Exploring VAEs performance over Mixture of Gaussians Distributions

J. Roberto Tello Ayala, Léo Benac
Harvard University
Cambridge, MA
{jtelloayala, lbenac}@g.harvard.edu

May 3, 2023

Abstract

Variational Auto-encoders (VAEs) are deep generative latent variable models that are widely used for a number of downstream tasks. In this paper, we investigate the performance of a VAE that utilizes a more expressive prior through Mixture of Gaussians (MoG) compared to a VAE that employs a more expressive posterior through MoG, and compare both to the traditional Mean Field Gaussian (MFG) VAE. We evaluate the performance on a dataset that we generated such that the true posterior distribution is multimodal. Our experimental results show that the VAE with a more expressive prior through MoG outperforms the traditional MFG VAE and MoG VAE in terms of data generation quality, inference and generative model. Furthermore, we observe that the VAE with a more expressive posterior through MoG performs similarly to the traditional MFG VAE, suggesting that a more expressive prior is more important than a more expressive posterior for VAE performance. Our findings suggest that using a more expressive prior through MoG can enhance the generative and inference capabilities of VAE and improve its ability to model complex distributions.

1 Introduction

Variational Autoencoders (VAEs) (Kingma and Welling, 2014) are a popular class of generative models that have been used in a wide range of downstream task, such as, anomaly detection (An and Cho, 2015), representation learning (Rezende and Mohamed, 2015), generating realistic looking synthetic data (Pu et al., 2016), learning compressed representations (Miao and Blunsom, 2016), adversarial defense using de-noising (Luo and Pfister, 2018), and, when expert knowledge is available, generating counter-factual data using weak or semi-supervision (Paige et al., 2017). VAEs consist of two parts, an encoder network that maps the input data into a latent space, and a decoder network that maps the latent representation back to the input space. The latent space is typically of lower dimensionality than the input space, allowing for a more compact and structured representation of the input data . VAEs are trained to maximize a lower bound on the log-likelihood of the data (ELBO). This lower bound is derived using the variational inference framework and involves an approximation of the intractable posterior distribution over the latent variables. However, as VAEs are increasingly being used in application where the data is numeric, e.g. in medical or financial domains (Pfohl et al., 2019), these intuitive qualitative checks no longer apply. For example, in many medical applications, the original data features themselves (e.g. biometric reading) are difficult to analyze by human experts in raw form. In these cases, where the application touches human lives and potential model error/pathologies are particularly consequential, we need to have a clear theoretical understanding of the failure modes of our models as well as the potential negative consequences on down-stream tasks. One important aspect of VAEs is the choice of the prior distribution over the latent variables. In the standard VAE, a simple

prior distribution such as a standard normal distribution is used. However, this prior distribution may not be appropriate for all data types, and modifying the prior can improve the generative performance of VAEs. For example, in the case of binary data, the Bernoulli distribution has been shown to be a more appropriate prior distribution than the standard normal distribution (see Kingma and Welling, 2014). Similarly, for data with a known structure or prior knowledge, using a more informed prior distribution can improve the quality of the generated samples. Another aspect of VAEs is the modification of the posterior distribution. The posterior distribution is the distribution over the latent variables given the observed data. In the standard VAE, the posterior distribution is approximated using a mean-field variational approximation. However, this approximation may not be able to capture the true posterior distribution, leading to suboptimal generative performance. Modifying the posterior distribution can improve the quality of the latent representation and the generated samples. Other ways of approximating the posterior are Wasserstein Autoencoder (WAE), Mixture of Gaussians VAE (MoG-VAE), Normalizing Flows VAE (NF-VAE), Importance Weighted Autoencoder (IWAE) (Arjovsky et al., 2017, Dilokthanakul et al., 2016, Rezende and Mohamed, 2015, Burda et al., 2015). Recent work (Yacoby et al., 2020) highlights a number of the pathologies when training MFG-VAEs, in particular, the objective may compromise learning a good generative model in order to learn a good inference model – in other words, the inference model over-regularizes the generative model.

2 Contribution

Extending Yacoby et al., 2020's work, we decided to explore if by having a more expressive prior versus a more expressive posterior distribution if we could improve the performance of the generative and inference model as they just used MFG-VAE. We also looked at avoiding certain pathologies as the non-identifiability of the decoder function f of the generative model since learning this function incorrectly can have high consequences in high stakes settings. In this work we trained our model on a low dimensional dataset such that the posterior distribution is multi-modal and easy to visualize. We then compared three VAE models: (MFG-VAE), (MoG-VAE) and (MFG-VAE with MoG prior) we showed how each of these three models performs on that dataset and showed for different model how much the inference model regularize the generative model.

3 Methodology

We present three different modelling paradigms in order to learn a simple data distribution. The goal will be to analyze how the expressiveness of the prior versus the expressiveness of the posterior of the latent variable affect the performance of the learnt inference and generative model.

3.1 True Data Generating Process

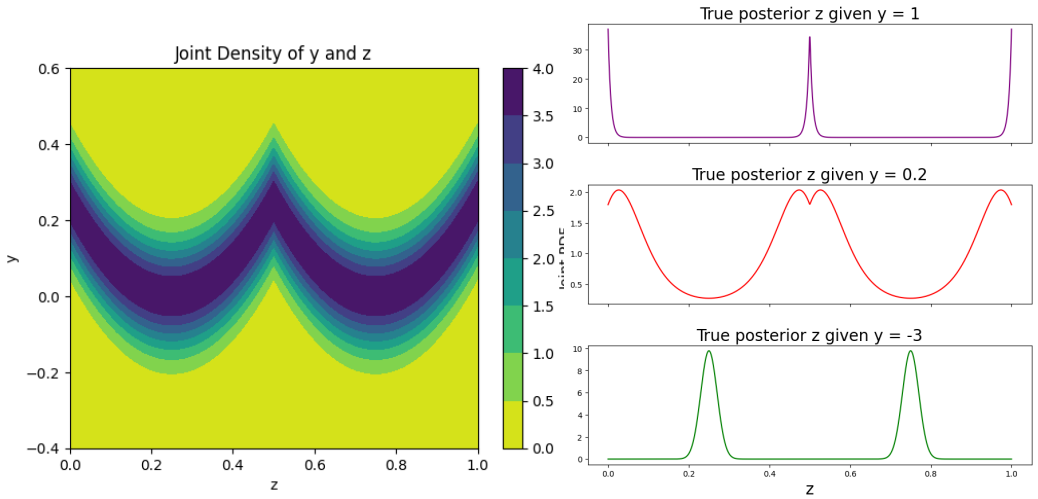
In order to visualize and compare our results, we assume that the true Data Generating Process is given. Let $z \sim \mathcal{U}(0,1)$ and $y|z \sim \mathcal{N}(f(z), 0.01)$ where we supposed y is observed and z is latent and f is given

$$f(z) = \mathbb{I}(z < 0.5) \cdot (2z - 0.5)^2 + \mathbb{I}(z \geq 0.5) \cdot (2z - 1.5)^2.$$

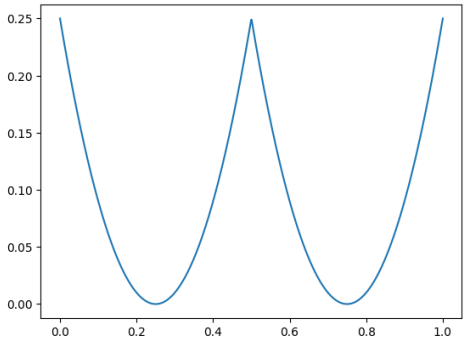
We see in Figure 1b that this data generating process yields a multimodal posterior distributions $p(z|y)$ for various values of y, $\mathbb{I}$ represents the indicator function.

3.2 MFG-VAE

We assume a standard Normal distribution as prior on z, f_ϕ is a neural network with weights ϕ and where $\mu_\theta(y)$ and $\Sigma_\theta(y)$ are output of another neural network with weights θ and where $\Sigma_\theta(y)$ is



(a) True joint density $p(y, z)$ (b) True posterior $p(z|y)$ for different y values



(c) True function $f(z)$

Figure 1: Ground truth data

diagonal, and Σ_{d_y} is a known covariance diagonal matrix. We have assumed that $z \sim \mathcal{N}(0_{d_z}, I_{d_z})$, $y|z \sim \mathcal{N}(f_\phi(z), \Sigma_{d_y})$, and $z|y \sim \mathcal{N}(\mu_\theta(y), \Sigma_\theta(y))$

3.2.1 Training

We learn the parameters Φ and θ by solving the following using SGD and the reparametrization trick on z:

$$\begin{aligned}\max_{\phi,\theta} ELBO(Y, Z; \phi, \theta) &= \max_{\phi,\theta} \mathbb{E}_{z\sim q_\theta(z|y)} \left[\log\left(\frac{p_\phi(y|z)p_\phi(z)}{q_\theta(z|y)}\right)\right] \\ &= \max_{\phi,\theta} \mathbb{E}_{z\sim q_\theta(z|y)} \left[\log\left(\frac{\mathcal{N}(f_\phi(z), \Sigma_{d_y})\mathcal{N}(0_{d_z}, I_{d_z})}{\mathcal{N}(\mu_\theta(y), \Sigma_\theta(y))}\right)\right]\end{aligned}$$

3.3 MFG-VAE with MoG prior

We assume a Gaussian Mixture Model (MoG) prior on z with equal proportion probabilities where $\Sigma_{d_z}^{(i)}$ are diagonal covariance matrix as well. Let $z \sim \frac{1}{M}\Sigma_{i=1}^{M}\mathcal{N}(\mu_{d_z}^{(i)}, \Sigma_{d_z}^{(i)})$, $y|z \sim \mathcal{N}(f_\phi(z), \Sigma_{d_y})$, and $z|y \sim \mathcal{N}(\mu_\theta(y), \Sigma_\theta(y))$ with learnable parameters: $\{\mu_{d_z}^{(1)}, \Sigma_{d_z}^{(1)}, \cdots \mu_{d_z}^{(M)}, \Sigma_{d_z}^{(M)}, \phi\} = \Phi$ and θ.

3.3.1 Training

We find the parameters Φ and θ by solving the following:

$$\begin{aligned}\max_{\Phi,\theta} ELBO(Y,Z;\Phi,\theta) &= \max_{\Phi,\theta} \mathbb{E}_{z\sim q_\theta(z|y)}\left[\log\left(\frac{p_\Phi(y|z)p_\Phi(z)}{q_\theta(z|y)}\right)\right] \\ &= \max_{\Phi,\theta} \mathbb{E}_{z\sim q_\theta(z|y)}\left[\log\left(\frac{\mathcal{N}(f_\phi(z),\Sigma_{d_y})\frac{1}{M}\Sigma_{i=1}^{M}\mathcal{N}(\mu_{d_z}^{(i)},\Sigma_{d_z}^{(i)})}{\mathcal{N}(\mu_\theta(y),\Sigma_\theta(y))}\right)\right]\end{aligned}$$

3.4 MoG-VAE

We now have a standard normal prior on z, and where $\mu_\theta^{(i)}(y)$ *and* $\Sigma_\theta^{(i)}(y)$ for each $i = 1 \dots M$ are output of another neural network with weights θ, we model the distribution $z|y$ as part of a MoG with equal proportion probabilities: $z \sim \mathcal{N}(0_{d_z}, I_{d_z})$, $y|z \sim \mathcal{N}(f_\phi(z), \Sigma_{d_y})$, and $z|y \sim \frac{1}{M}\Sigma_{i=1}^{M}\mathcal{N}(\mu_\theta^{(i)}(y), \Sigma_\theta^{(i)}(y))$.

3.4.1 Training

We find the parameters ϕ and θ by solving the following:

$$\begin{aligned}\max_{\Phi,\theta} ELBO(Y,Z;\Phi,\theta) &= \max_{\Phi,\theta} \mathbb{E}_{z\sim q_\theta(z|y)}\left[\log\left(\frac{p_\Phi(y|z)p_\Phi(z)}{q_\theta(z|y)}\right)\right] \\ &= \max_{\Phi,\theta} \mathbb{E}_{z\sim q_\theta(z|y)}\left[\log\left(\frac{\mathcal{N}(f_\phi(z),\Sigma_{d_y})\mathcal{N}(0_{d_z},I_{d_z})}{\frac{1}{M}\Sigma_{i=1}^{M}\mathcal{N}(\mu_\theta^{(i)}(y),\Sigma_\theta^{(i)}(y))}\right)\right]\end{aligned}$$

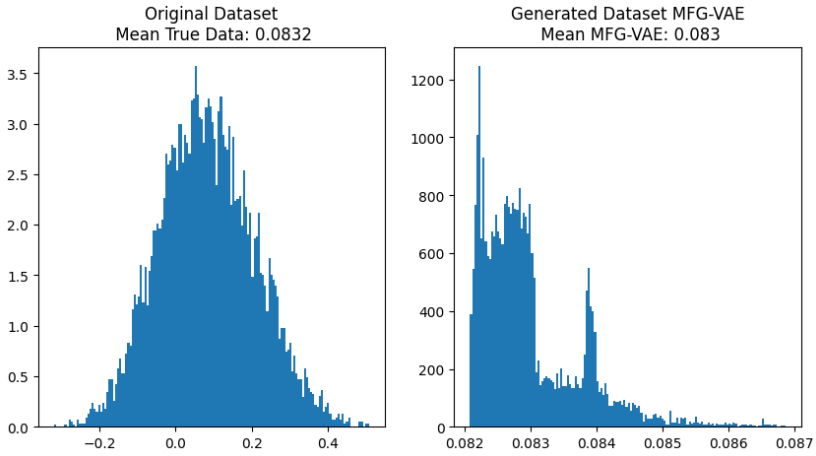
In order to compare both methods we will play around with the expressiveness of the prior and posterior and look how well they represent $p(y)$ and f through visualization. This would be possible when working with one dimensional datasets. For higher dimensional datasets we can calculate the true likelihood as well as the ELBO to asses the performance of each method, since we will know the true data generating process. Based on VAEs literature this seems like the most reasonable way to assess the performance of both methods.

4 Results

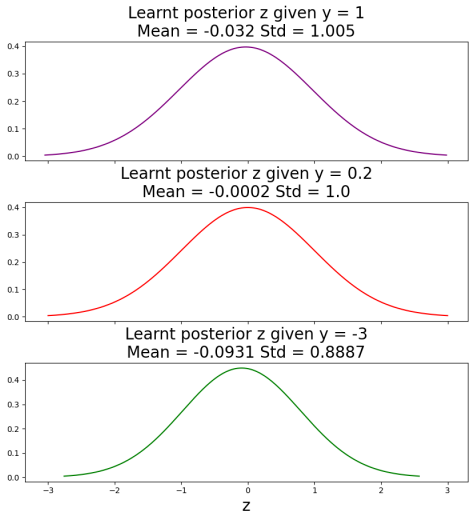
We trained the three models on the dataset explained in Section 3.1. For each of the models we visualized the learned decoder $f(z)$, the empirical learnt y's distribution and the learnt posterior for values of $y = 1, 0.2, -3$. For both of the models that utilized the Gaussian Mixtrure either as prior or posterior, we set the number of Gaussians to ten.

4.1 MFG-VAE

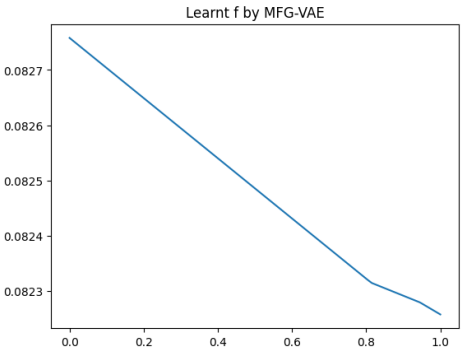
The vanilla VAE implementation underperforms compared to the other the other two models. Although the mean is captured correctly for the generated samples created by this model the distribution is multimodal and does not behave as a Normal distribution and does not learn the variance. The learnt decoder is almost linear and does not represent the true f. And given that under various values of y the posterior distribution barely changes which suggests that the inference model is not learnt correctly.



(a) True joint density p(y,z)



(b) True posterior $p(z|y)$ for different y values

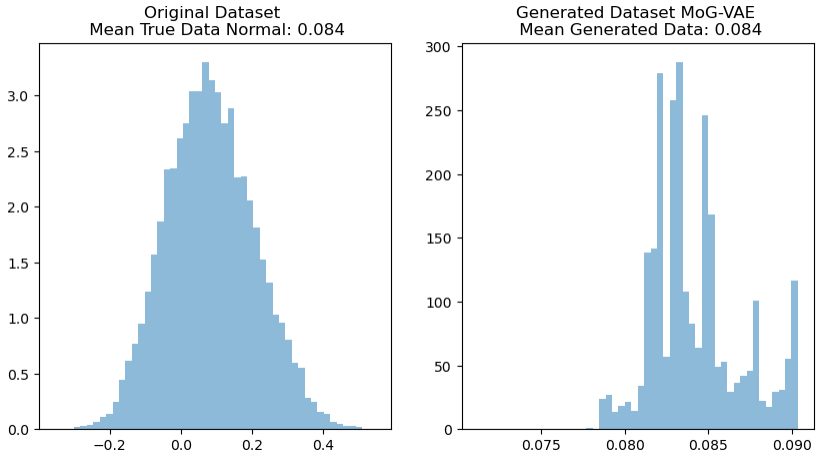


(c) True function $f(z)$

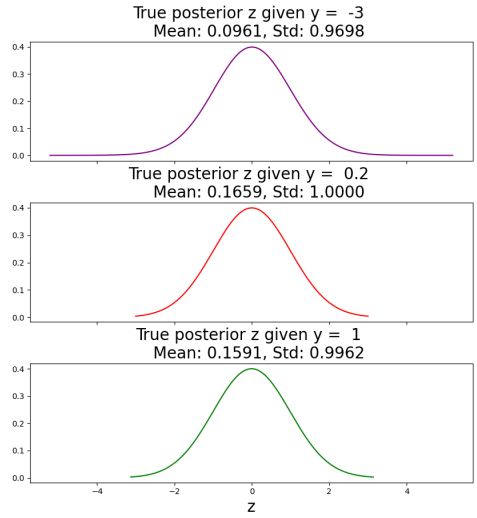
Figure 2: MFG-VAE

4.2 MoG-VAE

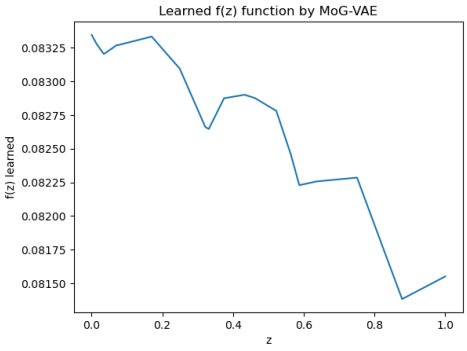
The VAE with a Mixture of Gaussians as a posterior implementation allows the model to be more expressive and as such performs better than the vanilla implementation. The generated samples captured the mean correctly as the vanilla implementation and the distribution is more normally behaved and does also not learn the variance. The learnt decoder mimics the best true f out of the three models, capturing the curvature. The same behaviour as the vanilla VAE we have that under various values of y the posterior distribution barely changes which suggests that the inference model is not learnt correctly.



(a) True joint density $p(y, z)$



(b) True posterior $p(z|y)$ for different y values

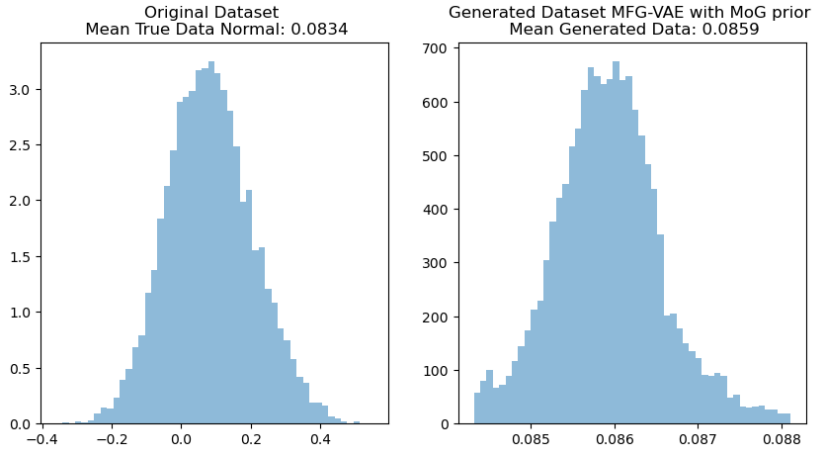


(c) Learned $f(z)$

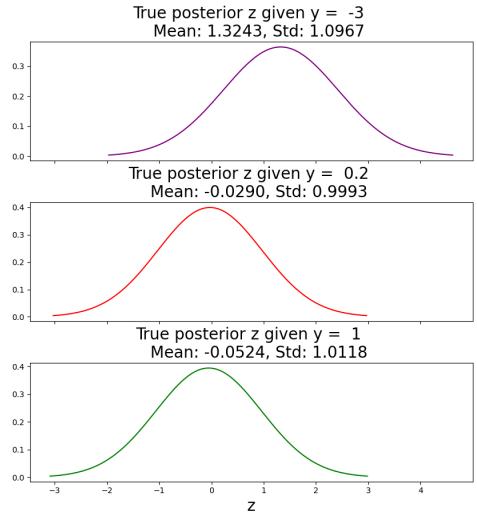
Figure 3: MoG-VAE

4.3 MFG-VAE with MoG prior

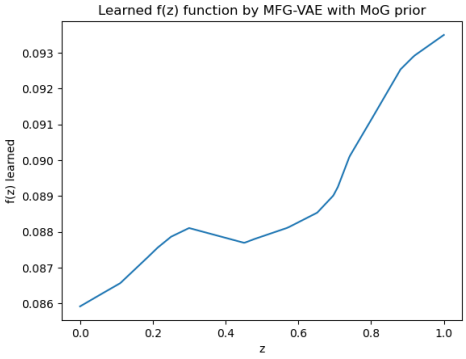
The VAE with a Mixture of Gaussians as a prior has the best inference properties. The prior being more expressive instead of the posterior seems to be better for the generation. The generated samples captured the mean correctly and is the one that looks actually normally distributed however it also does not learn the variance. The learned decoder does not capture the model learned f as well as the previous model but it is better than the vanilla implementation. Contrary to the other two models, under various values of y the learned posterior distribution does change with different values of y which suggests that the inference model better learned.



(a) True joint density $p(y, z)$



(b) True posterior $p(z|y)$ for different y values



(c) Learned $f(z)$

Figure 4: MFG-VAE with MoG as prior

One problem mentioned in Yacoby et al., 2020's work was the non-identifiability of the function f in the generative model. As we can see here this problem does not go away when using more expressive prior and posterior distribution which is interesting and this is mainly due to how low dimensional the data is. We see that for each different function f we are able to truly capture the means of the true distribution Y.

5 Conclusion

In this study, we presented an empirical comparison of three different Variational Autoencoder (VAE) models, including a VAE that utilizes a more expressive prior through Gaussian Mixture Model (MoG), a VAE that employs a more expressive posterior through MoG, and the traditional Mean Field Gaussian (MFG) VAE. Our aim was to investigate the impact of more expressive prior and posterior distributions on VAE performance, as well as to evaluate the effectiveness of these models for generative modeling and unsupervised learning tasks. Our experimental results showed that the VAE with a more expressive prior through MoG outperformed the traditional MFG VAE in terms of data generation, inference model

learnt as well as the a more informative f function. On the other hand, the VAE with a more expressive posterior through MoG performed only slightly better (in terms of the inference and generative model) than the MFG-VAE but could run into problem when training as the loss function could diverge for certain runs. These results suggest that for the particular dataset used in this study, a more expressive prior is more important than a more expressive posterior for VAE performance. Nonetheless, our findings demonstrate the importance of considering both the prior and posterior distributions in VAE modeling, and highlight the potential benefits of incorporating more expressive distributions.

6 Future Works

In addition to the promising results obtained in this study, there are several directions for future research that could extend and improve the models we presented. One possible avenue of investigation would be to explore the use of more sophisticated expressive distributions, such as hierarchical or non-parametric models, for both the prior and posterior in VAE. These models could capture more complex structures in the data distribution, and potentially further improve the quality of generated samples. Another potential direction for future research is to experience similar experiments but on more complicated and higher dimensional dataset and see how the results differs. Finally, it would also be interesting to explore the application of these models to real-world datasets in various domains, such as healthcare that could potentially lead to new insights and discoveries in these fields.

References

Kingma, D. P., & Welling, M. (2014). Auto-Encoding Variational Bayes. *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

An, J., & Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. *Special lecture on IE*, *2*(1), 1–18.

Burda, Y., Grosse, R., & Salakhutdinov, R. (2015). Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*.

Rezende, D., & Mohamed, S. (2015). Variational inference with normalizing flows. *International conference on machine learning*, 1530–1538.

Dilokthanakul, N., Mediano, P. A., Garnelo, M., Lee, M. C., Salimbeni, H., Arulkumaran, K., & Shanahan, M. (2016). Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*.

Miao, Y., & Blunsom, P. (2016). Discrete generative models for sentence compression. *Empirical Methods on Natural Language Processing (EMNLP)*.

Pu, Y., Gan, Z., Henao, R., Yuan, X., Li, C., Stevens, A., & Carin, L. (2016). Variational autoencoder for deep learning of images, labels and captions. *Advances in neural information processing systems*, *29*.

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. *International conference on machine learning*, 214–223.

Paige, B., van de Meent, J.-W., Desmaison, A., Goodman, N., Kohli, P., Wood, F., Torr, P., et al. (2017). Learning disentangled representations with semi-supervised deep generative models. *Advances in neural information processing systems*, *30*.

Luo, Y., & Pfister, H. (2018). Adversarial defense of image classification using a variational auto-encoder. *arXiv preprint arXiv:1812.02891*.

Pfohl, S. R., Duan, T., Ding, D. Y., & Shah, N. H. (2019). Counterfactual reasoning for fair clinical risk prediction. *Machine Learning for Healthcare Conference*, 325–358.

Yacoby, Y., Pan, W., & Doshi-Velez, F. (2020). Failure modes of variational autoencoders and their effects on downstream tasks. *arXiv preprint arXiv:2007.07124*.

7 Contribution statement

Regarding the code/experiments Leo focused on adapting and creating the dataset with its figures Figure 1 as well as the implementation for the vanilla VAE and its figures Figure 2. Roberto focused on the code/implementation and visualization for the VAEs that use the Mixture of Gaussians Figure 3 and Figure 4. The writing and analysis of the final submission was done equally between the two of us, as we only worked on the writing while we were together and jointly agreed on the conclusions.