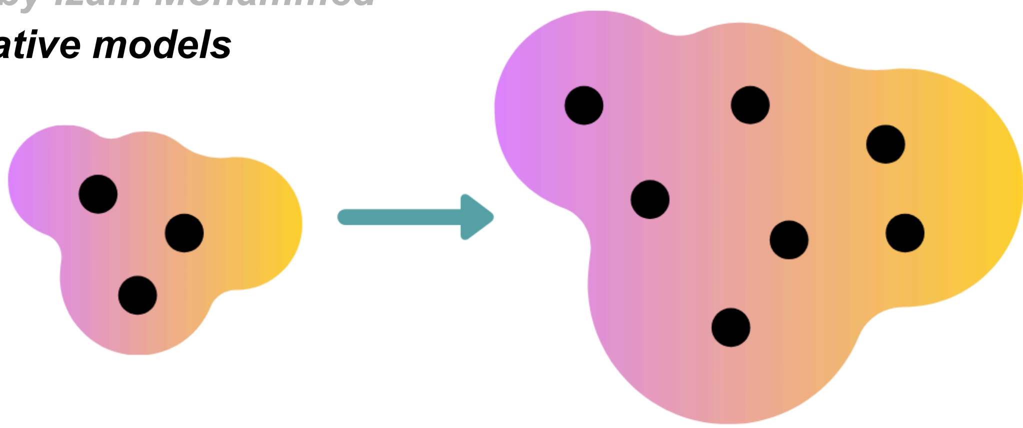
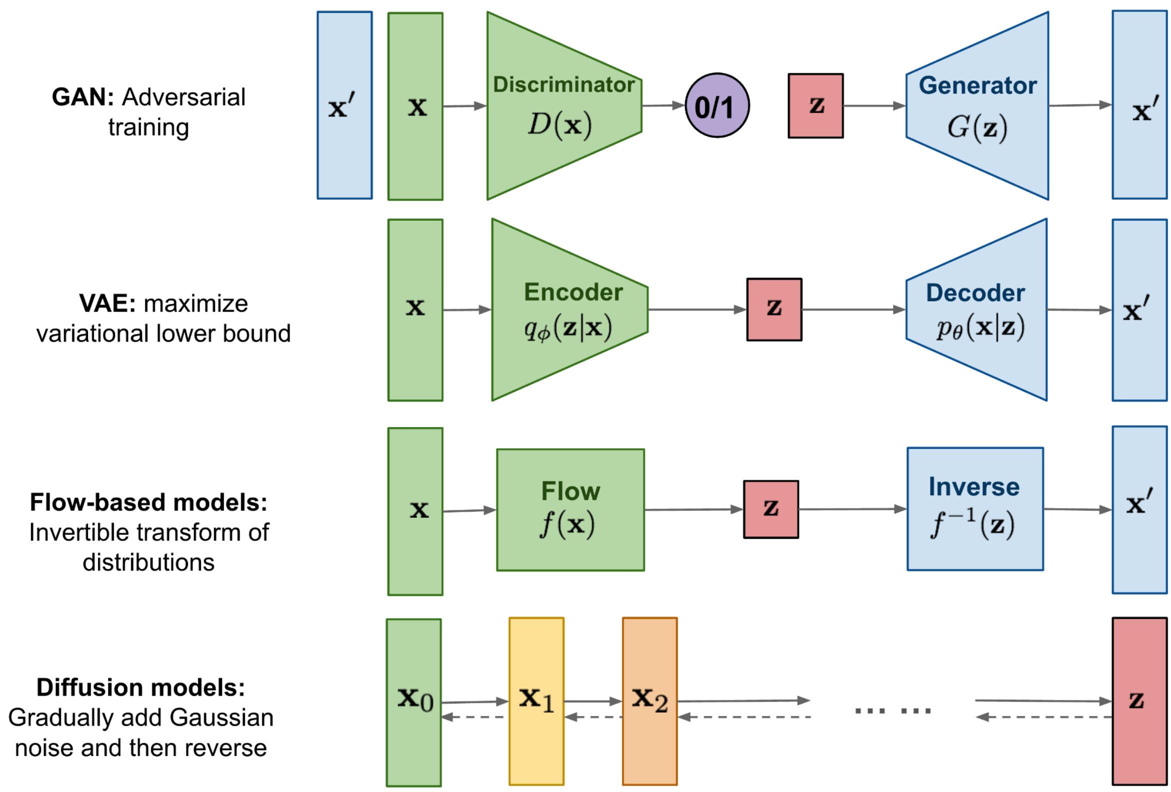
Notes by Izam Mohammed

# Generative models



*It is a class of machine learning models used to generate new data points that can be used as training datasets. These models are widely used in various fields such as computer vision, and natural language processing.*




### *Advantages of Generative models*

- *Data Generation*
  - *It can generate new data points that are exactly similar to the training data*
- *Unsupervised Learning*
  - *No need for labeled data*
- *Data Imputation*

    - *Generative models can fill in missing or corrupted parts of data, a process known as data imputation.*
- ➢ *Domain adaptation*
    - *Generative models can be used for domain adaptation, where the goal is to transfer knowledge from a source domain to a target domain with different distributions.*

***Disadvantages of Generative models***

- ➢ *Training complexity*
    - *Challenging due to its inheritance complexity. Sometimes, it involves optimizing two neural networks simultaneously*
- ➢ *Quality controls*
    - *Ensuring and evaluating the quality of generated samples in GAN is difficult.*
- ➢ *Overfitting*
    - *If we are training on a small or noisy dataset, there is a possibility of overfitting.*
- ➢ *Lack of Interpretability*
    - *Can be challenging to interpret and understand. This lack of interpretability can hinder model debugging, troubleshooting, and trustworthiness in real-world applications.*
- ➢ *Evaluation Difficulty*
    - *Assessing the performance of generative models can be challenging.*

***Generative vs Discriminative models***

| | ***Generative*** | ***Discriminative*** |
|---|---|---|
| *Objective* | *learn the underlying data distribution and generate new samples from that distribution.* | *learn the boundary between different classes in the input space.* |
| *Data Generation* | *can generate new data samples that resemble the training data distribution.* | *do not generate new data; instead, they classify or predict labels for given input data.* |
| *Use Cases* | *data augmentation, generating synthetic data, anomaly detection* | *classification, regression* |
| *Model complexity* | *Often more complex than discriminative models* | *simpler compared to generative models* |
| *Handling missing data* | *Can handle missing data by learning a probabilistic model of the data distribution* | *do not handle missing data directly but may require preprocessing steps such as imputation* |
| *Robustness to* | *Can be sensitive to* | *Tend to be more robust to* |

| | | |
|---|---|---|
| *irrelevant features* | *irrelevant features* | *irrelevant features* |
| *examples* | ➢ *Variational Autoencoders (VAEs)*<br>➢ *Generative Adversarial Networks (GANs)*<br>➢ *Autoregressive Models (e.g., PixelCNN)*<br>➢ *Hidden Markov Models (HMMs)* | ➢ *Logistic Regression*<br>➢ *Support Vector Machines (SVMs)*<br>➢ *Decision Trees*<br>➢ *Random Forests* |

***Types of generative models***

- *Autoregressive models*
  - *These are probabilistic models used to predict the next value based on the previous value*
  - *Commonly used in time series data and NLP*
  - *It produces coherent patterns within the domains*
- *Variational autoencoder*
  - *Type of autoencoder that learns to encode input data into a low dimensional latent space and decode as well.*
  - *These are probabilistic models that learn the probability distribution of the latent space.*
  - *Consists of an encoder and a decoder*
- *Generative adversarial network*
  - *Consists of a generator and discriminator which trained simultaneously through adversarial learning*
  - *widely used for generating synthetic data, such as images, texts, and music.*
- *Flow-based models*
  - *Learn to transform a simple distribution (Gaussian) into a complex data distribution through a series of invertible transformations*
  - *Eg:- RealNVP(Non volume preserving) and GLOW (generative flow with invertable 1x1 transformation)*
- *Energy-based models → EBM*
  - *A probabilistic model that is associated with an energy function with each configuration of data*
  - *Learns to give low energy to certain data points and high energy to other data points*
  - *They are trained to minimize the energy of training data points and increase the energy of other data points.*
- *Autoencoding Variational Bayes → AEVB*
  - *Combines the variational inference with autoencoding to perform Bayesian inference in complex probabilistic models*

***Real-World use cases***

- ➢ *Art creation*
- ➢ *Drug discovery*
- ➢ *Content creation*
- ➢ *Video games.*

# *GAN*

***What it is***

*Gan means Generative adversarial Network. It is widely used to generate synthetic data, such as images, texts, and music. The primary goal is to generate data similar to the training data.*

***How it works***

*It has 2 components*

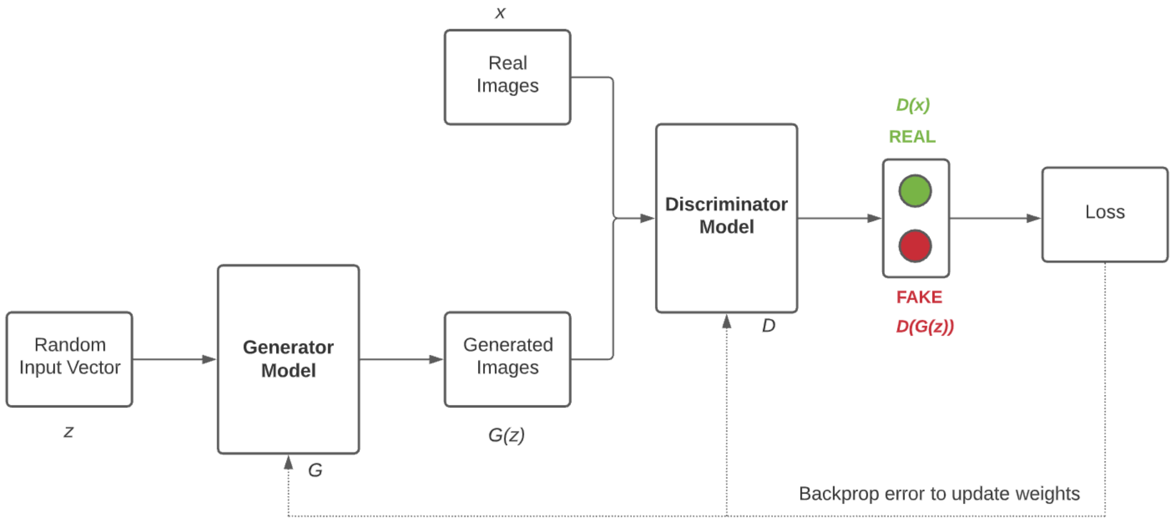
1. *Generator*

   *The generator model generates new images by taking fixed-size random noise as an input. Generated images are fed into the discriminator*

2. *Discriminator*

   *The discriminator model takes an image as input (generated and real) and classifies it as real or fake*

***Work in detail***




- ➢ *The generator model G takes a random input vector z as an input and generates the image G(z).*
- ➢ *These generated images along with real images x from training data are then fed into the discriminator model D*
- ➢ *The discriminator model then classifies the generated image as real or fake.*
- ➢ *Then we can measure the loss and this loss is backpropagated to update the weights of the generator and discriminator*
- ➢ *When we are training the Discriminator, we have to freeze the generator and back-propagate errors to only update the discriminator*

- *The fight between generator and discriminator can be expressed mathematically as*

$$\min_G \max_D V(D, G)$$

$$V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z))]$$

| | |
|---|---|
| G | Generator Model |
| D | Discriminator Model |
| z | Random Noise (Fixed size input vector) |
| x | Real Image |
| G(z) | Image generated by Generator (Fake Image) |
| $p_{data}(x)$ | Probability Distribution of Real Images |
| $p_z(z)$ | Probability Distribution of Fake Images |
| D(G(z)) | Discriminator's output when the generated image is an input |
| D(x) | Discriminator's output when the real image is an input |

1. $E_{x\sim pdata(x)}$ [log D(x)]: Average log probability of D when real image is input.
2. $E_{z\sim pz(z)}$ [log(1 – D(G(z)))]: Average log probability of D when the generated image is input.

*In here …..*

***Advantages***

- *Data generation*
  - *Great at the generation of new data samples that closely resemble the training data.*
- *Unsupervised learning*
  - *Do not require labeled data for training*
- *Feature Learning*
  
  *Generator and discriminator network in gan learn a useful representation of data during the training time*
- *Robustness*
  
  *GANs are inherently robust to noise and variation in the input data*

***Disadvantages***

- *Mode Collapse*
  - *the generator produces limited variations of samples, ignoring large portions of the data distribution.*
- *Training instability*
  - *GAN training can be unstable and sensitive to hyperparameters, architecture choices, and initialization.*
- *Evaluation Difficulty*

- *challenging due to the lack of well-defined metrics for assessing sample quality and diversity.*

➢ *Generator-Discriminator Imbalance*
- *If one component becomes too dominant, it can hinder convergence and degrade sample quality.*

➢ *Mode dropping*
- *occurs when the generator fails to capture certain modes or regions of the data distribution.*

➢ *Require large dataset*
- *to get reasonable performance*

***Types of GAN***

➢ *Vanilla GAN*
- *The original gan*

➢ *Deep Convolution GAN*
- *Introduced the convolutional layers in both generator and discriminator making them suitable for image generation tasks*

➢ *Conditional GAN*
- *Extends vanilla GANs to include conditional information, such as class labels or additional features, during both training and generation.*

➢ *Cycle GAN*
- *Introduces cycle consistency loss to enable unsupervised image-to-image translation between two domains without paired data.*

➢ *Style GAN*
- *Introduces a style-based generator architecture that decouples the latent space from the generator network, allowing for fine-grained control over image attributes.*

➢ *Big GAN*
- *Scale up GAN architecture to generate high-quality images at higher resolutions.*

➢ *Progressive GAN*
- *Proposes a progressive growing training strategy where both the generator and discriminator grow progressively in resolution during training.*

➢ *Self-attention GAN*
- *Incorporates self-attention mechanisms into the GAN architecture to capture long-range dependencies and improve image quality.*

# *Diffusion models*

***What it is***

*Diffusion models belong to the class of autoregressive generative models. The key principle behind diffusion models is to transform a simple initial distribution (e.g., Gaussian noise) into the target data distribution through a series of transformations.*

***Application***

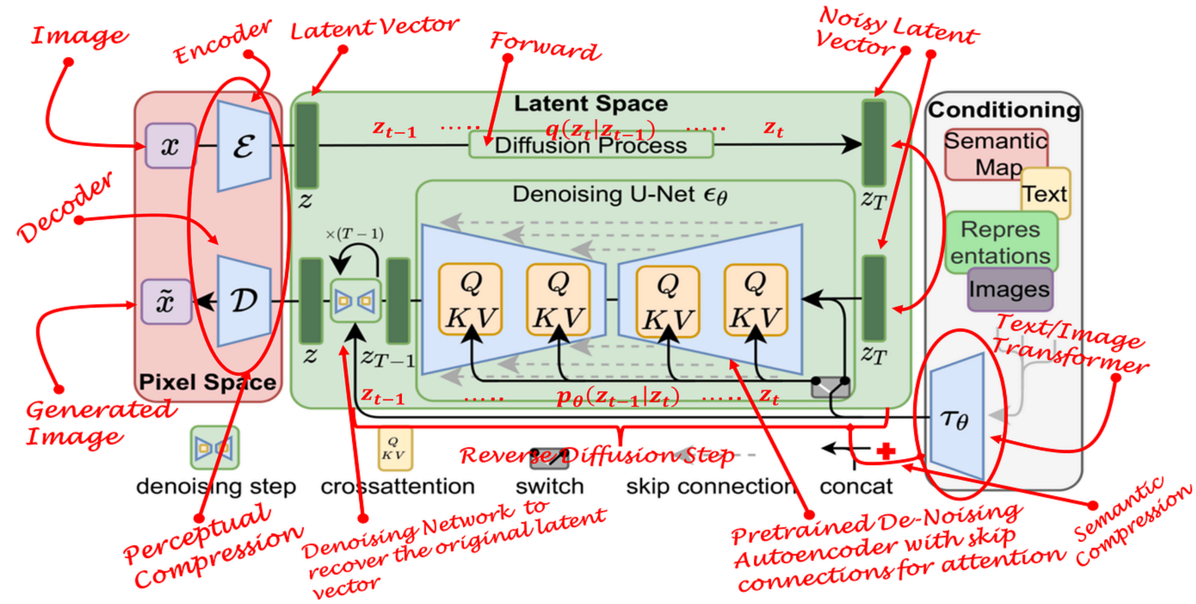
➢ *Image Generation: Generating high-quality images with diverse content and styles.*

➢ *Image Denoising: Removing noise from corrupted images while preserving essential details.*

➢ *Image Inpainting: Filling in missing or occluded regions of images.*

➢ *Super-Resolution: Upscaling low-resolution images to higher resolutions while maintaining image quality.*

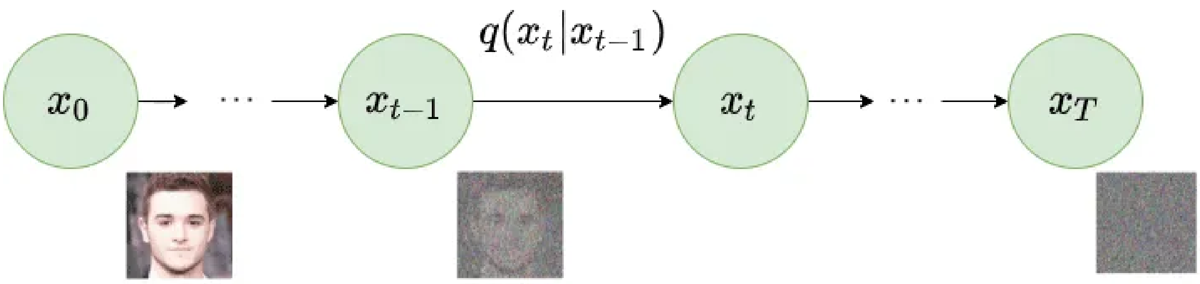
***How it works***



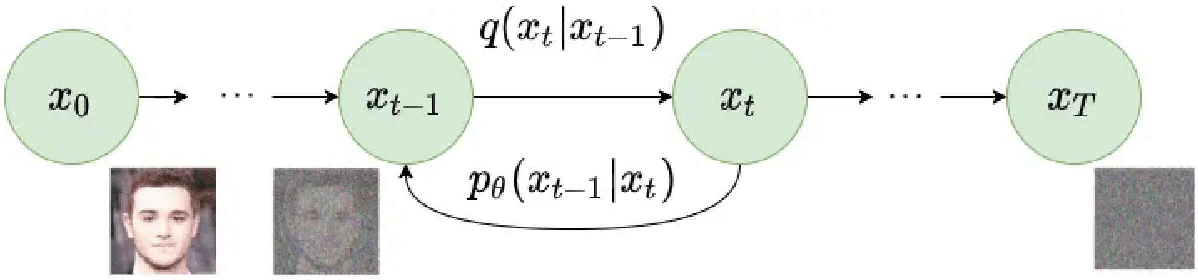

**am gonna explain each point, hope it will make sense at the end*

- *Mainly has 2 parts: forward diffusion and reverse diffusion.*
- *In forward diffusion, the image is corrupted by gradually introducing noise until the image becomes completely random noise*
- *In the reverse process, a series of Markov chains are used to recover the data from the Gaussian noise by gradually removing the predicted noise at each time step*

- *There are a lot of diffusion models out there. The very recently proposed method leverages the perceptual power of GAN. This model is known as the Latent Diffusion Model*
- *Understand the forward diffusion process in a much better way*
- *In the forward diffusion process, we slowly and gradually add Gaussian noise to the data input image x0 through a series of T steps*




- *After that phase, we have to recover the image and have to learn from it*
- *This step is known as the reverse diffusion process.*
- *It is the process of training a neural network to recover the original data by reversing the noising process applied in the forward pass.*




- ➢ *The training process of diffusion models is to find the reverse Markov transitions that maximize the likelihood of the training data*
- ➢ *In practice, training consists of minimizing the variational upper bound on the negative log-likelihood*

***Advantages***

- ➢ *Better Handling of Out-of-Distribution Samples*
- ➢ *tend to have more stable training*
- ➢ *Interpretable noise level*
- ➢ *Likelihood-based training*

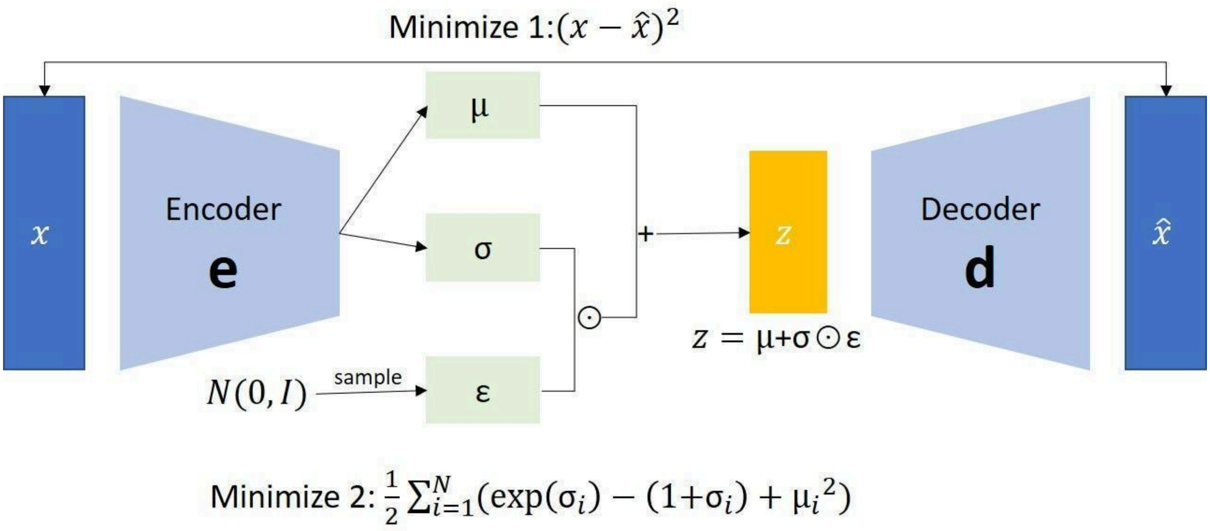
***Disadvantages***

- ➢ *Slower sampling and computationally expensive*
- ➢ *Limited scalability when dealing with high-dimensional data*
- ➢ *Less establishment compared to the GAN*

***types***

- ➢ *Autoregressive Diffusion Model*
- ➢ *Continous diffusion model*
- ➢ *Discrete diffusion model*
- ➢ *Variational diffusion model*
- ➢ *Hierarchical diffusion model*

## ***VAE - variational Auto Encoder***




***What it is***

*It is a neural network-based approach to probabilistic modeling and is widely used in applications such as image generation, data compression, and representation learning.*

***How it works***

*A Variational Autoencoder is composed of two main components: an encoder and a decoder.*

- *Encoder → The encoder network takes input data and maps it to a latent space representation. This latent space is typically of lower dimensionality compared to the input data.*
- *Decoder: The decoder network takes samples from the latent space (usually drawn from the distribution outputted by the encoder) and reconstructs the input data from these samples.*

*Some more details …*

- *The VAE aims to maximize the evidence lower bound (ELBO), which is a lower bound on the log-likelihood of the data*
- *Reconstruction Loss → Measures the difference between the input data and the data reconstructed by the decoder.*
- *KL Divergence Regularization Term: This term ensures that the learned latent space is close to a prior distribution (usually a unit Gaussian distribution).*
- *In simple terms, the "latent space" refers to a hidden space where meaningful features or patterns of data are encoded.*
- *Latent space is like a simplified, abstract representation of the data where each point in this space represents a different "version" of a cat.*

***Advantages***

- *Generative Modeling:*
  - *VAEs are generative models capable of generating new data samples from a learned latent space distribution.*
- *Probabilistic Framework:*
  - *VAEs are built on a probabilistic framework, allowing them to model complex data distributions and capture uncertainty in the data.*
- *End-to-End Training:*

    - *VAEs can be trained end-to-end using backpropagation and stochastic gradient descent, making them easy to implement and train.*
- ➢ *Regularization:*
    - *The VAE objective includes a regularization term that encourages the learned latent space to follow a prior distribution, such as a Gaussian distribution.*

***Disadvantages***

- ➢ *Blurry Reconstructions:*
    - *VAEs often produce blurry reconstructions of input data, especially for high-resolution images.*
- ➢ *Posterior Collapse:*
    - *VAEs are prone to a phenomenon known as posterior collapse, where the model ignores the latent variable and relies solely on the decoder network to generate samples.*
- ➢ *Limited Expressiveness:*
    - *The latent space learned by VAEs may not capture all the variations present in the data, especially if the data distribution is highly complex or non-linear.*
- ➢ *Difficulty in Latent Space Interpretation:*
    - *While VAEs learn a latent space representation of the data, interpreting individual dimensions of the latent space can be challenging, especially when the dimensions are entangled or not easily interpretable.*