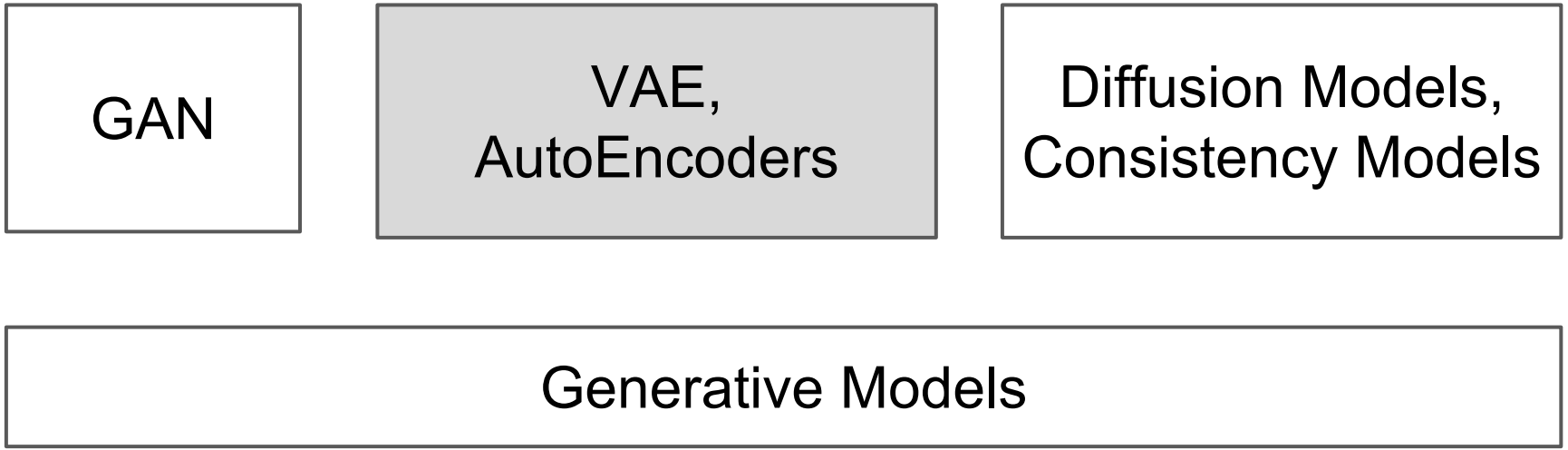
VAE: Variational Autoencoder

Rowel Atienza
github.com/roatienza
University of the Philippines
2023

Generative Models Landscape (Vision)



GAN vs VAE

GAN focuses on modeling the input distribution, P(x)

VAE focuses on the continuous latent space code to indirectly model the input distribution, P(x)

$$\boldsymbol{\theta}^* = arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{m} p_{model}(\boldsymbol{x}^{(i)}; \boldsymbol{\theta})$$

$$\boldsymbol{\theta}^* = arg\max_{\boldsymbol{\theta}} \log \prod_{i=1}^{m} p_{model}(\boldsymbol{x}^{(i)}; \boldsymbol{\theta})$$

$$\boldsymbol{\theta}^* = arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{m} \log p_{model}(\boldsymbol{x}^{(i)}; \boldsymbol{\theta})$$

GAN

$$P_\theta(\boldsymbol{x}) = \int P_\theta(\boldsymbol{z}|\boldsymbol{x}) P(\boldsymbol{x}) d\boldsymbol{z}$$

$$Q_\phi(\boldsymbol{z}|\boldsymbol{x}) \approx P_\theta(\boldsymbol{z}|\boldsymbol{x})$$

VAE

GAN vs VAE

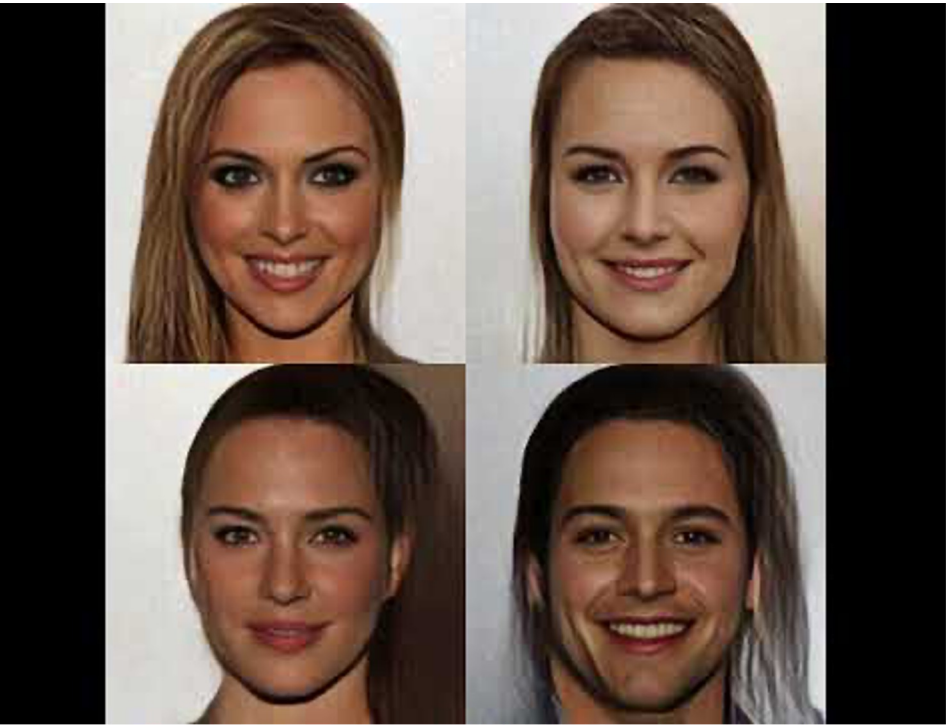
GAN tends to generate more realistic synthetic signals

~~VAE generated signals appear more blurry~~

> Very Deep VAEs Generalize Autoregressive Models and can Outperform them on Images (ICLR 2021)

Figure 1: **Selected samples from our very deep VAE on FFHQ-256, and a demonstration of the learned generative process.** VAEs can learn to first generate global features at low resolution,

VAE



VAE

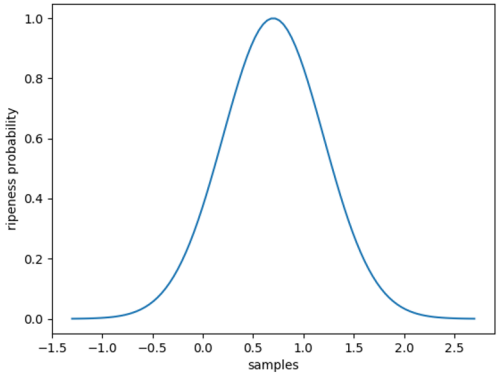
GAN

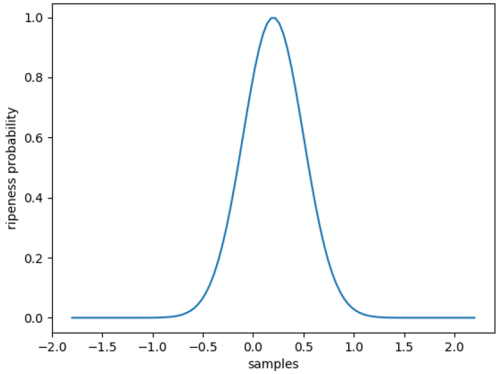
GAN vs VAE

Unlike GAN, VAE provides a suitable variational framework for Bayesian inference and learning with latent variables

Robot Inference:

If the mango is ripe, I'll harvest it.





Input distribution $P_\theta(\boldsymbol{x})$ and Joint distribution $P_\theta(\boldsymbol{x},\boldsymbol{z})$

$$\boldsymbol{x} \sim P_\theta(\boldsymbol{x})$$

In machine learning, we are interested in finding $P_\theta(\boldsymbol{x}, \boldsymbol{z})$, the joint distribution between $\boldsymbol{x}$ and latent code $\boldsymbol{z}$

$$P_\theta(\boldsymbol{x}) = \int P_\theta(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}$$

However, $P_\theta(\boldsymbol{x}, \boldsymbol{z})$ is intractable or it does not have an analytic form or a good estimator.

z
as vector
of
attributes

Source: http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

Using Bayes Theorem

$$P_\theta(\boldsymbol{x}) = \int P_\theta(\boldsymbol{x}|\boldsymbol{z})P(\boldsymbol{z})d\boldsymbol{z}$$

A neural network can easily ignore $P_\theta(\boldsymbol{x}|\boldsymbol{z})$ to come up with a trivial solution $P_\theta(\boldsymbol{x}|\boldsymbol{z}) = P_\theta(\boldsymbol{x})$.

$$P_\theta(\boldsymbol{x}) = \int P_\theta(\boldsymbol{z}|\boldsymbol{x})P(\boldsymbol{x})d\boldsymbol{z}$$

However, $P_\theta(\boldsymbol{z}|\boldsymbol{x})$ is also intractable.

Variational Inference Model (Encoder)

$$Q_\phi(\boldsymbol{z}|\boldsymbol{x}) \approx P_\theta(\boldsymbol{z}|\boldsymbol{x})$$

$Q_\phi(\boldsymbol{z}|\boldsymbol{x})$ provides a good estimate of $P_\theta(\boldsymbol{z}|\boldsymbol{x})$

$Q_\phi(\boldsymbol{z}|\boldsymbol{x})$ is parametric and tractable.

$Q_\phi(\boldsymbol{z}|\boldsymbol{x})$ can be approximated by deep neural networks by optimizing the parameters ϕ.

Variational Inference Model (Encoder)

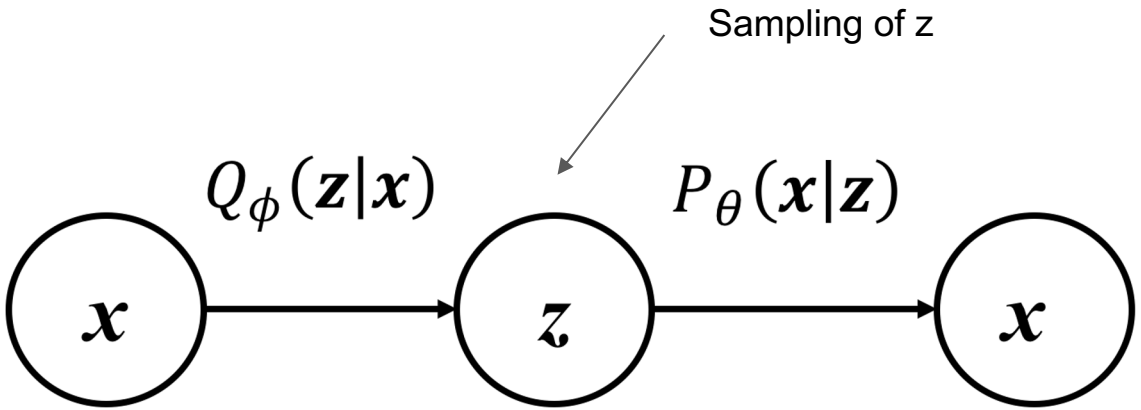
Typically, $Q_\phi(\boldsymbol{z}|\boldsymbol{x})$ is chosen to be a multivariate Gaussian:

$$Q_\phi(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{z}; \boldsymbol{\mu}(\boldsymbol{x}), diag\left(\boldsymbol{\sigma}(\boldsymbol{x})\right)\right)$$

Both mean, $\boldsymbol{\mu}(\boldsymbol{x})$, and standard deviation, $\boldsymbol{\sigma}(\boldsymbol{x})$, are computed by the encoder neural network from the input data points.

The diagonal matrix implies that the elements of $\boldsymbol{z}$ are independent.

Probabilistic Graphical Model of VAE



To estimate $P_\theta(\boldsymbol{x})$, we must identify its relationship with $Q_\phi(\boldsymbol{z}|\boldsymbol{x})$ and $P_\theta(\boldsymbol{x}|\boldsymbol{z})$.

Core Equation of VAE

If $Q_\phi(\boldsymbol{z}|\boldsymbol{x})$ is an estimate of $P_\theta(\boldsymbol{z}|\boldsymbol{x})$, the Kullback-Leibler (KL) divergence determines the distance between these two conditional densities:

$$D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x})\middle\|P_\theta(\boldsymbol{z}|\boldsymbol{x})\right) = \mathbb{E}_{\boldsymbol{z}\sim Q}\left[\log Q_\phi(\boldsymbol{z}|\boldsymbol{x}) - \log P_\theta(\boldsymbol{z}|\boldsymbol{x})\right]$$

Using Bayes Theorem:

$$D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x})\middle\|P_\theta(\boldsymbol{z}|\boldsymbol{x})\right)$$
$$= \mathbb{E}_{\boldsymbol{z}\sim Q}\left[\log Q_\phi(\boldsymbol{z}|\boldsymbol{x}) - \log P_\theta(\boldsymbol{x}|\boldsymbol{z}) - \log P_\theta(\boldsymbol{z})\right] + \log P_\theta(\boldsymbol{x})$$

Core Equation of VAE

Recognizing $\log Q_\phi(\boldsymbol{z}|\boldsymbol{x}) - \log P_\theta(\boldsymbol{z}) = D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x}) \| P_\theta(\boldsymbol{z})\right)$:

$$\log P_\theta(\boldsymbol{x}) - D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x}) \| P_\theta(\boldsymbol{z}|\boldsymbol{x})\right) = \mathbb{E}_{\boldsymbol{z}\sim Q}[\log P_\theta(\boldsymbol{x}|\boldsymbol{z})] - D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x}) \| P_\theta(\boldsymbol{z})\right)$$

Term being maximized

Distance between Q and true P: Approx Zero

Decoder (Reconstruction)

Distance between Q and Prior P(z)

Left Side is known as Evidence Lower Bound (ELBO) of $\log_\theta P(\boldsymbol{x})$

Core Equation of VAE

Maximizing ELBO by optimizing the parameters ϕ and θ of the neural network means:

- $D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x})\middle\|P_\theta(\boldsymbol{z}|\boldsymbol{x})\right) \rightarrow 0$ or the inference model is getting better in encoding the attributes of $\boldsymbol{x}$ in $\boldsymbol{z}$.
- $\log P_\theta(\boldsymbol{x}|\boldsymbol{z})$ on the right-hand side of Equation 8.1.10 is maximized or the decoder model is getting better in reconstructing $\boldsymbol{x}$ from the latent vector $\boldsymbol{z}$.

Optimization

- The decoder term $\mathbb{E}_{\boldsymbol{z}\sim Q}[\log P_\theta(\boldsymbol{x}|\boldsymbol{z})]$ means that the generator takes $\boldsymbol{z}$ samples from the output of the inference model to reconstruct the inputs.
- Maximizing this term implies that we minimize a *Reconstruction Loss*, $\mathcal{L}_R$.
- If the image distribution is assumed to be Gaussian, MSE can be used.
- If every pixel is considered a Bernoulli distribution, the loss function is a binary cross entropy.

Optimization

- The second term, $-D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x})\middle\|P_\theta(\boldsymbol{z})\right)$, turns out to be straightforward to evaluate.
- Q_ϕ is a Gaussian distribution.
- Typically, $P_\theta(\boldsymbol{z}) = P(\boldsymbol{z}) = \mathcal{N}(0, I)$ is also a Gaussian with zero mean and standard deviation equal to 1.0. The KL term simplifies to:

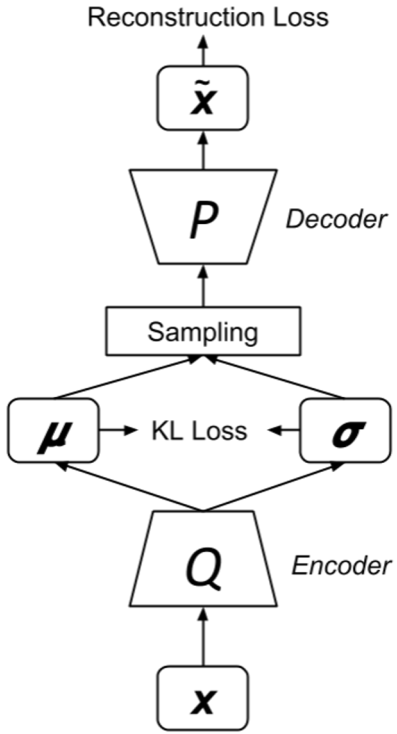
$$-\mathcal{L}_{KL} = -D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x})\middle\|P_\theta(\boldsymbol{z})\right) = \frac{1}{2}\sum_{j=1}^{J}\left(1 + \log\left(\sigma_j\right)^2 - \left(\mu_j\right)^2 - \left(\sigma_j\right)^2\right)$$

where J is the dimensionality of $\boldsymbol{z}$. Both μ_j and σ_j are functions of $\boldsymbol{x}$ computed through the inference model. To maximize $-D_{KL}$, $\sigma_j \to 1$ and $\mu_j \to 0$.

VAE Loss Function

$$\mathcal{L}_{VAE} = \mathcal{L}_{R} + \mathcal{L}_{KL}$$

Training VAE



Without Reparameterization Trick

Training VAE

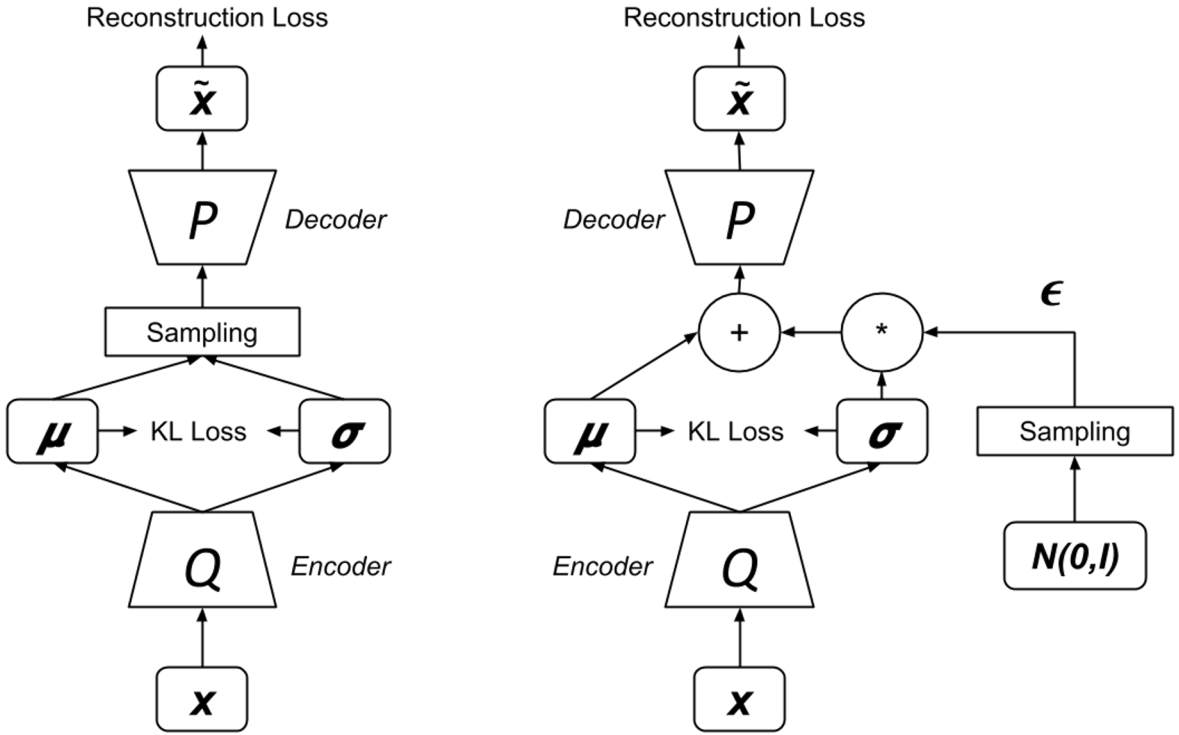
- No problem with forward computation in the network
- Problem is backpropagation gradients will not pass through the stochastic **Sampling** block
- While it is fine to have stochastic inputs for neural networks, it is not possible for gradients to go through a stochastic layer.

Reparameterization Trick

- The solution to this problem is to push out the **Sampling** process
- The solution is to sample ϵ from an isotropic Gaussian distribution and adjust the value using the encoder predicted μ and σ
- Compute the sample as:

$$\text{Sample} = \boldsymbol{\mu} + \boldsymbol{\epsilon\sigma}$$

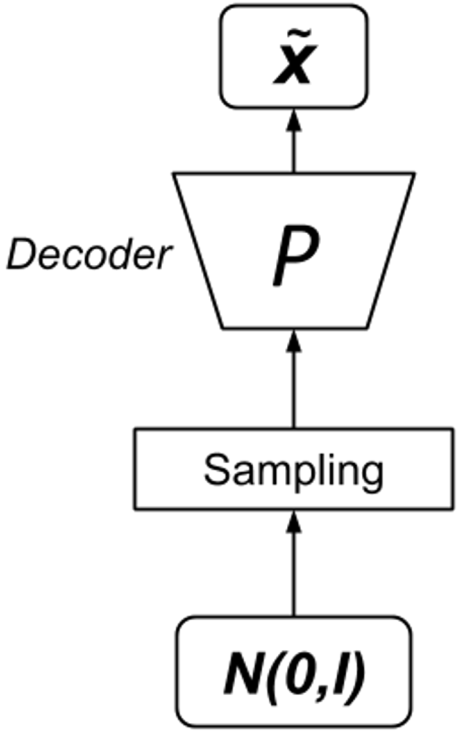
Training VAE using Reparameterization Trick



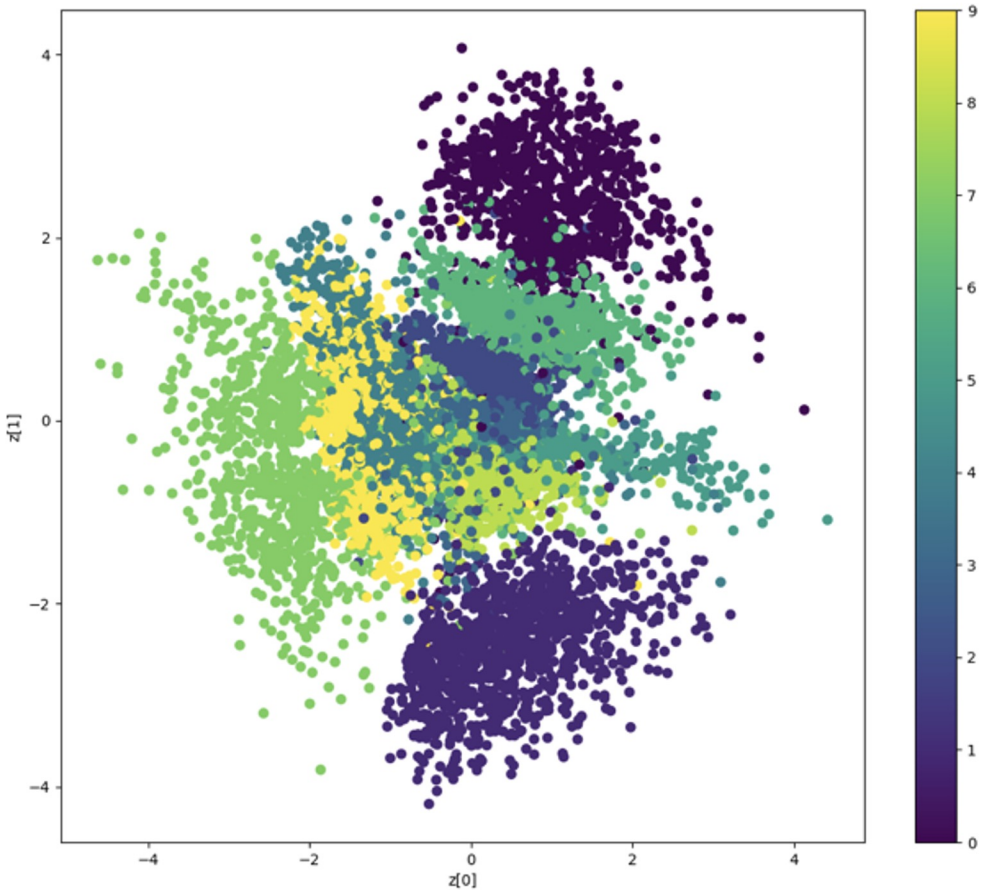
Decoder Testing

No need for the encoder

Sample directly from isotropic unit Gaussian to generate ***x***

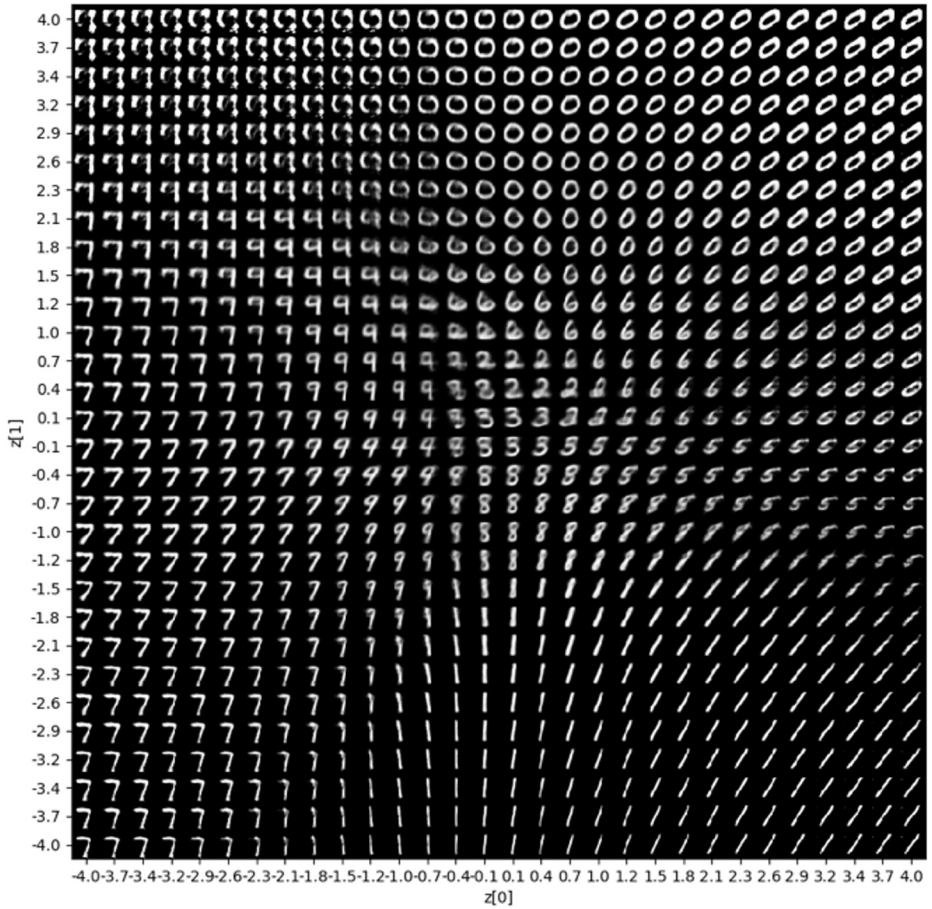


Example distribution
for 2-dim z (VAE MLP)

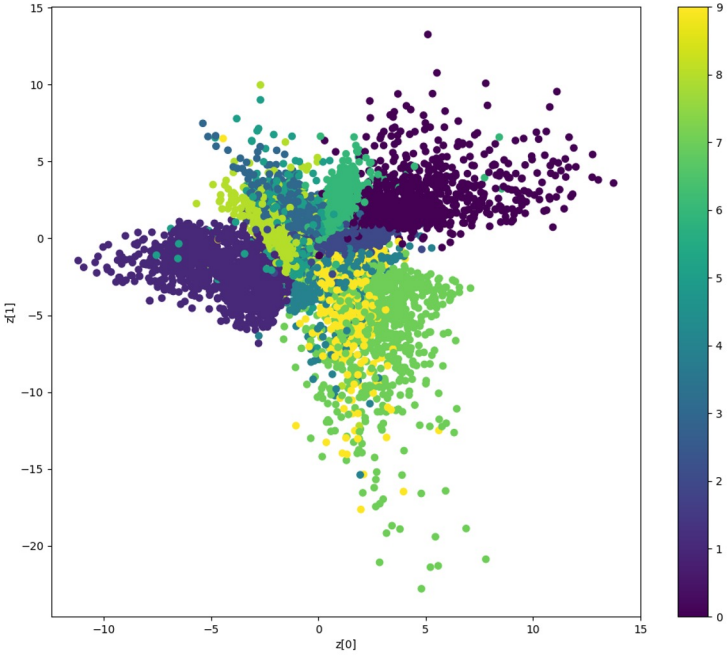


Example distribution for 2-dim z

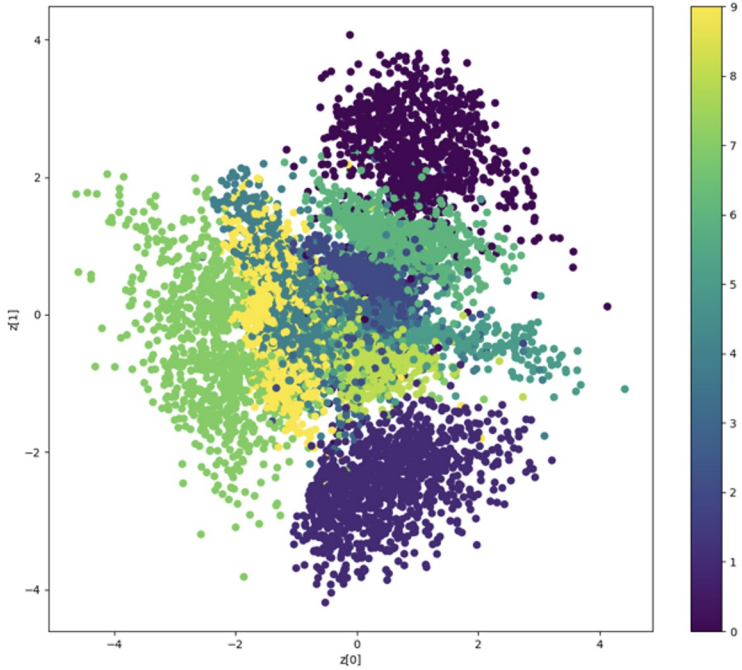
(VAE MLP)



2-dim Autoencoder vs 2-dim VAE

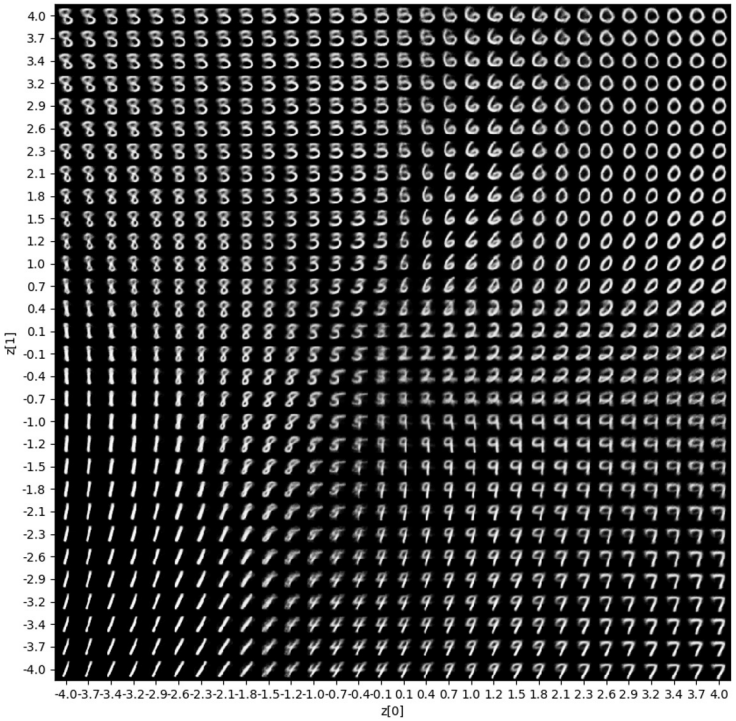


Autoencoder

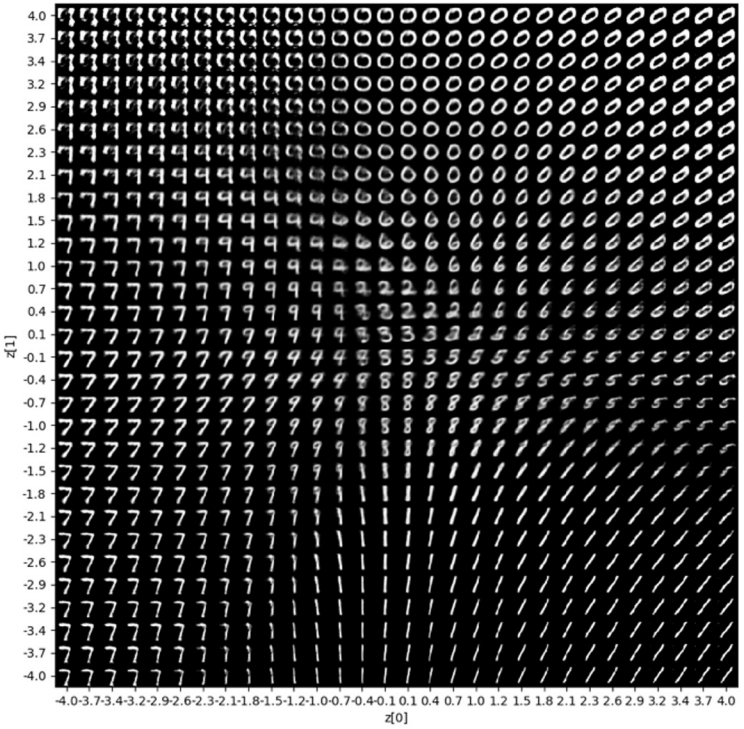


VAE

2-dim Autoencoder vs 2-dim VAE



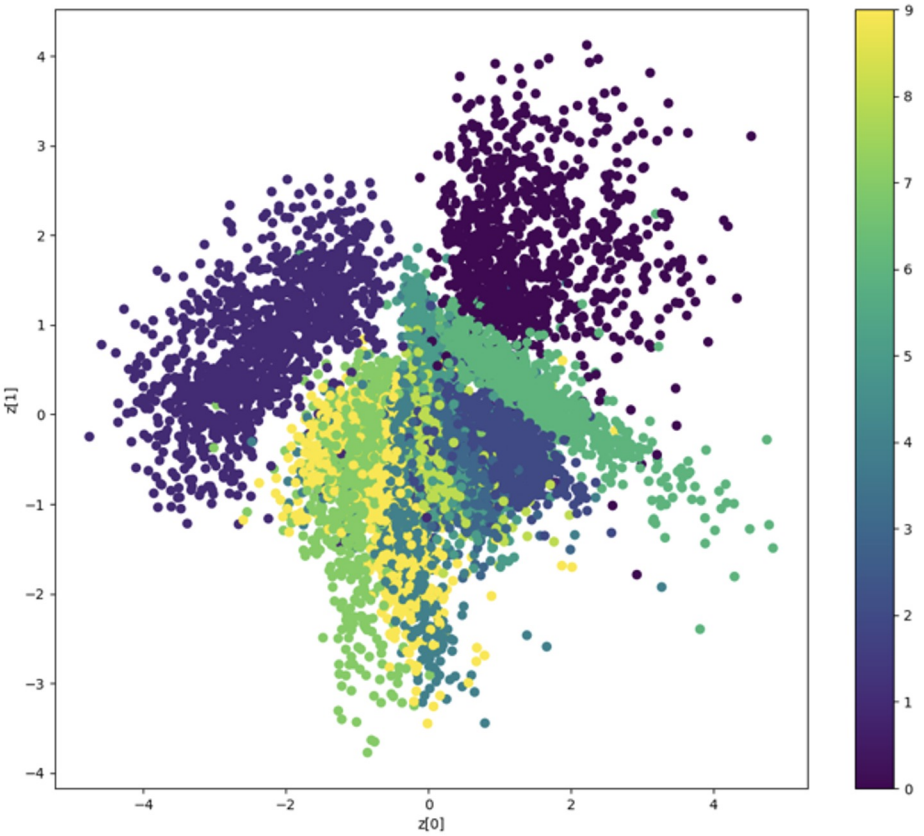
Autoencoder



VAE

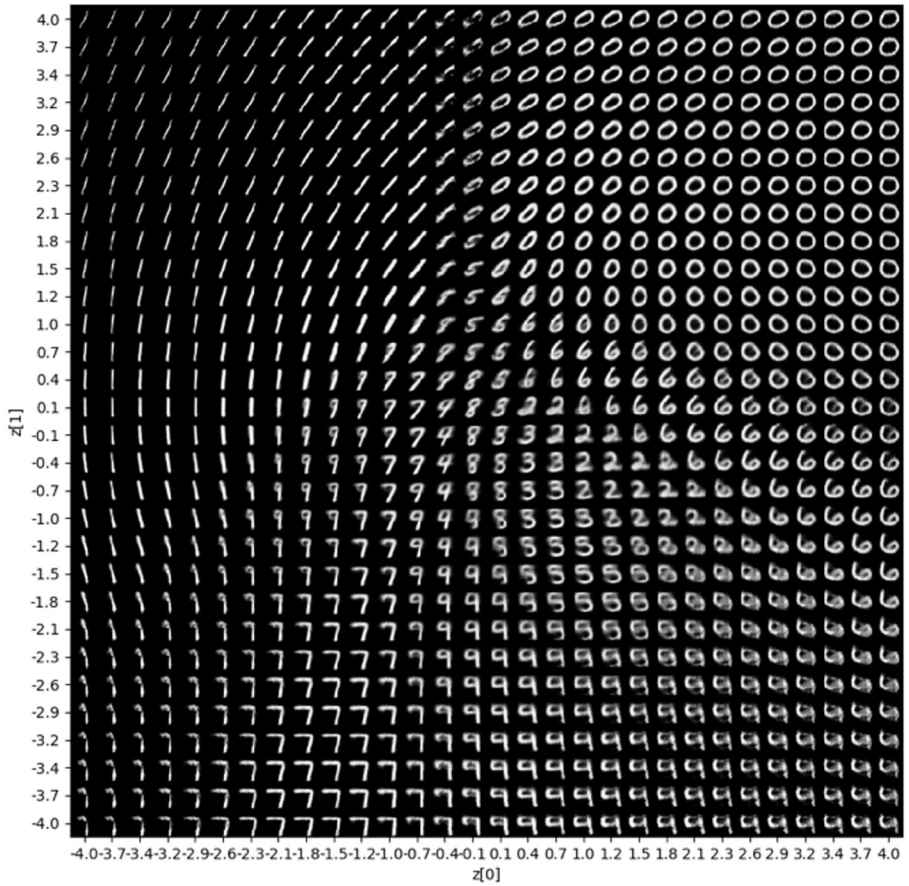
Example distribution for 2-dim z

(VAE CNN)



Example distribution for 2-dim z

(VAE CNN)



Conditional VAE

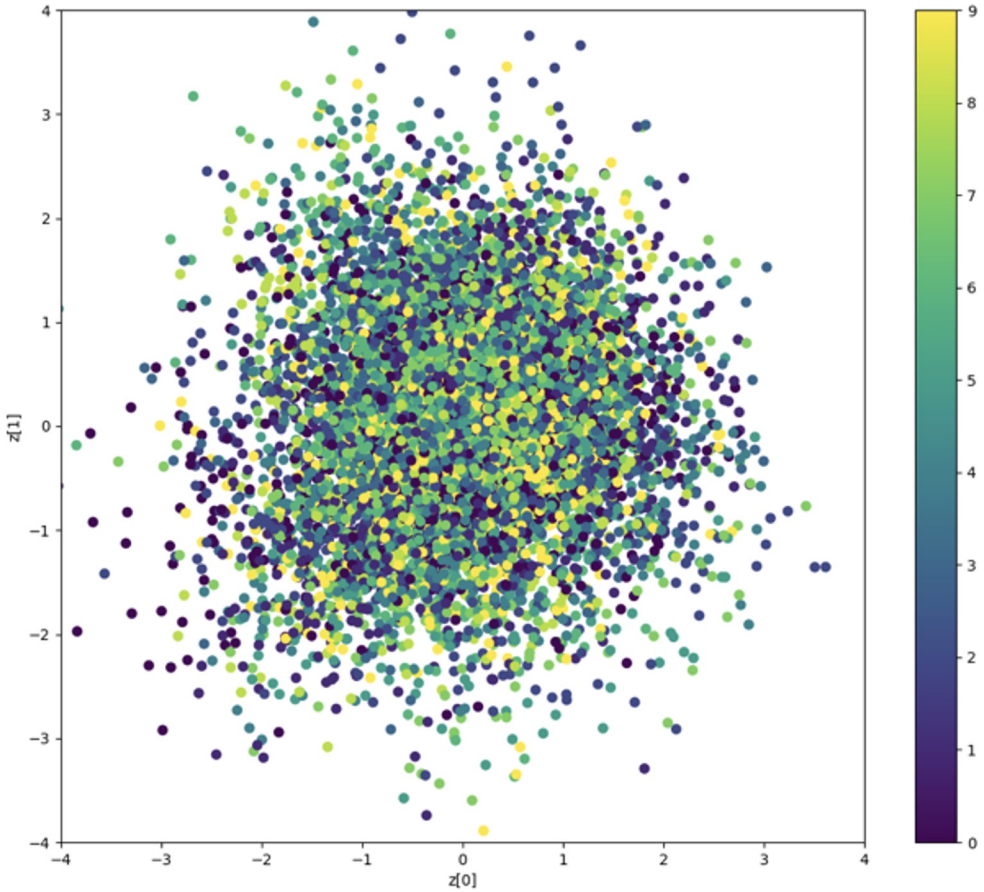
$$\log P_\theta(\boldsymbol{x}|\boldsymbol{c}) - D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{c})\middle\|P_\theta(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{c})\right) = \mathbb{E}_{\boldsymbol{z}\sim Q}[\log P_\theta(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{c})] - D_{KL}\left(Q_\phi(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{c})\middle\|P_\theta(\boldsymbol{z}|\boldsymbol{c})\right)$$

No change in the loss function

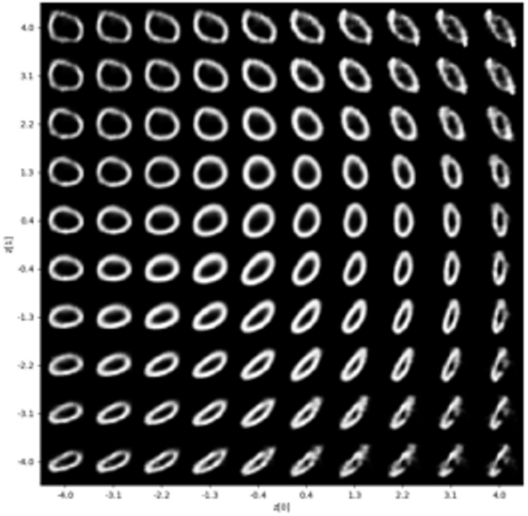
- *Reconstruction Loss* of the decoder given both the latent vector and the condition
- *KL Loss* between the encoder given both the latent vector and the condition and the prior distribution given the condition. Similar to VAE, typically we choose $P_\theta(\boldsymbol{z}|\boldsymbol{c}) = P(\boldsymbol{z}|\boldsymbol{c}) = \mathcal{N}(0, I)$.

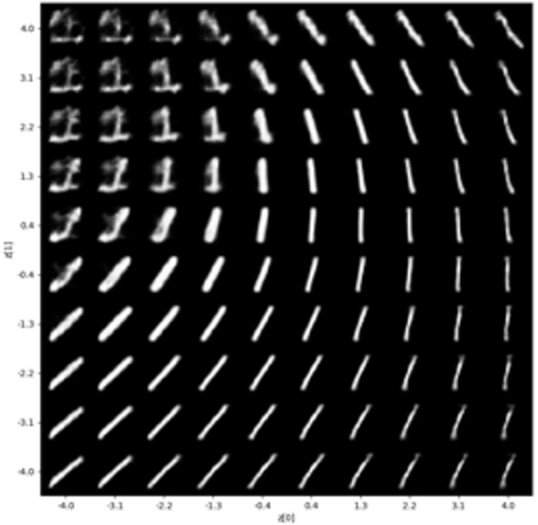
Example distribution for 2-dim z

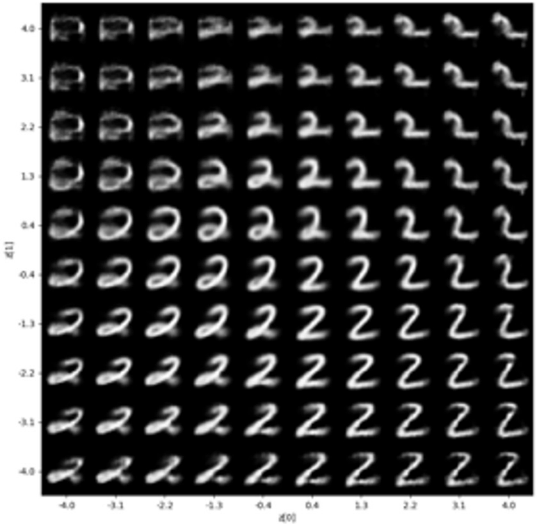
(VAE CNN)



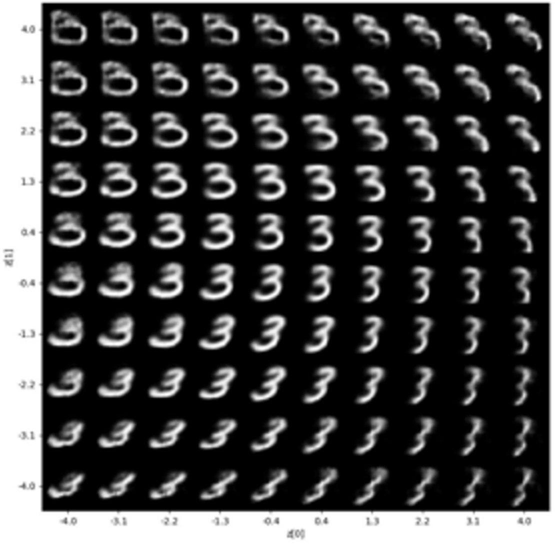
CVAE Sample Outputs

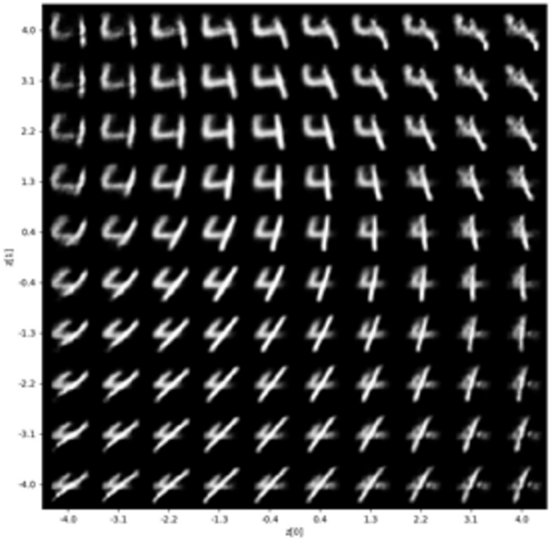


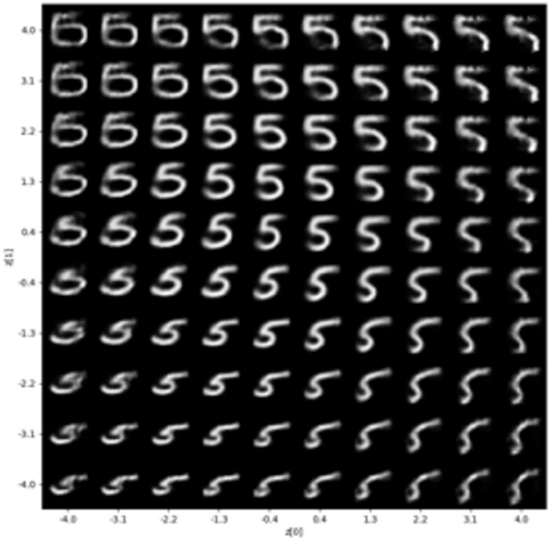


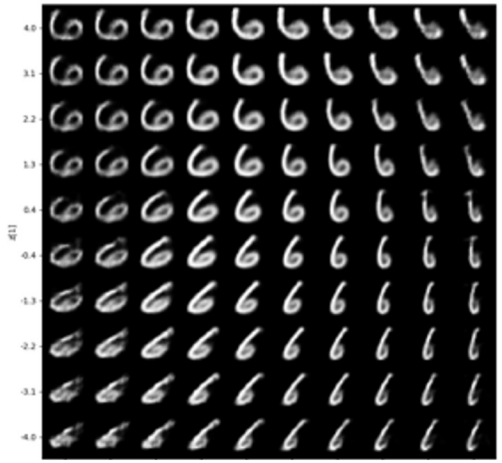


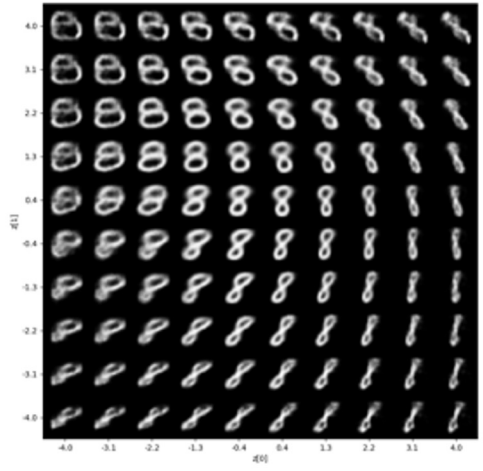
CVAE Sample Outputs

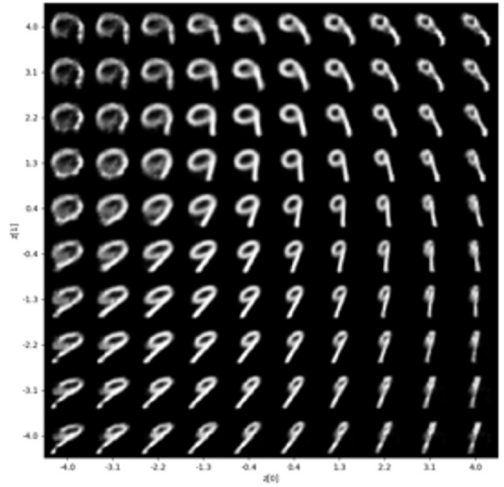












CVAE Sample Outputs

Beta-VAE - Disentangled Representation VAE

- Intuitively, we define a vector representation as disentangled, if it can be decomposed into a number of subspaces, each one of which is compatible with, and can be transformed independently by a unique symmetry transformation. [1]
- A disentangled code or representation is a tensor that can change a specific feature or attribute of the output data while not affecting the other attributes. [2]
- Capture the independent features of a given data in such a way that if one feature changes, the others remain unaffected [3]

1. Higgins, Irina, et al. "Towards a Definition of Disentangled Representations." arXiv preprint arXiv:1812.02230 (2018).
2. Atienza, R. Advanced Deep Learning with Keras. Packt Pub (2018)
3. Locatello et al. “Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations.” ICML 2019

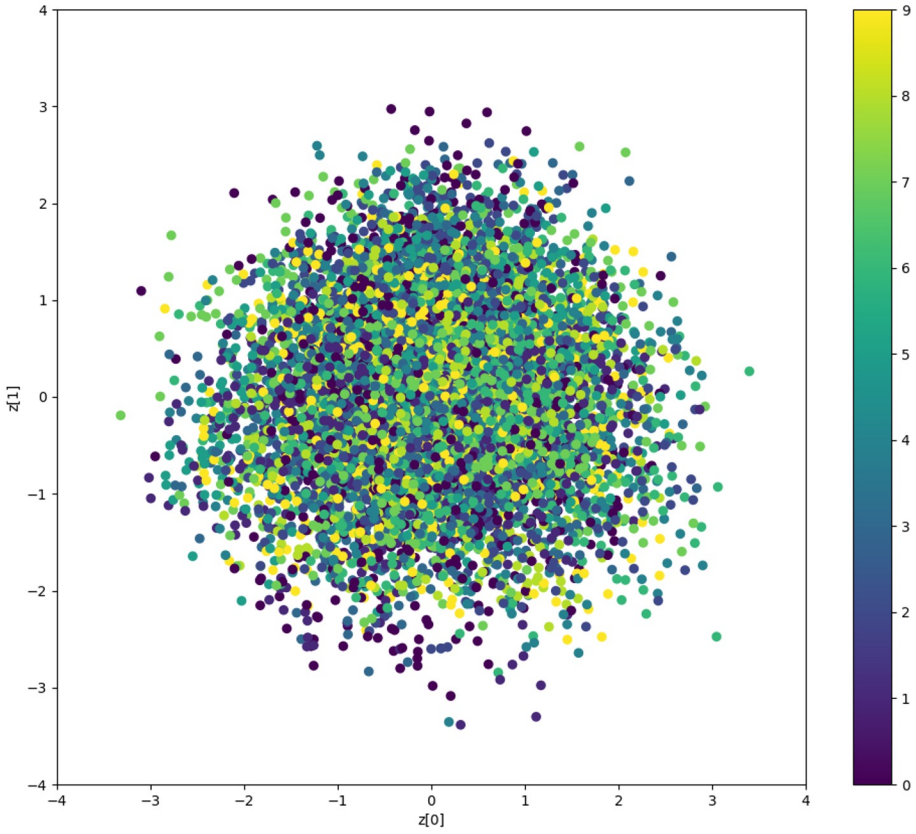
β-VAE - Change in Loss Function

$$\mathcal{L}_{\beta-VAE} = \mathcal{L}_R + \beta\mathcal{L}_{KL}$$

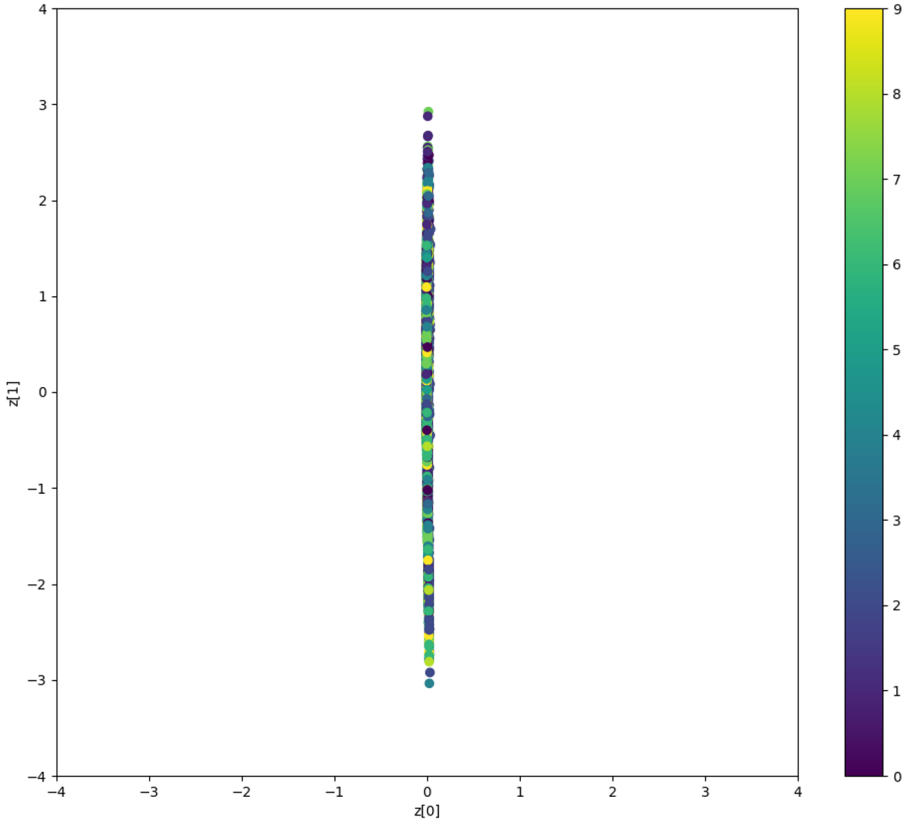
β>1 acts as a loss regularizer. The effect is to enforce tighter standard deviation resulting to disentanglement of latent codes. CVAE and VAE are special cases where β=1.

How to determine β? Answer: Trial and error

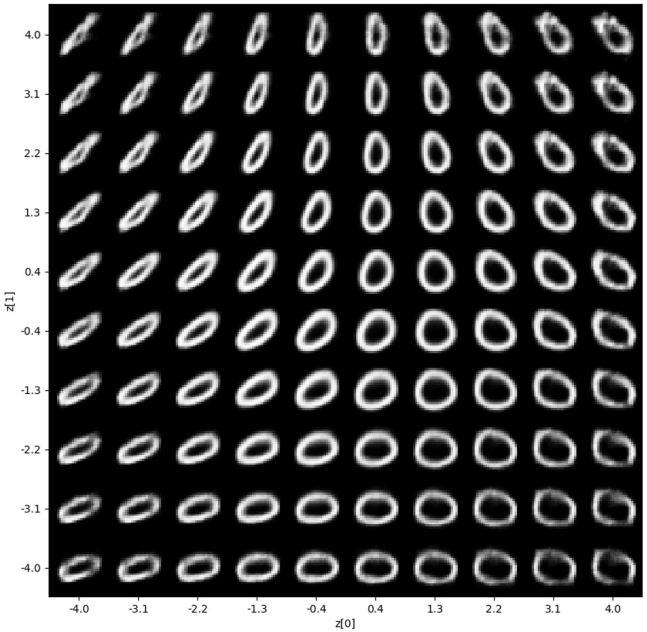
β=7

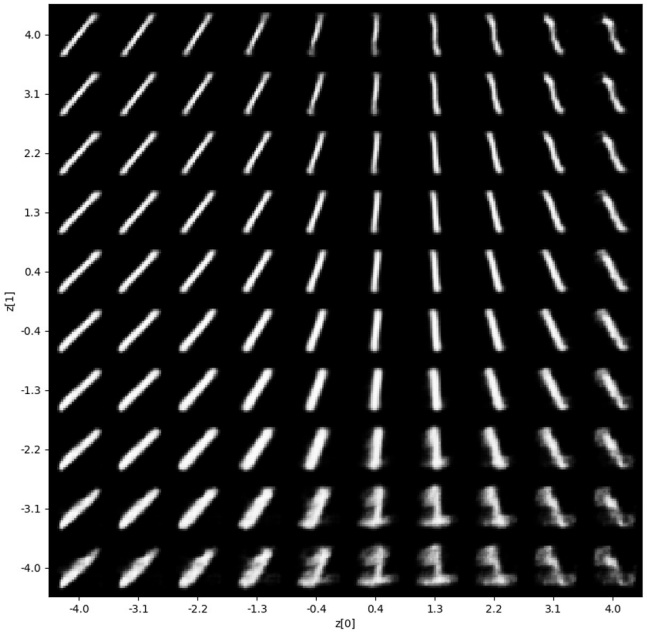


β=10

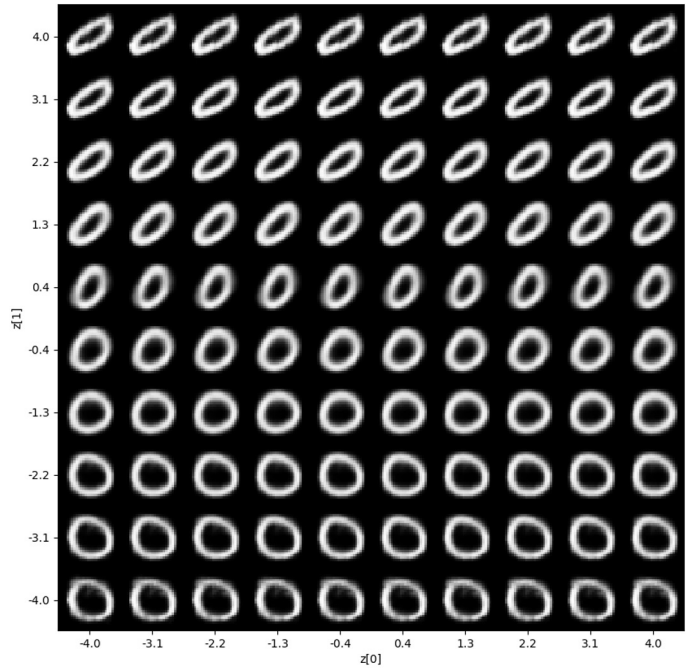


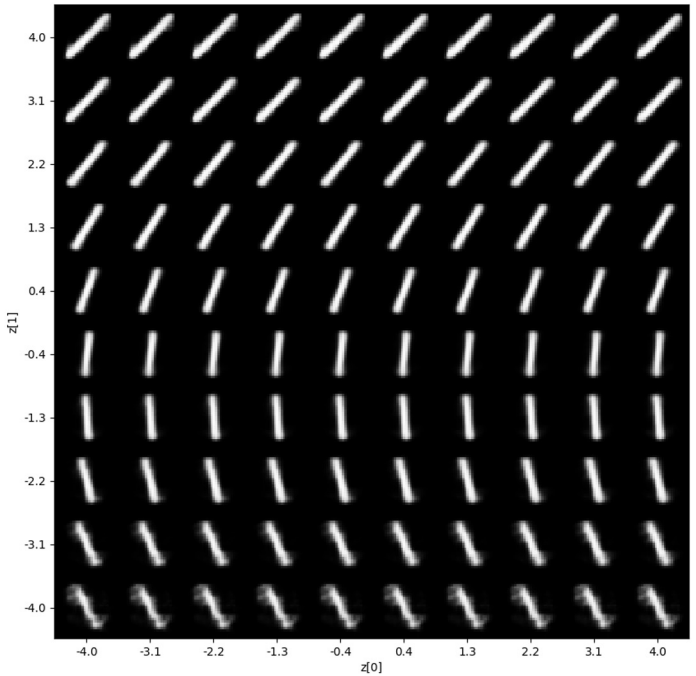
β=7


4.0
3.1
2.2
1.3
0.4
-0.4
-1.3
-2.2
-3.1
-4.0
z[1]
-4.0 -3.1 -2.2 -1.3 -0.4 0.4 1.3 2.2 3.1 4.0
z[0]


4.0
3.1
2.2
1.3
0.4
-0.4
-1.3
-2.2
-3.1
-4.0
z[1]
-4.0 -3.1 -2.2 -1.3 -0.4 0.4 1.3 2.2 3.1 4.0
z[0]

β=10


4.0
3.1
2.2
1.3
0.4
-0.4
-1.3
-2.2
-3.1
-4.0
z[1]
-4.0 -3.1 -2.2 -1.3 -0.4 0.4 1.3 2.2 3.1 4.0
z[0]


4.0
3.1
2.2
1.3
0.4
-0.4
-1.3
-2.2
-3.1
-4.0
z[1]
-4.0 -3.1 -2.2 -1.3 -0.4 0.4 1.3 2.2 3.1 4.0
z[0]

References

Kingma, Diederik P., and Max Welling. "Auto-encoding Variational Bayes." arXiv preprint arXiv:1312.6114 (2013).

Sohn, Kihyuk, Honglak Lee, and Xinchen Yan. "Learning structured output representation using deep conditional generative models." Advances in Neural Information Processing Systems. 2015.

Doersch, Carl. "Tutorial on variational autoencoders." arXiv preprint arXiv:1606.05908 (2016).

Atienza, R. “Advanced Deep Learning with TF2 and Keras.” Packt Pub 2020

End