^{(i)}$ nearest*

*Table 1.* Comparative analysis of approximation errors for the local decoder when modeled as linear versus non-linear.

| Dataset | Error | $\beta$-TCVAE | $\beta$-VAE | VAE |
|---|---|---|---|---|
| dSprites | Zietlow et al. (2021) | 0.9209 | 0.8394 | 0.4693 |
| | **Ours** | 0.8502 | 0.7829 | 0.4166 |
| 3DFaces | Zietlow et al. (2021) | 0.8679 | 0.8080 | 0.4956 |
| | **Ours** | 0.8088 | 0.7694 | 0.4236 |
| 3DShapes | Zietlow et al. (2021) | 0.8848 | 0.8244 | 0.4856 |
| | **Ours** | 0.8122 | 0.7944 | 0.4266 |
| MPI3D | Zietlow et al. (2021) | 0.9646 | 0.5240 | 0.5024 |
| | **Ours** | 0.8624 | 0.8043 | 0.4832 |

*Table 2.* Comparison of error incurred in approximating the local decoder for evaluating $L_{rec}^{stoch}$ using a linearized approach versus a non-linear approach. Note: *p<.05, **p<.01

| Dataset | Error | $\beta$-TCVAE | $\beta$-VAE | VAE |
|---|---|---|---|---|
| dSprites | Rolinek et al. (2019) | 0.4866 | 0.5243 | 0.6680 |
| | **Ours** | 0.4532** | 0.4907* | 0.6643 |
| 3DFaces | Rolinek et al. (2019) | 0.5690 | 0.5799 | 0.6039 |
| | **Ours** | 0.5498* | 0.5296* | 0.5887** |
| 3DShapes | Rolinek et al. (2019) | 0.7257 | 0.7215 | 0.7344 |
| | **Ours** | 0.6879* | 0.7010** | 0.7316 |
| MPI3D | Rolinek et al. (2019) | 0.7836 | 0.8001 | 0.8042 |
| | **Ours** | 0.7818* | 0.7908* | 0.8014* |

*neighbours of $\boldsymbol{x}^{(i)}$, we define $Dist(L_{KL})$ as follows:*

$$Dist(L_{KL}) = \mathbb{E}_{\boldsymbol{x}^{(i)},\boldsymbol{z}^{(i)},\boldsymbol{z}^{(k)}}\left[\sum_{k=1}^{k^{(i)}} ||\boldsymbol{z}^{(i)} - \boldsymbol{z}^{(k)}||^2\right]$$

*The following hold:*

*(a) Given $Enc_\phi(\boldsymbol{x}^{(i)}) \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ overlaps (is close) with $k^{(i)}$ posterior probabilities, they must be posterior probabilities generated by the $k^{(i)}$ nearest neighbours of $\boldsymbol{x}^{(i)}$ in $X$, i.e. $Enc_\phi(\boldsymbol{x}^{(k)}) \sim q_\phi(\boldsymbol{z}^{(k)}|\boldsymbol{x}^{(k)})$. Here, for every $\boldsymbol{x}^{(i)}$, we have $k^{(i)}$ number of $\boldsymbol{x}^{(k)}$-s, whose posterior probabilities, $q_\phi(\boldsymbol{z}^{(k)}|\boldsymbol{x}^{(k)})$, overlap with the posterior probability $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ in the latent space.*

*(b) Given, $L_{KL'} < L_{KL}$, $Dist(L_{KL'}) < Dist(L_{KL})$.*

From this result, given any point in the latent space, $\boldsymbol{z}^{(i)} = \sum_{j=1}^{d} z_j^{(i)} \boldsymbol{e}_j$, where $z_j^{(i)} = \boldsymbol{z}^{(i)} \cdot \boldsymbol{e}_j$ (with "$\cdot$" indicating the dot product), adjusting $z_l^{(i)}$ by $\Delta z_l^{(i)}$ for any $l \in \{1, \ldots, d\}$ (while keeping all other $z_j^{(i)}$s constant) results in a new point $\boldsymbol{z}^{(k)}$. Specifically, $\boldsymbol{z}^{(k)} = \boldsymbol{z}^{(i)} + \Delta z_l^{(i)} \boldsymbol{e}_l$. Finally, using Sect. 3.3, Lem. 5, Theorem 2, we observe that orthogonality in $\boldsymbol{M}_D$ promotes disentanglement. Particularly, we observe:

(a) the output derived from $\boldsymbol{z}^{(k)}$ differs from that of $\boldsymbol{z}^{(i)}$ solely in terms of a single generative factor linked to the latent variable $z_l^{(i)}$ in the $\boldsymbol{e}_l$ direction; see Sect. 3.3.

(b) orthogonality promotes lower $L_{KL}$. According to Theorem 2, as $L_{KL}$ decreases, the samples close in data space come closer in latent space too. Hence, the variance between the two outputs is directly proportional to $\Delta z_l^{(i)}$, which is the variance in the latent variable $z_l^{(i)}$; see Lem. 5.

These findings substantiate that the orthogonality of $\boldsymbol{M}_D$ columns promote disentanglement; detailed proof in A.7.

### 3.5. How do $\sigma$ & $L_{KL}$ relate to the data principal axes?

Moving on, we explain (a) the relationship between the principal axes (and consequently the generative factors) and $\sigma_j^2 = \mathbb{E}_i[{\sigma_j^{(i)}}^2]$, and (b) how the $L_{KL}$ loss relates to the principal axes and generative factors. From part (d) of Theorem 1, ${\sigma_j^{(i)}}^2 \propto \frac{1}{||\boldsymbol{c}_j||^2}$ for all $i$. This implies that

$$\sigma_j^2 \propto \mathbb{E}_i\left[\frac{1}{||\boldsymbol{c}_j||^2}\right] = \frac{1}{||\boldsymbol{c}_j||^2}.$$

Referring to Lem. 4, the singular values of $\boldsymbol{M}_D$ are given by $||\boldsymbol{c}_j||$, and $\boldsymbol{\Sigma}_D$ stretches the latent space distribution along the standard basis. Principal axes with higher variances correspond to the smaller singular values. Hence, principal axes vital for image generation are associated with columns of $\boldsymbol{M}_D$ with greater $||\boldsymbol{c}_j||$ and lower $\sigma_j^2$. Given that $\sigma_j^{(i)} < 1$, $-\log({\sigma_j^{(i)}}^2) > 0$. As $\sigma_j^{(i)}$ decreases, significant principal axes contribute more to the $L_{KL}$ loss.

## 4. Experiments

In this section, we discuss the experimental setup and the results to verify our theoretical findings. We experimentally verify how introducing local non-linearity makes the VAE modeling more realistic. Furthermore, we show that this local-nonlinearity modeling technique is better than the linearization of $L_{rec}^{stoch}$. We define a metric, Orthogonality Deviation Score (OD-Score) to calculate the extent of orthogonality of the linear component of the local-decoder matrix. Finally, we show that disentanglement (measured using MIG and MIG-Sup scores) is directly proportional to Orthogonality (measured using OD-Score.) The code is available at `https://github.com/criticalml-uw/Disentanglement-in-VAE`.

### 4.1. Datasets

We study the VAE architectures using four widely used datasets, namely, dSprites, 3D Faces (Paysan et al., 2009), 3D shapes (Burgess & Kim, 2018), and MPI 3D complex real-world shapes dataset (Gondal et al., 2019).

### 4.2. Metrics

For evaluating disentanglement, we use MIG and MIG-sup metrics introduced in Chen et al. (2018a) and Li et al. (2020) respectively. While for analyzing the efficacy of the non-linearity $f_D$, we introduce an error function to compare the deviation of the analysis from the actual

scenario with and without the non-linearity, for calculating the orthogonality of the linear part of the local decoder, we device a metric based on Lem. 4. Further, in A.8.6, we experimentally show that principal axes with the highest variance are associated with the generative factors most significant for reconstruction.

### 4.3. Models and Implementation Details

To evaluate the efficacy of our analysis we consider three VAE-based models, namely, VAE, $\beta$-VAE, and $\beta$-TCVAE. In the subsequent experiments, we show that our analysis holds for all the VAE-based architectures. In Appendix A.8.1, we summarize implementation details.

### 4.4. Main Experimental Findings

**Contribution of the non-linearity $f(\cdot)$:** From (8), $Dec_\theta(Enc_\phi(\boldsymbol{x}^{(i)})) \approx Dec_\theta(\boldsymbol{\mu}^{(i)}) + f_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})$. On the other hand, in the work by Zietlow et al. (2021) the local decoder is approximated to be linear, and so $Dec_\theta(Enc_\phi(\boldsymbol{x}^{(i)})) \approx Dec_\theta(\boldsymbol{\mu}^{(i)}) + f_{linear}(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})$. To show that introducing non-linearity improves the approximation, in the experiment, we compare the squared error between the non-linear approximation and the ground truth in (11) as $\delta$ with the squared error between the linear approximation and ground truth in (12) as $\delta_{linear}$,

$$\delta = ||\text{Dec}_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - \text{Dec}_\theta(\boldsymbol{\mu}^{(i)}) - f_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})||^2, \quad (11)$$

$$\delta_{\text{linear}} = ||\text{Dec}_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - \text{Dec}_\theta(\boldsymbol{\mu}^{(i)}) - f_{\text{linear}}(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})||^2 \quad (12)$$

In the experiment, $f_{linear}(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})$ is approximated by a neural network without non-linearity, while, $f_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})$ is approximated by a neural network consisting of one non-linearity. We define $\boldsymbol{x}^{(i)} \in \mathcal{X}_{val}$, and for each $\boldsymbol{x}^{(i)}$, we estimate the parameters $f_D$ and $\boldsymbol{M}_D$ (as these parameters are local for each $\boldsymbol{x}^{(i)}$) using (11) as the loss function. For random $\boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(0, {\boldsymbol{\sigma}^{(i)}}^2)$ for each $\boldsymbol{x}^{(i)}$, we calculate the error for each $\boldsymbol{x}^{(i)}$ using (11) and finally take the average.

In Table 1, we demonstrate using three different VAE-based architectures on four different datasets that the non-linearity makes the local decoder approximation much more accurate.

**Comparisions across different approximations:** We compare two approximations: the linearization assumption made by Rolinek et al. (2019) which approximates the stochastic part of the reconstruction loss as $J\boldsymbol{\epsilon}^{(i)}$, where $J$ is the Jacobian approximation of the decoder around $\boldsymbol{\mu}^{(i)}$, and our modeling, where we assume the decoder to be non-linear. Table 2 summarizes that modelling VAEs as non-linear has lower error on real-world dataset than the linearization approximation; details in A.8.2.

**Disentanglement across VAE architectures:** As explained above, we have used the Mutual Information Gap (MIG) and (MIG-sup) for quantifying the disentanglement of the different VAE architectures in the datasets mentioned. Panels (a)-(d) in Fig. 4, illustrate the MIG scores for the different VAE-based architectures while Panels (e)-(h) illustrate the MIG-sup scores. In Appendix A.8.4, we provide the scores in a tabular form.

**Understanding Orthogonality:** We use Lem. 4 to calculate the distance between the linear component of decoder function and the closest orthogonal matrix. We take average of the normalized distances as follows.

$$\text{d}_U(\boldsymbol{U_D}, \widehat{\boldsymbol{U}}_{\boldsymbol{D}}) = \frac{||\boldsymbol{U_D} - \widehat{\boldsymbol{U}}_{\boldsymbol{D}}||_F^2}{\max_\phi(||\boldsymbol{U_D} - \widehat{\boldsymbol{U}}_{\boldsymbol{D},\phi}||_F^2)},$$

$$\text{d}_\Sigma(\boldsymbol{\Sigma_D}, \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{D}}) = \frac{||\boldsymbol{\Sigma_D} - \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{D}}||_F^2}{\max_\phi(||\boldsymbol{\Sigma_D} - \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{D},\phi}||_F^2)}$$

$$\text{d}_V(\boldsymbol{V_D}, \widehat{\boldsymbol{V}}_{\boldsymbol{D}}) = \frac{||\boldsymbol{V_D} - \widehat{\boldsymbol{V}}_{\boldsymbol{D}}||_F^2}{\max_\phi(||\boldsymbol{V_D} - \widehat{\boldsymbol{V}}_{\boldsymbol{D},\phi}||_F^2)},$$

where $\hat{\boldsymbol{U}}_{\boldsymbol{D}}, \hat{\boldsymbol{\Sigma}}_{\boldsymbol{D}}$, and $\hat{\boldsymbol{V}}_{\boldsymbol{D}}$ are calculated using Lem. 4 and Appendix A.8.3. The norms are calculated using the Frobenius norm, and the equations are normalized across all VAE-based models used for evaluating the distance. The Orthogonal Deviation Score, denoted as OD-Score($\boldsymbol{M}_D$), is defined as:

$$\text{OD-Score}(\boldsymbol{M}_D) = \frac{\text{d}_U(\boldsymbol{U_D}, \hat{\boldsymbol{U}}_{\boldsymbol{D}}) + \text{d}_\Sigma(\boldsymbol{\Sigma_D}, \hat{\boldsymbol{\Sigma}}_{\boldsymbol{D}}) + \text{d}_V(\boldsymbol{V_D}, \hat{\boldsymbol{V}}_{\boldsymbol{D}})}{3}$$

In Fig. 4, panels (i)-(l) demonstrate the MIG versus OD-Score($\boldsymbol{M}_D$) scores for all the datasets for each of the three VAE-based architectures. Further, panels (m)-(p) demonstrate the MIG-Sup versus OD-Score($\boldsymbol{M}_D$) scores. We note that as the MIG and the MIG-Sup scores increase, the orthogonality as measured by OD-Score($\boldsymbol{M}_D$) decreases (lower is better). This establishes that orthogonality promotes disentanglement. In Appendix A.8.5, we record the OD-Score($\boldsymbol{M}_D$) values of the different VAE-based architectures for the datasets.

## 5. Discussion and Conclusion

In this work, build on existing works which use local linearity for explaining the behavior of Variational Autoencoders (VAEs), to propose an analysis that incorporates local non-linearity. We provide theoretical analysis and extensive experimental evaluations to show that the stochastic part of the loss function promotes orthogonality among the columns of the linear component in the decoder's function. Furthermore, we establish both mathematically and empirically that this orthogonality is instrumental in promoting disentanglement, a link previously observed only through experimental evidence.

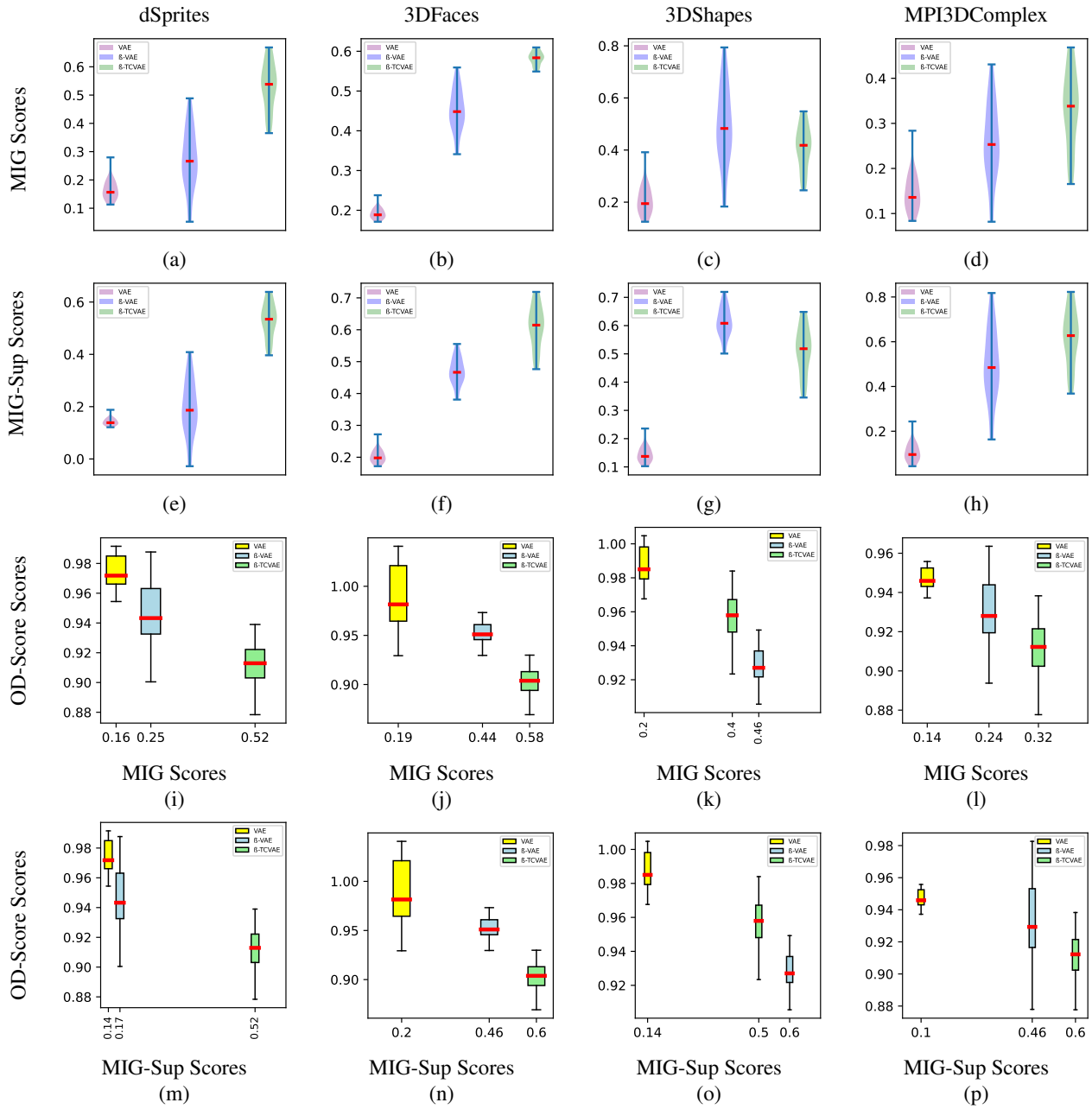



*Figure 4.* **Performance of different VAE architectures** Panels (a)-(d) illustrate the MIG scores (a higher score promotes disentanglement) of different VAE architectures for dSprites, 3DFaces, 3DShapes and MPI3DComplex, respectively. Panels (e)-(h) illustrate the MIG-Sup scores (a higher score promotes disentanglement). Panels (i)-(l) illustrate positive correlation between orthogonality, measured by OD-Score($\boldsymbol{M}_D$) (a lower score promotes orthogonality) and MIG scores for specified models and datasets. Finally, Panels (m)-(p) illustrate the OD-Score($\boldsymbol{M}_D$) vs the MIG-Sup score.

Most previous studies suggest that the imposition of a diagonal posterior on the encoder is the primary driver for VAEs to learn disentangled representations. Our work expands on this notion by demonstrating that the reconstruction loss, when constrained by the KL-Divergence loss, also facilitates disentanglement. Nevertheless, the precise alignment of embeddings within the latent space remains an open question. Unraveling this aspect could significantly enhance the understanding of VAEs, and other generative models, for disentangled representation learning.

## Acknowledgement

Sirisha Rambhatla would like to acknowledge support of the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant, RGPIN-2022-03512.

## Impact Statement

As generative models grow in popularity and find applications across different disciplines, it is critical to equip them with the ability to break spurious correlations between features to be able to generate unbiased data. For instance, a dataset with correlations between a protected attribute and a feature (in ambient or the latent space) can learn to generate data that reinforces such patterns. Constructing generative models with distentanglement capability is therefore key for fair data generation to break historic biases in the data. Therefore, understanding why certain architectures inherently promote disentanglement is important to incorporate such properties in contemporary and future generative models.

## References


Achille, A. and Soatto, S. Emergence of invariance and disentanglement in deep representations. *The Journal of Machine Learning Research*, 19(1):1947–1980, 2018.

Adel, T., Ghahramani, Z., and Weller, A. Discovering interpretable representations for both deep generative and discriminative models. In *International Conference on Machine Learning*, pp. 50–59. PMLR, 2018.

Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Bojanowski, P., Joulin, A., Lopez-Paz, D., and Szlam, A. Optimizing the latent space of generative networks. *arXiv preprint arXiv:1707.05776*, 2017.

Burgess, C. and Kim, H. 3d shapes dataset. https://github.com/deepmind/3dshapes-dataset/, 2018.

Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., and Lerchner, A. Understanding disentangling in $\beta$-vae. *arXiv preprint arXiv:1804.03599*, 2018.

Chen, R. T., Li, X., Grosse, R., and Duvenaud, D. Isolating sources of disentanglement in vaes. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, volume 2615, pp. 2625, 2018a.

Chen, R. T., Li, X., Grosse, R. B., and Duvenaud, D. K. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018b.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in neural information processing systems*, 29, 2016.

Creager, E., Madras, D., Jacobsen, J.-H., Weis, M., Swersky, K., Pitassi, T., and Zemel, R. Flexibly fair representation learning by disentanglement. In *International conference on machine learning*, pp. 1436–1445. PMLR, 2019.

Ding, Z., Xu, Y., Xu, W., Parmar, G., Yang, Y., Welling, M., and Tu, Z. Guided variational autoencoder for disentanglement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7920–7929, 2020.

Engel, J., Hoffman, M., and Roberts, A. Latent constraints: Learning to generate conditionally from unconditional generative models. *arXiv preprint arXiv:1711.05772*, 2017.

Fan, S., Wang, X., Mo, Y., Shi, C., and Tang, J. Debiasing graph neural networks via learning disentangled causal substructure. *Advances in Neural Information Processing Systems*, 35:24934–24946, 2022.

Gao, L., Mao, Q., Dong, M., Jing, Y., and Chinnam, R. On learning disentangled representation for acoustic event detection. In *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 2006–2014, 2019.

Gondal, M. W., Wuthrich, M., Miladinovic, D., Locatello, F., Breidt, M., Volchkov, V., Akpo, J., Bachem, O., Schölkopf, B., and Bauer, S. On the transfer of inductive bias from simulation to the real world: a new disentanglement dataset. *Advances in Neural Information Processing Systems*, 32, 2019.

Gonzalez-Garcia, A., Van De Weijer, J., and Bengio, Y. Image-to-image translation for cross-domain disentanglement. *Advances in neural information processing systems*, 31, 2018.

Han, L., Lyu, Y., Peng, C., and Zhou, S. K. Gan-based disentanglement learning for chest x-ray rib suppression. *Medical Image Analysis*, 77:102369, 2022.

Härkönen, E., Hertzmann, A., Lehtinen, J., and Paris, S. Ganspace: Discovering interpretable gan controls. *Advances in neural information processing systems*, 33: 9841–9850, 2020.

He, S., Ding, H., and Jiang, W. Primitive generation and semantic-related alignment for universal zero-shot segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11238–11247, 2023.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. beta-vae:

Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*, 2016.

Higgins, I., Amos, D., Pfau, D., Racaniere, S., Matthey, L., Rezende, D., and Lerchner, A. Towards a definition of disentangled representations. *arXiv preprint arXiv:1812.02230*, 2018.

Hu, Q., Szabó, A., Portenier, T., Favaro, P., and Zwicker, M. Disentangling factors of variation by mixing them. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3399–3407, 2018.

Jeong, Y. and Song, H. O. Learning discrete and continuous factors of data via alternating disentanglement. In *International Conference on Machine Learning*, pp. 3091–3099. PMLR, 2019.

Jha, A. H., Anand, S., Singh, M., and Veeravasarapu, V. R. Disentangling factors of variation with cycle-consistent variational auto-encoders. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 805–820, 2018.

Jia, M., Cheng, X., Lu, S., and Zhang, J. Learning disentangled representation implicitly via transformer for occluded person re-identification. *IEEE Transactions on Multimedia*, 25:1294–1305, 2022.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pp. 2207–2217. PMLR, 2020.

Kim, H. and Mnih, A. Disentangling by factorising. In *International Conference on Machine Learning*, pp. 2649–2658. PMLR, 2018.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Klindt, D., Schott, L., Sharma, Y., Ustyuzhaninov, I., Brendel, W., Bethge, M., and Paiton, D. Towards nonlinear disentanglement in natural data with temporal sparse coding. *arXiv preprint arXiv:2007.10930*, 2020.

Kramer, M. A. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2):233–243, 1991.

Kumar, A. and Poole, B. On implicit regularization in $\beta$-vaes. *International Conference on Machine Learning*, 2020.

Kumar, A., Sattigeri, P., and Balakrishnan, A. Variational inference of disentangled latent concepts from unlabeled observations. *arXiv preprint arXiv:1711.00848*, 2017.

Lachapelle, S., Deleu, T., Mahajan, D., Mitliagkas, I., Bengio, Y., Lacoste-Julien, S., and Bertrand, Q. Synergies between disentanglement and sparsity: Generalization and identifiability in multi-task learning. In *International Conference on Machine Learning*, pp. 18171–18206. PMLR, 2023.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017.

Li, H., Xu, K., Li, J., and Yu, Z. Dual-stream reciprocal disentanglement learning for domain adaptation person re-identification. *Knowledge-Based Systems*, 251:109315, 2022.

Li, Z., Murkute, J. V., Gyawali, P. K., and Wang, L. Progressive learning and disentanglement of hierarchical representations. *arXiv preprint arXiv:2002.10549*, 2020.

Liao, Y., Schwarz, K., Mescheder, L., and Geiger, A. Towards unsupervised learning of generative models for 3d controllable image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5871–5880, 2020.

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pp. 4114–4124. PMLR, 2019.

Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., and Bachem, O. A sober look at the unsupervised learning of disentangled representations and their evaluation. *arXiv preprint arXiv:2010.14766*, 2020.

Mathieu, M. F., Zhao, J. J., Zhao, J., Ramesh, A., Sprechmann, P., and LeCun, Y. Disentangling factors of variation in deep representation using adversarial training. *Advances in neural information processing systems*, 29, 2016.

Mita, G., Filippone, M., and Michiardi, P. An identifiable double vae for disentangled representations. In *International Conference on Machine Learning*, pp. 7769–7779. PMLR, 2021.

Nie, J., Zhang, T., Li, T., Yu, S., Li, X., and Wei, Z. Image-based 3d model retrieval via disentangled feature learning and enhanced semantic alignment. *Information Processing & Management*, 60(2):103159, 2023.

Paysan, P., Knothe, R., Amberg, B., Romdhani, S., and Vetter, T. A 3d face model for pose and illumination invariant face recognition. In *2009 sixth IEEE international conference on advanced video and signal based surveillance*, pp. 296–301. Ieee, 2009.

Peters, J., Janzing, D., and Schölkopf, B. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Qin, Y., Wang, Y., Sun, F., Ju, W., Hou, X., Wang, Z., Cheng, J., Lei, J., and Zhang, M. Disenpoi: Disentangling sequential and geographical influence for point-of-interest recommendation. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pp. 508–516, 2023.

Rolinek, M., Zietlow, D., and Martius, G. Variational autoencoders pursue pca directions (by accident). In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12406–12415, 2019.

Ruan, D., Mo, R., Yan, Y., Chen, S., Xue, J.-H., and Wang, H. Adaptive deep disturbance-disentangled learning for facial expression recognition. *International Journal of Computer Vision*, 130(2):455–477, 2022.

Scholz, M. and Vigário, R. Nonlinear pca: a new hierarchical approach. In *Esann*, pp. 439–444, 2002.

Shen, X., Liu, F., Dong, H., Lian, Q., Chen, Z., and Zhang, T. Weakly supervised disentangled generative causal representation learning. *The Journal of Machine Learning Research*, 23(1):10994–11048, 2022.

Shu, Z., Yumer, E., Hadap, S., Sunkavalli, K., Shechtman, E., and Samaras, D. Neural face editing with intrinsic image disentangling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5541–5550, 2017.

Song, J., Kalluri, P., Grover, A., Zhao, S., and Ermon, S. Learning controllable fair representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2164–2173. PMLR, 2019.

Stammer, W., Memmel, M., Schramowski, P., and Kersting, K. Interactive disentanglement: Learning concepts by interacting with their prototype representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10317–10328, 2022.

Szabó, A., Hu, Q., Portenier, T., Zwicker, M., and Favaro, P. Challenges in disentangling independent factors of variation. *arXiv preprint arXiv:1711.02245*, 2017.

Tran, L., Yin, X., and Liu, X. Disentangled representation learning gan for pose-invariant face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1415–1424, 2017.

Träuble, F., Creager, E., Kilbertus, N., Locatello, F., Dittadi, A., Goyal, A., Schölkopf, B., and Bauer, S. On disentangled representations learned from correlated data. In *International conference on machine learning*, pp. 10401–10412. PMLR, 2021.

Voynov, A. and Babenko, A. Unsupervised discovery of interpretable directions in the gan latent space. In *International conference on machine learning*, pp. 9786–9796. PMLR, 2020.

Wang, D., Deng, Y., Yin, Z., Shum, H.-Y., and Wang, B. Progressive disentangled representation learning for fine-grained controllable talking head synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17979–17989, 2023.

Wu, Q., Liu, Y., Zhao, H., Kale, A., Bui, T., Yu, T., Lin, Z., Zhang, Y., and Chang, S. Uncovering the disentanglement capability in text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1900–1910, 2023.

Xie, Q., Li, Y., He, N., Ning, M., Ma, K., Wang, G., Lian, Y., and Zheng, Y. Unsupervised domain adaptation for medical image segmentation by disentanglement learning and self-training. *IEEE Transactions on Medical Imaging*, 2022.

Xu, J., Ren, Y., Tang, H., Pu, X., Zhu, X., Zeng, M., and He, L. Multi-vae: Learning disentangled view-common and view-peculiar visual representations for multi-view clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9234–9243, 2021.

Yang, T., Wang, Y., Lv, Y., and Zh, N. Disdiff: Unsupervised disentanglement of diffusion probabilistic models. *arXiv preprint arXiv:2301.13721*, 2023.

Yao, W., Chen, G., and Zhang, K. Temporally disentangled representation learning. *Advances in Neural Information Processing Systems*, 35:26492–26503, 2022.

Zhang, X., Li, X., Sultani, W., Zhou, Y., and Wshah, S. Cross-view geo-localization via learning disentangled geometric layout correspondence. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 3480–3488, 2023.

Zhang, Z., Zhao, Z., and Lin, Z. Unsupervised representation learning from pre-trained diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 35:22117–22130, 2022.

Zhu, X., Xu, C., and Tao, D. Commutative lie group vae for disentanglement learning. In *International Conference on Machine Learning*, pp. 12924–12934. PMLR, 2021.

Zhu, Y., Min, M. R., Kadav, A., and Graf, H. P. S3vae: Self-supervised sequential vae for representation

disentanglement and data generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6538–6547, 2020.

Zietlow, D., Rolinek, M., and Martius, G. Demystifying inductive biases for (beta-) vae based architectures. In *International Conference on Machine Learning*, pp. 12945–12954. PMLR, 2021.

# A. Appendix

## A.1. Background

### A.1.1. Variational Autoencoders

Let $\{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ is a dataset consisting N elements, such that $\boldsymbol{x}^{(i)} \in \mathbb{X} = \mathbb{R}^n$. A VAE consists of a probabilistic encoder, $Enc_\phi : \mathbb{X} \to \mathbb{Z}$ and a decoder $Dec_\theta : \mathbb{Z} \to \mathbb{X}$, where $\mathbb{Z} = \mathbb{R}^d$ is called the Latent Space. The distribution of the input, namely $q(\boldsymbol{x}^{(i)})$ is fixed as it is the actual data distribution. We also define a fixed prior distribution $p(\boldsymbol{z}^{(i)})$ over $\mathbb{Z}$. The idealized loss function of the VAE is the marginalized log-likelihood which is defined as follows:

$$\sum_{i=1}^{N} \log(p(\boldsymbol{x}^{(i)}))$$

However, this loss function is not tractable and is approximated by its lower bound, the Evidence Lower Bound (ELBO) loss function defined as follows:

$$\mathbb{E}_{\boldsymbol{z}^{(i)} \sim q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})}[\log(p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)}))] - D_{KL}(q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})||p(\boldsymbol{z}^{(i)}))$$

where the first term is the reconstruction loss while the second term is KL divergence, which calculates the similarity between the probability distributions $q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ and $p(\boldsymbol{z}^{(i)})$. Hence, VAE performs a trade-off between reconstruction and the ability to mimic the prior probability distribution.

All the probability distributions are assumed to be Gaussian with the prior probability distribution being defined as follows:

$$p(\boldsymbol{z}^{(i)}) = \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{13}$$

The encoder is defined as follows:

$$Enc_\phi(\boldsymbol{x}^{(i)}) \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)}) = \mathcal{N}(\mu_\phi(\boldsymbol{x}^{(i)}), diag(\sigma^2_\phi(\boldsymbol{x}^{(i)})))$$

where $\mu_\phi$ and $diag(\sigma_\phi)$ are the parameter $\phi$ dependent mappings. Parametrizing the distributions in this way allows for the use of the reparametrization trick to estimate gradients of the lower bound with respect to the parameters $\phi$. The latent variables, hence are defined as follows $\boldsymbol{z}^{(i)} \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ and hence can be reparametrized as follows:

$$\boldsymbol{z}^{(i)} = \boldsymbol{\mu}^{(i)} + \boldsymbol{\sigma}^{(i)}\boldsymbol{\epsilon}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Finally, it is to be noted that the posterior distribution $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ has a diagonal covariance matrix.

Under the Gaussian assumptions, the KL divergence can be written in closed form as follows:

$$L_{KL}^{(i)} = \frac{1}{2}\sum_j (\mu_j^{(i)^2} + \sigma_j^{(i)^2} - log(\sigma_j^{(i)^2}) - 1) \tag{14}$$

### A.1.2. Understanding Disentanglemnt for log-likelihood loss function

It is assumed in case of interpretable representation that some generating factors are responsible for the generation of data. For example in the case of the dSprites dataset, data is generated from generative factors like position, shape, scale or size, rotational orientation etc. Disentangled representation is the situation, when, a single latent variable is responsible for the changes in a single generative factor and is non sensitive to the changes in other generative factors. In the case of unsupervised algorithms, the generative factors are not known and the methods rely on statistical procedures. One category of such methods is the VAE-based methods which we are analyzing.

Rotation matrices, ($\boldsymbol{U}$), are defined as as orthogonal matrices ($\boldsymbol{U}^\top = \boldsymbol{U}^{-1}$) with determinant equal to 1 ($|\boldsymbol{U}| = 1$). We define rotational invariance or rotational symmetry for a probability as $p(\boldsymbol{z}) = p(\boldsymbol{U}\boldsymbol{z})$. It is important to note that Disentanglement is sensitive to rotations of the latent embedding. For example, consider a disentangle latent representation $\begin{pmatrix} a \\ b \end{pmatrix}$. When acted upon by the rotation matrix in 2-dimensions, namely $\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$, it becomes, $\begin{pmatrix} a\cos\theta - b\sin\theta \\ a\sin\theta + b\cos\theta \end{pmatrix}$.

Hence, it can be seen that rotating disentangled latent representation essentially destroys the disentanglement property of the latent representation.

Earlier works like (Rolinek et al., 2019) record in detail that the log-likelihood loss and the ELBO losses are rotationally invariant, meaning that if the prior probability is rotationally symmetric, then, even if a rotational matrix and its transpose are multiplied to the encoder and decoder respectively the log-likelihood objective and the ELBO objective do not change. However, this rotational invariance is disrupted by forcing a diagonal posterior on the encoder of the VAE. This gives rise to the KL Loss function that has been recorded in (14)

### A.2. Derivation of Loss Functions $L_{KL}^{(i)}$, $L_{MLE}^{(i)}$ and Equation 8

**Proposition 2.** *Assuming the probabilities, $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$-s to be Gaussian distributions $\mathcal{N}(\mu_\phi(\boldsymbol{x}^{(i)}), diag(\sigma_\phi^2(\boldsymbol{x}^{(i)})))$-s, the KL divergence Loss function can be written as follows:*

$$L_{KL}^{(i)} = \frac{1}{2}\sum_j (\mu_j^{(i)^2} + \sigma_j^{(i)^2} - log(\sigma_j^{(i)^2}) - 1) \tag{15}$$

*where, $\mu_j^{(i)}$ is the $j$-th element of $\mu_\phi(\boldsymbol{x}^{(i)})$ and $\sigma_j^{(i)}$ is the $j$-th element of $diag(\sigma_\phi(\boldsymbol{x}^{(i)}))$*

*Proof.* Given, $p(\boldsymbol{z}^{(i)}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)}) = \mathcal{N}(\mu_\phi(\boldsymbol{x}^{(i)}), \text{diag}(\sigma_\phi^2(\boldsymbol{x}^{(i)})))$. In short, we refer to $\mathcal{N}(\mu_\phi(\boldsymbol{x}^{(i)}), \text{diag}(\sigma_\phi^2(\boldsymbol{x}^{(i)})))$ as $\mathcal{N}(\mu^{(i)}, \boldsymbol{\Sigma}^{(i)})$ The multivariate normal distributions are given by,

$$\mathcal{N}(\mu^{(i)}, \boldsymbol{\Sigma}^{(i)}) = \frac{1}{(2\pi)^{\frac{k}{2}}|\boldsymbol{\Sigma}^{(i)}|^{\frac{1}{2}}} exp(-\frac{1}{2}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}^{(i)})^\intercal \boldsymbol{\Sigma}^{(i)^{-1}}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}^{(i)}))$$

and

$$\mathcal{N}(\mathbf{0}, \mathbf{I}) = \frac{1}{(2\pi)^{\frac{k}{2}}|\mathbf{I}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\boldsymbol{z}^{(i)\intercal}\boldsymbol{z}^{(i)}\right)$$

We refer to $\mathcal{N}(\boldsymbol{\mu}^{(i)}, \boldsymbol{\sigma}^{(i)})$ as $a(\boldsymbol{z}^{(i)})$ and $\mathcal{N}(\mathbf{0}, \mathbf{I})$ as $b(\boldsymbol{z}^{(i)})$. We know that,

$$\begin{aligned}
L_{KL}^{(i)} &= D_{KL}(a||b) = \mathbb{E}_a[\log(a) - \log(b)] \\
&= \mathbb{E}_a\left[\frac{1}{2}\log\left(\frac{1}{|\Sigma_a|}\right) - \frac{1}{2}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal\Sigma_a^{-1}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a) + \frac{1}{2}\boldsymbol{z}^{(i)\intercal}\boldsymbol{z}^{(i)}\right] \\
&= \mathbb{E}_a\left[\frac{1}{2}\log\left(\frac{1}{|\Sigma_a|}\right)\right] - \mathbb{E}_a\left[\frac{1}{2}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal\Sigma_a^{-1}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)\right] + \mathbb{E}_a\left[\frac{1}{2}\boldsymbol{z}^{(i)\intercal}\boldsymbol{z}^{(i)}\right]
\end{aligned} \tag{16}$$

Now, since, $(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal\boldsymbol{\Sigma}_a^{-1}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a) \in \mathbb{R}$, we have,

$$(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal\boldsymbol{\Sigma}_a^{-1}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a) = Tr\{(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal\boldsymbol{\Sigma}_a^{-1}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)\}.$$

Again,

$$Tr\{(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal\boldsymbol{\Sigma}_a^{-1}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)\} = Tr\{(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)\boldsymbol{\Sigma}_a^{-1}\} \tag{17}$$

Finally, since we can swap $Tr$ and $\mathbb{E}_a$,

$$\begin{aligned}
\mathbb{E}_a\left[\text{Tr}\left\{\frac{1}{2}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)\boldsymbol{\Sigma}_a^{-1}\right\}\right] &= \text{Tr}\left\{\mathbb{E}_a\left[\frac{1}{2}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)\boldsymbol{\Sigma}_a^{-1}\right]\right\} \\
&= \text{Tr}\left\{\mathbb{E}_a\left[\frac{1}{2}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)\right]\boldsymbol{\Sigma}_a^{-1}\right\}
\end{aligned}$$

Now, since, $\mathbb{E}_a\left[(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)\right] = \boldsymbol{\Sigma}_a$,

$$\mathbb{E}_a\left[\frac{1}{2}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)^\intercal\boldsymbol{\Sigma}_a^{-1}(\boldsymbol{z}^{(i)} - \boldsymbol{\mu}_a)\right] = tr\{\boldsymbol{\Sigma}_a\boldsymbol{\Sigma}_a^{-1}\} = Tr\{\mathbf{I}_j\} = j \tag{18}$$

Again, we know that, $\mathbb{E}[\boldsymbol{x}^\intercal \boldsymbol{x}] = Tr\{\boldsymbol{\Sigma}\} + \boldsymbol{\mu}^\intercal \boldsymbol{\mu}$

$$\mathbb{E}_a\Big[\frac{1}{2}\boldsymbol{z}^{(i)^\intercal}\boldsymbol{z}^{(i)}\Big] = \frac{1}{2}(Tr\{\boldsymbol{\Sigma}_a\} + \boldsymbol{\mu}_a^\intercal \boldsymbol{\mu}_a) \tag{19}$$

Finally,

$$\mathbb{E}_a\Big[\frac{1}{2}\log(\frac{1}{|\boldsymbol{\Sigma}_a|})\Big] = -\frac{1}{2}\log(|\boldsymbol{\Sigma}_a|) \tag{20}$$

Substituting 18, 19 and 20 in 16, we get,

$$L_{KL}^{(i)} = \frac{1}{2}\big(Tr\{\boldsymbol{\Sigma}_a\} + \boldsymbol{\mu}_a^\intercal \boldsymbol{\mu}_a - j - \log(|\boldsymbol{\Sigma}_a|\big)$$

Simplifying, we get,

$$L_{KL}^{(i)} = \frac{1}{2}\sum_j (\mu_j^{(i)^2} + \sigma_j^{(i)^2} - \log(\sigma_j^{(i)^2}) - 1)$$

□

**Proposition 3.** *Given a VAE, with Gaussian distributions, the maximum likelihood estimate,* $L_{MLE}^{(i)} = \mathbb{E}_{\boldsymbol{z}^{(i)}\sim q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})}[\log(p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)}))]$ *can be expressed as follows:*

$$L_{MLE}^{(i)} = -\frac{\log(2\pi)}{2} - \frac{\log(|\boldsymbol{\Sigma}_\theta|)}{2} - \mathbb{E}_{\boldsymbol{z}^{(i)}\sim q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})}\Big[\frac{||\boldsymbol{x}^{(i)} - \tilde{\boldsymbol{x}}||^2 \boldsymbol{\Sigma}_\theta^{-1}}{2}\Big]$$

*Proof.* Given that the distribution $p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)})$ in the VAE is Gaussian, it can be expressed as:

$$p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)}) = \mathcal{N}(Dec_\theta(\boldsymbol{z}^{(i)}), \boldsymbol{\Sigma}_\theta)$$

where $\boldsymbol{\Sigma}_\theta = \text{diag}(\sigma_\theta^2(\boldsymbol{z}^{(i)}))$.

The multivariate Gaussian distribution can be written as:

$$\mathcal{N}(Dec_\theta(\boldsymbol{z}^{(i)}), \boldsymbol{\Sigma}_\theta) = \frac{1}{(2\pi)^{\frac{k}{2}}|\boldsymbol{\Sigma}_\theta|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))^\intercal \boldsymbol{\Sigma}_\theta^{-1}(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))\right)$$

Therefore, the log-likelihood is:

$$\log p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)}) = -\frac{k}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}_\theta| - \frac{1}{2}(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))^\intercal \boldsymbol{\Sigma}_\theta^{-1}(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))$$

Since $(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))^\intercal \boldsymbol{\Sigma}_\theta^{-1}(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))$ is a scalar, it can be represented as a trace:

$$(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))^\intercal \boldsymbol{\Sigma}_\theta^{-1}(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)})) = \text{Tr}\{(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))^\intercal \boldsymbol{\Sigma}_\theta^{-1}(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))\}$$

Using the cyclic property of the trace, we get:

$$\text{Tr}\{(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))^\intercal \boldsymbol{\Sigma}_\theta^{-1}(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))\} = \text{Tr}\{(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))^\intercal \boldsymbol{\Sigma}_\theta^{-1}\}$$

Therefore, the maximum likelihood estimation (MLE) loss $L_{MLE}^{(i)}$ can be written as:

$$L_{MLE}^{(i)} = \mathbb{E}_{\boldsymbol{z}^{(i)}\sim q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})}\left[-\frac{k}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}_\theta| - \frac{1}{2}\text{Tr}\{(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))(\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)}))^\intercal \boldsymbol{\Sigma}_\theta^{-1}\}\right]$$

Simplifying, we get:

$$L_{MLE}^{(i)} = -\frac{k}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}_\theta| - \mathbb{E}_{\boldsymbol{z}^{(i)}\sim q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})}\left[\frac{1}{2}||\boldsymbol{x}^{(i)} - Dec_\theta(\boldsymbol{z}^{(i)})||^2 \boldsymbol{\Sigma}_\theta^{-1}\right]$$

Replacing $Dec_\theta(\boldsymbol{z}^{(i)})$ with $\tilde{\boldsymbol{x}}$, we obtain:

$$L_{MLE}^{(i)} = -\frac{k}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}_\theta| - \mathbb{E}_{\boldsymbol{z}^{(i)}\sim q(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})}\left[\frac{1}{2}||\boldsymbol{x}^{(i)} - \tilde{\boldsymbol{x}}||^2_{\boldsymbol{\Sigma}_\theta^{-1}}\right]$$

□

**Proposition 4.** *Given a VAE operating in the polarized regime, such that its decoder can be expressed as a combination of linear and non-linear transformation, $Dec_\theta(\boldsymbol{z}^{(i)}) = g_D^{(i)}(\boldsymbol{M}_D^{(i)}\boldsymbol{z}^{(i)})$ where $\boldsymbol{M}_D^{(i)}$ is a finite matrix, the local decoder can be approximately expressed as*

$$Dec_\theta(\boldsymbol{z}^{(i)}) = g_D^{(i)}(\boldsymbol{M}_D^{(i)}\boldsymbol{z}^{(i)}) \approx Dec_\theta(\boldsymbol{\mu}^{(i)}) + f_D^{(i)}(\boldsymbol{M}_D^{(i)}\boldsymbol{\epsilon}^{(i)})$$

*Proof.* Since the VAE is operating in the polarized regime, for all active latent variables , ${\sigma_j^{(i)}}^2 \ll 1$. Given that the matrix $\boldsymbol{M}_D^{(i)}$ is finite, $\boldsymbol{M}_D^{(i)}\boldsymbol{\epsilon}^{(i)} \ll 1$.

Further,

$$g_D^{(i)}(\boldsymbol{M}_D^{(i)}\boldsymbol{z}^{(i)}) = g_D^{(i)}(\boldsymbol{M}_D^{(i)}(\mu^{(i)} + \epsilon^{(i)}))$$

Applying Taylor Series approximation around the point $\boldsymbol{M}_D^{(i)}\mu^{(i)}$, we have,

$$g_D^{(i)}(\boldsymbol{M}_D^{(i)}(\mu^{(i)} + \epsilon^{(i)})) = g_D^{(i)}(\boldsymbol{M}_D^{(i)}\mu^{(i)}) + (\boldsymbol{M}_D^{(i)}\epsilon^{(i)})g_D^{(i)'}(\boldsymbol{M}_D^{(i)}\mu^{(i)}) + (\boldsymbol{M}_D^{(i)}\epsilon^{(i)})^2\frac{g_D^{(i)''}(\boldsymbol{M}_D^{(i)}\mu^{(i)})}{2!} + ...$$

$$= g_D^{(i)}(\boldsymbol{M}_D^{(i)}\mu^{(i)}) + f_D^{(i)}(\boldsymbol{M}_D^{(i)}\epsilon^{(i)})$$

where,

$$f_D^{(i)}(\boldsymbol{M}_D^{(i)}\epsilon^{(i)}) = (\boldsymbol{M}_D^{(i)}\epsilon^{(i)})g_D^{(i)'}(\boldsymbol{M}_D^{(i)}\mu^{(i)}) + (\boldsymbol{M}_D^{(i)}\epsilon^{(i)})^2\frac{g_D^{(i)''}(\boldsymbol{M}_D^{(i)}\mu^{(i)})}{2!} + ...$$

Again,

$$g_D^{(i)}(\boldsymbol{M}_D^{(i)}\mu^{(i)}) = \text{Dec}_\theta(\boldsymbol{\mu}^{(i)})$$

Hence,

$$\text{Dec}_\theta(\boldsymbol{z}^{(i)}) = g_D^{(i)}(\boldsymbol{M}_D^{(i)}\boldsymbol{z}^{(i)}) \approx \text{Dec}_\theta(\boldsymbol{\mu}^{(i)}) + f_D^{(i)}(\boldsymbol{M}_D^{(i)}\boldsymbol{\epsilon}^{(i)})$$

□

### A.3. Proof of Proposition 1, and Lemmas 1, 2, 3 and 6

**Proposition 1.** *Given $L_{rec}^{(i)} := \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[||Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - \boldsymbol{x}^{(i)}||^2]$, and assuming that the stochastic estimate, $Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)})$ is unbiased around $Dec_\theta(\boldsymbol{\mu}^{(i)})$, $L_{rec}^{(i)}$ can be decomposed into deterministic and stochastic parts:*

$$L_{rec}^{(i)} = L_{rec}^{\mu^{(i)}} + L_{rec}^{stoch^{(i)}}, \text{ where,}$$

$$\begin{aligned} L_{rec}^{stoch^{(i)}} &:= \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}||Dec_\theta(Enc_\phi(\boldsymbol{x}^{(i)})) - Dec_\theta(\boldsymbol{\mu}^{(i)})||^2, \\ L_{rec}^{\mu^{(i)}} &:= \left[||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||^2\right] \end{aligned} \tag{7}$$

*Proof.* As defined, $Enc_\phi(\boldsymbol{x}^{(i)}) = \boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}$, where $\boldsymbol{\epsilon}^{(i)}$ is the Gaussian noise while $\boldsymbol{\mu}^{(i)}$ is the deterministic part of the encoder derived from $\boldsymbol{x}^{(i)}$ as $\boldsymbol{\mu}^{(i)} = f_E(\boldsymbol{M_E}\boldsymbol{x}^{(i)})$. We write $L_{rec}^{(i)}$ in the following way

$$L_{rec}^{(i)} = \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[||Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - Dec_\theta(\boldsymbol{\mu}^{(i)}) + Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||^2]$$

$$\begin{aligned} L_{rec}^{(i)} = \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[&||Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - Dec_\theta(\boldsymbol{\mu}^{(i)})||^2 + ||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||^2 + \\ &2||Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - Dec_\theta(\boldsymbol{\mu}^{(i)})||||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||] \end{aligned} \tag{21}$$

Simplifying the terms in (21), we have,

$$\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||^2 = ||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||^2 \tag{22}$$

$$\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[2||Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - Dec_\theta(\boldsymbol{\mu}^{(i)})||||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||] =$$
$$2||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[||Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - Dec_\theta(\boldsymbol{\mu}^{(i)})||]$$

Again, since stochastic estimate, $Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)})$ is unbiased around $Dec_\theta(\boldsymbol{\mu}^{(i)})$, $\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)})] = Dec_\theta(\boldsymbol{\mu}^{(i)})$. Hence,

$$2||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[||Dec_\theta(\boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)}) - Dec_\theta(\boldsymbol{\mu}^{(i)})||] = 0 \tag{23}$$

Plugging (22) and (23) into (21), we get,

$$L_{rec}^{(i)} = [||Dec_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||^2] + \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}||Dec_\theta(Enc_\phi(\boldsymbol{x}^{(i)})) - Dec_\theta(\boldsymbol{\mu}^{(i)})||^2$$

Hence,

$$L_{rec}^{(i)} = L_{rec}^{\mu^{(i)}} + L_{rec}^{stoch^{(i)}}$$

□

**Lemma 1.** *With the approximation of the decoder being locally non-linear such that it can be expressed as $g_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})$, $L_{rec}^{stoch^{(i)}}$ can be expressed as follows:*

$$L_{rec}^{stoch^{(i)}} = \sum_{j=1}^{n} \{var[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] + f_D^2(\mathbf{0}) + f_D(\mathbf{0})f_D''(\mathbf{0})var[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]\} \tag{9}$$

*Proof.* Under the assumption that the decoder of the VAE is locally non-linear, s.t. it can be expressed as $f_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})$, $L_{rec}^{stoch^{(i)}}$ can be expressed as

$$L_{rec}^{stoch^{(i)}} = \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[||Dec_\theta(\boldsymbol{\mu}^{(i)}) + f_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)}) - Dec_\theta(\boldsymbol{\mu}^{(i)})||^2] = \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}||f_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})||^2$$

Further, $\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[||f_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})||^2]$ can be expressed as

$$\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[||f_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})||^2] = \sum_{j=1}^{n} \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[(f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}))^2]$$

Given a random variable, $o$, $var(o) = \mathbb{E}[o^2] - (\mathbb{E}[o])^2$. Hence,

$$L_{rec}^{stoch^{(i)}} = \sum_{j=1}^{n} \{var[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] + (\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[(f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}))])^2\} \tag{24}$$

From Lem. 6,

$$\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = f_D(\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + \frac{f_D''(\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2}\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2] \tag{25}$$

From Lem. 9, $\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] = \mathbf{0}$. Plugging it into (25), we have

$$\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = f_D(\mathbf{0}) + \frac{f_D''(\mathbf{0})}{2}\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})^2] \tag{26}$$

Again,

$$\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})^2] = var[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] + (\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2 = var[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]$$

Substitution the value in (26), we get,

$$\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = f_D(\mathbf{0}) + \frac{f_D^{''}(\mathbf{0})}{2} var[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]$$

Substituting into (24), and ignoring the higher order term, we have,

$$L_{rec}^{stoch^{(i)}} = \sum_{j=1}^{n} \left\{ var[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] + f_D^2(\mathbf{0}) + f_D(\mathbf{0}) f_D^{''}(\mathbf{0}) var[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] \right\}$$

□

**Lemma 2.** *Given the local decoder matrix $\boldsymbol{M}_D = \boldsymbol{U}_D\boldsymbol{\Sigma}_D\boldsymbol{V}_D^\top$, local encoder matrix $\boldsymbol{M}_E = \boldsymbol{U}_E\boldsymbol{\Sigma}_E\boldsymbol{V}_E^\top$, local decoder non-linearity $g_D$, local encoder non-linearity $g_E$ , the minimization of $L_{rec}^{\mu^{(i)}}$ depends either only on $\boldsymbol{V}_E$ or only on $\boldsymbol{U}_D$ and $f_D$, i.e., fixing $L_{rec}^{\mu^{(i)}}$ fixes $\boldsymbol{V}_E$, $\boldsymbol{U}_D$ and $f_D$.*

*Proof.* From 1, the loss function, $L_{\text{rec}}^{\mu^{(i)}} = \left[||\text{Dec}_\theta(\boldsymbol{\mu}^{(i)}) - \boldsymbol{x}^{(i)}||^2\right]$. Given $\boldsymbol{M}_D = \boldsymbol{U}_D\boldsymbol{\Sigma}_D\boldsymbol{V}_D^\top$, $\boldsymbol{M}_E = \boldsymbol{U}_E\boldsymbol{\Sigma}_E\boldsymbol{V}_E^\top$, $g_D$ and $g_E$, $L_{\text{rec}}^{\mu^{(i)}} = ||\boldsymbol{x}^{(i)} - g_D(\boldsymbol{M}_D(g_E(\boldsymbol{M_E}\boldsymbol{x}^{(i)})))||^2$. This can further be expressed as $||\boldsymbol{x}^{(i)} - F_D(F_D^{-1}(\boldsymbol{x}^{(i)}))||^2$, where $F_D(\boldsymbol{x}^{(i)})$ is defined as follows:

$$F_D^{-1}(\boldsymbol{x}^{(i)}) = g_E(\boldsymbol{M}_E(\boldsymbol{x}^{(i)})) \tag{27}$$

Hence,

$$F_D(\boldsymbol{x}^{(i)}) = \boldsymbol{M}_E^+ g_E^{-1}(\boldsymbol{x}^{(i)}) \tag{28}$$

where $\boldsymbol{M_E^+}$ is the pseudo inverse of $\boldsymbol{M}_E$.

Hence, the SVD decomposition of $\boldsymbol{M_E^+}$ is

$$\boldsymbol{M_E^+} = \boldsymbol{V_E}\boldsymbol{\Sigma_E^+}\boldsymbol{U_E^\top} \tag{29}$$

Substituting the (27), (28), (29) into $F_D(F_D^{-1}(\boldsymbol{x}^{(i)}))$, we get,

$$\begin{aligned}
\boldsymbol{M}_E^+ g_E^{-1}(g_E(\boldsymbol{M_E}\boldsymbol{x})) &= \boldsymbol{M}_E^+\boldsymbol{M_E}\boldsymbol{x}^{(i)} \\
&= \boldsymbol{V_E}\boldsymbol{\Sigma_E^+}\boldsymbol{U_E^\top}\boldsymbol{U_E}\boldsymbol{\Sigma_E}\boldsymbol{V_E^\top}\boldsymbol{x}^{(i)} \\
&= \boldsymbol{V_E}\boldsymbol{I}\boldsymbol{V_E^\top}\boldsymbol{x}^{(i)} \\
&= \boldsymbol{V_E}\boldsymbol{I_{d\times n}}\boldsymbol{I_{n\times d}}\boldsymbol{V_E^\top}\boldsymbol{x}^{(i)} \\
&= \boldsymbol{V_{E_d}}\boldsymbol{V_{E_d}^\top}\boldsymbol{x}^{(i)}
\end{aligned}$$

Hence, replacing $\boldsymbol{M}_E$ with $\boldsymbol{V_{E_d}}$, does not affect the loss. Further, the loss function is not dependent on $\boldsymbol{U_E}$ and $\boldsymbol{\Sigma_E}$.

Again, expressing the loss function, $||\boldsymbol{x}^{(i)} - g_D(\boldsymbol{M}_D(g_E(\boldsymbol{M_E}\boldsymbol{x}^{(i)})))||^2$ as $||\boldsymbol{x} - F_D(F_D^{-1}(\boldsymbol{x}^{(i)}))||^2$, $F_D(\boldsymbol{x}^{(i)})$ can be defined as:

$$F_D(\boldsymbol{x}^{(i)}) = g_D(\boldsymbol{M}_D(\boldsymbol{x}^{(i)})) \tag{30}$$

Hence,

$$F_D^{-1}(\boldsymbol{x}^{(i)}) = \boldsymbol{M_D^+} g_D^{-1}(\boldsymbol{x}^{(i)}) \tag{31}$$

where $\boldsymbol{M_D^+}$ is the pseudo inverse of $\boldsymbol{M}_D$.

Using SVD decomposition on $\boldsymbol{M_D^+}$ gives

$$\boldsymbol{M_D^+} = \boldsymbol{V_D}\boldsymbol{\Sigma_D^+}\boldsymbol{U_D^T} \tag{32}$$

Substituting the (30), (31), (32) into $F_D(F_D^{-1}(\boldsymbol{x}^{(i)}))$,

$$\begin{aligned} g_D(\boldsymbol{M_D}\boldsymbol{M_D^+}g_D^{-1}(\boldsymbol{x}^{(i)})) &= g_D(\boldsymbol{U_D}\boldsymbol{\Sigma_D}\boldsymbol{V_D^T}\boldsymbol{V_D}\boldsymbol{\Sigma_D^+}\boldsymbol{U_D^T}g_D^{-1}(\boldsymbol{x}^{(i)})) \\ &= g_D(\boldsymbol{U_D}\boldsymbol{I}\boldsymbol{U_D^T}g_D^{-1}(\boldsymbol{x}^{(i)})) \\ &= g_D(\boldsymbol{U_D}\boldsymbol{I_{n\times d}}\boldsymbol{I_{d\times n}}\boldsymbol{U_D^T}g_D^{-1}(\boldsymbol{x}^{(i)})) \\ &= g_D(\boldsymbol{U_{D_n}}\boldsymbol{U_{D_n}^T}g_D^{-1}(\boldsymbol{x}^{(i)})) \end{aligned}$$

Hence, replacing $\boldsymbol{M}_D$ with $\boldsymbol{U_{D_n}}$, does not change the loss. Again, the loss function is only dependent on $g_D$ and $\boldsymbol{U_D}$ and not on $\boldsymbol{V}_D$ and $\boldsymbol{\Sigma}_D$. Again, since $f_D$ is a polynomial function of $g_D$, the loss is dependent on $f_D$ too.

Hence, the minimization of the loss function depends either only on $\boldsymbol{V}_E$ or only on $\boldsymbol{U}_D$, $g_D$ and $f_D$, i.e., fixing $L_{rec}^{\mu^{(i)}}$ fixes $\boldsymbol{V}_E$, $\boldsymbol{U}_D$ and $f_D$. □

**Lemma 3.** *Fixing the deterministic part of the reconstruction loss ($L_{rec}^{\mu^{(i)}}$) and assuming the VAE is operating in a polarized regime, $L_{KL}$ can be expressed as:*

$$L_{KL} = \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}}\sum_{j\in\mathbb{V}_a} -\log({\sigma_j^{(i)}}^2) = \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{KL}^{(i)}$$

*Proof.* Operating the VAE to operate in a polarized regime, the passive latent variables are ignored, and for the active latent variables, $\sigma_j^2(x^{(i)}) \ll \log({\sigma_j^{(i)}}^2)$, which simplifies the KL loss function to:

$$L_{KL}^{(i)} = \frac{1}{2}\sum_{j\in\mathbb{V}_a}({\mu_j^{(i)}}^2 - \log({\sigma_j^{(i)}}^2) - 1) \tag{33}$$

From Lem. 2, the $L_{rec}^{\mu^{(i)}}$ depends either only on $\boldsymbol{V}_E$ or only on $f_D$ and $\boldsymbol{U}_D$, and not on the entire $g_E(\boldsymbol{M_E}\boldsymbol{x^{(i)}})$. Hence, fixing it fixes $\boldsymbol{V}_E$, $\boldsymbol{U}_D$, and $f_D$. The matrices $\boldsymbol{V}_D$, $\boldsymbol{\Sigma}_D$, $\boldsymbol{U_E}$ and $\boldsymbol{\Sigma_E}$ are not fixed and minimizing the stochastic loss under the constraint of $L_{KL}^{(i)}$ forces constraints on these matrices.

Fixing $\boldsymbol{V}_E$ and $\boldsymbol{U}_D$, and for a fixed $i$, 33 can be written as follows:

$$L_{KL}^{(i)} = ||\boldsymbol{\mu^{(i)}}||^2 + \sum_{j\in\mathbb{V}_a} -\log(\boldsymbol{\sigma_j^{(i)^2}})$$

Again, since $\boldsymbol{U}_D$ is fixed, $\boldsymbol{\mu}^{(i)}$ can only be affected by $\boldsymbol{V}_D$. However, since, $\boldsymbol{V}_D$ is orthogonal and hence norm-preserving, $||\boldsymbol{\mu}^{(i)}||^2$ is fixed. As a result, the only portion of $L_{KL}$ that affects the minimization of the stochastic reconstruction loss can be expressed as:

$$L_{KL} = \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}}\sum_{j\in\mathbb{V}_a} -\log({\boldsymbol{\sigma}_j^{(i)}}^2)$$

□

**Lemma 6.** *The expectation of a function $f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})$, $\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})]$ can be approximately written in terms of the expectation of $\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}$ as follows:*

$$\begin{aligned} \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] &= f_D(\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) \\ + \frac{f_D''(\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2}&\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2] \end{aligned}$$

*Proof.* Using Chebyshev's inequality, where $\mathcal{P}$ stands for probability, we have,

$$\mathcal{P}(|\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]| > a) \leq \frac{var(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})}{a^2} \tag{34}$$

Hence, given any $\delta > 0$, we can find an $a$, such that,

$$\mathcal{P}(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} \in [\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] - a, \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] + a] = \mathcal{P}(|\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]| \leq a) < 1 - \delta \tag{35}$$

Calculating $\mathbb{E}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})]$, we have,

$$\begin{aligned}\mathbb{E}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = \int_{|x-\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]|\leq a} f_D(x)dF_D(x) + \\ \int_{|x-\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]|> a} f_D(x)dF_D(x)\end{aligned} \tag{36}$$

where, $F_D(x)$ is the distribution function for $x$.

Since the domain of the first integral, $[\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] - a, \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] + a]$, is bounded and closed, Taylor Expansion can be applied in that interval. Applying Taylor's expansion in the interval $[\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] - a, \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] + a]$, and considering only the first four terms, we have,

$$\begin{aligned}f_D(x) = f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) \\ + \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2}(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2 \\ + \frac{f_D^{'''}(U)}{3!}(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^3\end{aligned} \tag{37}$$

where $U \in [\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] - a, \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] + a]$. Substituting (37) in (36), we get,

$$\begin{aligned}\mathbb{E}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = \int_{|x-\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]|\leq a} \Big\{ f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) \\ + \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2}(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2 \Big\} dF_D(x) + \int_{|x-\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]|\leq a} \frac{f_D^{'''}(U)}{3!}(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^3 dF_D(x) + \\ \int_{|x-\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]|> a} f_D(x)dF_D(x)\end{aligned} \tag{38}$$

The interval or the domain is increased by increasing the value of $a$, i.e., $a \to \infty$. From (34) and (35), $\mathcal{P}(|\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]| > a) \to 0$ and $\mathcal{P}(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} \in [\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] - a, \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] + a]) \to 1$. This simplifies (38) as follows:

$$\begin{aligned}\mathbb{E}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] \approx \int_{-\infty}^{+\infty} \Big\{ f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) \\ + \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2}(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2 \Big\} dF_D(x) + \int_{-\infty}^{+\infty} \frac{f_D^{'''}(U)}{3!}(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^3 dF_D(x)\end{aligned} \tag{39}$$

as $\int_{|x-\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]|>a} f_D(x)dF_D(x) \to 0$.

Simplifying the terms in (39) further, we get,

$$\int_{-\infty}^{+\infty} f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])dF_D(x) = f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) \int_{-\infty}^{+\infty} dF_D(x) = f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) \tag{40}$$

$$\int_{-\infty}^{+\infty} f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])dF_D(x) = \Big\{ f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) \int_{-\infty}^{+\infty} xdF_D(x) - (\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) \Big\} = 0 \tag{41}$$

and

$$\begin{aligned}\int_{-\infty}^{+\infty} \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2}(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2 dF_D(x) = \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2} \int_{-\infty}^{+\infty} (x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2 dF_D(x) = \\ \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2} \mathbb{E}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2]\end{aligned} \tag{42}$$

Substituting (40), (41) and (42) in (39), we get,

$$\mathbb{E}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2}\mathbb{E}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2] + R_E \tag{43}$$

Ignoring the higher order integral, we get,

$$\mathbb{E}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2}\mathbb{E}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2] \tag{44}$$

□

### A.4. The Proof of the proposed Theorem-1 and the relevant Lemmas

**Theorem 1.** *Given independent data samples $\mathbf{x}^{(i)}$, if we fix the $\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{KL}^{(i)}$ for a constant $C_{KL}^{(i)}$, and $\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{rec}^{\mu^{(i)}}$, then the minimization of the VAE loss $L$ in (5) reduces to the minimization of the stochastic reconstruction loss $L_{rec}^{stoch^{(i)}}$:*

$$\min_{\sigma_j^{(i)}>0, \boldsymbol{V}_D,} \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log L_{rec}^{stoch^{(i)}} \quad s.t. \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} L_{KL}^{(i)} = C_{KL}. \tag{10}$$

*Then, the following hold for the local minima:*

*(a) Every local minimum is a global minimum.*
*(b) In every global minimum, the columns of every $\boldsymbol{M}_D$ are orthogonal.*

*Further, the variance of a latent variable is inversely proportional to the norm of the corresponding column in the linear part of the local decoder:*

$$(c) \qquad \sigma_j^{(i)^2} \propto \frac{1}{||\boldsymbol{c}_j||^2} \;\; \forall i$$

*where $\boldsymbol{c}_j$ is the $j$-th column of $\boldsymbol{M}_D$.*

*Proof.* Proof of part (b):

From Lemma 1 we have that

$$L_{rec}^{stoch^{(i)}} = \sum_{j=1}^{n} \Big\{ \text{var}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] + f_D^2(\mathbf{0}) + f_D(\mathbf{0}) f_D''(\mathbf{0}) \text{var}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] \Big\} \tag{45}$$

where $\boldsymbol{M}_{Dj}$ is the $j$-th row and $n$ is the total number of rows in $\boldsymbol{M}_D$.

Then from Lemma 7, we know that

$$\text{var}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = \Big( f_D'(\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})) \Big)^2 \text{var}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] \tag{46}$$

Again, from Lemma 8, we know that

$$\text{var}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] = \sum_{k=1}^{d} a_{j,k}^2 \boldsymbol{\sigma}_k^{(i)^2} \tag{47}$$

where $d$ is the total number of columns in $\boldsymbol{M}_D$.

From Lemma 9,

$$\mathbb{E}_i[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] = \mathbf{0} \tag{48}$$

Plugging (47) and (48) into (46), we have

$$\text{var}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = [f_D'(\mathbf{0})]^2 \sum_{k=1}^{d} a_{j,k}^2 \boldsymbol{\sigma}_k^{(i)^2} \tag{49}$$

Plugging (49) into (45), we have

$$\begin{aligned} L_{rec}^{stoch^{(i)}} &= \sum_{j=1}^{n} \left\{ [f'_D(\mathbf{0})]^2 + f_D(\mathbf{0}) f''_D(\mathbf{0}) \right\} \sum_{k=1}^{d} a_{j,k}^2 \boldsymbol{\sigma}_k^{(i)^2} + n f_D^2(\mathbf{0}) \\ &= \left\{ [f'_D(\mathbf{0})]^2 + f_D(\mathbf{0}) f''_D(\mathbf{0}) \right\} \sum_{j=1}^{n} \sum_{k=1}^{d} a_{j,k}^2 \boldsymbol{\sigma}_k^{(i)^2} + n f_D^2(\mathbf{0}) \end{aligned} \tag{50}$$

Let $\boldsymbol{c}_k$ be the $k$-th column of the $\boldsymbol{M}_D$ matrix. We know that

$$||\boldsymbol{c}_k||^2 = \sum_{j=1}^{n} a_{j,k}^2 \tag{51}$$

Substituting (51) in (50), we have

$$L_{rec}^{stoch^{(i)}} = \left\{ [f'_D(\mathbf{0})]^2 + f_D(\mathbf{0}) f''_D(\mathbf{0}) \right\} \sum_{k=1}^{d} ||\boldsymbol{c}_k||^2 \boldsymbol{\sigma}_k^{(i)^2} + n f_D^2(\mathbf{0})$$

Since $n f_D^2(\mathbf{0})$ is a constant for a fixed $\boldsymbol{x}^{(i)}$, minimizing $L_{rec}^{stoch^{(i)}}$ does not depend on it. Hence, we minimize $D_{rec}^{stoch^{(i)}} = L_{rec}^{stoch^{(i)}} - n f_D^2(\mathbf{0})$. The value of the minima of the two functions would differ by $n f_D^2(\mathbf{0})$.

Using the AM-GM inequality on $D_{rec}^{stoch^{(i)}}$, we have

$$\begin{aligned} D_{rec}^{stoch^{(i)}} &= \left\{ [f'_D(\mathbf{0})]^2 + f_D(\mathbf{0}) f''_D(\mathbf{0}) \right\} \sum_{k=1}^{d} ||\boldsymbol{c}_k||^2 \boldsymbol{\sigma}_k^{(i)^2} \\ &\geq \left\{ [f'_D(\mathbf{0})]^2 + f_D(\mathbf{0}) f''_D(\mathbf{0}) \right\} d \left( \prod_{k=1}^{d} ||\boldsymbol{c}_k||^2 \boldsymbol{\sigma}_k^{(i)^2} \right)^{\frac{1}{d}} \end{aligned}$$

with equality if and only if

$$\frac{||\boldsymbol{c}_j||^2}{||\boldsymbol{c}_k||^2} = \frac{\boldsymbol{\sigma}_k^{(i)^2}}{\boldsymbol{\sigma}_j^{(i)^2}} \tag{52}$$

for any $j$ and $k$ in $\{1, \ldots, d\}$.

Taking the logarithm, we have

$$\log(D_{rec}^{stoch^{(i)}}) = \log \left\{ [f'_D(\mathbf{0})]^2 + f_D(\mathbf{0}) f''_D(\mathbf{0}) \right\} + \log(d) + \tfrac{1}{d} \sum_{k=1}^{d} \log(\boldsymbol{\sigma}_k^{(i)^2}) + \tfrac{2}{d} \log \left( \prod_{k=1}^{d} ||\boldsymbol{c}_k|| \right)$$

Taking the summation over all the values of $x_i \in \mathbb{X}$, we have

$$\sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \log(D_{rec}^{stoch^{(i)}}) = \sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \log \left\{ [f'_D(\mathbf{0})]^2 + f_D(\mathbf{0}) f''_D(\mathbf{0}) \right\} + N \log(d) - \tfrac{C_{KL}}{d} + \tfrac{2}{d} \sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \log \left( \prod_{k=1}^{d} ||\boldsymbol{c}_k|| \right)$$

From Lemma 10, we have

$$\sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \log(D_{rec}^{stoch^{(i)}}) \geq \sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \log \left\{ [f'_D(\mathbf{0})]^2 + f_D(\mathbf{0}) f''_D(\mathbf{0}) \right\} + N \log(d) - \tfrac{C_{KL}}{d} + \tfrac{2}{d} \sum_{\boldsymbol{x}^{(i)} \in \mathbb{X}} \log(\text{Sing}(\boldsymbol{M}_D)) \tag{53}$$

It can be seen that the RHS of the above equation does not depend on $\boldsymbol{\sigma}_k^{(i)^2}$s. Also, it is independent of the orthogonal matrices $\boldsymbol{V}_D$ as these do not influence the singular values of $\boldsymbol{M}_D$.

However, according to Lemma 10, to minimize the LHS, we set the matrices $\boldsymbol{V}_D$ such that the columns of $\boldsymbol{M}_D$ are orthogonal to each other. Also, the values of ${\boldsymbol{\sigma}_k^{(i)}}^2$s are chosen so as to achieve equality in the AM-GM inequality. Hence, in every global minimum, the columns of $\boldsymbol{M}_D$ are orthogonal.

Proof of part (c):
Part (c) of the Theorem follows directly from (52).

Proof of part (a): To prove part (a), we show that at least one small change can be made in the objective function that would minimize it until it has reached the global minimum, which is the RHS of (53). Since the RHS of (53) is dependent only on the AM-GM inequality and Lemma 10, it suffices to show that local minima do not exist for both of these inequalities and that at least one small perturbation would always improve the LHS of these inequalities until they become equal to their RHS, which is their global minimum and that no other local minima exist. We have shown this in Lemma 11 and Lemma 12. □

**Lemma 7.** *The variance of a function $f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})$, denoted as $var[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})]$, can be expressed in terms of the variance of $\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}$ as follows:*

$$var[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = (f_D^{'}(\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]))^2 \cdot var[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]$$

*Proof.* From (43), in Lem. 6, we get,

$$\mathbb{E}[f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{2}\mathbb{E}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2] + R_E$$

where $R_E$ is the remaining integral.
Again,

$$\begin{aligned} f_D^2(x) = f_D^2(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + 2f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + \\ [(f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2 + f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])](x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2 + \\ \frac{(f_D^2(K))^{'''}}{3!}(x - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^3 \end{aligned}$$

From Lem. 6, taking expectation over $f_D^2$, we have,

$$\begin{aligned} \mathbb{E}[f_D^2(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})] = f_D^2(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]) + [(f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]))^2 + f_D(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])] \\ \mathbb{E}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)} - \mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2] + \tilde{R_E} \end{aligned}$$

As we know $Var(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}) = \mathbb{E}[(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})^2] - (\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])^2$. Evaluating, we have,

$$Var(f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})) = (f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]))^2 Var(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}) + \frac{f_D^{''}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}])}{4}Var^2(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}) + T_E$$

Approximating to only the first term, we have,

$$Var(f_D(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})) \approx (f_D^{'}(\mathbb{E}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]))^2 Var(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})$$

□

**Lemma 8.** *The variance of $\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}$, $var[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}]$, can be expressed as follows :*

$$var[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] = \sum_{k=1}^{d} a_{j,k}^2 {\boldsymbol{\sigma}_k^{(i)}}^2$$

*where $\boldsymbol{M}_{Dj}$ are the rows of the decoder matrix $\boldsymbol{M}_D$ and $a_{j,k}$ is the element in the $j$-th row and $k$-th column of $\boldsymbol{M}_D$.*

*Proof.* We know that, $var(cB) = c^2 var(B)$, where, $c$ is a constant.

Now,

$$var(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}) = var(\sum_{k=1}^{d} a_{jk}\boldsymbol{\epsilon}_k^{(i)}) = \sum_{k=1}^{d} a_{jk}^2 var(\boldsymbol{\epsilon}_k^{(i)}) = \sum_{k=1}^{d} a_{jk}^2 {\boldsymbol{\sigma}_k^{(i)}}^2$$

□

**Lemma 9.** *The expectation of $\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}$, $\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)})$ can be expressed as follows:*

$$\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}(\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}) = \mathbf{0}$$

*Proof.* For a Gaussian distribution, $D \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$, $\mathbb{E}[D] = \boldsymbol{\mu}$.

Again, $\epsilon_k^{(i)} \sim \mathcal{N}(0, {\sigma_k^{(i)}}^2)$, $\mathbb{E}_{\epsilon_k^{(i)}}[\epsilon_k^{(i)}] = 0$.
Hence,

$$\mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[\boldsymbol{M}_{Dj}\boldsymbol{\epsilon}^{(i)}] = \mathbb{E}_{\boldsymbol{\epsilon}^{(i)}}[(\sum_{k=1}^{d} a_{jk}\epsilon_k^{(i)})] = \sum_{k=1}^{d} a_{jk}\mathbb{E}_{\epsilon_k^{(i)}}[\epsilon_k^{(i)}] = 0$$

□

**Proposition 5.** *For a matrix $\boldsymbol{M}_D \in \mathbb{R}^{n\times d}$ with SVD $\boldsymbol{M}_D = \boldsymbol{U_D}\boldsymbol{\Sigma_D}\boldsymbol{V_D}^\top$, the following statements are equivalent.*

*a) The columns of $\boldsymbol{M}_D$ are pairwise orthogonal.*

*b) The matrix $\boldsymbol{M_D}^\top\boldsymbol{M_D}$ is diagonal.*

*c) The columns of $\boldsymbol{\Sigma_D}\boldsymbol{V_D}^\top$ are pairwise orthogonal.*

*Proof.* The statements (a) and (b) are equivalent as the columns of $\boldsymbol{M}_D$, $\boldsymbol{c}_i$ are orthogonal and hence $\boldsymbol{c}_i^\top\boldsymbol{c}_j = 0\ \forall\ i \neq j$ while $\boldsymbol{c}_i^\top\boldsymbol{c}_i \neq \mathbf{0}$.

The equivalence of statements (a) and (c) can be proved as follows. Suppose we define $\boldsymbol{M_D'} = \boldsymbol{\Sigma_D}\boldsymbol{V_D}^\top$. We can see that,

$$\boldsymbol{M_D'}^\top\boldsymbol{M_D'} = \boldsymbol{V_D}\boldsymbol{\Sigma_D}^\top\boldsymbol{\Sigma_D}\boldsymbol{V_D}^\top = \boldsymbol{V_D}\boldsymbol{\Sigma_D}^\top\boldsymbol{U_D}^\top\boldsymbol{U_D}\boldsymbol{\Sigma_D}\boldsymbol{V_D}^\top = \boldsymbol{M_D}^\top\boldsymbol{M_D}$$

Since, the columns of $\boldsymbol{M}_D$ are orthogonal, from the equivalence of (a) and (b), $\boldsymbol{M_D}^\top\boldsymbol{M_D}$ is a diagonal matrix which also implies that $\boldsymbol{M_D'}^\top\boldsymbol{M_D'}$ is diagonal. Again from the equivalence of (a) and (b) the columns of $\boldsymbol{M_D'}$ are orthogonal. □

**Lemma 10.** *Let $\boldsymbol{M}_D \in \mathbb{R}^{n\times d}$ be a matrix where, $d < n$, be a matrix with column vectors $\boldsymbol{c}_1...\boldsymbol{c}_d$ and non-zero singular vectors $s_1...s_d$. It can be claimed that,*

$$\prod_{j=1}^{d} ||\boldsymbol{c}_j|| \geq \mathit{Sing}(\boldsymbol{M}_D)$$

*where, $\mathit{Sing}(\boldsymbol{M}_D)$ is the product of singular values of $\boldsymbol{M}_D$. The condition of equality is when $\boldsymbol{c}_1...\boldsymbol{c}_d$ are pairwise orthogonal.*

*Proof.* Suppose $\boldsymbol{M}_D = \boldsymbol{U_D}\boldsymbol{\Sigma_D}\boldsymbol{V_D}^\top$. We first show that multiplying both sides of the equation by $\boldsymbol{U}_D$ does not change the inequality. Firstly, for the RHS, the singular value of $\boldsymbol{U_D}\boldsymbol{M_D}$ is the same as that of $\boldsymbol{M}_D$. In the LHS, the $\boldsymbol{c}_j$s are the images of the $\boldsymbol{e_j}$s, as $\boldsymbol{c}_j$s can be expressed as $\boldsymbol{c}_j = \boldsymbol{M}_D\boldsymbol{e_j}\ \forall\ j \in \{1...d\}$. However, $\boldsymbol{U}_D$, being an orthogonal matrix and hence an isometry, we get $||\boldsymbol{U_D}\boldsymbol{M_D}\boldsymbol{e}_j|| = ||\boldsymbol{M_D}\boldsymbol{e}_j|| = ||\boldsymbol{c}_j||\ \forall\ j$ and hence, the column norms of $\boldsymbol{M}_D$ remains same as the column norms of $\boldsymbol{U_D}\boldsymbol{M_D}$.

Since we can see that both sides of the inequality are invariant to multiplication by $\boldsymbol{U}_D$, we restrict to the values of $\boldsymbol{M}_D$, which can be expressed as $\boldsymbol{M}_D = \boldsymbol{\Sigma_D V_D^\top}$. Since, we assume that $d < n$, $\boldsymbol{\Sigma_{D_{D\times d}}}$, the $d \times d$ left corner submatrix of $\boldsymbol{\Sigma}_D$ contains all the nonzero singular values. We can define $\boldsymbol{M_D^{sq}} = \boldsymbol{\Sigma_{D_{D\times d}} V_D^\top}$, which is basically a square matrix consisting of all the nonzero rows of the matrix $\boldsymbol{M}_D$. Again, this implies,

$$(\boldsymbol{M_D^{sq}})^\top \boldsymbol{M_D^{sq}} = \boldsymbol{M_D^\top M_D}$$

More specifically, the column norms of $\boldsymbol{M}_D$ and $\boldsymbol{M_D^{sq}}$ are equal and hence according to Hadamard's Inequality, we have,

$$\prod_{k=1}^{d} ||\boldsymbol{c}_k|| = \prod_{k=1}^{d} ||\boldsymbol{c}_k^{sq}|| \geq |det(\boldsymbol{M_D^{sq}})| = \text{Sing}(\boldsymbol{M}_D)$$

where, $\boldsymbol{c}_k$ are the columns of $\boldsymbol{M}_D$ and $\boldsymbol{c}_k^{sq}$ are the columns of $\boldsymbol{M_D^{sq}}$.

Further, according to Hadamard's inequality, equality occurs only when the columns of $\boldsymbol{M_D^{sq}}$ are orthogonal. Again that implies that the columns of $\boldsymbol{M}_D$ are orthogonal and from Proposition 5, all possible $\boldsymbol{M}_D$s and not just the simplified ones assumed are orthogonal. Hence, the equality holds when the columns of any matrix $\boldsymbol{M}_D$ are orthogonal. □

**Lemma 11.** *Given non-negative values $a_1, a_2, a_3....a_N$ for which,*

$$\frac{1}{N}\sum_{i=1}^{N} a_i > (\prod_{i=1}^{N} a_i)^{\frac{1}{N}}$$

*there exists at least one perturbation of $a_i$s, $a_i^{'}$s such that,*

$$\frac{1}{N}\sum_{i=1}^{N} a_i > \frac{1}{N}\sum_{i=1}^{N} a_i^{'} \geq (\prod_{i=1}^{N} a_i^{'})^{\frac{1}{N}} = (\prod_{i=1}^{N} a_i)^{\frac{1}{N}} \tag{54}$$

*Proof.* Selecting $i \neq j$, we set $a_i^{'} = \frac{a_i}{d+\delta}$ and $a_j^{'} = a_j(d+\delta)$, where, $d \geq 1$ and $\delta \leq 1$. For all other $k \in \{1, ...N\}$, $a_k^{'} = a_k$. We can see that,

$$a_i + a_j - a_i^{'} - a_j^{'} = (a_i + a_j)\frac{\delta}{1+\delta} > 0$$

Hence, we have, $a_i + a_j > a_i^{'} + a_j^{'}$. Again, we can see that, $a_i^{'}a_j^{'} = a_i a_j$ This ensures that at least one small perturbation of $a_i$s would satisfy (54). □

**Lemma 12.** *For a matrix $\boldsymbol{M_n} \in \mathbb{R}^{n\times n}$ with SVD $\boldsymbol{M_n} = \boldsymbol{U\Sigma V}^\top$, and column vectors $\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3...\boldsymbol{c}_n$, for which,*

$$\prod_{j=1}^{n} ||\boldsymbol{c}_j|| > |det(\boldsymbol{M_n})| \tag{55}$$

*there exists at least one $\boldsymbol{V}^{'}$, which is a small perturbation of $\boldsymbol{V}$, and a matrix $\boldsymbol{M_n^{'}}$with columns $\boldsymbol{c}_1', \boldsymbol{c}_2', \boldsymbol{c}_3'...\boldsymbol{c}_n'$ such that,*

$$\prod_{j=1}^{n} ||\boldsymbol{c}_j|| > \prod_{j=1}^{n} ||\boldsymbol{c}_j'|| \tag{56}$$

*Proof.* We establish the proof by induction on n. For n=2, we define a rotation matrix $\boldsymbol{R_\theta}$ where theta is the angle of rotation for the rotation matrix. By setting $\boldsymbol{V'} = \boldsymbol{V R_\theta}$, we can verify that (56) is satisfied.

For $n > 2$, we can see that, (55) implies $\boldsymbol{c}_i^T \boldsymbol{c}_j \neq 0$ for some $i \neq j$. Let us assume that $i = 1$ and $j = 2$ in this case. Let for $k > 2$, $\boldsymbol{c}_k' = \boldsymbol{c}_k$. We now consider, $\boldsymbol{R_\theta^{2D}}$, where,

$$\boldsymbol{R_\theta^{2D}} = \begin{pmatrix} \boldsymbol{R_\theta} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\mathrm{I}_{n-2}} \end{pmatrix}$$

is a block diagonal matrix. $\boldsymbol{R_\theta}$ is the rotational matrix that we defined previously. Further, since $\boldsymbol{U}$ can be set to $\boldsymbol{\mathrm{I}_n}$ as either side of (55) is not influenced by an isometry, we can reduce the above situation to $n = 2$ case. Hence by induction, the proof is complete. □

### A.5. Proof of Lem. 4

**Lemma 4.** *Given $\boldsymbol{M}_D = \boldsymbol{U}_D\boldsymbol{\Sigma}_D\boldsymbol{V}_D^\top$, such that the columns of $\boldsymbol{M}_D$ are orthogonal, and $\boldsymbol{M}_D$ has unique non-zero singular values, the following hold: (a) $\boldsymbol{U}_D$ is an orthogonal matrix, (b) the diagonal elements of $\boldsymbol{\Sigma}_D$ are the norms of the columns of $\boldsymbol{M}_D$, and (c) $\boldsymbol{V}_D = \boldsymbol{I}$.*

*Proof.* From Proposition 5, $\boldsymbol{M_D^\top M_D}$ is a diagonal matrix. Moreover, the diagonal elements are $\boldsymbol{c}_k^\top \boldsymbol{c}_k = ||\boldsymbol{c}_k||_2^2$. Furthermore, from the characteristic equation of the matrix $\boldsymbol{M_D^\top M_D}$, the eigenvalues of $\boldsymbol{M_D^\top M_D}$ are $||\boldsymbol{c}_k||_2^2 \; \forall \; k \in \{1..d\}$. Hence the singular values are $||\boldsymbol{c_k}||_2$. Which proves part (b) of the lemma.

Again solving for $\boldsymbol{M_D^\top M_D x}_k = \lambda_k \boldsymbol{x}_k$, and given the singular values are non-zero and unique, gives us that the solution only consists of the respective $\boldsymbol{e}_k$s. Hence, the matrix $\boldsymbol{V}_D = \mathbf{I}$. This proves part (c) of the lemma.

Further, we use the equation, $\boldsymbol{M}_D = \boldsymbol{U_D\Sigma_D V_D^\top}$. Considering the RHS, since $\boldsymbol{V_D^\top} = \mathbf{I}$, the RHS becomes $\boldsymbol{U_D\Sigma_D}$ and the individual elements of the product matrix become $u_{ij}||\boldsymbol{c}_j||_2$ where $u_{ij}$ are the elements of matrix $\boldsymbol{U}_D$. Also, let the individual elements of the matrix $\boldsymbol{M}_D$, be $a_{ij}$. Hence $u_{ij} = \frac{a_{ij}}{||\boldsymbol{c}_j||_2}$. Hence, it is clear that the columns of $\boldsymbol{U}_D$ are normalized. Also the columns of $\boldsymbol{U}_D$ are orthogonal to each other as the columns of $\boldsymbol{M}_D$ are orthogonal to each other. Hence, the matrix $\boldsymbol{U}_D$ is orthogonal. This proves part (a) of the lemma.

□

### A.6. Proof of Lem. 5

**Lemma 5.** *Given a $L_{rec}^{stoch^{(i)}}$, orthogonality in the linear component of the Decoder's function transformation, $\boldsymbol{M}_D$, promotes a lower $L_{KL}$.*

*Proof.* Let $\boldsymbol{M}_D$ be a non-orthogonal decoder matrix. Also, let $\boldsymbol{M}_{D_{ortho}}$ be the matrix whose columns are orthogonal to each other, solution to the optimization in Theorem 1, by setting the matrix $\boldsymbol{V}_D$. From, (53), we have,

$$\begin{aligned}\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log(D_{rec}^{stoch^{(i)}}) &= \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log\left\{[f_D^{'}(\mathbf{0})]^2 + f_D(\mathbf{0})f_D^{''}(\mathbf{0})\right\} + N\log(d) - \frac{C_{KL}}{d} + \frac{2}{d}\sum_{\boldsymbol{x^{(i)}}\in\mathbb{X}} \log\prod_{k=1}^{d}||\boldsymbol{c}_k|| \\ &\geq \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log\left\{[f_D^{'}(\mathbf{0})]^2 + f_D(\mathbf{0})f_D^{''}(\mathbf{0})\right\} + N\log(d) - \frac{C_{KL}}{d} + \frac{2}{d}\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log(\text{Sing}(\boldsymbol{M}_D))\end{aligned}$$

where, from Theorem 1, $D_{rec}^{stoch^{(i)}} = L_{rec}^{stoch^{(i)}} - nf_D^2(\mathbf{0})$.

Given a fixed $L_{rec}^{stoch^{(i)}}$, we have,

$$\begin{aligned}\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log(D_{rec}^{stoch^{(i)}}) &= \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log\left\{[f_D^{'}(\mathbf{0})]^2 + f_D(\mathbf{0})f_D^{''}(\mathbf{0})\right\} + N\log(d) - \frac{C_{KL}}{d} + \frac{2}{d}\sum_{\boldsymbol{x^{(i)}}\in\mathbb{X}} \log\prod_{k=1}^{d}||\boldsymbol{c}_k|| \\ &= \sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log\left\{[f_D^{'}(\mathbf{0})]^2 + f_D(\mathbf{0})f_D^{''}(\mathbf{0})\right\} + N\log(d) - \frac{C_{KLortho}}{d} + \frac{2}{d}\sum_{\boldsymbol{x}^{(i)}\in\mathbb{X}} \log(\text{Sing}(\boldsymbol{M}_{D_{ortho}}))\end{aligned}$$

Hence,

$$\begin{aligned}C_{KL} - C_{KLortho} &= 2\Big\{\sum_{\boldsymbol{x^{(i)}}\in\mathbb{X}} \log\prod_{k=1}^{d}||\boldsymbol{c}_k|| - \log(\text{Sing}(\boldsymbol{M}_{D_{ortho}}))\Big\} \\ &\geq 2\Big\{\log(\text{Sing}(\boldsymbol{M}_D)) - \log(\text{Sing}(\boldsymbol{M}_{D_{ortho}}))\Big\}\end{aligned}$$

The term on the RHS is greater than 0. Also due to polarized regime, $L_{KL} > 0$, and hence, $C_{KL}$ and $C_{KLortho}$ are $> 0$. Hence, we have,

$$C_{KL} - C_{KLortho} > 0$$

Hence, $L_{KLortho}$ corresponding to the decoder $\boldsymbol{M}_{D_{ortho}}$ has a lower positive value as compared to $L_{KL}$ corresponding to decoder the decoder $\boldsymbol{M}_D$ proving our lemma.

□

### A.7. Proof of Theorem 2

**Theorem 2.** *For a VAE, given $\boldsymbol{z}^{(i)} \sim Enc_\phi(\boldsymbol{x}^{(i)})$ and $\boldsymbol{z}^{(k)} \sim Enc_\phi(\boldsymbol{x}^{(k)})$, where, $\boldsymbol{x}^{(k)}$ are the $k^{(i)}$ nearest neighbours of $\boldsymbol{x}^{(i)}$, we define $Dist(L_{KL})$ as follows:*

$$Dist(L_{KL}) = \mathbb{E}_{\boldsymbol{x}^{(i)},\boldsymbol{z}^{(i)},\boldsymbol{z}^{(k)}}\left[\sum_{k=1}^{k^{(i)}} ||\boldsymbol{z}^{(i)} - \boldsymbol{z}^{(k)}||^2\right]$$

*The following hold:*

*(a) Given $Enc_\phi(\boldsymbol{x}^{(i)}) \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ overlaps (is close) with $k^{(i)}$ posterior probabilities, they must be posterior probabilities generated by the $k^{(i)}$ nearest neighbours of $\boldsymbol{x}^{(i)}$ in $X$, i.e. $Enc_\phi(\boldsymbol{x}^{(k)}) \sim q_\phi(\boldsymbol{z}^{(k)}|\boldsymbol{x}^{(k)})$. Here, for every $\boldsymbol{x}^{(i)}$, we have $k^{(i)}$ number of $\boldsymbol{x}^{(k)}$-s, whose posterior probabilities, $q_\phi(\boldsymbol{z}^{(k)}|\boldsymbol{x}^{(k)})$, overlap with the posterior probability $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ in the latent space.*

*(b) Given, $L_{KL'} < L_{KL}$, $Dist(L_{KL'}) < Dist(L_{KL})$.*

*Proof.* For a VAE, loss $L$ is defined as $L = L_{rec} + \beta L_{KL}$, where $L_{rec}$ and $L_{KL}$ are defined as:

$$L_{rec} = \sum_{\boldsymbol{x}^{(i)} \in X} [||Dec_\theta(Enc_\phi(\boldsymbol{x}^{(i)})) - \boldsymbol{x}^{(i)}||^2] = \sum_{\boldsymbol{x}^{(i)} \in X} [||\tilde{\boldsymbol{x}}^{(i)} - \boldsymbol{x}^{(i)}||^2] \tag{57}$$

$$L_{KL} = \frac{1}{2} \sum_{\boldsymbol{x}^{(i)} \in X} \sum_{j \in \mathbb{V}_a} (\mu_j^{(i)^2} + \sigma_j^{(i)^2} - \log(\sigma_j^{(i)^2}) - 1) \tag{58}$$

To minimize the loss $L$, given a $L_{KL}$, the architecture aligns the latent space such that $L_{rec}$ can be minimized.

From (58), decreasing $L_{KL}$ causes $\mu_j^{(i)}$ to approach 0 (decrease) and $\sigma_j^{(i)}$ to approach 1. The decreased means and the broadened variances cause the posterior probabilities, $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$-s to overlap.

We consider a random point $\boldsymbol{x}^{(i)} \in \mathbb{X}$, with posterior probability $Enc_\phi(\boldsymbol{x}^{(i)}) \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$. Considering posterior probabilities $q_\phi(\boldsymbol{z}^{(j)}|\boldsymbol{x}^{(j)})$s which overlap with $Enc_\phi(\boldsymbol{x}^{(i)}) \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$, we define $\boldsymbol{z}$ as follows:

$$\mathcal{Z} = \{\boldsymbol{z} \sim Enc_\phi(\boldsymbol{x}^{(i)}) | z \in \boldsymbol{z}_{sam}^{(i)} \cap \boldsymbol{z}_{sam}^{(j)}, \boldsymbol{z}_{sam}^{(i)} \in q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)}), \boldsymbol{z}_{sam}^{(j)} \in q_\phi(\boldsymbol{z}^{(j)}|\boldsymbol{x}^{(j)}) \, \forall j \ni \boldsymbol{z}_{sam}^{(i)} \cap \boldsymbol{z}_{sam}^{(j)} \neq \phi\}$$

Hence, $\boldsymbol{z}$ can be represented in 2 different ways as follows:

$$\boldsymbol{z} = \boldsymbol{\mu}^{(i)} + \boldsymbol{\epsilon}^{(i)} = \boldsymbol{\mu}^{(j)} + \boldsymbol{\epsilon}^{(j)}$$

Given an optimal decoder, $Dec_\theta(\boldsymbol{z}) = \tilde{\boldsymbol{x}}^{(i)}$ or $Dec_\theta(\boldsymbol{z}) = \tilde{\boldsymbol{x}}^{(j)_{(i)}}$, where $\tilde{\boldsymbol{x}}^{(j)_{(i)}}$ is wrongly regenerated by $Dec_\theta$ as $\tilde{\boldsymbol{x}}^{(j)}$ instead of $\tilde{\boldsymbol{x}}^{(i)}$. Since, $\boldsymbol{z} \sim Enc_\phi(\boldsymbol{x}^{(i)})$, $Dec_\theta(\boldsymbol{z}) = \tilde{\boldsymbol{x}}^{(j)_{(i)}}$ generates reconstruction loss $||\tilde{\boldsymbol{x}}^{(j)_{(i)}} - \boldsymbol{x}^{(i)}||^2$. From Proposition 6, the loss $||\tilde{\boldsymbol{x}}^{(j)_{(i)}} - \boldsymbol{x}^{(\boldsymbol{i})}||^2$ would be minimum when $\boldsymbol{x}^{(j)}$ is the nearest element to $\boldsymbol{x}^{(i)}$ in $X$.

In (57), given ideal decoder, $||\tilde{\boldsymbol{x}}^{(i)} - \boldsymbol{x}^{(i)}||^2 \neq 0$ only when $\boldsymbol{z}^{(i)} \in \boldsymbol{z}$ and $Dec_\theta(\boldsymbol{z}^{(i)}) = \tilde{\boldsymbol{x}}^{(j)_{(i)}}$. Hence, (57) simplifies to

$$L_{rec} = \sum_{\substack{\boldsymbol{x}^{(i))} \in X \\ \boldsymbol{x}^{(j)} \neq \boldsymbol{x}^{(i)}}} ||\tilde{\boldsymbol{x}}^{(j)_{(i)}} - \boldsymbol{x}^{(i)}||^2$$

Further considering that each $\boldsymbol{x}^{(i)}$ is sampled multiple times, $L_{rec}$ can be expressed as ,

$$L_{rec} = \sum_{\substack{\boldsymbol{x}^{(i))} \in X \\ \boldsymbol{x}^{(\boldsymbol{j})} \neq \boldsymbol{x}^{(i)}}} ||\tilde{\boldsymbol{x}}^{(j)_{(i)}} - \boldsymbol{x}^{(i)}||^2 = \sum_{\substack{\boldsymbol{x}^{(i))} \in X \\ \boldsymbol{x}^{(\boldsymbol{j})} \neq \boldsymbol{x}^{(i)}}} \sum_{l=0}^{k_i} ||\tilde{\boldsymbol{x}}_l^{(j)_{(i)}} - \boldsymbol{x}^{(i)}||^2 \tag{59}$$

where $k_i = |\boldsymbol{z}^{(i)} \in \mathcal{Z}| \; \forall \; \boldsymbol{x}^{(i)} \in X$ and $\tilde{\boldsymbol{x}}_l^{(j)_{(i)}}$ is the $l$-th $\tilde{\boldsymbol{x}}^{(j)_{(i)}}$.

From (59), given ideal $Enc_\phi$ and $Dec_\theta$, $L_{rec}$ depends only on $\tilde{\boldsymbol{x}}_l^{(j)_{(i)}} = Dec_\theta(\boldsymbol{z}^{(i)})$ where $\boldsymbol{z}^{(i)} \in \mathcal{Z}$. From Lem. 13, $\mathcal{X}_k^* = \bigcup_{i=1}^{|X|} \mathcal{X}_{k^{(i)}}^{(i)^*}$ minimizes $L_{rec}$, where $\mathcal{X}_{k^{(i)}}^{(i)^*}$ is the set of $k^{(i)} \leq k_i$ nearest elements to any given $\boldsymbol{x}^{(i)} \in \mathbb{X}$. Hence, the overlap between $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ and the $k^{(i)}$ posterior probabilities, must be posterior probabilities generated by the $k^{(i)}$ nearest neighbours of $\boldsymbol{x}^{(i)}$ in $\mathbb{X}$. This proves part (a) of the Theorem.

As $L_{KL}$ decreases, since the overlap between $q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ and $q_\phi(\boldsymbol{z}^{(k)}|\boldsymbol{x}^{(k)}) \; \forall \, k^{(i)} - nearest \;\; neighbours$ of $\boldsymbol{x}^{(i)}$ increases, given $\boldsymbol{z}^{(i)} \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$ and $\boldsymbol{z}^{(k)} \sim q_\phi(\boldsymbol{z}^{(k)}|\boldsymbol{x}^{(k)})$, $\mathbb{E}_{x^{(i)},z^{(i)},z^{(k)}}[\sum_k ||z^{(i)} - z^{(k)}||^2]$ decreases. Hence, given $L_{KL'} < L_{KL}$, $Dist(L_{KL'}) < Dist(L_{KL})$. This conclusively proves part (b) of the Theorem.

□

**Lemma 13.** *For $\boldsymbol{x}^{(i)} \in \mathbb{X}$ (data space), the number of elements in the dataset $N = |\mathbb{X}|$, an ideal VAE consisting of $Enc_\phi(\boldsymbol{x}^{(i)})$, $Dec_\theta(\boldsymbol{z}^{(i)})$, $\boldsymbol{z}^{(i)} \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$, $\tilde{\boldsymbol{x}}^{(i)} = Dec_\theta(\boldsymbol{z}^{(i)})$ and a set of numbers $k_i < |\mathbb{X}|$ where $i = \{1, 2...N\}$, given, $x_l^{(j)}$-s $\in \mathbb{X}$ s.t. $\boldsymbol{x}_l^{(j)} \neq \boldsymbol{x}^{(i)}$*

*for the optimization problem,*

$$\min_{\boldsymbol{x}_l^{(j)}} \sum_{x^{(i)} \in X} \sum_{l=0}^{k_i} ||\tilde{\boldsymbol{x}}_l^{(j)} - \boldsymbol{x}^{(i)}||^2 \tag{60}$$

*the following hold:*

- *$\mathcal{X}_k^* = \bigcup_{i=1}^{|\mathbb{X}|} \mathcal{X}_{k^{(i)}}^{(i)^*}$ is the solution set to the optimization, where $\mathcal{X}_{k^{(i)}}^{(i)^*}$ is the set of $k^{(i)} \leq k_i$ nearest elements to $\boldsymbol{x}^{(i)}$ in $\mathbb{X}$.*

*Proof.* First, solve the optimization problem $\min_{\boldsymbol{x}_l^{(j)}} \sum_{l=0}^{k_i} ||\tilde{\boldsymbol{x}}_l^{(j)} - \boldsymbol{x}^{(i)}||^2$ for a random $\boldsymbol{x}^{(i)} \in \mathbb{X}$. Given an ideal VAE, $\tilde{\boldsymbol{x}}^{(i)} = \boldsymbol{x}^{(i)}$ and hence, $||\tilde{\boldsymbol{x}}^{(i)} - \boldsymbol{x}^{(i)}||^2 = 0$. We use induction to prove that $\mathcal{X}_{k^{(i)}}^{(i)^*}$ contains the $k^{(i)}$ nearest elements to $\boldsymbol{x}^{(i)}$ in $\mathbb{X}$. From Proposition 6, the element closest to $\boldsymbol{x}^{(i)}$ in $\mathbb{X}$ (say $\boldsymbol{x}^{(\text{nearest}\mathbb{X})}$) has the lowest value for $||\tilde{\boldsymbol{x}}_l^{(j)} - \boldsymbol{x}^{(i)}||^2$. We prove the rest by induction.

**Base Case**: Removing $\boldsymbol{x}^{(\text{nearest}\mathbb{X})}$ from $\mathbb{X}$, we have $\mathbb{X}_{-1} = \mathbb{X} \setminus \boldsymbol{x}^{(\text{nearest}\mathbb{X})}$. From Proposition 6, the element closest to $\boldsymbol{x}^{(i)}$ in $\mathbb{X}_{-1}$ (say $\boldsymbol{x}^{(\text{nearest}\mathbb{X}_{-1})}$) has the lowest value for $||\tilde{\boldsymbol{x}}_l^{(j)} - \boldsymbol{x}^{(i)}||^2$. This is also the second nearest element to $\boldsymbol{x}^{(i)}$ in $\mathbb{X}$.

**Induction Hypothesis**: Removing $d$ nearest elements to $\boldsymbol{x}^{(i)}$ from $\mathbb{X}$, generates $\mathbb{X}_{-d}$, where the $(d+1)$-th nearest element to $\boldsymbol{x}^{(i)}$, $\boldsymbol{x}^{(\text{nearest}\mathbb{X}_{-d})}$ in $\mathbb{X}$, generates the lowest value for $||\tilde{\boldsymbol{x}}_l^{(j)} - \boldsymbol{x}^{(i)}||^2$.

**Induction Step**: We remove the $(d+1)$-th nearest element from $\mathbb{X}_{-d}$, $\boldsymbol{x}^{(\text{nearest}\mathbb{X}_{-d})}$ (from Induction Hypothesis) to generate $\mathbb{X}_{-(d+1)}$. From Proposition 6, the element closest to $\boldsymbol{x}^{(i)}$ in $\mathbb{X}_{-(d+1)}$ (say $\boldsymbol{x}^{(\text{nearest}\mathbb{X}_{-(d+1)})}$) has the lowest value for $||\tilde{\boldsymbol{x}}_l^{(j)} - \boldsymbol{x}^{(i)}||^2$. This is also the $(d+1)$-th nearest element to $\boldsymbol{x}^{(i)}$ in $\mathbb{X}$.

In the case when there are no repeated elements, i.e., $\boldsymbol{x}^{(\text{nearestX})} \neq \boldsymbol{x}^{(\text{nearestX}_{-1})} \ldots \neq \boldsymbol{x}^{(\text{nearestX}_{-d})} \neq \boldsymbol{x}^{(\text{nearestX}_{-(d+1)})}$, $d$ elements form the solution set $\mathcal{X}_d^{(i)^*}$. However, in the case of repetition of a single element, i.e., $\boldsymbol{x}^{(\text{nearestX}_{-i})} = \boldsymbol{x}^{(\text{nearestX}_{-j})}$ (say), there exists a $\mathcal{X}_{d'}^{(i)^*}$ such that $d' < d$. Hence, given any $\boldsymbol{x}^{(i)}$, $\mathcal{X}_{k^{(i)}}^{(i)^*}$ is the solution set for the optimization problem $\min_{\boldsymbol{x}_l^{(j)}} \sum_{l=0}^{k_i} ||\tilde{\boldsymbol{x}}_l^{(j)} - \boldsymbol{x}^{(i)}||^2$ where $k^{(i)} \leq k_i$. Hence, $\mathcal{X}_k^* = \bigcup_{i=1}^{|\mathbb{X}|} \mathcal{X}_{k^{(i)}}^{(i)^*}$ is the solution set to the optimization problem in (60).

□

**Proposition 6.** *Given $\boldsymbol{x}^{(i)} \in \mathbb{X}$ (data space), an ideal encoder $Enc_\phi(\boldsymbol{x}^{(i)})$, an ideal decoder $Dec_\theta(\boldsymbol{z}^{(i)})$, $\boldsymbol{z}^{(i)} \sim q_\phi(\boldsymbol{z}^{(i)}|\boldsymbol{x}^{(i)})$, $\tilde{\boldsymbol{x}}^{(i)} = Dec_\theta(\boldsymbol{z}^{(i)})$, and there exists an $x_k$ s.t.*

$$||\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(i)}||^2 < ||\boldsymbol{x}^{(j)} - \boldsymbol{x}^{(i)}||^2 \; \forall j \neq k \neq i \tag{61}$$

*then the following holds:*

- $\boldsymbol{x}^{(k)}$ *is the element in* $\mathbb{X}$ *nearest to* $\boldsymbol{x}^{(i)}$
- $||\tilde{\boldsymbol{x}}^{(k)} - \boldsymbol{x}^{(i)}||^2 < ||\tilde{\boldsymbol{x}}^{(j)} - \boldsymbol{x}^{(i)}||^2$

*Proof.* Part (a) of the proposition follows directly from (61). Given an ideal encoder and decoder, $||\tilde{\boldsymbol{x}}^{(k)} - \boldsymbol{x}^{(k)}||^2 = 0$. Hence, $\tilde{\boldsymbol{x}}^{(k)} = \boldsymbol{x}^{(k)}$. Similarly, $\tilde{\boldsymbol{x}}^{(j)} = \boldsymbol{x}^{(j)}$. Hence, in (61), we can replace $\boldsymbol{x}^{(k)}$ by $\tilde{\boldsymbol{x}}^{(k)}$ and $\boldsymbol{x}^{(j)}$ by $\tilde{\boldsymbol{x}}^{(j)}$ giving us part (b) of the proposition. □

## A.8. Appendix Related to the experiments

### A.8.1. Network Architectures used in Experimental Setup

Table 3 summarizes the network architectures and the implementation details of the models that we trained for each of the datasets.

*Table 3.* The architectures of the VAE-based models used for the different datasets.

| Dataset | Component | Architecture |
|---|---|---|
| dSprites | Encoder | 1200 (ReLU) - 1200 (ReLU) - Latent Space (ReLU) |
| | Decoder | 1200 (ReLU) - 1200 (ReLU) - 4096 (ReLU) |
| | $\beta$ | 6 |
| | Optimizer | Adam (lr = $10^{-3}$) |
| 3DFaces | Encoder | Conv [32, 4, 2, 1] (BN) (ReLU), [32, 4, 2, 1] (BN) (ReLU), [64, 4, 2, 1] (BN) (ReLU), [64, 4, 2, 1] (BN) (ReLU), [512, 4, 1, 0] (BN) (ReLU), [Latent Space, 1, 1] (BN) (ReLU) |
| | Decoder | ConvTrans [512, 1, 1, 0] (BN) (ReLU), [64, 4, 2, 1] (BN) (ReLU), [64, 4, 2, 1] (BN) (ReLU), [32, 4, 2, 1] (BN) (ReLU), [1, 4, 2, 1] (BN) (ReLU) |
| | $\beta$ | 6 |
| | Optimizer | Adam (lr = $10^{-3}$, betas = (0.9, 0.999)) |
| 3D shape | Encoder | Conv [32, 4, 2, 1] (ReLU), [32, 4, 2, 1] (ReLU), [64, 4, 2, 1] (ReLU), [64, 4, 2, 1] (BN) (ReLU), [256, 4, 1, 0] (BN) (ReLU), [Latent Space, 1, 1] (ReLU) |
| | Decoder | Conv[64, 1, 1, 0] (Relu), ConvTrans [64, 4, 1, 0] (ReLU), [64, 4, 2, 1] (ReLU), [64, 4, 2, 1] (ReLU), [32, 4, 2, 1] (ReLU), [32, 4, 2, 1] (ReLU), [3, 4, 2, 1] |
| | $\beta$ | 6 |
| | Optimizer | Adam (lr = $10^{-3}$, betas = (0.9, 0.999)) |
| MPI 3D complex | Encoder | Conv [32, 4, 2, 1] (ReLU), [32, 4, 2, 1] (ReLU), [64, 4, 2, 1] (ReLU), [64, 4, 2, 1] (BN) (ReLU), [256, 4, 1, 0] (BN) (ReLU), [Latent Space, 1, 1] (ReLU) |
| | Decoder | Conv[64, 1, 1, 0] (Relu), ConvTrans [64, 4, 1, 0] (ReLU), [64, 4, 2, 1] (ReLU), [64, 4, 2, 1] (ReLU), [32, 4, 2, 1] (ReLU), [32, 4, 2, 1] (ReLU), [3, 4, 2, 1] |
| | $\beta$ | 6 |
| | Optimizer | Adam (lr = $10^{-3}$, betas = (0.9, 0.999)) |

### A.8.2. Comparing Jacobian approximation of stochastic loss with loss calculated from approximated local decoder

In this experiment, we compare two approximations: the linearization approximation by (Rolinek et al., 2019), where, $\tilde{L}_{\mathrm{J}}^{\mathrm{stoch}^{(i)}} = J\boldsymbol{\epsilon}^{(i)}$ and our modeling, where, $\tilde{L}_{\mathrm{dec}}^{\mathrm{stoch}^{(i)}} = g_D(\boldsymbol{M}_D\boldsymbol{\epsilon}^{(i)})$. The comparison focuses on determining the approximate loss closer to the actual stochastic loss, denoted as $\hat{L}_{\mathrm{rec}}^{\mathrm{stoch}^{(i)}}$. A validation set $\boldsymbol{x}^{(i)} \in \mathcal{X}_{\mathrm{val}}$ is defined. For each $\boldsymbol{x}^{(i)}$, $g_D$ and $\boldsymbol{M}_D$ are estimated as neural networks, considering that these are local approximations unique to each $\boldsymbol{x}^{(i)}$. (11) is employed to train these networks, as detailed in Sect. 4.4.

Subsequently, we compute $\hat{L}_{\text{rec}}^{\text{stoch}} = \sum_{\boldsymbol{x}^{(i)} \in \mathcal{X}_{\text{val}}} \hat{L}_{\text{rec}}^{\text{stoch}^{(i)}}$. This is followed by the calculation of $\tilde{L}_{\text{J}}^{\text{stoch}} = \sum_{\boldsymbol{x}^{(i)} \in \mathcal{X}_{\text{val}}} \tilde{L}_{\text{J}}^{\text{stoch}^{(i)}}$ and $\tilde{L}_{\text{dec}}^{\text{stoch}} = \sum_{\boldsymbol{x}^{(i)} \in \mathcal{X}_{\text{val}}} \tilde{L}_{\text{dec}}^{\text{stoch}^{(i)}}$. The final step involves comparing the squared errors $\delta_{\text{dec}} = ||\tilde{L}_{\text{dec}}^{\text{stoch}} - \hat{L}_{\text{rec}}^{\text{stoch}}||^2$, and $\delta_{\text{J}} = ||\tilde{L}_{\text{J}}^{\text{stoch}} - \hat{L}_{\text{rec}}^{\text{stoch}}||^2$, to ascertain the more accurate approximation.

In Table 2, we summarize the two differences which demonstrates that $\tilde{L}_{\text{dec}}^{\text{stoch}}$ is a much better approximation as compared to $\tilde{L}_{\text{J}}^{\text{stoch}}$.

### A.8.3. Calculating the values $\hat{\boldsymbol{U}}_{\boldsymbol{D}}, \hat{\boldsymbol{\Sigma}}_{\boldsymbol{D}}$ for determining the OD-Score for different VAE Architectures

**Proposition 7.** *Given a matrix $\boldsymbol{M}_D$, the nearest orthogonal matrix $\hat{\boldsymbol{M}}_D$ can be estimated as follows:*

$$\hat{\boldsymbol{M}}_D = \boldsymbol{M}_D(\boldsymbol{M}_D^\intercal \boldsymbol{M}_D)^{\frac{1}{2}}$$

*Proof.* We minimize $||\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D||^2$ subject to $\hat{\boldsymbol{M}}_D^\intercal \hat{\boldsymbol{M}}_D = \mathbf{I}$. Introducing a symmetric Lagrangian multiplier matrix $\boldsymbol{\Lambda}$, we look for the stationary values of

$$e(\hat{\boldsymbol{M}}_D, \boldsymbol{\Lambda}) = Tr\{(\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D)^\intercal(\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D)\} + Tr\{\boldsymbol{\Lambda}(\hat{\boldsymbol{M}}_D^\intercal \hat{\boldsymbol{M}}_D - \mathbf{I})\}$$

Differentiating $e(\hat{\boldsymbol{M}}_D, \boldsymbol{\Lambda})$ w.r.t $\hat{\boldsymbol{M}}_D$ and setting it to 0, we have,

$$\frac{\partial e}{\partial \hat{\boldsymbol{M}}_D} = \frac{\partial(Tr\{(\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D)^\intercal(\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D)\})}{\partial \hat{\boldsymbol{M}}_D} + \frac{\partial(Tr\{\boldsymbol{\Lambda}(\hat{\boldsymbol{M}}_D^\intercal \hat{\boldsymbol{M}}_D - \mathbf{I})\})}{\partial \hat{\boldsymbol{M}}_D} = 0$$

We know that $\frac{\partial(Tr\{\boldsymbol{X}^\intercal \boldsymbol{X}\})}{\partial \boldsymbol{X}} = 2\boldsymbol{X}$.
Hence,

$$\frac{\partial(Tr\{(\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D)^\intercal(\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D)\})}{\partial \hat{\boldsymbol{M}}_D} = -2(\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D)$$

Further, $\frac{\partial(Tr\{\boldsymbol{\Lambda}\boldsymbol{X}^\intercal \boldsymbol{X}\})}{\partial \boldsymbol{X}} = \boldsymbol{X}(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^\intercal)$.
Hence,

$$\frac{\partial(Tr\{\boldsymbol{\Lambda}(\hat{\boldsymbol{M}}_D^\intercal \hat{\boldsymbol{M}}_D - \mathbf{I})\})}{\partial \hat{\boldsymbol{M}}_D} = \hat{\boldsymbol{M}}_D(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^\intercal)$$

Replacing the terms, we get,

$$-2(\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D) + \hat{\boldsymbol{M}}_D(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^\intercal) = 0$$

Since, $\boldsymbol{\Lambda}^\intercal = \boldsymbol{\Lambda}$,

$$-(\boldsymbol{M}_D - \hat{\boldsymbol{M}}_D) + \hat{\boldsymbol{M}}_D\boldsymbol{\Lambda} = 0$$

$$\boldsymbol{M}_D = \hat{\boldsymbol{M}}_D(\mathbf{I} + \boldsymbol{\Lambda})$$

Calculating $\boldsymbol{M}_D^\intercal \boldsymbol{M}_D$, we have,

$$\boldsymbol{M}_D^\intercal \boldsymbol{M}_D = (\mathbf{I} + \boldsymbol{\Lambda})\hat{\boldsymbol{M}}_D^\intercal \hat{\boldsymbol{M}}_D(\mathbf{I} + \boldsymbol{\Lambda})$$

Hence,

$$(\mathbf{I} + \boldsymbol{\Lambda}) = (\boldsymbol{M}_D^\intercal \boldsymbol{M}_D)^{\frac{1}{2}}$$

Hence, solving for $\hat{\boldsymbol{M}}_D$, we have,

$$\hat{\boldsymbol{M}}_D = \boldsymbol{M}_D(\boldsymbol{M}_D^\intercal \boldsymbol{M}_D)^{\frac{1}{2}}$$

□

### A.8.4. MIG and MIG-sup Scores for different Architectures

The MIG score for the different VAE-based architectures for the datasets have been summarized in Table 4. The MIG-sup score for the different VAE-based architectures for the datasets have been summarized in Table 5

*Table 4.* The MIG scores of the different VAE-based models for the Datasets

| | $\beta$-TCVAE | $\beta$-VAE | VAE |
|---|---|---|---|
| dSprites | 0.52±0.1 | 0.25±0.1 | 0.16±0.05 |
| 3DFaces | 0.58±0.02 | 0.44±0.05 | 0.19±0.02 |
| 3DShapes | 0.40±0.1 | 0.46±0.14 | 0.20±0.08 |
| MPI3DComplex | 0.32±0.1 | 0.24±0.08 | 0.14±0.06 |

*Table 5.* The MIG-Sup scores of the different VAE-based models for the Datasets

| | $\beta$-TCVAE | $\beta$-VAE | VAE |
|---|---|---|---|
| dSprites | 0.52±0.08 | 0.17±0.1 | 0.14±0.02 |
| 3DFaces | 0.60±0.08 | 0.46±0.04 | 0.20±0.03 |
| 3DShapes | 0.50±0.1 | 0.60±0.05 | 0.14±0.04 |
| MPI3DComplex | 0.60±0.15 | 0.46±0.15 | 0.10±0.06 |

### A.8.5. OD-SCORE FOR DIFFERENT VAE ARCHITECTURES

Table 6 summarizes the OD-Score($\boldsymbol{M}_D$) scores for the different VAE based architectures for the datasets.

### A.8.6. DEMONSTRATING THE CONNECTION BETWEEN GENERATIVE FEATURES AND PRINCIPAL AXES

In this section, we experimentally establish a connection between the generative factors and the principal components of the data. Further, we show that the principal components with the highest variance are associated with the generative factors most significant for the reconstruction.

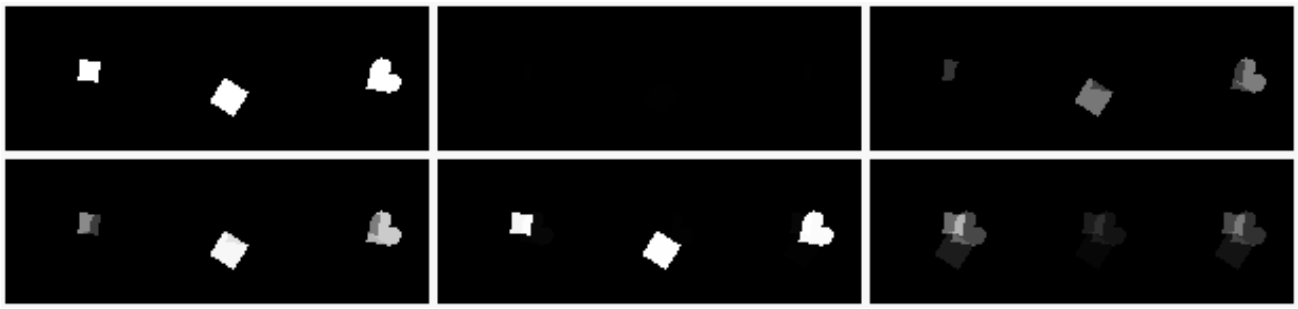


*Figure 5.* The first image is the image subset input to the PCA. The second image is the reconstruction with the first 1500 principal axes, the third with first 2000 principal axes, the fourth with first 2500 axes, the $5th$ with 3000 and the $6th$ with 2500th to 3000th axes.

Fig. 5 shows the original image, which is the first image, and the rest reconstructed images from the different sets of principal axes (PA) the details of which are provided in Fig. 5. Given that the second picture (PA = first 1500) is blank, the third (PA = first 2000), fourth (PA = first 2500), and fifth (PA = first 3000) images capture both position and parts of the shape, and the sixth image (PA = 2500th to 3000th) does not capture position, indicates that the first 1500 axes summarize just the position and not the shape. Given that the sixth image lacks both position and shape information while the fifth lacks intricate details, we conclude that the principal axes from 1500 to 2500 capture position, while 2500 to 3000 capture intricate position details.

Table 7 summarizes the reconstruction error from different sets of principal axes. While the error decreases as more axes are added, it increases rapidly when the initial high variance axes, which convey greater information are removed.

*Table 6.* The OD-Score($M_D$) scores of the different VAE based models for the Datasets

| | $\beta$-TCVAE | $\beta$-VAE | VAE |
|---|---|---|---|
| dSprites | 0.9093±0.02 | 0.9400±0.02 | 0.9732±0.02 |
| 3DFaces | 0.9002± 0.02 | 0.9494±0.01 | 0.9857±0.01 |
| 3DShapes | 0.9543±0.02 | 0.9254±0.01 | 0.0.9864±0.02 |
| MPI3DComplex | 0.9086±0.014 | 0.9254±0.016 | 0.9466±0.01 |

*Table 7.* Comparing the reconstruction error for the images generated from the different sets of principal axes.

| Axes used | first 1500 | first 2000 | first 2500 | first 3000 | 2000 to 3000 | 2500 to 3000 |
|---|---|---|---|---|---|---|
| Reconstruction Loss | 21.23 | 12.43 | 6.41 | 0.54 | 12.31 | 17.64 |