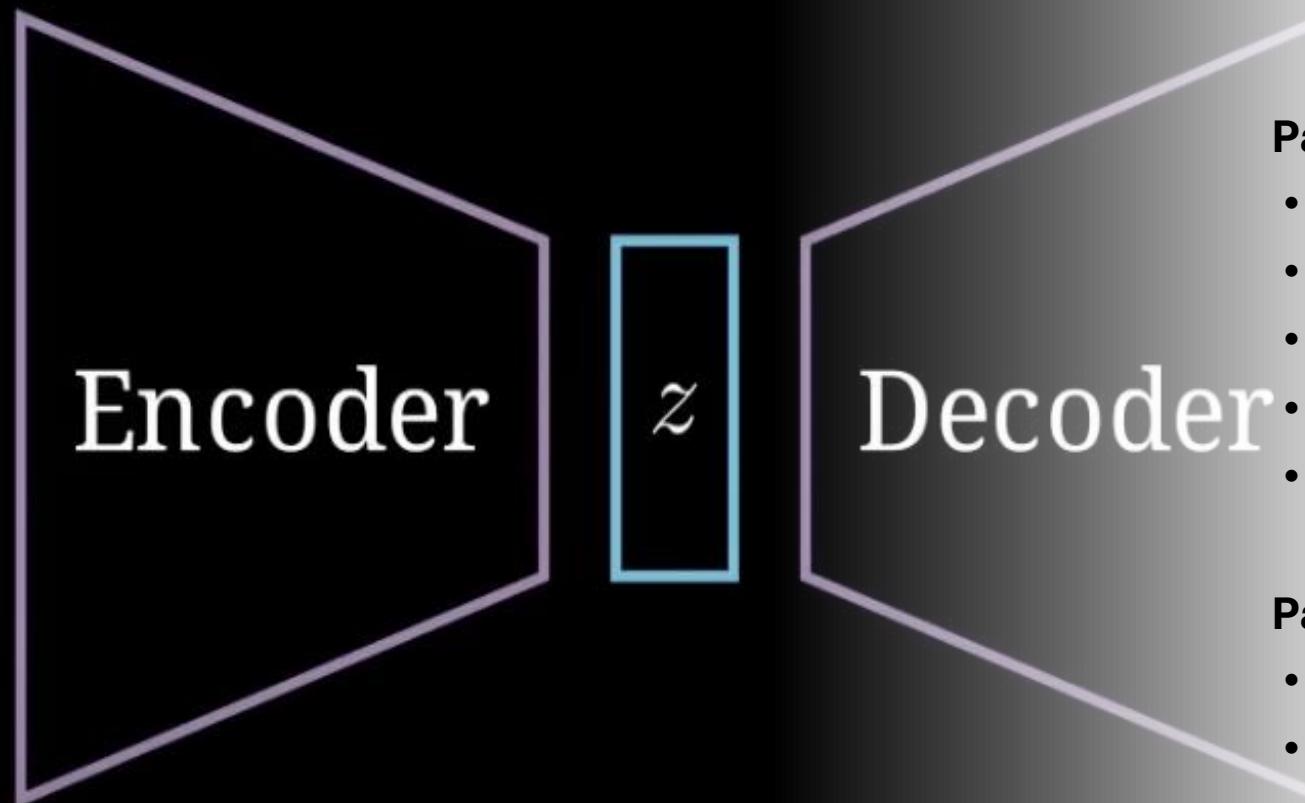


Autoencoders (Basic) & Variational autoencoders (VAE)

Sanjaya Edirisinghe
Sanjaya@bu.edu

AGENDA



Part 1 – Basic Autoencoders

- Why Autoencoders and the use of it
- Architecture of Autoencoders
- Components of the Autoencoders
- Information loss of Autoencoders
- Applications of Autoencoders

Part 2 - Variational Autoencoders (VAEs)

- Why VAEs and their purpose
 - Architecture of VAEs
 - Mathematical representation of VAEs
 - How VAE's works
 - Limitations of VAE's
- Different types of Autoencoders

The inventor of Autoencoder – 2013

Diederik P. Kingma



arXiv:1312.6114v1 [stat.ML] 10 Dec 2022

Auto-Encoding Variational Bayes

Diederik P. Kingma
Machine Learning Group
Universiteit van Amsterdam
dpkingma@gmail.com

Max Welling
Machine Learning Group
Universiteit van Amsterdam
welling.max@gmail.com

Abstract

How can we perform efficient inference and learning in directed probabilistic models, in the presence of continuous latent variables with intractable posterior distributions, and large datasets? We introduce a stochastic variational inference and learning algorithm that scales to large datasets and, under some mild differentiability assumptions, even works in the intractable case. Our theoretical contributions are two-fold. First, we show that a reparameterization of the variational lower bound yields a lower bound estimator that can be straightforwardly optimized using standard stochastic gradient methods. Second, we show that for i.i.d. datasets with continuous latent variables per datapoint, posterior inference can be made especially efficient by fitting an approximate inference model (also called a ‘recognition model’) to the intractable posterior using the proposed lower bound estimator. Theoretical advantages are reflected in experimental results.

1 Introduction

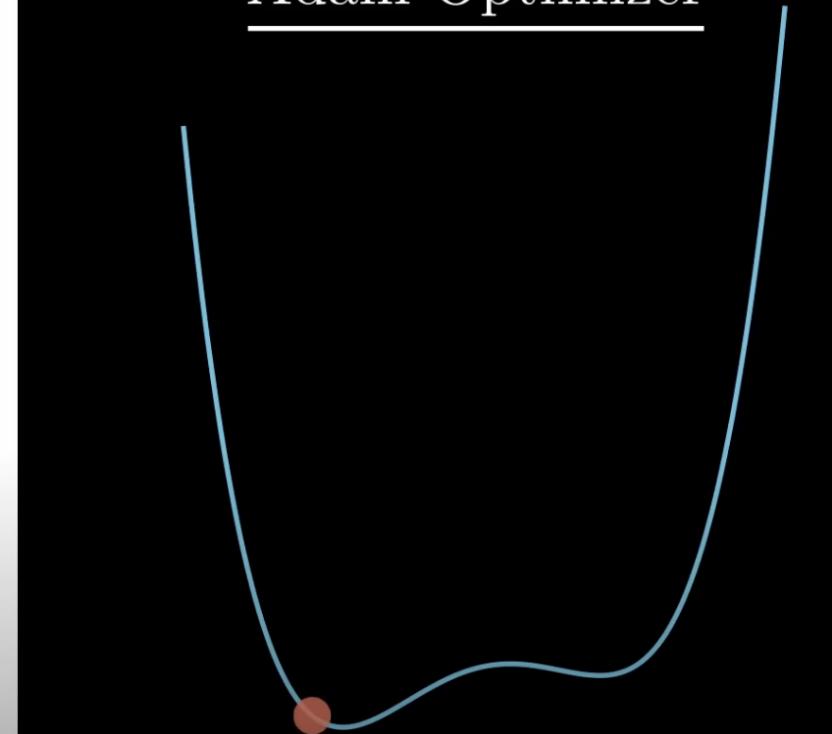
How can we perform efficient approximate inference and learning with directed probabilistic models whose continuous latent variables and/or parameters have intractable posterior distributions? The variational Bayesian (VB) approach involves the optimization of an approximation to the intractable posterior distribution. Specifically, the VB approach requires the computation of expectations w.r.t. the approximate posterior, which are also intractable in the general case. We show how a reparameterization of the variational lower bound yields a simple differentiable unbiased estimator of the lower bound; this SGVB (Stochastic Gradient Variational Bayes) estimator can be used for efficient approximate posterior inference in almost any model with continuous latent variables and/or parameters, and is especially useful to minimize standard stochastic gradient ascent techniques.

For the case of an i.i.d. dataset of continuous latent variable data, we propose the Auto-Encoding Variational Bayes (AEVB) algorithm. In the AEVB algorithm we make inference and learning especially efficient by using the SGVB estimator to optimize a recognition model that allows us to perform very efficient approximate posterior inference using simple ancestral sampling, which in turn allows us to efficiently learn the model parameters, without the need of expensive iterative inference schemes (such as MCMC) per datapoint. The learned approximate posterior inference model can also be used for a host of tasks such as recognition, denoising, representation and visualization purposes. When a neural network is used for the recognition model, we arrive at the *variational auto-encoder*.

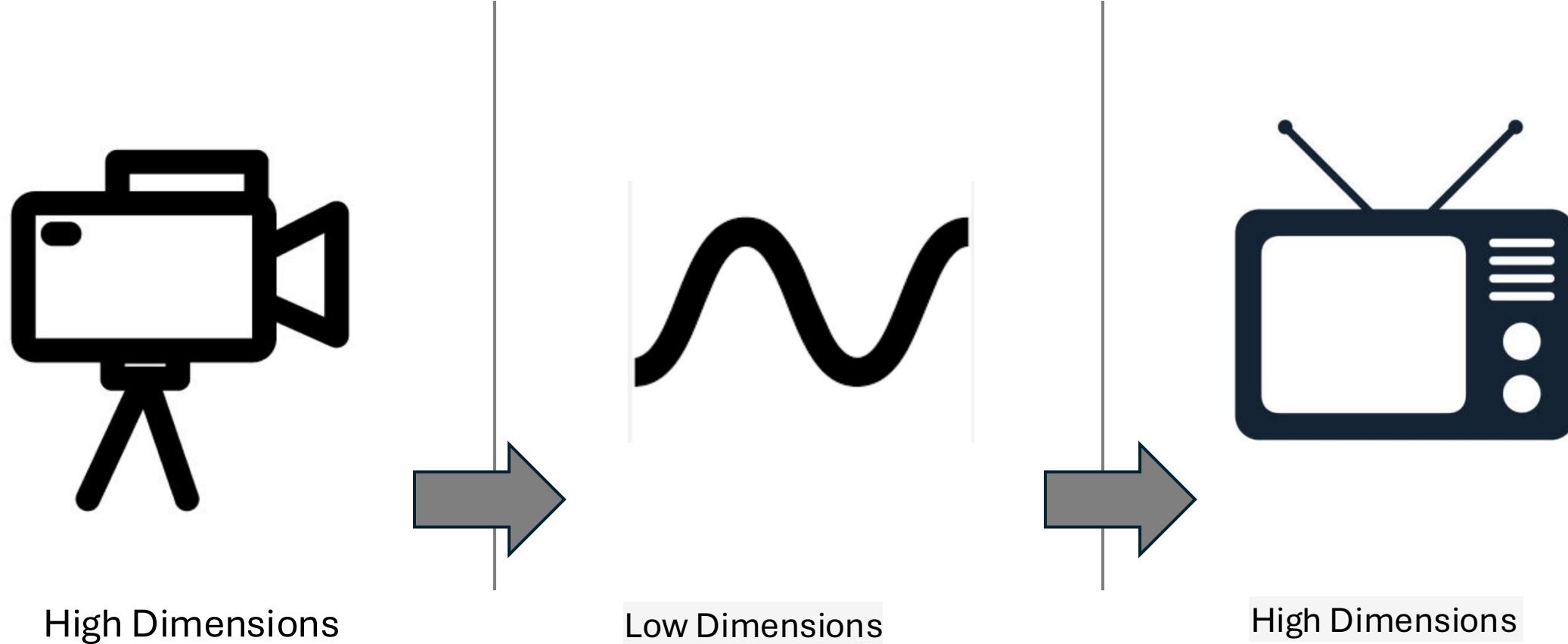
2 Method

The strategy in this section can be used to derive a lower bound estimator (a stochastic objective function) for a variety of directed graphical models with continuous latent variables. We will restrict ourselves here to the common case where we have an i.i.d. dataset with latent variables per datapoint, and where we like to perform maximum likelihood (ML) or maximum a posteriori (MAP) inference on the (global) parameters, and variational inference on the latent variables. It is, for example,

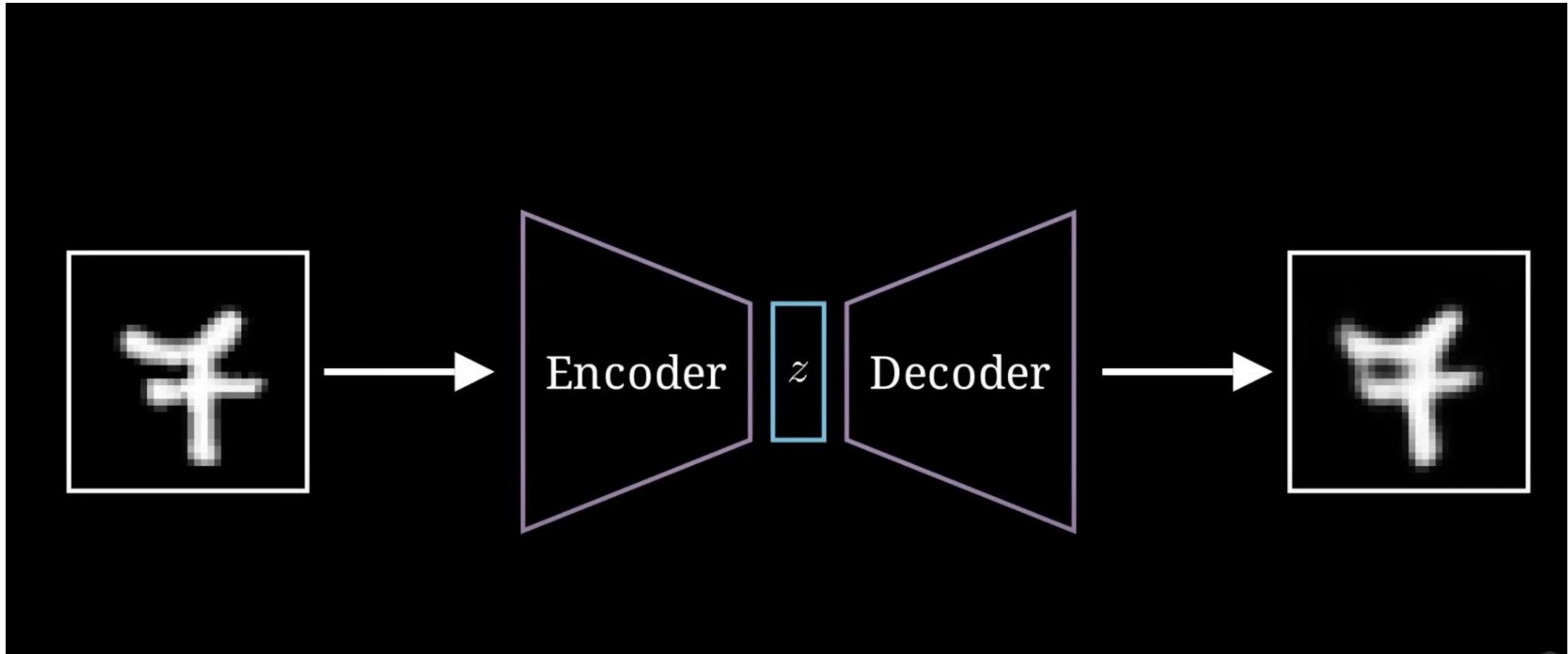
Adam Optimizer



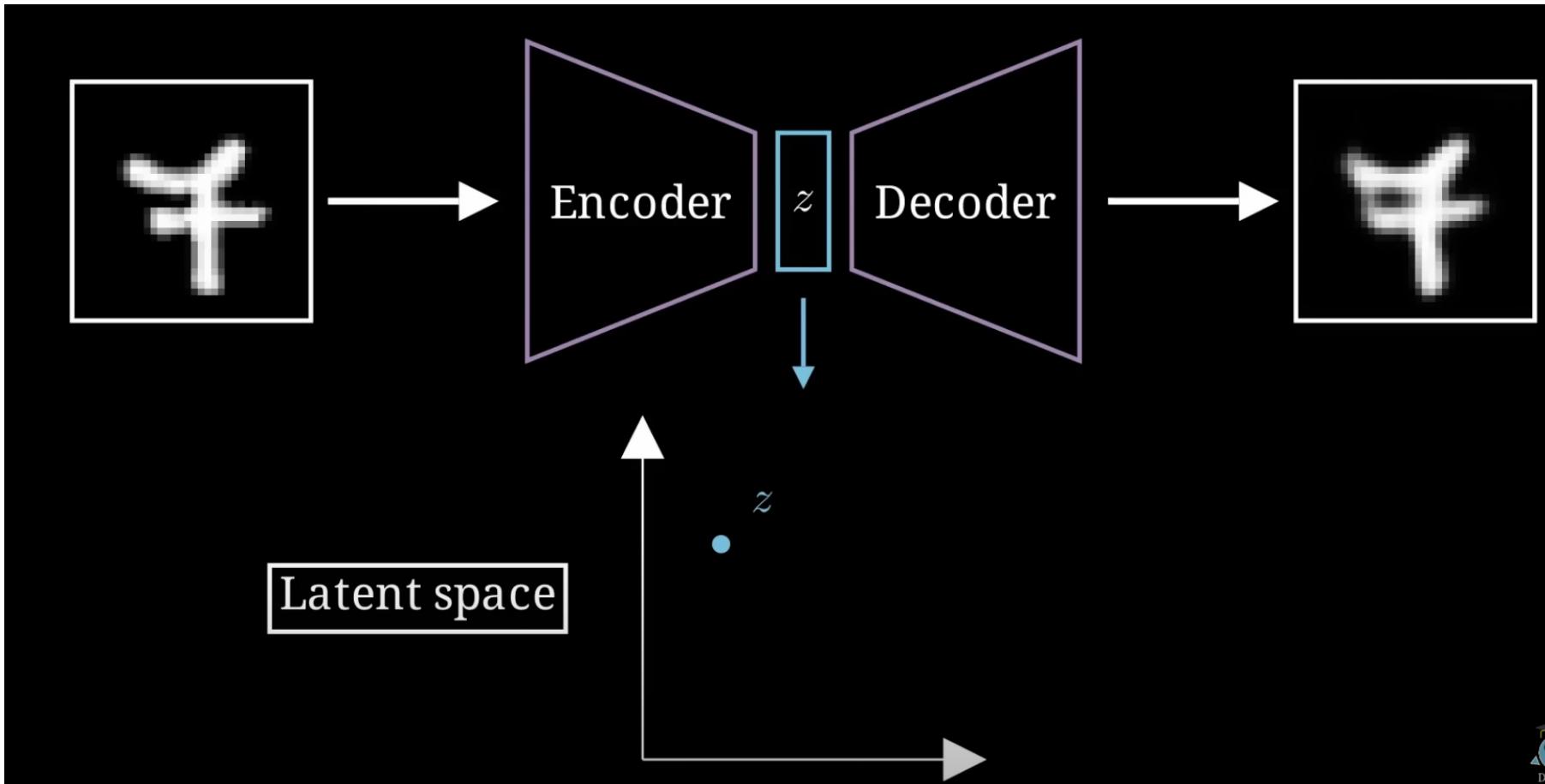
Do you know how TV is working ..



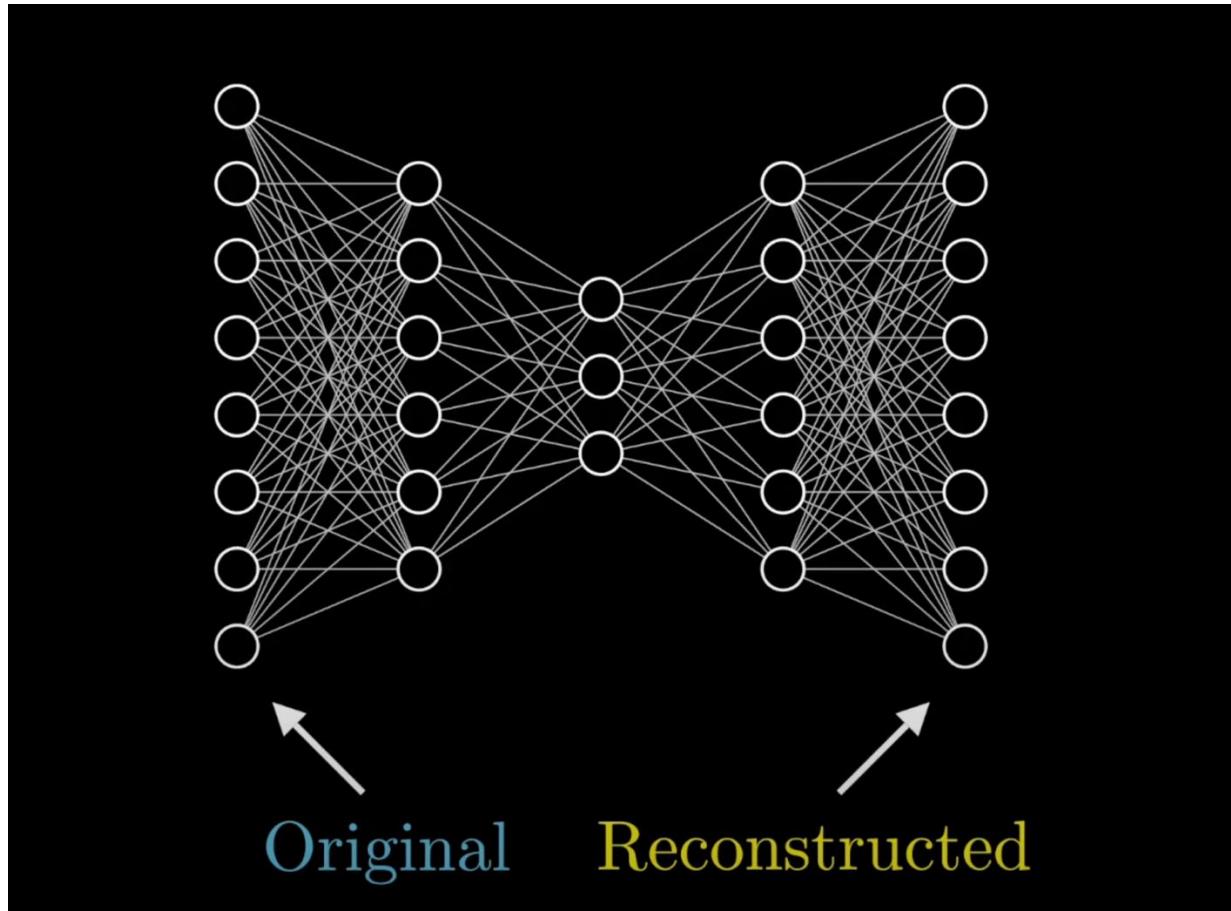
Why Autoencoders and what does it do ?



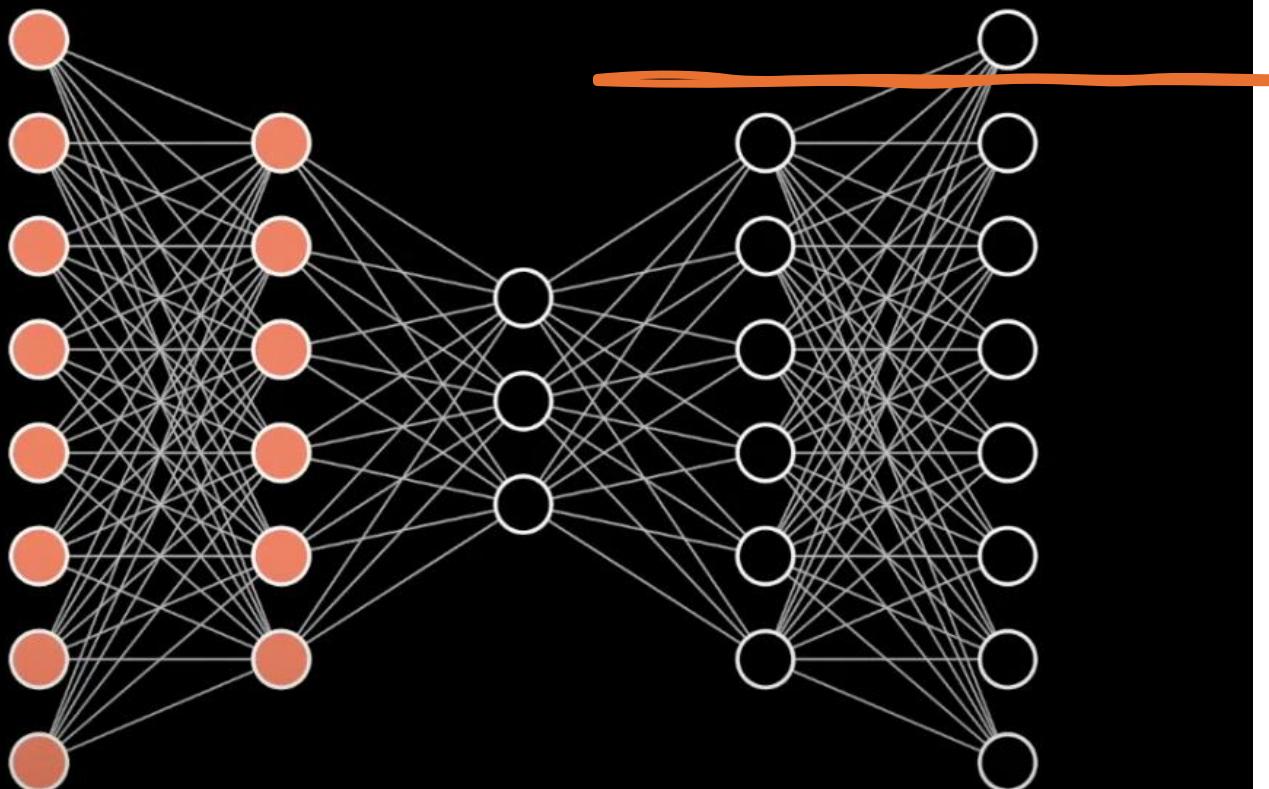
Components of Autoencoders ...



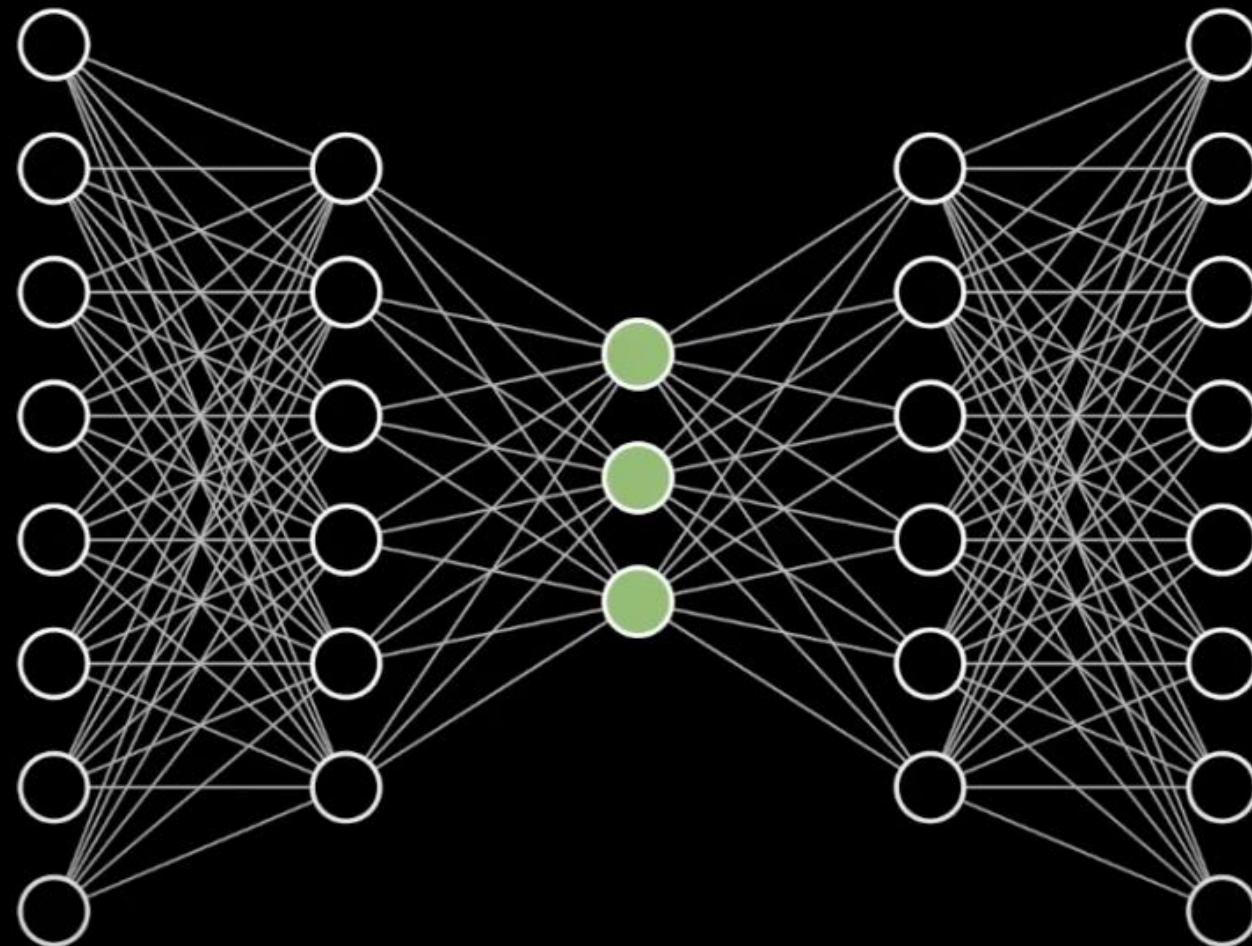
Autoencoders are NN based deeplearning models ..



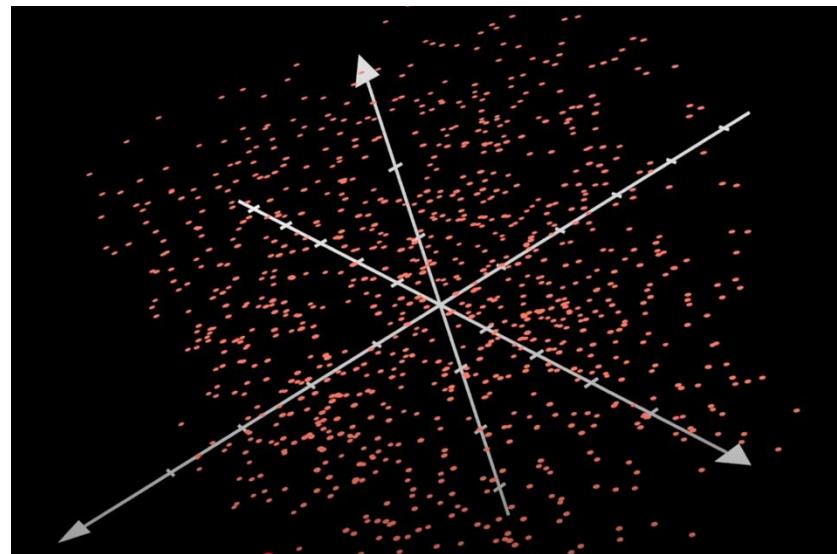
Encoder Network



Embedding Vector

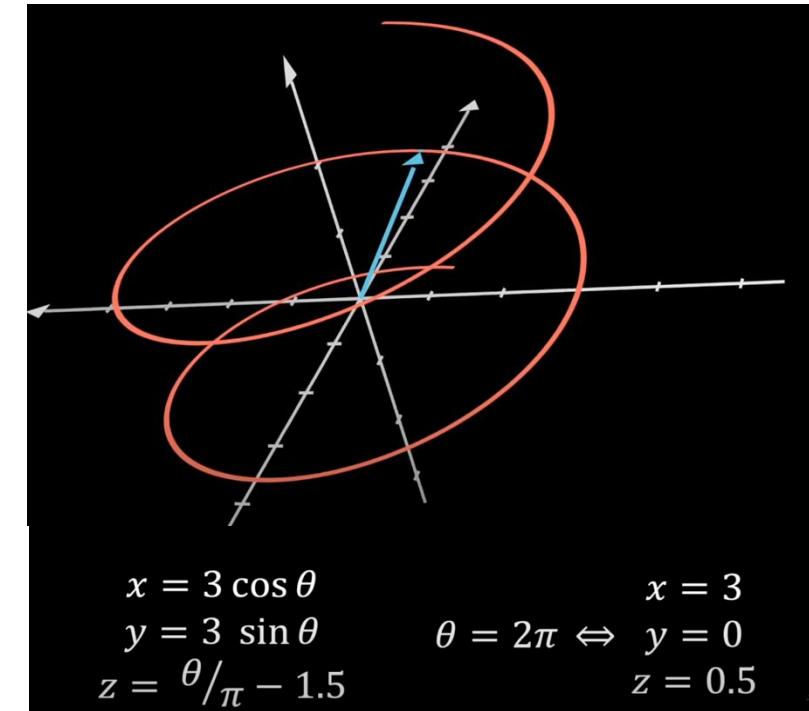


- Bottleneck
- Low Dimensions



Random Distribution

Nature has a structure,
therefore most of the natural
data followed a pattern behind
them

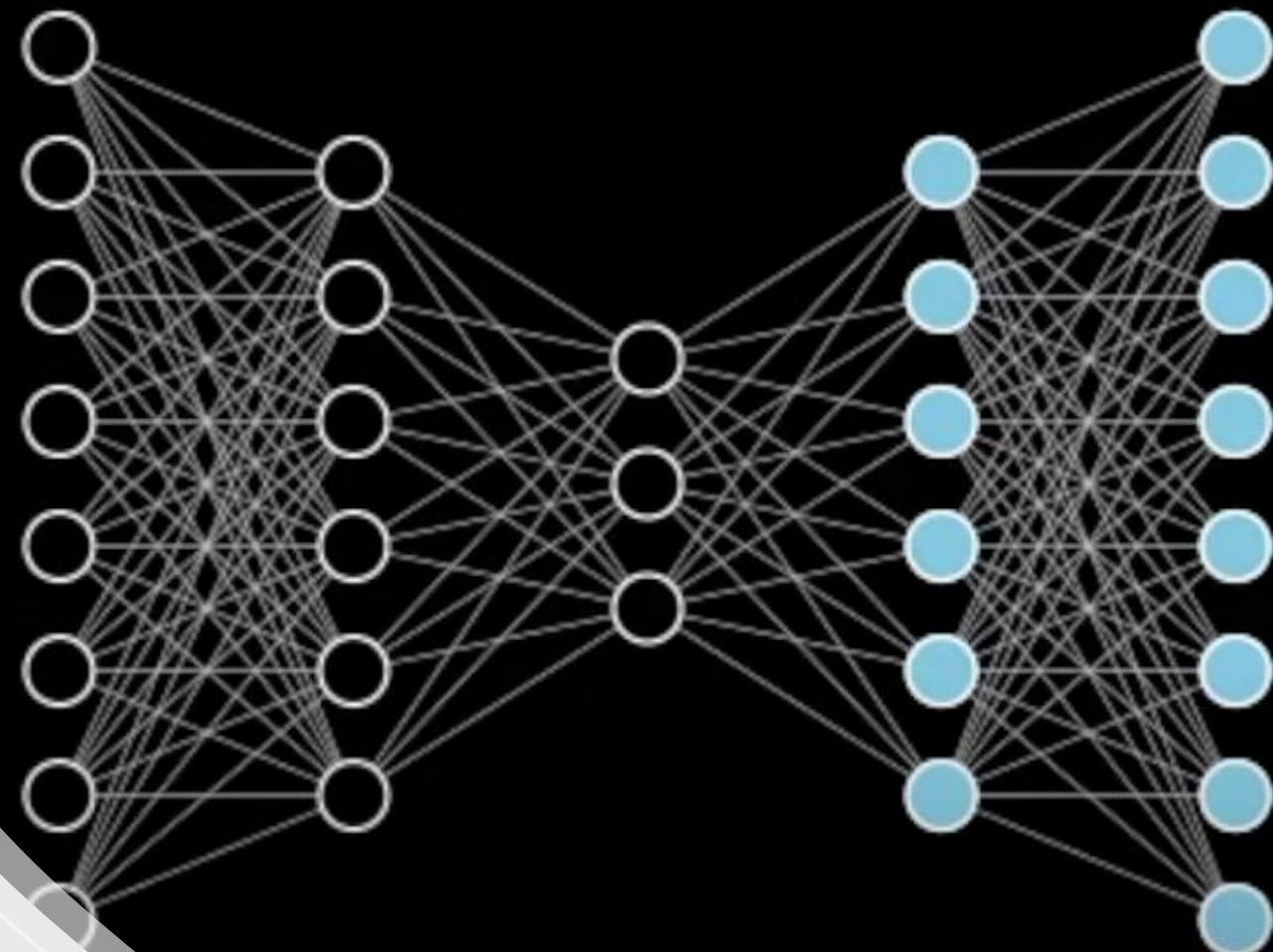


(City, Country)

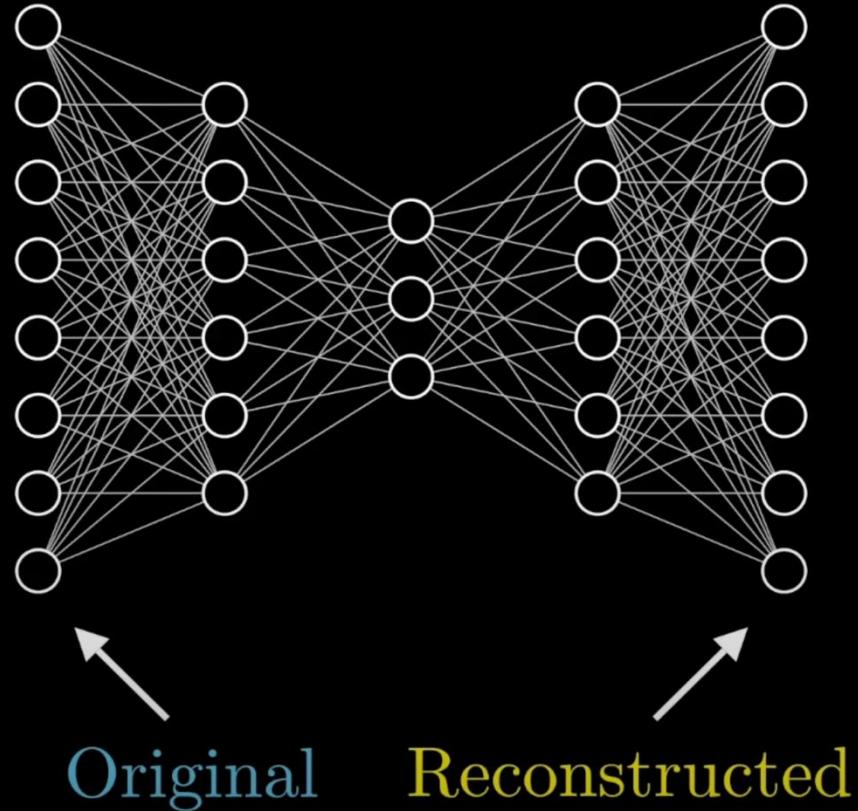
- (Tokyo, Japan)
- (Paris, France)
- (Hong Kong, Spain) ?



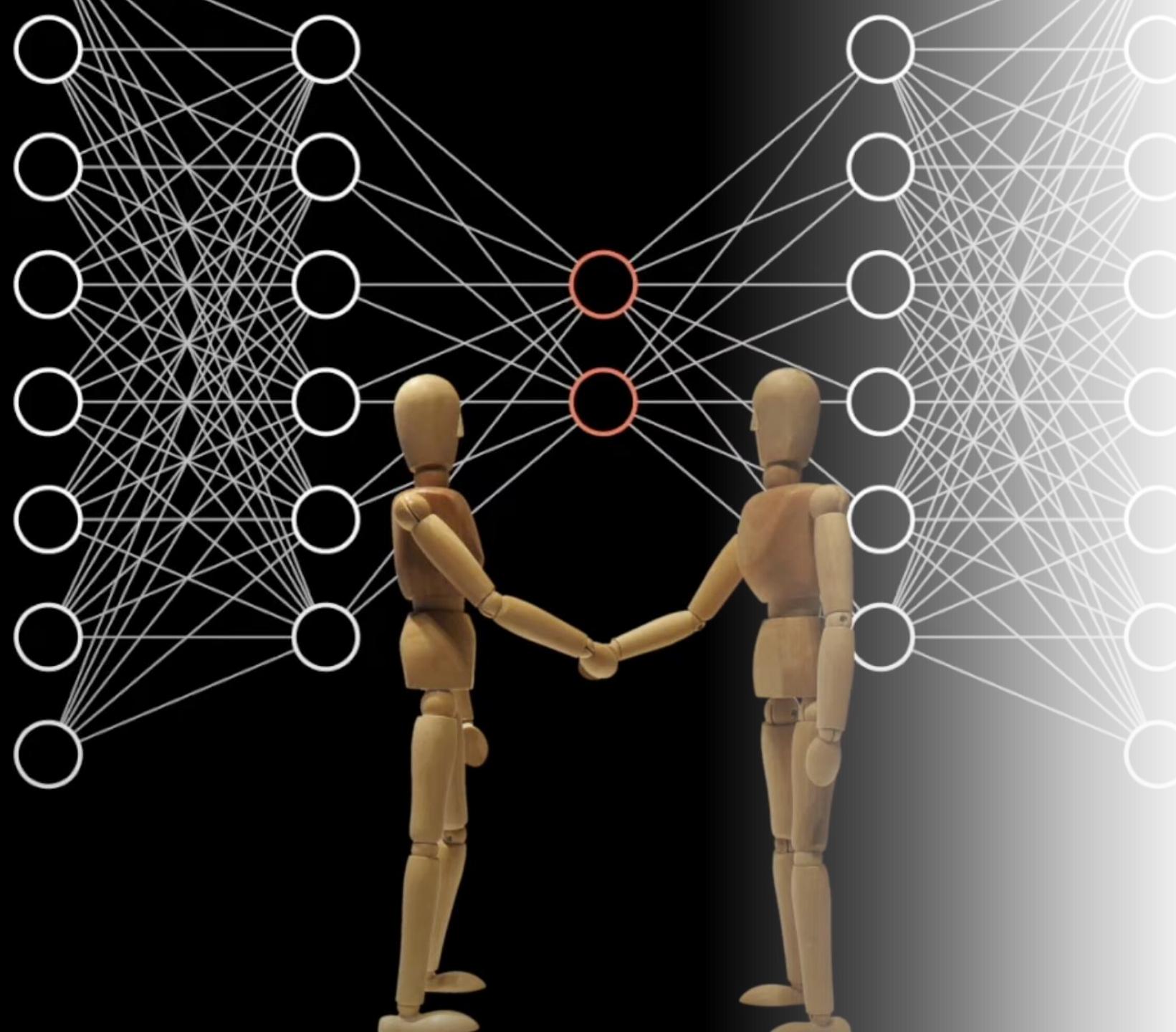
Decoder Network



Reconstruction Error = Reconstructed – Original

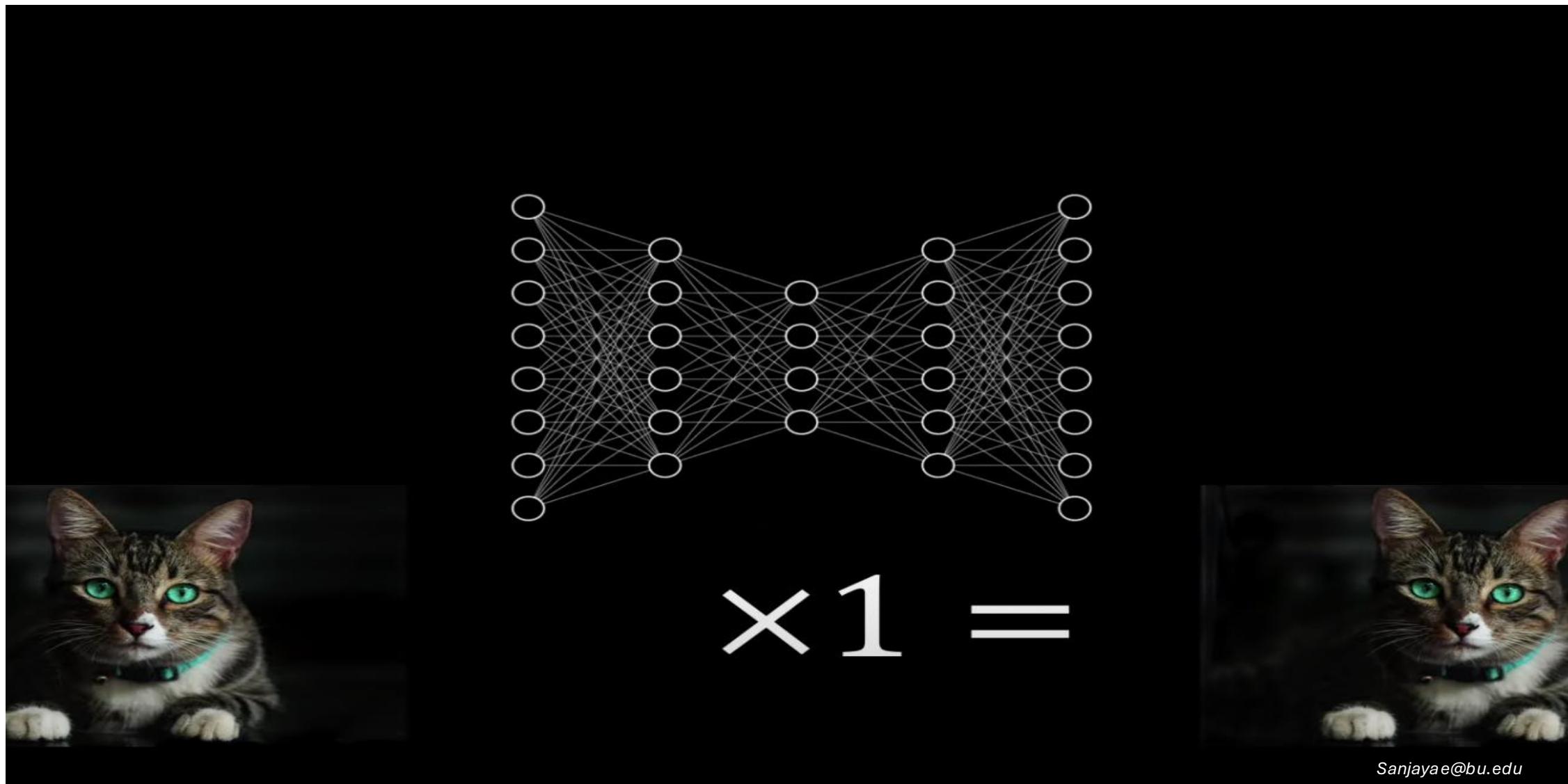


Autoencoders
have
information
losses



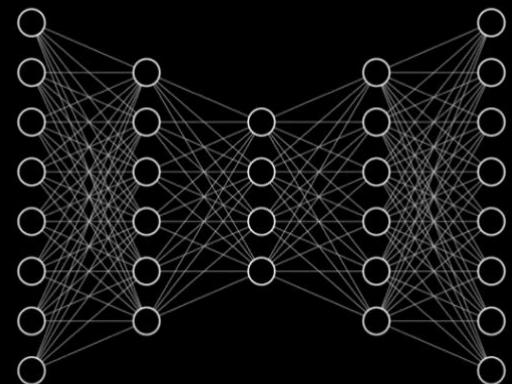
Encoder and
Decoder works
together to
minimize the
information loss

Standard (simple) autoencoders are $\times 1$ pass through system and cannot generate new data



Applications of Autoencoders

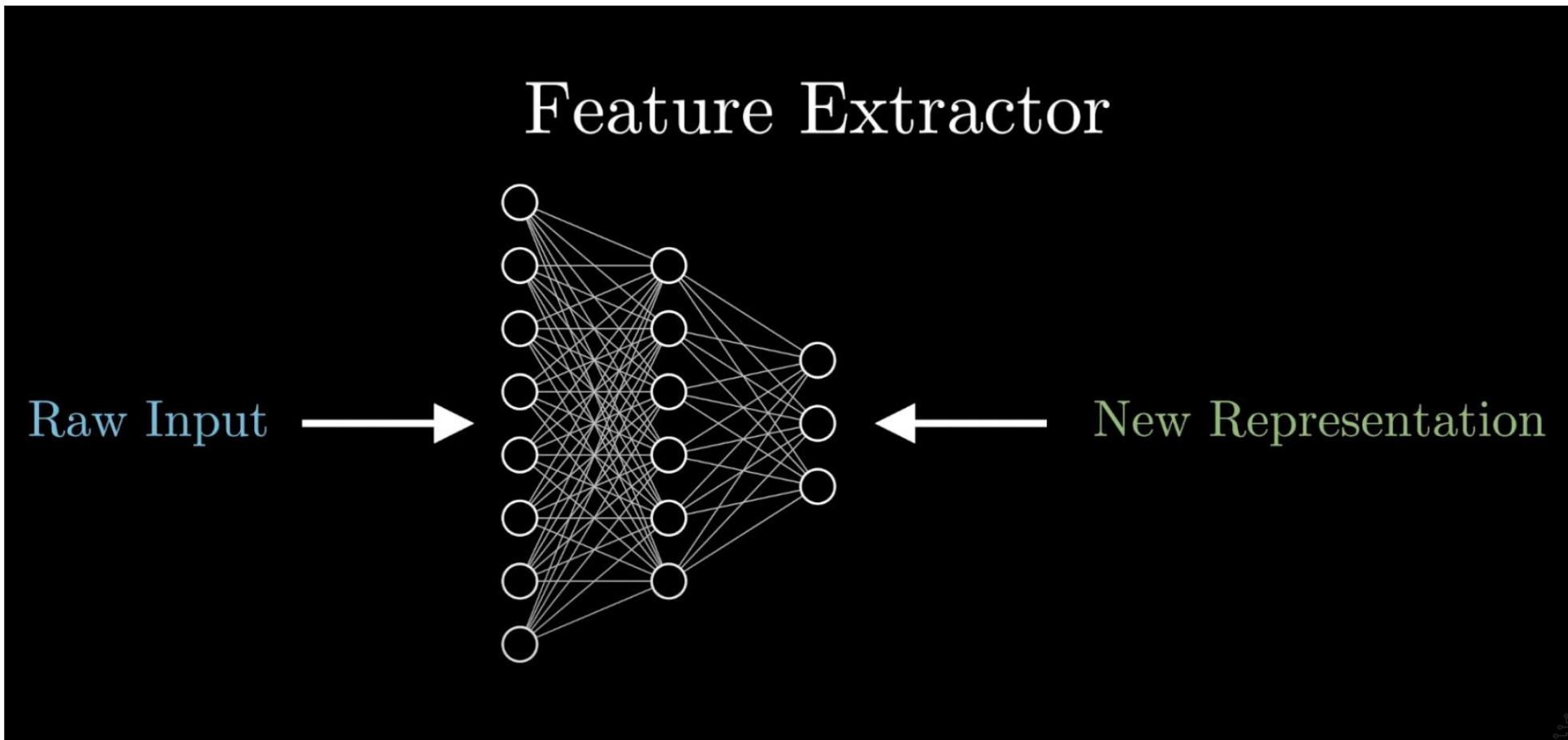
$$Error = Decoder(Encoder(X + noise)) - X$$



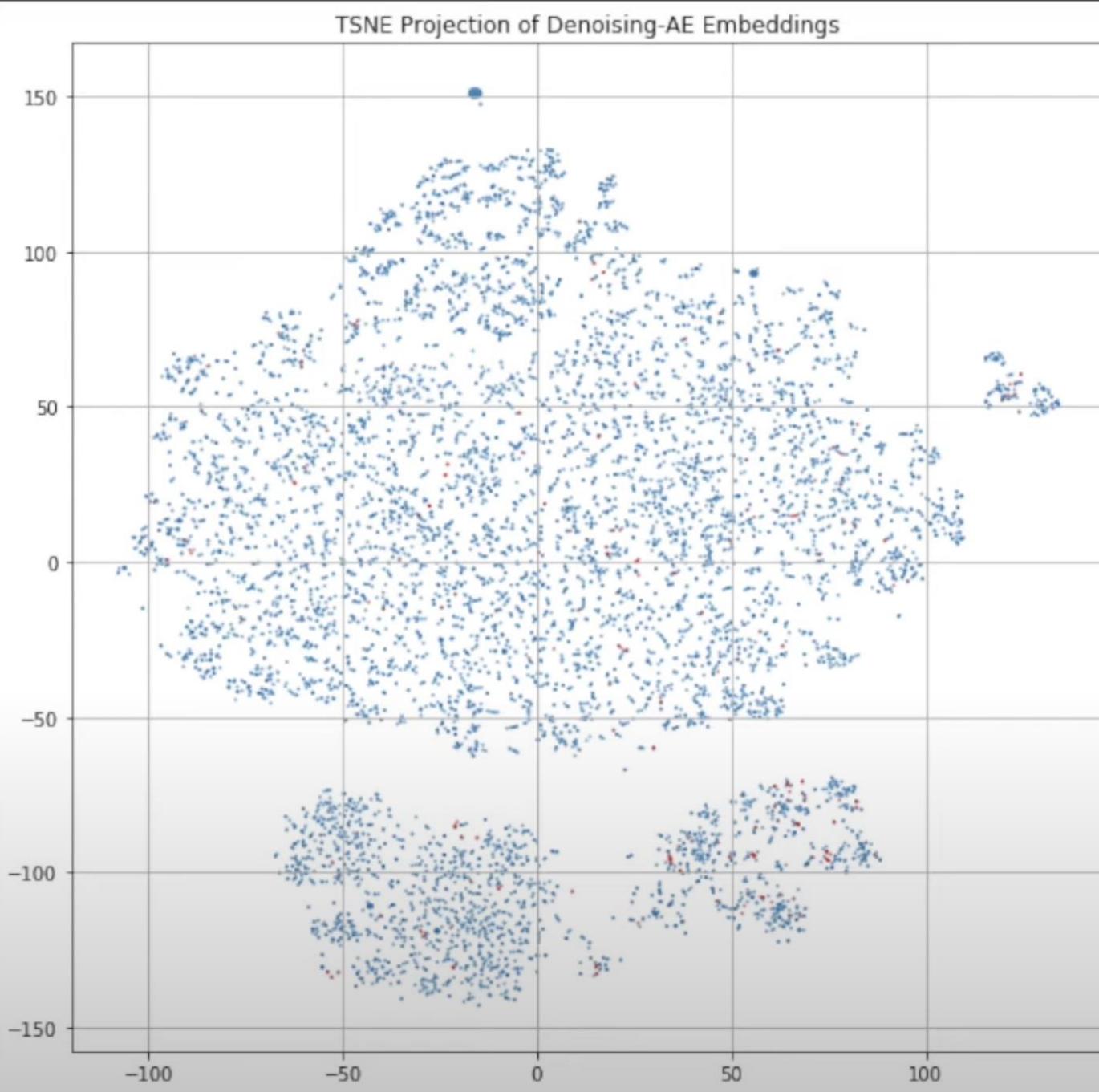
Denoising Autoencoders

- Trained to reconstruct the original input from a **corrupted** version.
- Learns robust features by handling noisy data.
- **Use case:** Noise reduction, data denoising.

Applications of Autoencoders



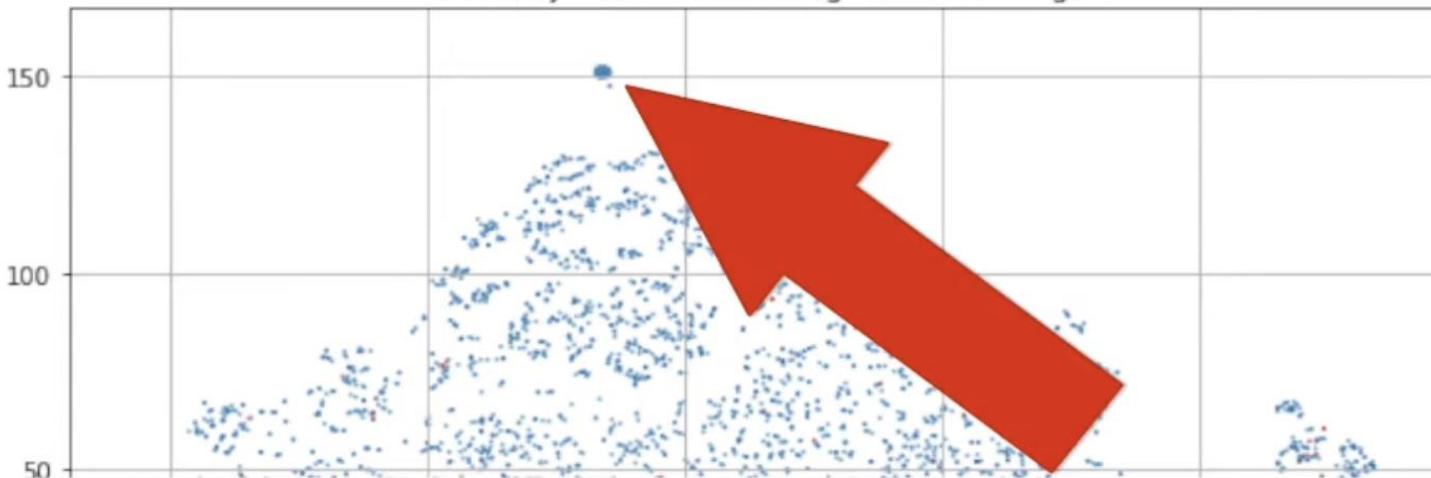
TSNE Projection of Denoising-AE Embeddings



Applications of
Autoencoders

Fraud Detection
Systems

TSNE Projection of Denoising-AE Embeddings



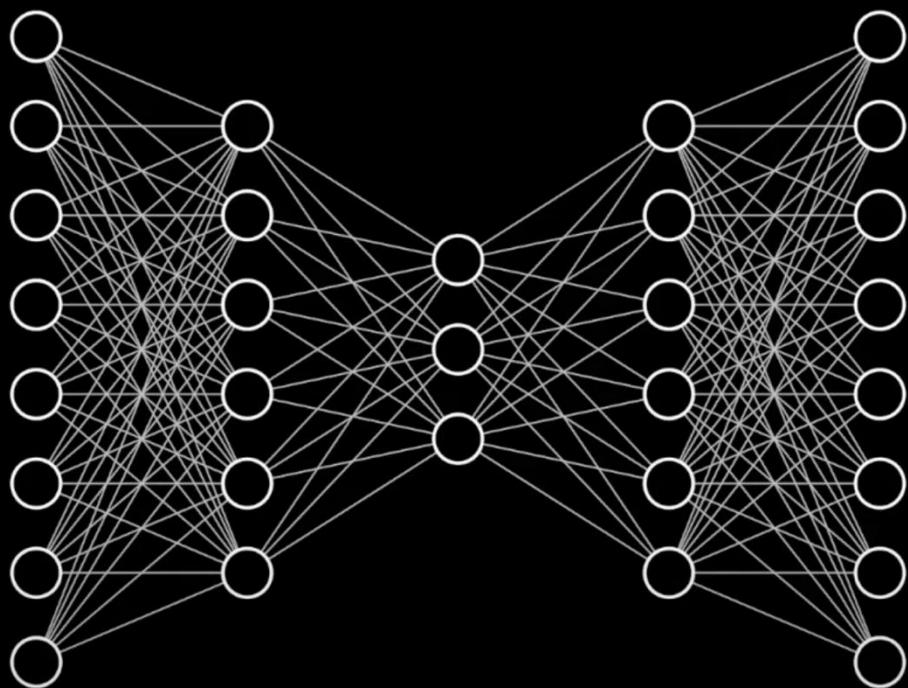
Applications of Auto Encoders

Fraud Detection

	ProductCD	card1	card2	card3	card4	card5	card6	addr1	addr2	P_emaildomain	R_emaildomain	M1	M2	M3	...
940	W	9323	111.0	150.0	visa	226.0	debit	191.0	87.0	charter.net	<UNK>	<UNK>	<UNK>	<UNK>	
977	W	9323	111.0	150.0	visa	226.0	debit	191.0	87.0	charter.net	<UNK>	<UNK>	<UNK>	<UNK>	
999	W	9323	111.0	150.0	visa	226.0	debit	191.0	87.0	charter.net	<UNK>	<UNK>	<UNK>	<UNK>	
059	W	9323	111.0	150.0	visa	226.0	debit	191.0	87.0	charter.net	<UNK>	<UNK>	<UNK>	<UNK>	
114	W	9323	111.0	150.0	visa	226.0	debit	191.0	87.0	charter.net	<UNK>	<UNK>	<UNK>	<UNK>	
385	W	9323	111.0	150.0	visa	226.0	debit	191.0	87.0	charter.net	<UNK>	<UNK>	<UNK>	<UNK>	
405	W	9323	111.0	150.0	visa	226.0	debit	191.0	87.0	charter.net	<UNK>	<UNK>	<UNK>	<UNK>	
	...	---	---	---	---	---	---	---	---	---	---	---	---	---	
	-100	-50	0	50	100										
	-150														

Applications of Autoencoders

Anomaly Detection



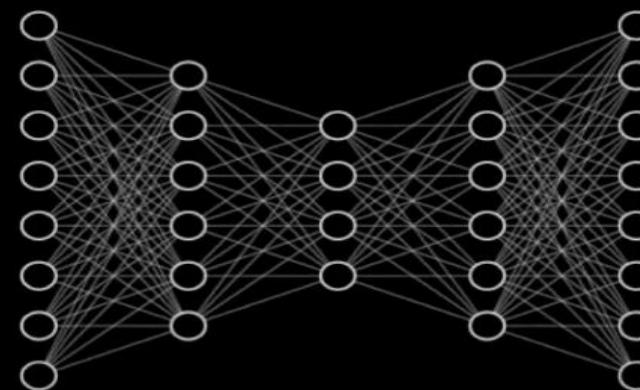
Anomaly Score = Reconstruction Error

Applications of Autoencoders

Missing value
recreation

RU	-15.0	yandex.ru
US	8.3	gmail.com
IN	11.2	rediff.com
RU	<MISSING>	yandex.ru

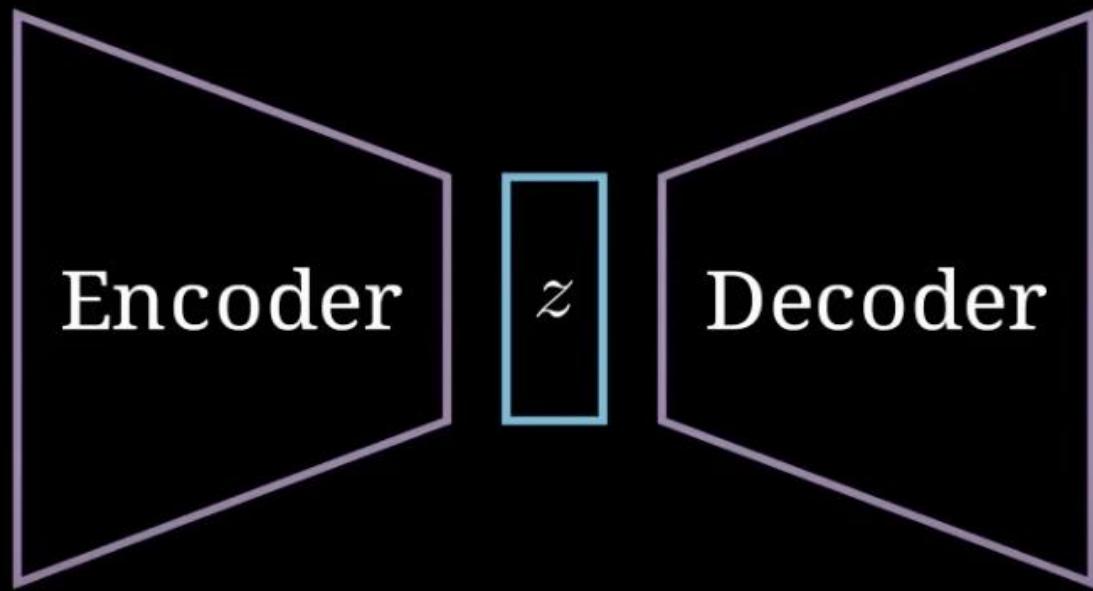
<MISSING>	-15.0	yandex.ru
US	<MISSING>	gmail.com
IN	11.2	<MISSING>



RU	-15.0	yandex.ru
US	8.3	gmail.com
IN	11.2	rediff.com

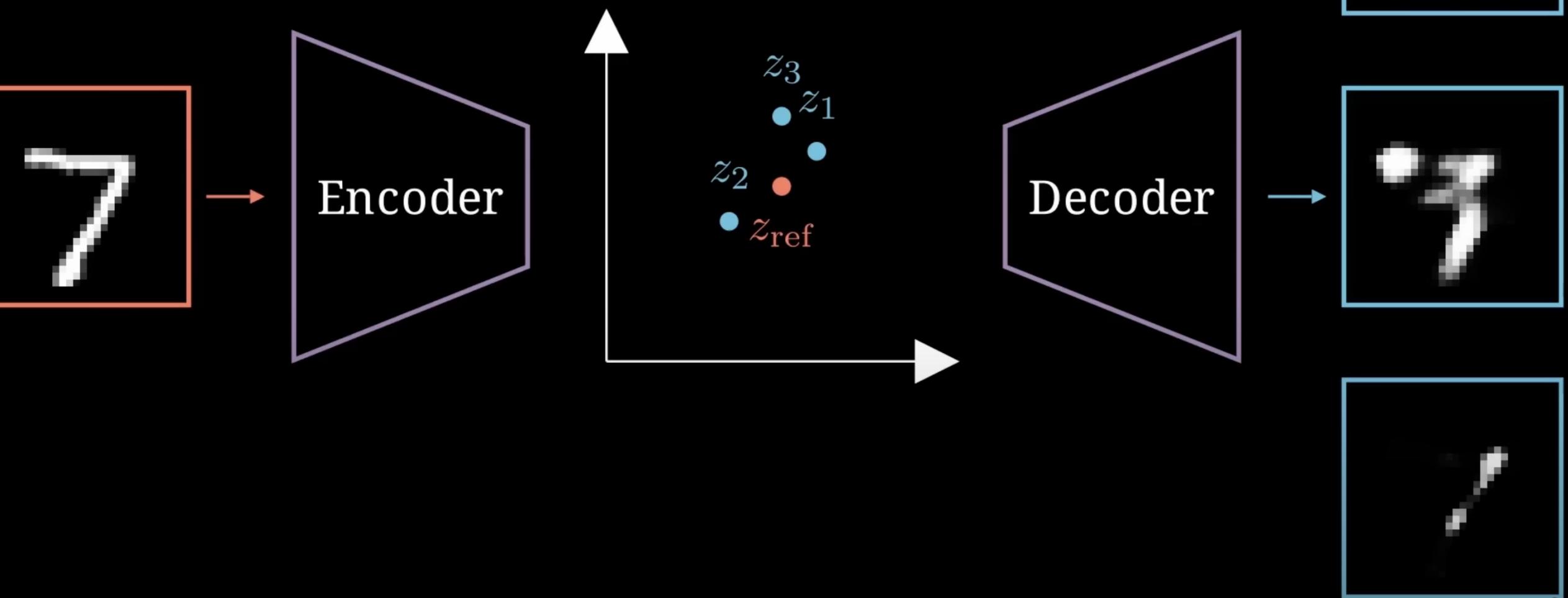
Variational Autoencoder (VAE)

Why Variational Autoencoder (VAE) ?



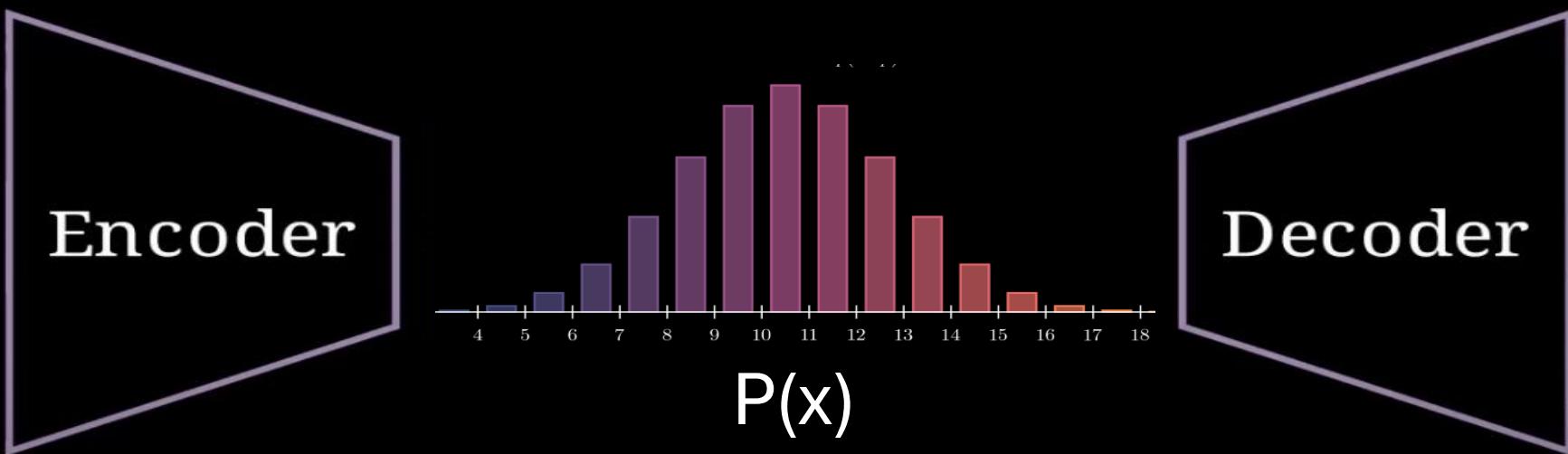
Standard Autoencoders can't generate new data :(

Sampling latent vectors near a reference

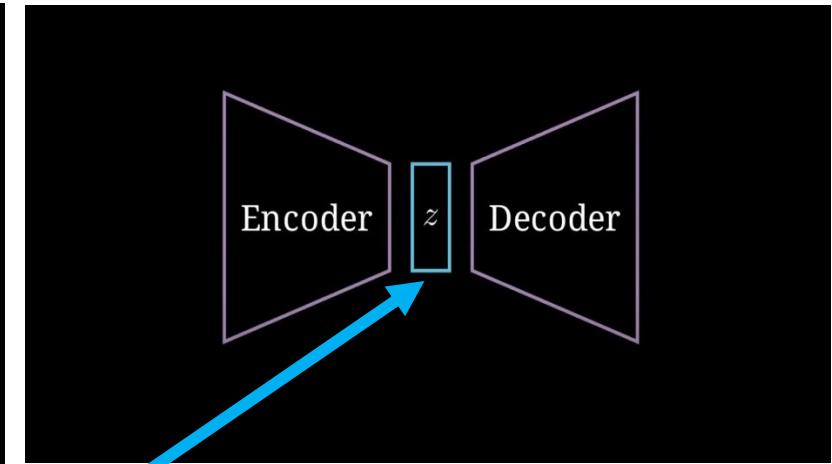
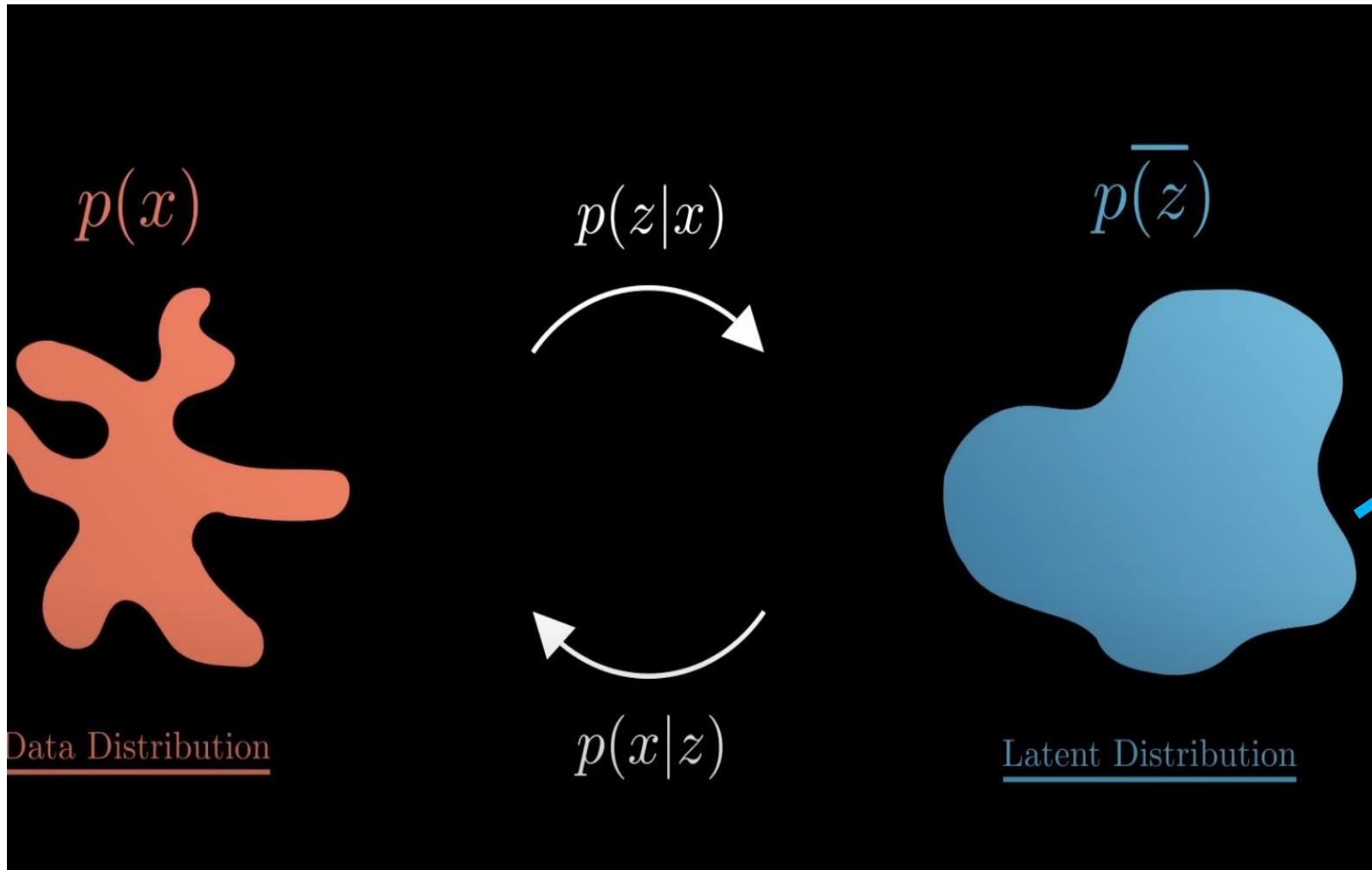


Variational Autoencoder

Variational Autoencoder

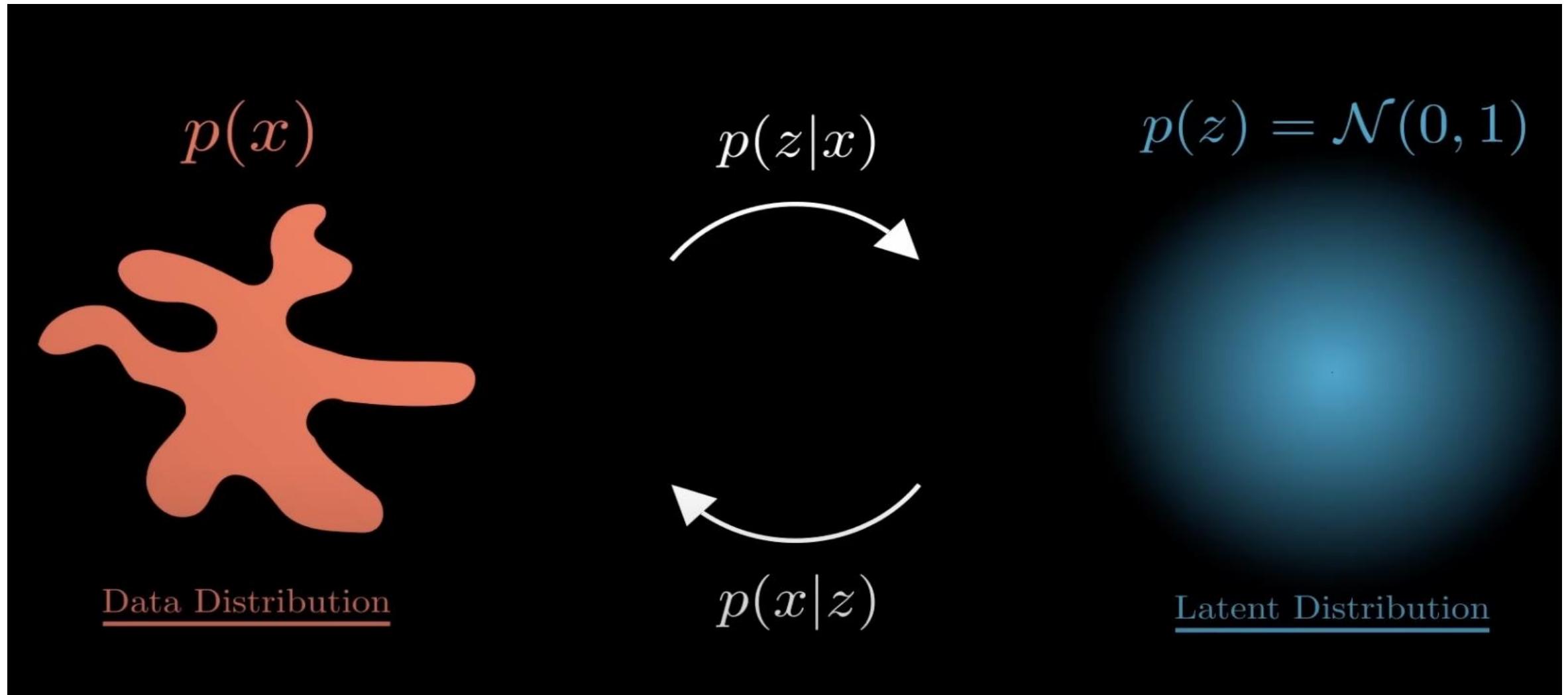


we know the x , so we can calculate the $p(z|x)$ and if we know the $P(x|z)$. We can recreate the image ..
but ...



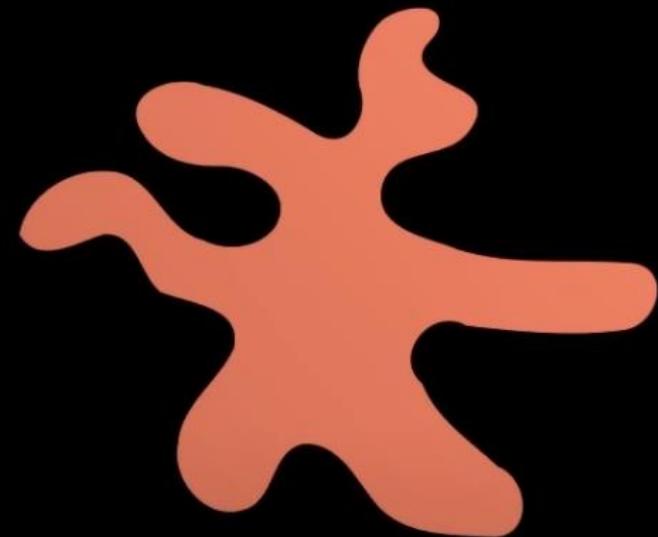
How do we know the probability distribution of $P(z) ???/$

We can assume that $p(z)$ will have a Normal (Gaussian) distribution



We can use the medium and standard deviation of X distribution to create P(z) using training parameter tuning....

$$p(x)$$



Data Distribution

$$q(z|x) = \mathcal{N}(\mu, \sigma)$$



$$p(x|z)$$

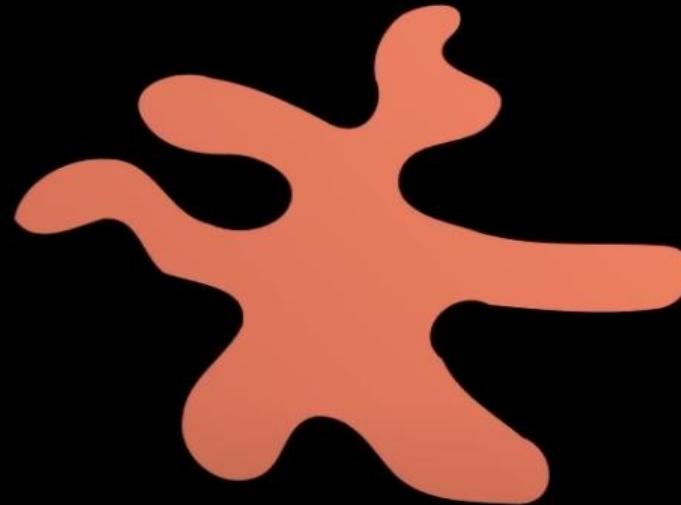
$$p(z) = \mathcal{N}(0, 1)$$

Latent Distribution

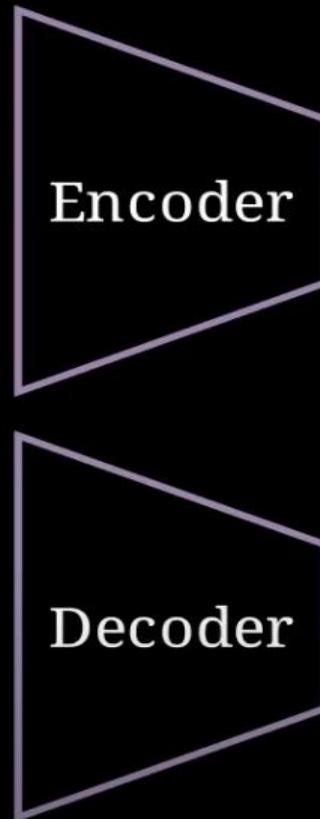
We can use encoder and decoder to train the autoencoder end to end and fine tune the weights (parameters)

Autoencoder Variational Bayes

$$p(x)$$



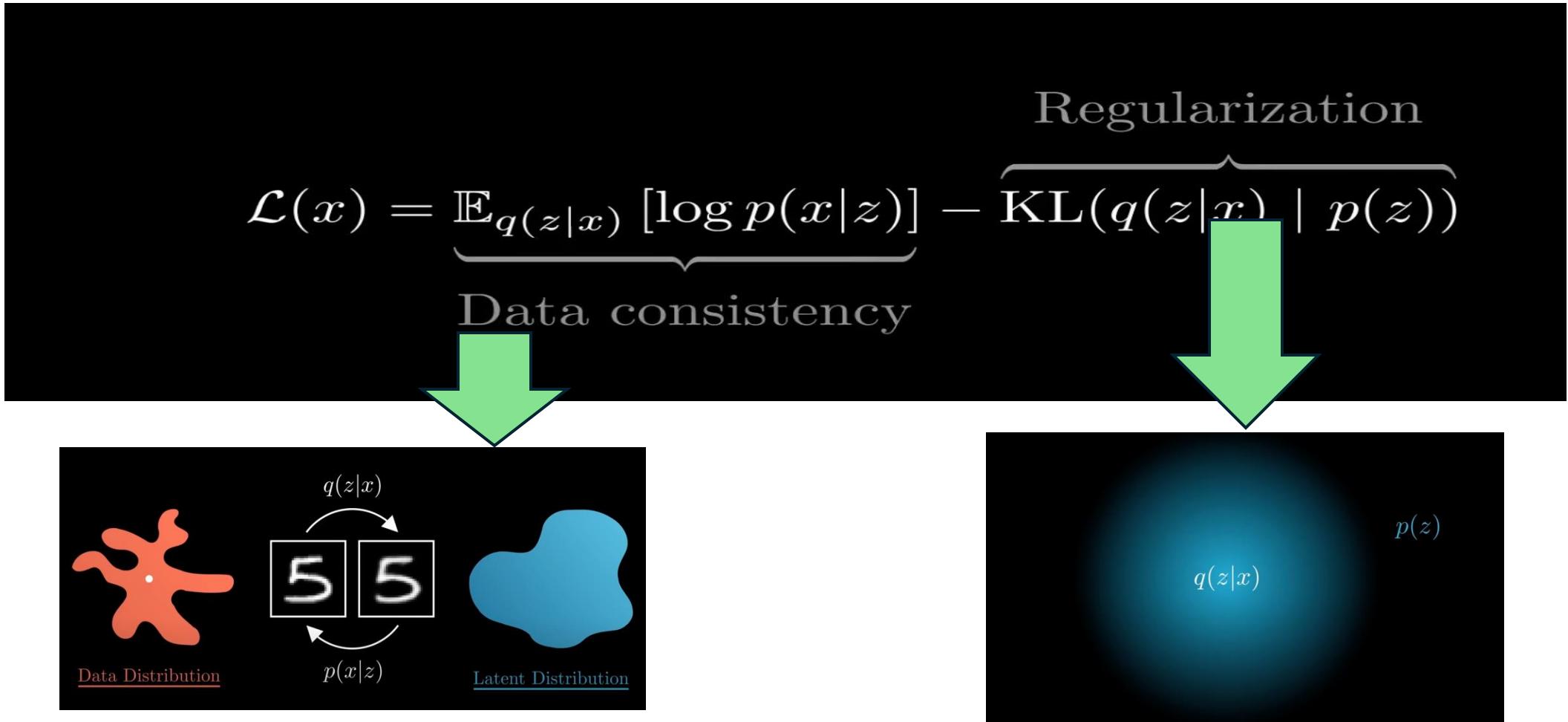
Data Distribution



$$p(z) = \mathcal{N}(0, 1)$$

Latent Distribution

We can represent this process and calculate the loss function with the following equation ..



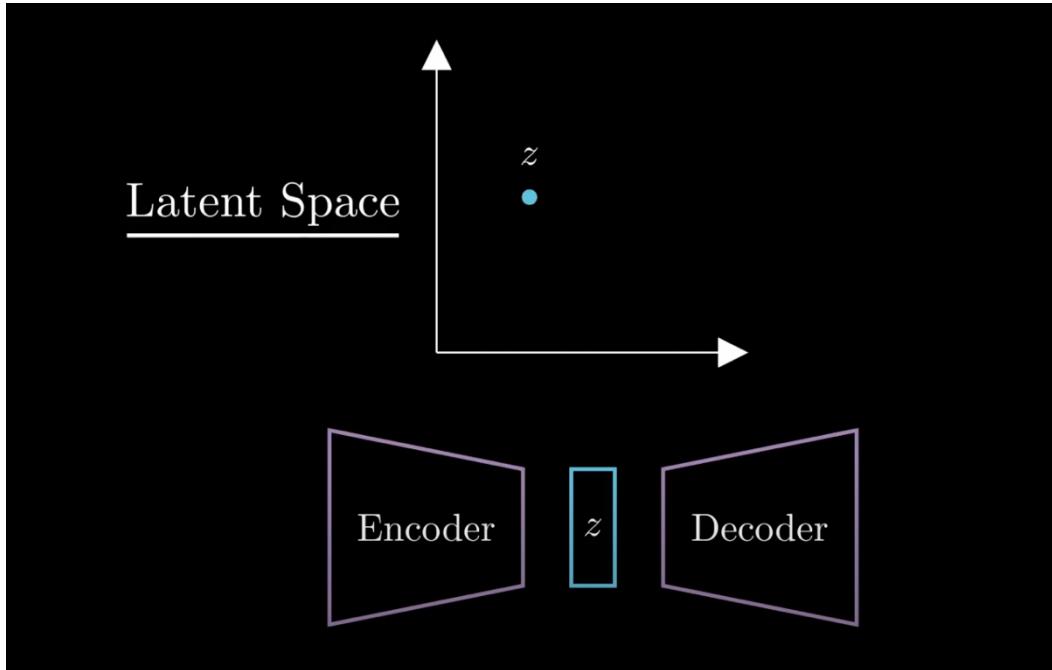
L2 Difference between two
dimensional spaces

Measure how close the two
probability distribution

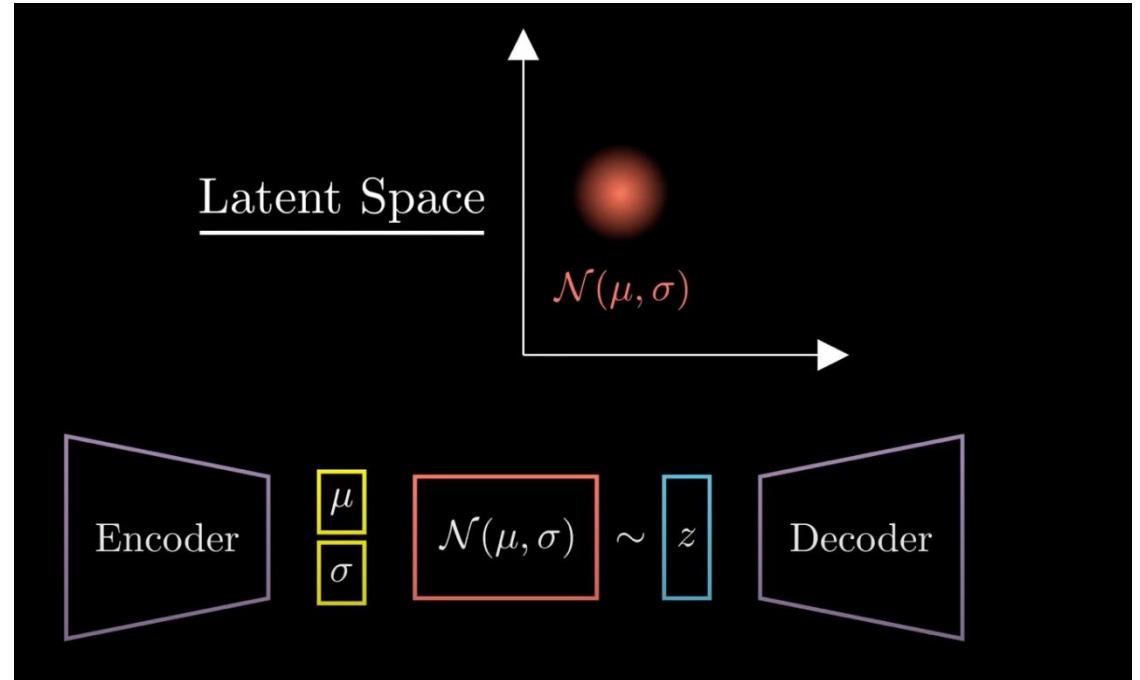
Variational Autoencoder

Let's look at how Variational
Autoencoders practically works ?

How this works in practically ?

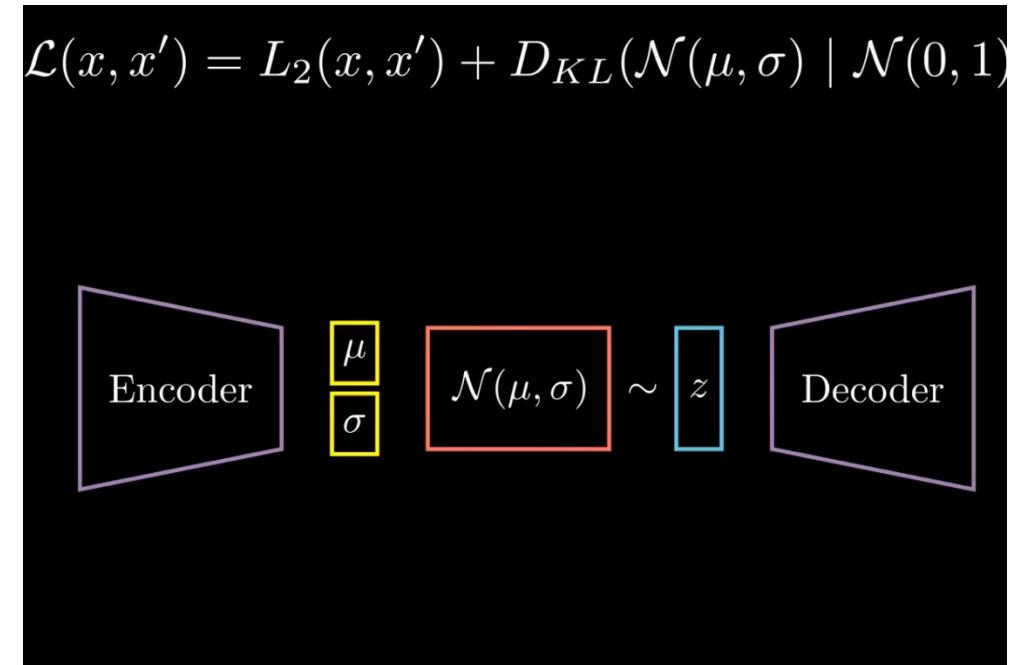
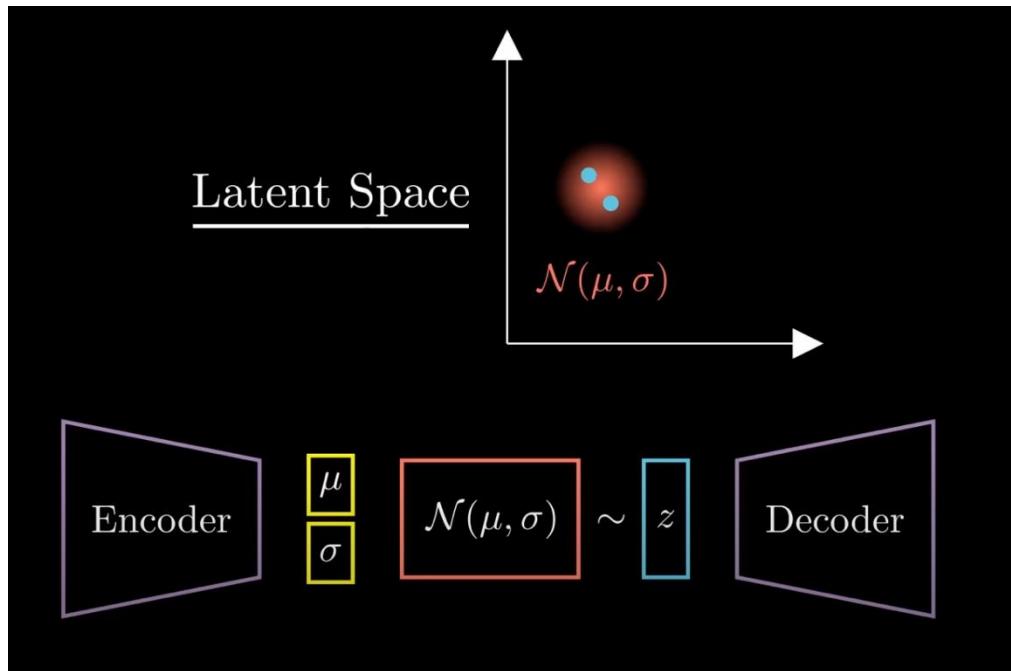


In a **standard autoencoder** an input (e.g. image) is a 1 point in the latent space



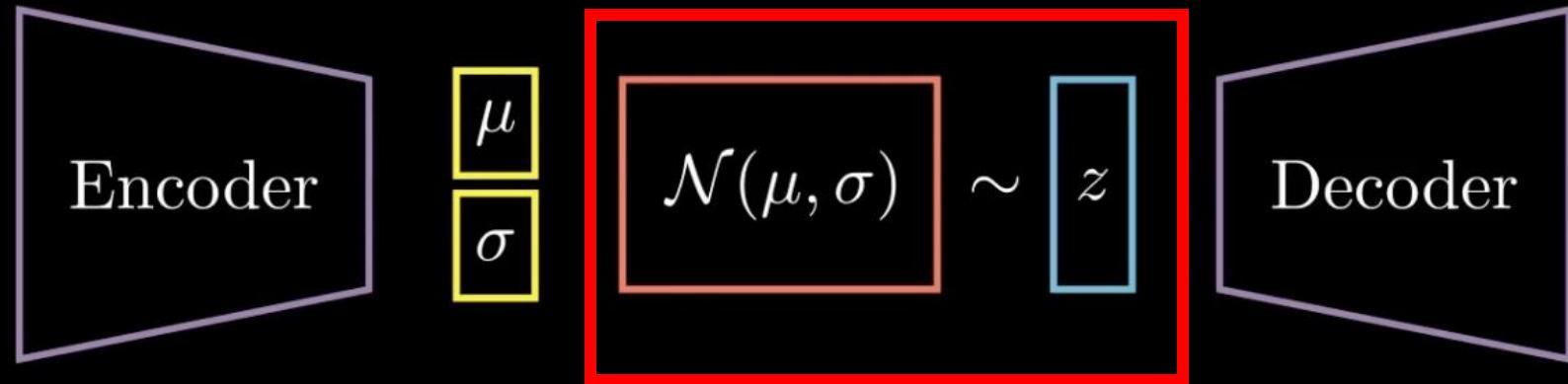
In a **Variational Autoencoder** an input (e.g. image) represented as a Gaussian distribution with parameters defined.

How this works in practically ?



Select two samples (x, x') from the latent space and calcite the loss and backpropagate to the input space

How to backpropagate through the sampling process?



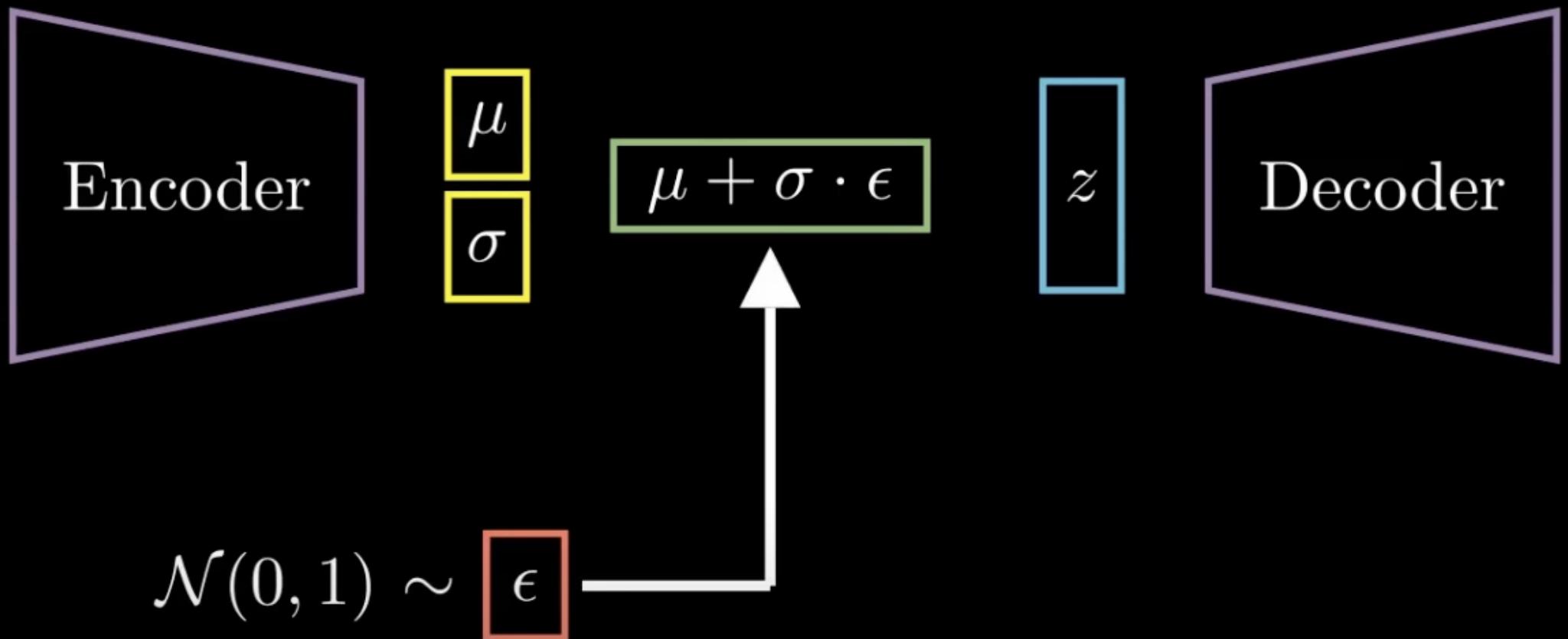
? ?



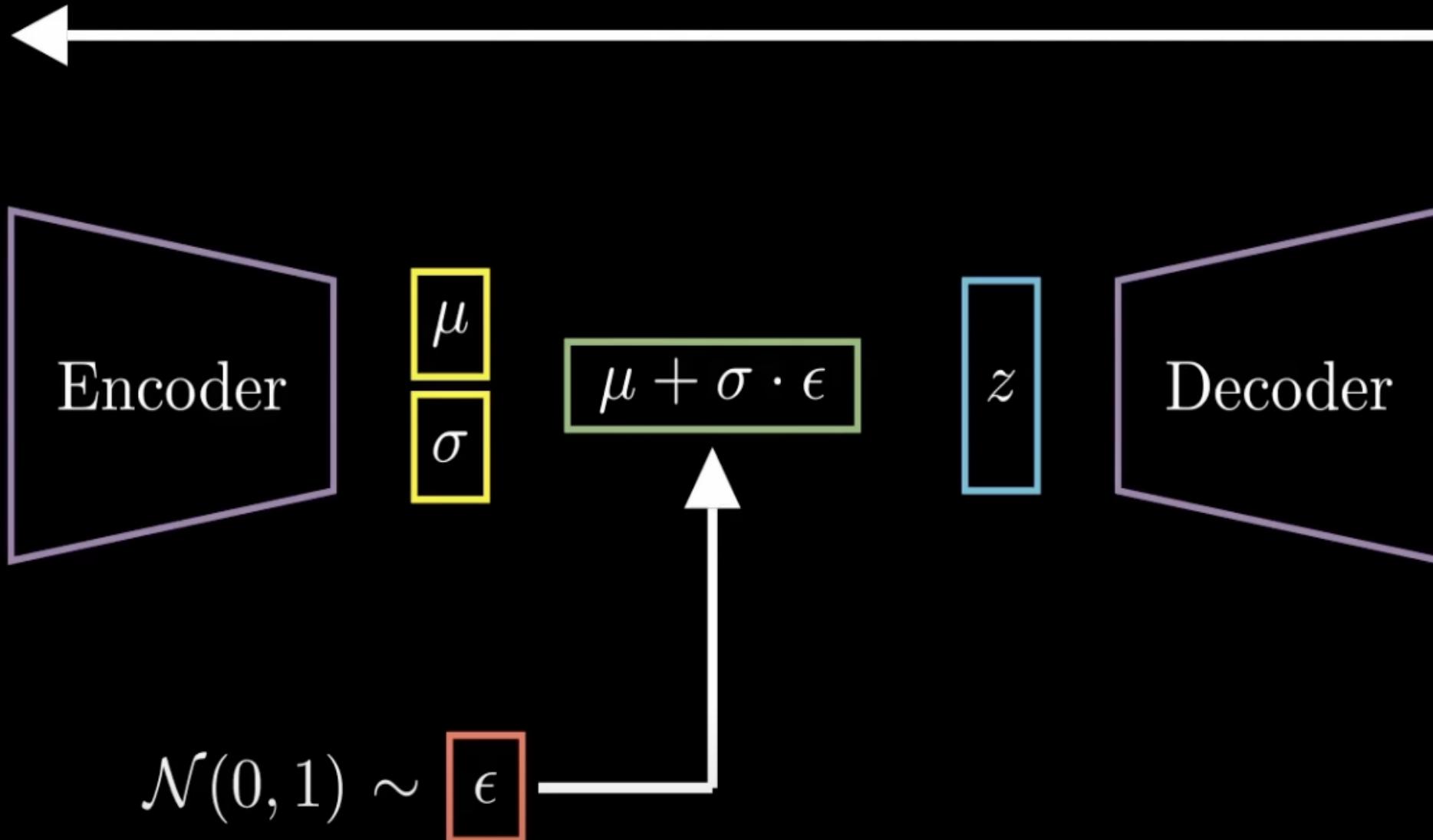
We can't backpropagate through a distribution

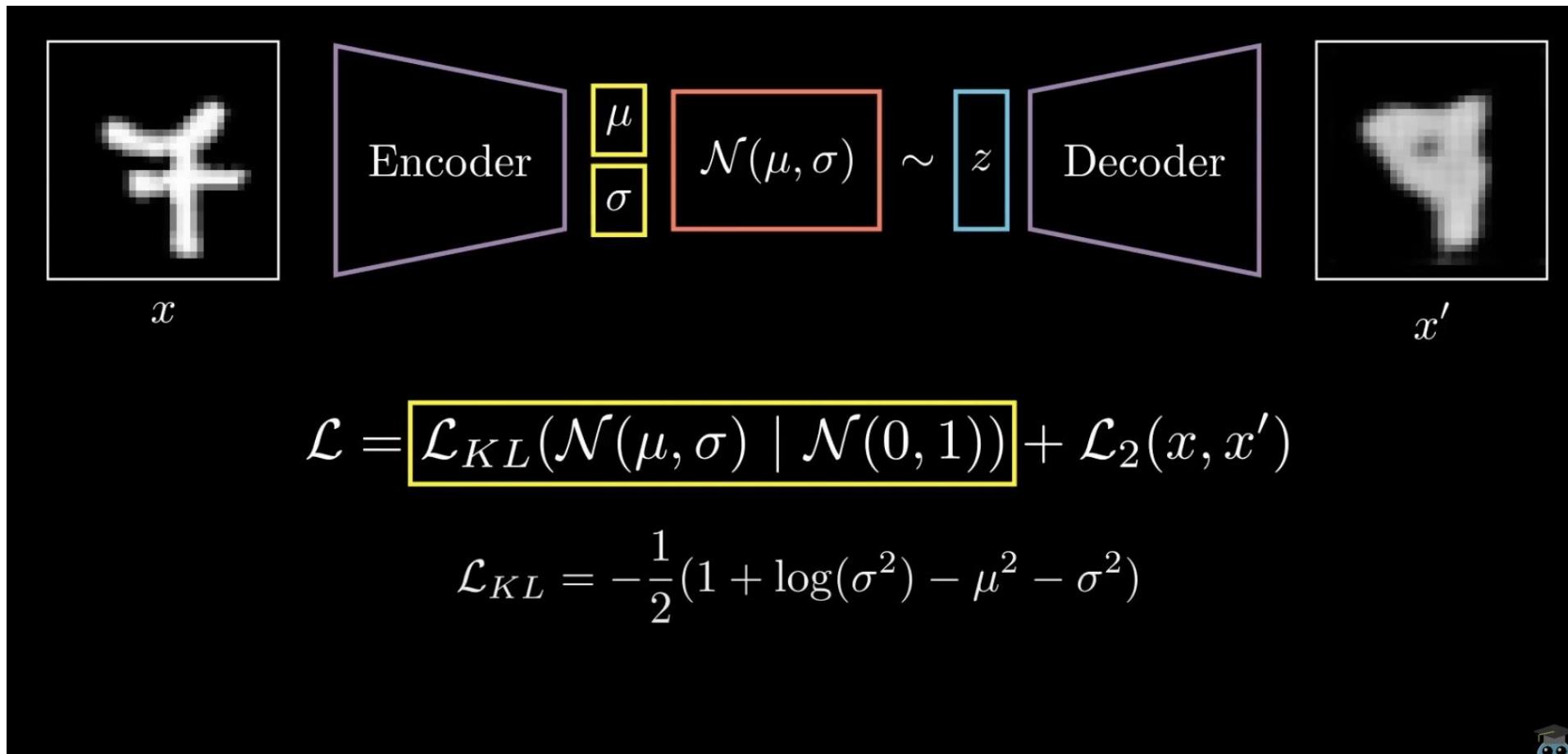
Sanjaya@bu.edu

Reparameterization Trick



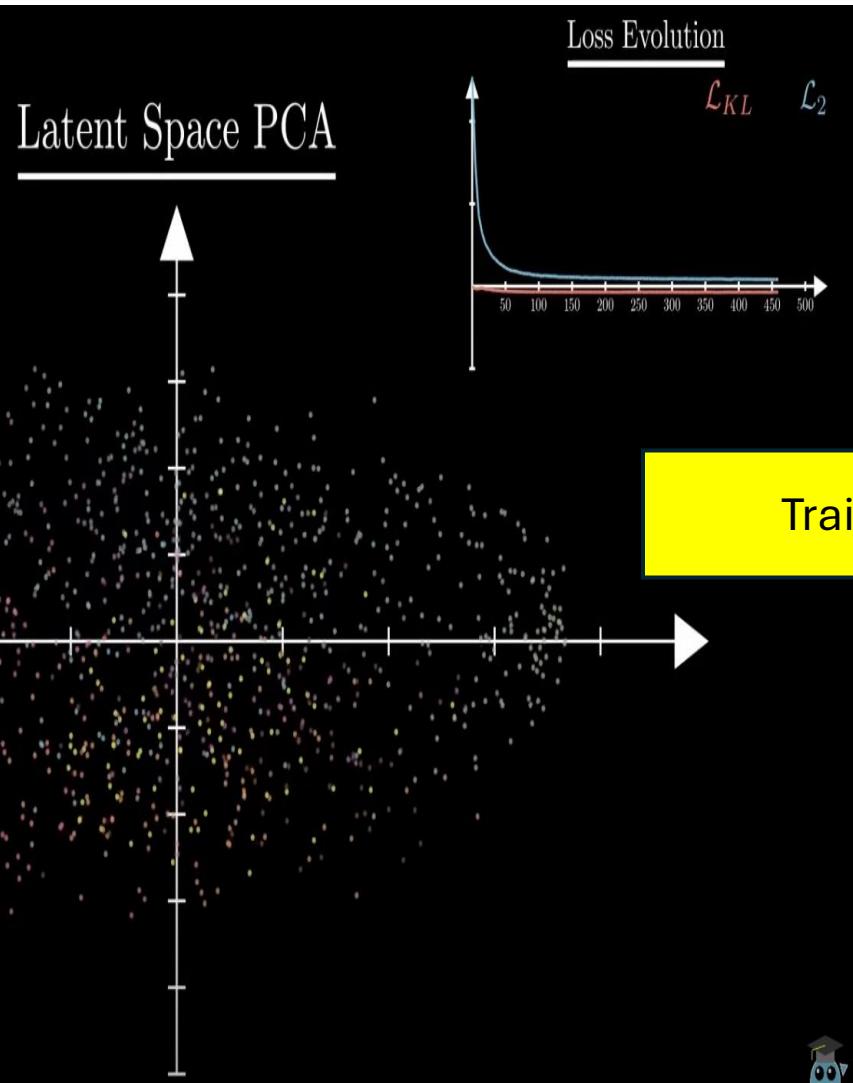
Backpropagation





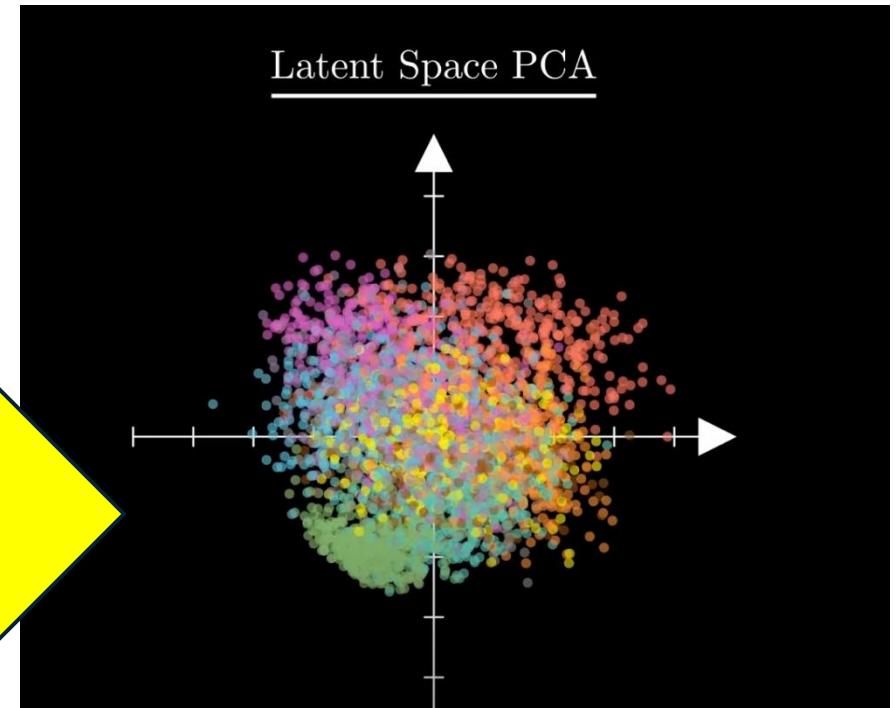
Now we can train the VAE similar to a standard AE , and finetune the parameters for each image (input) of the data set.

Gradient Step: 460

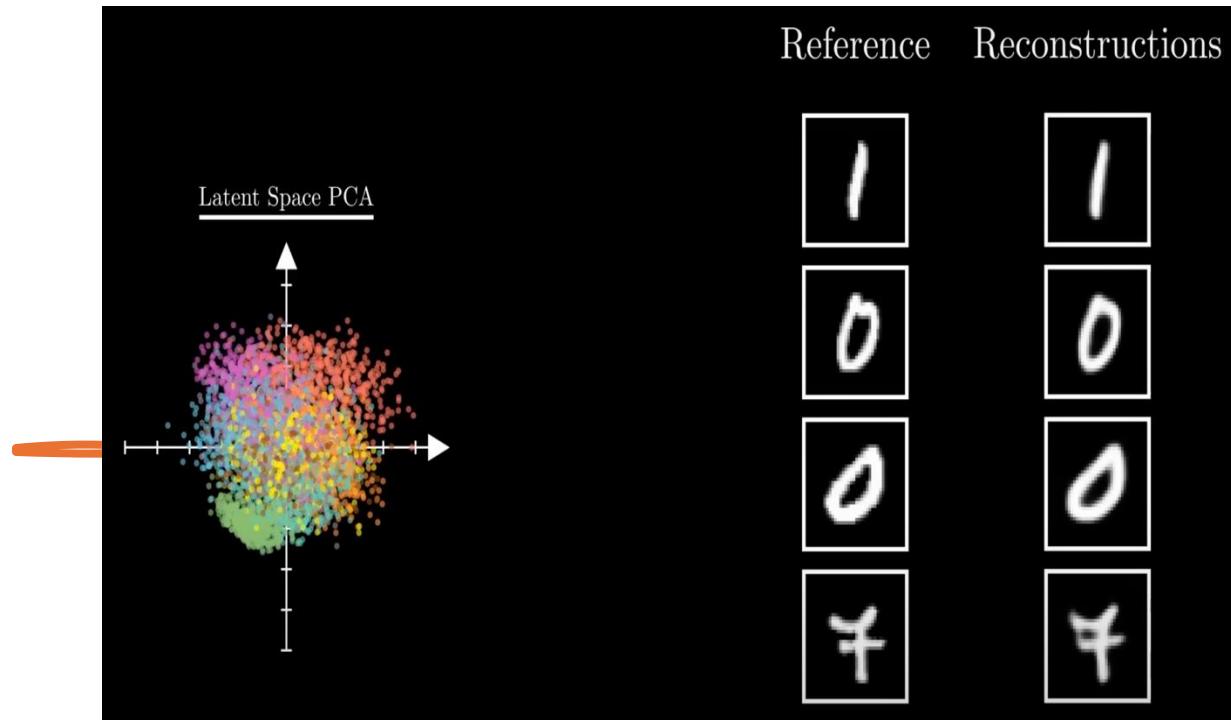


Initially the VAE latent space is more spread (high variance) and distributed a lot.

Training



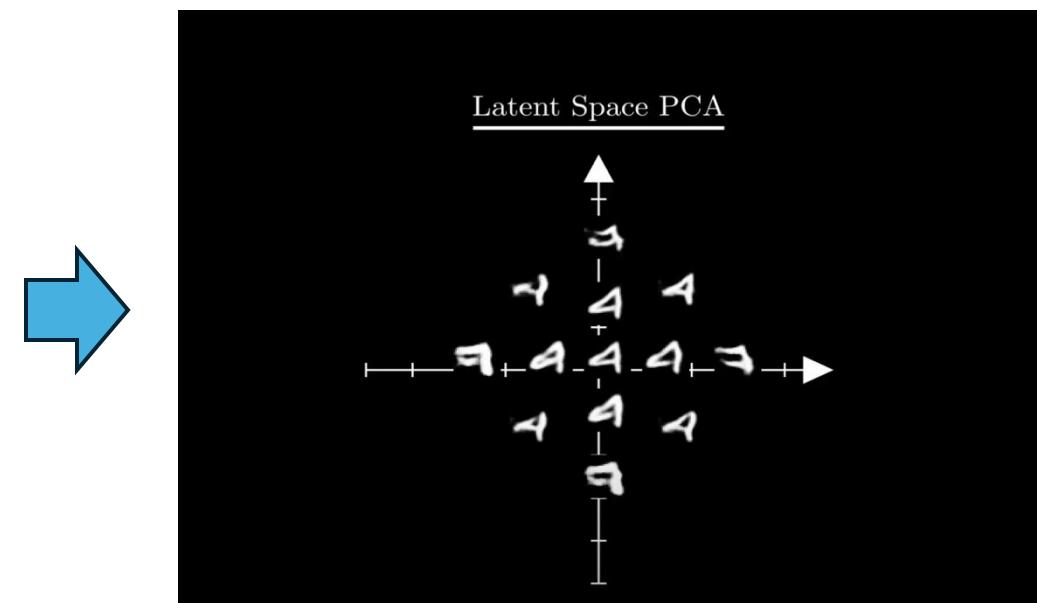
Once the training is completed, VAE latent space is compact, high dense vector embeddings with proximity to each other. This means, most of the points in latent space is present a real data in the input space. Additionally this will enable generating new data

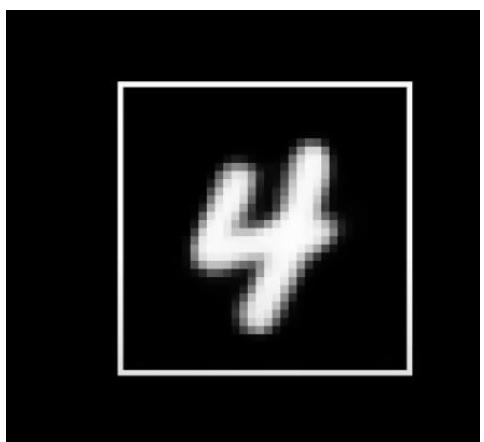


When the distance increase from the original image latent space point , the image get different than its original creation.

- Most of the points in latent space is present a real data in the input space. Additionally this will enable generating new data

- Very rarely we can have non-meaning less reconstructions in well trained VAE's , this will be the cases where we have hallucination and completely new creativity

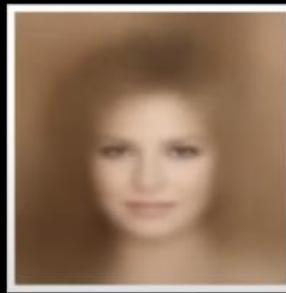
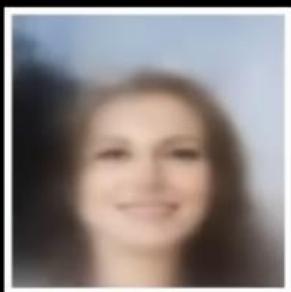




Limitations with VAEs ...

- They create blurry images
- How do we decide which image to generate ?
- Advanced VAE's like conditional VAE's address this with class specified image generations.

VAE Samples



Diffusion samples



GAN Samples



Various types of Autoencoders

Autoencoder Type	Key Feature	Common Use Cases
Basic Autoencoder	Simple encoder-decoder	Dimensionality reduction
VAE	Probabilistic latent space	Generative modeling
Denoising Autoencoder	Learns to remove noise	Image & audio denoising
Sparse Autoencoder	Sparse activations	Feature learning, anomaly detection
Contractive Autoencoder	Regularized representation	Robust feature learning
Convolutional Autoencoder	Uses CNN layers	Image compression & feature extraction
Stacked Autoencoder	Deep multi-layer model	Deep feature extraction
AAE	Autoencoder + GAN	High-quality data generation
LSTM Autoencoder	LSTM-based sequence learning	Time series anomaly detection
Transformer Autoencoder	Uses transformer layers	NLP feature extraction