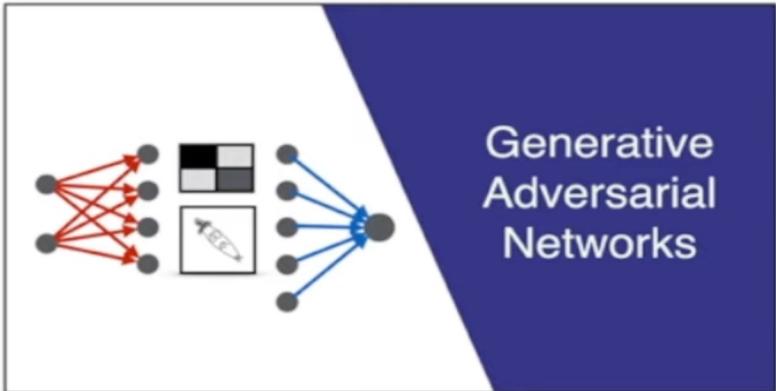


Generative models

Agenda: Autoencoders, VAE
2 Hands-on Case Studies



Generative
Adversarial
Networks



Restricted
Boltzmann
Machines

Understanding a book



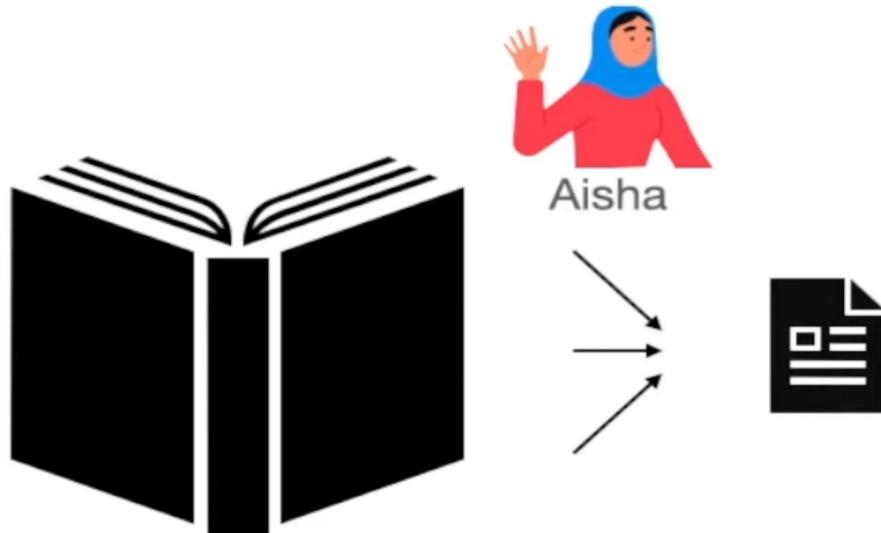
Understanding a book



Understanding a book



Understanding a book



Understanding a book

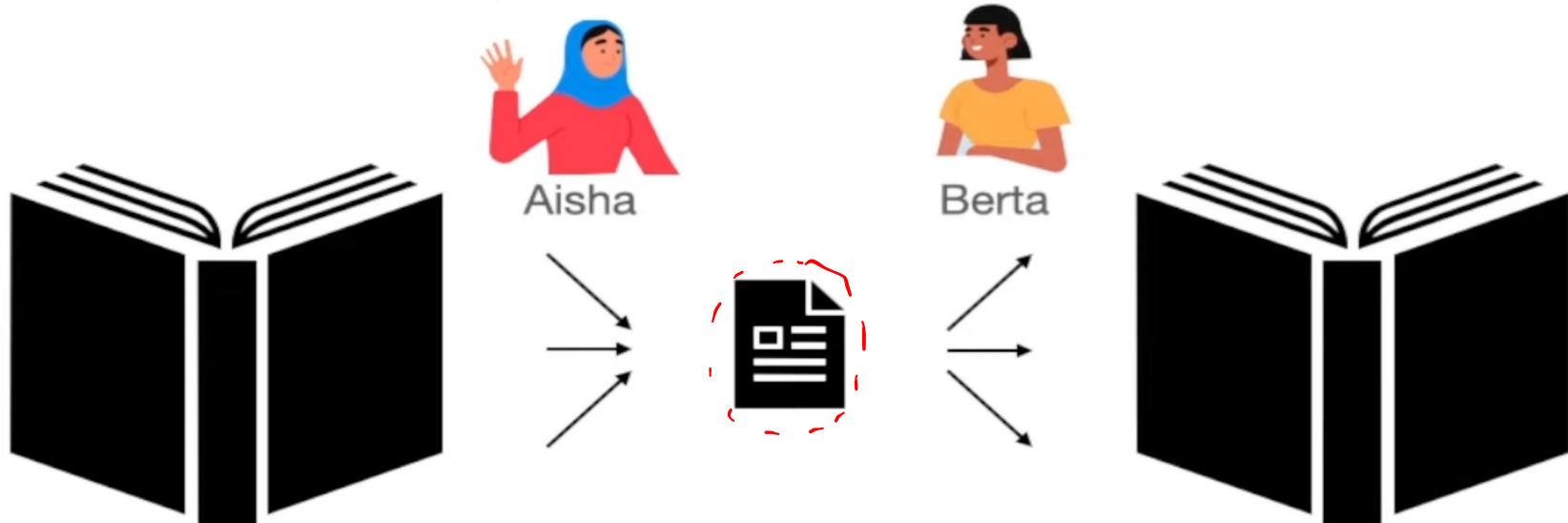


Aisha

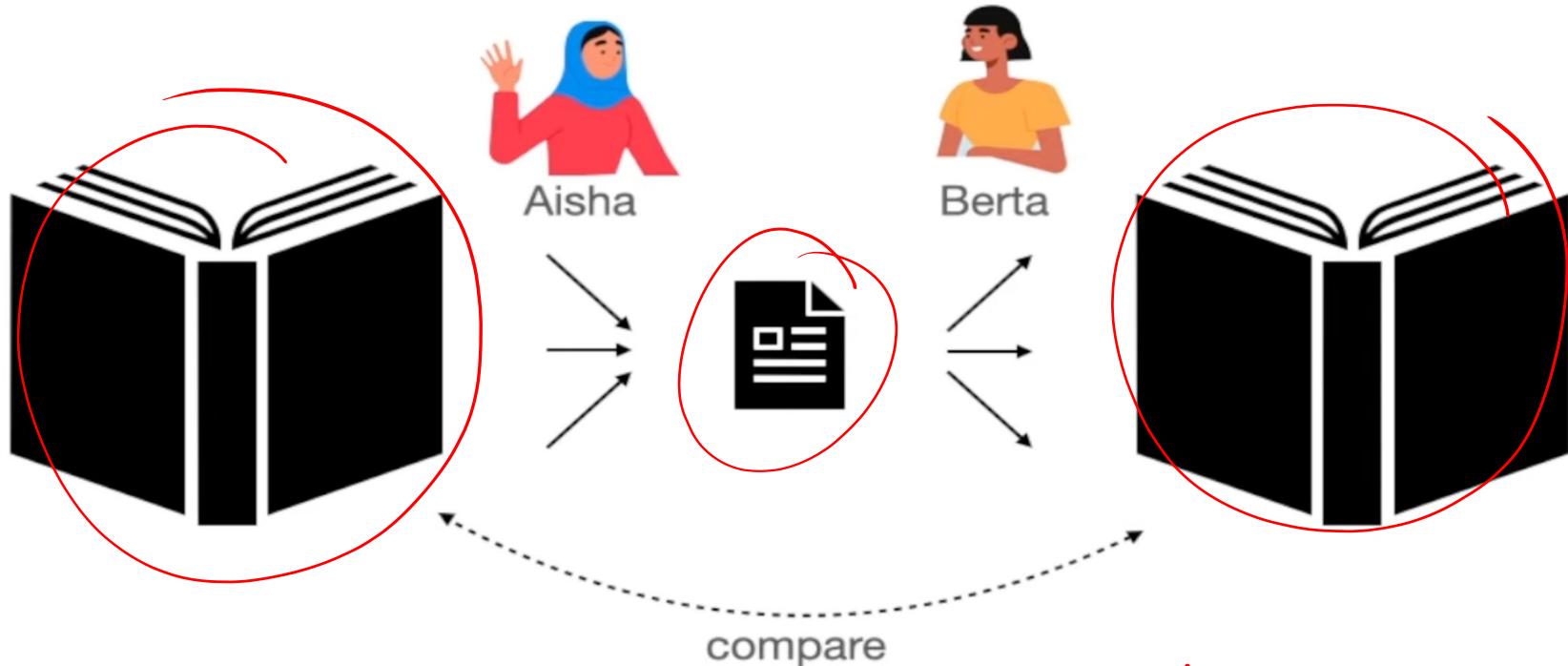


Berta

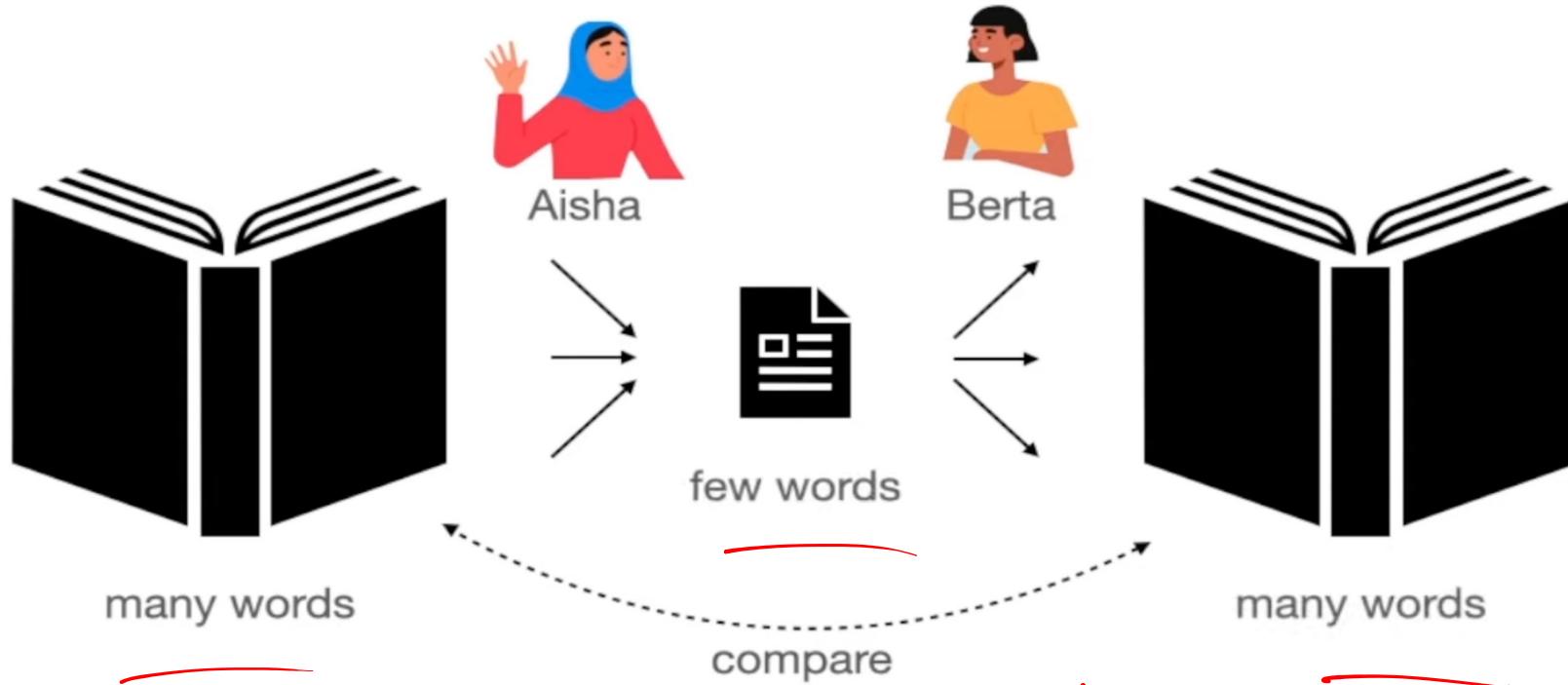
Understanding a book



Understanding a book



Understanding a book



Autoencoder

Autoencoder



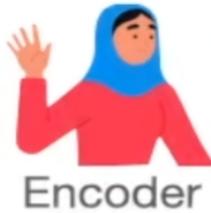
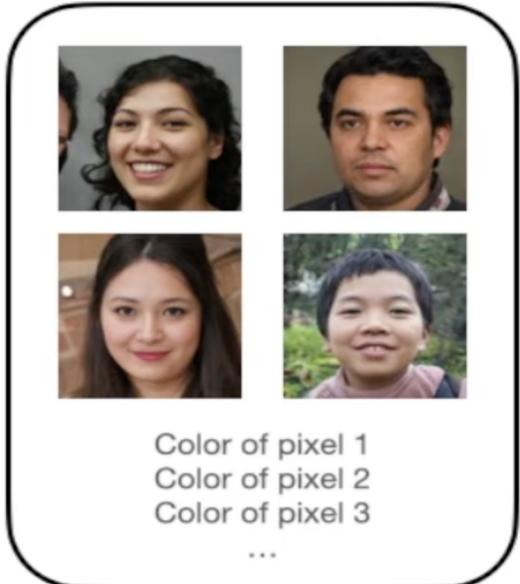
Color of pixel 1

Color of pixel 2

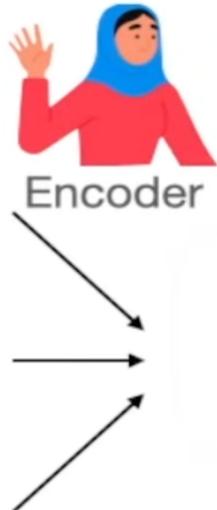
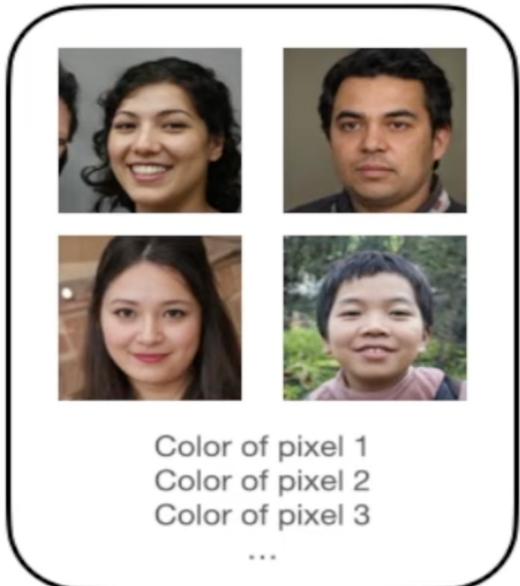
Color of pixel 3

...

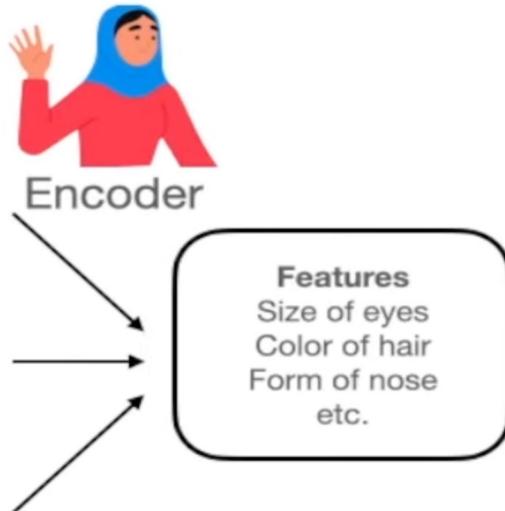
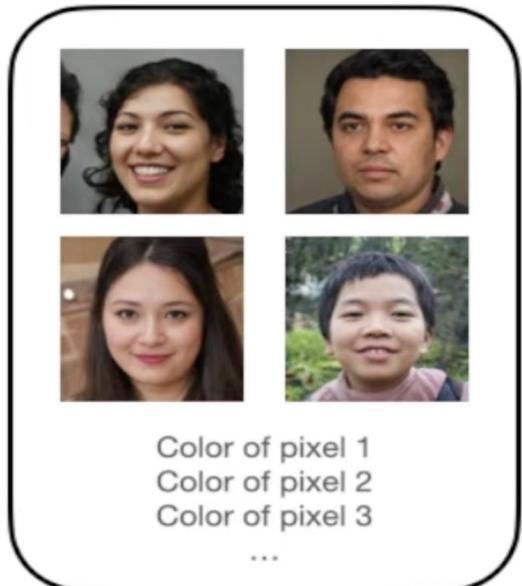
Autoencoder



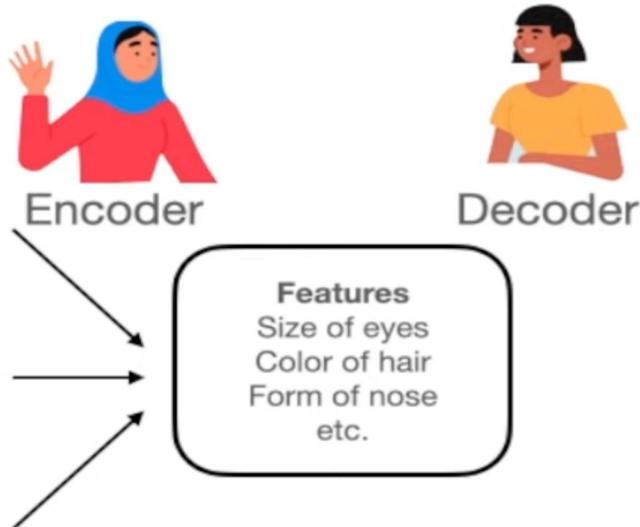
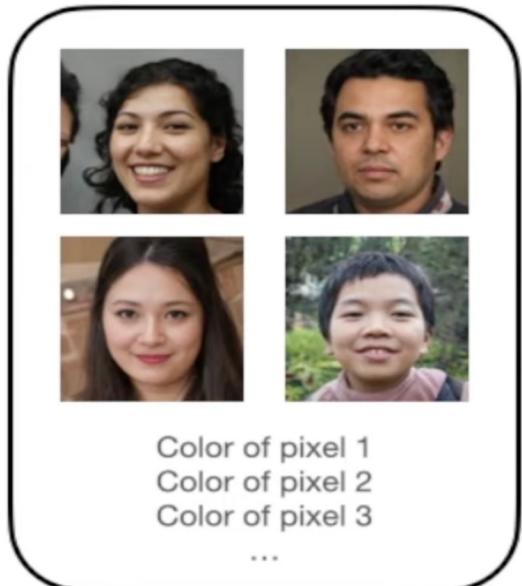
Autoencoder



Autoencoder

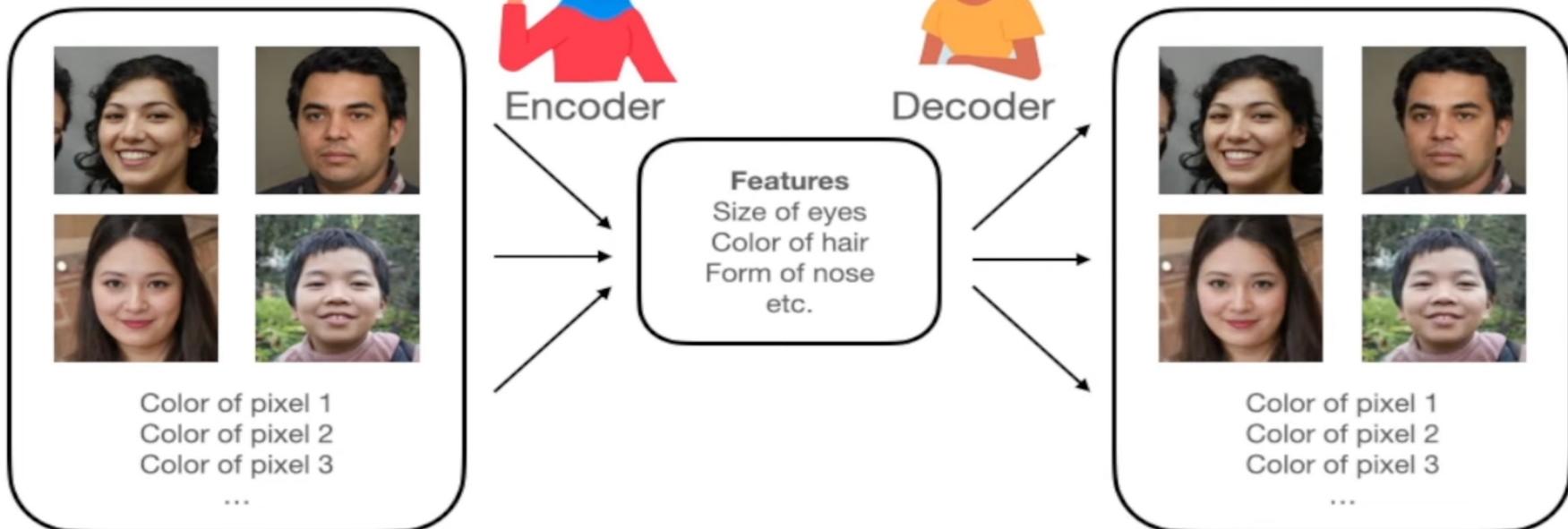


Autoencoder

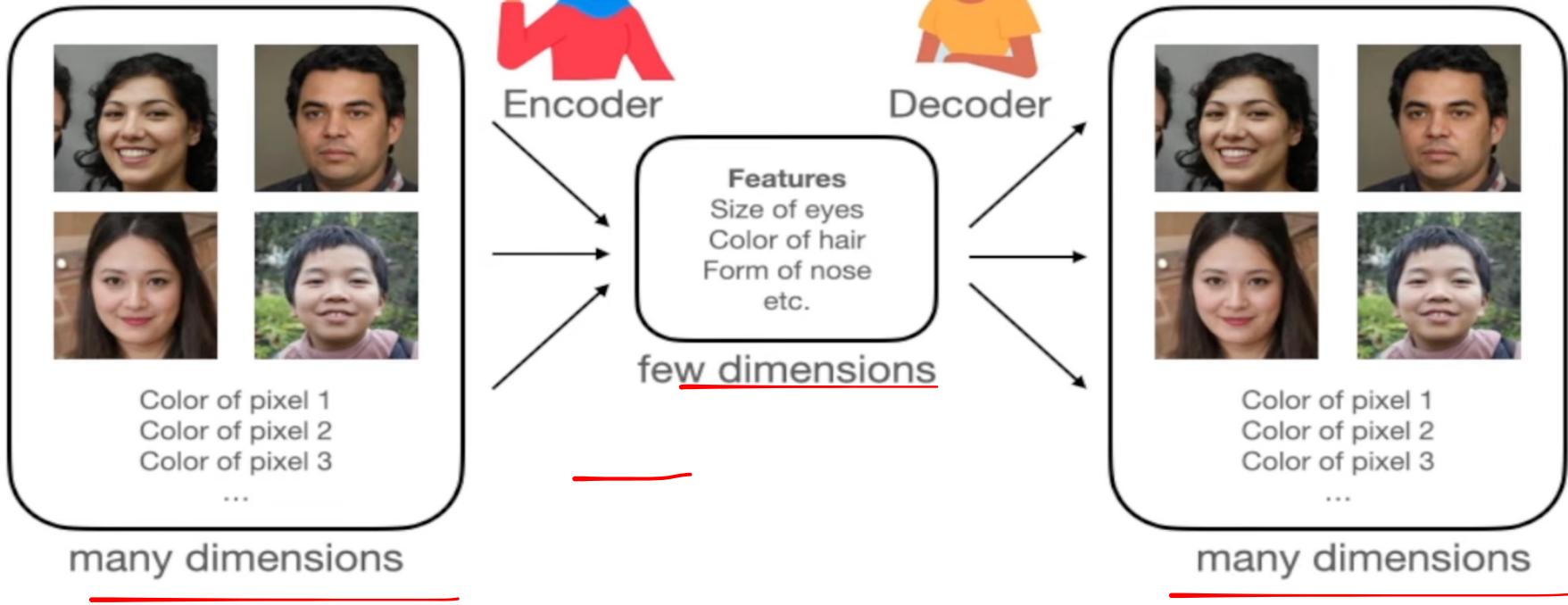


Decoder

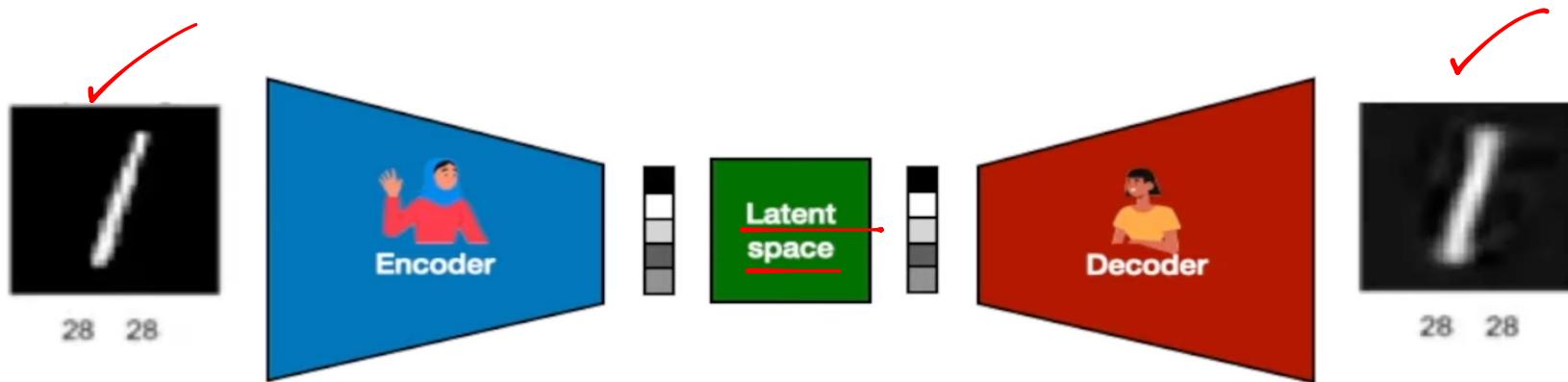
Autoencoder



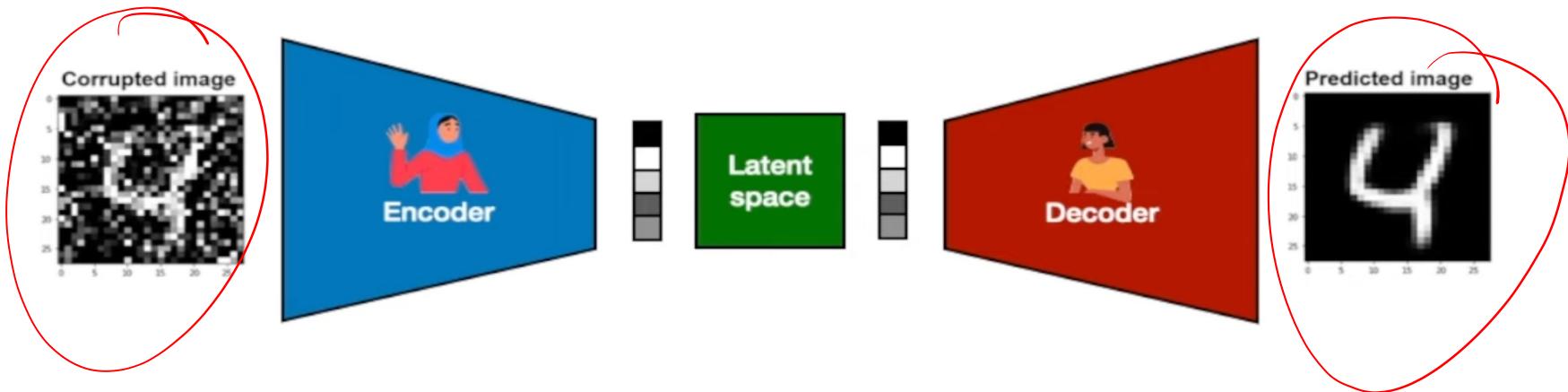
Autoencoder



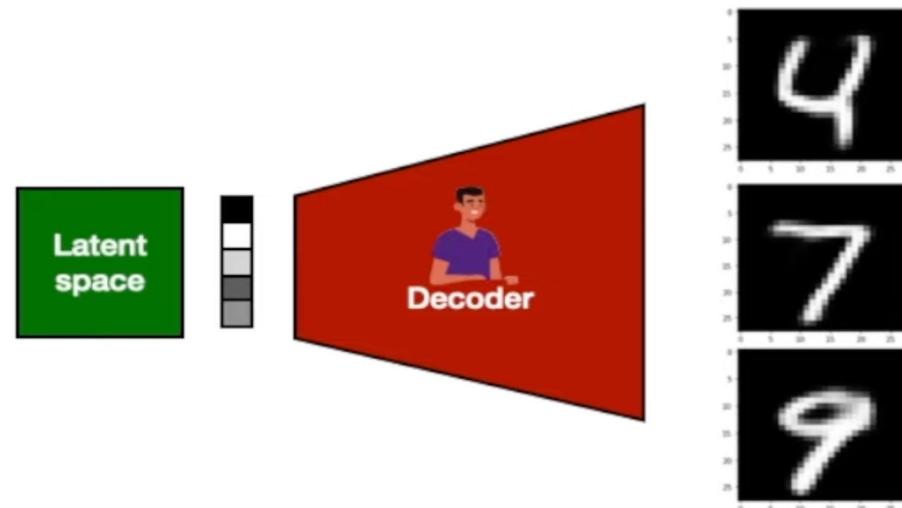
Autoencoders



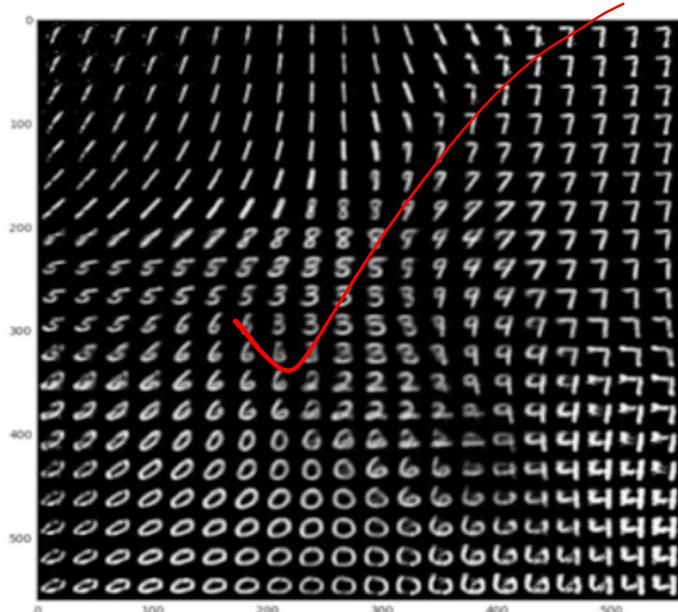
Denoising autoencoders



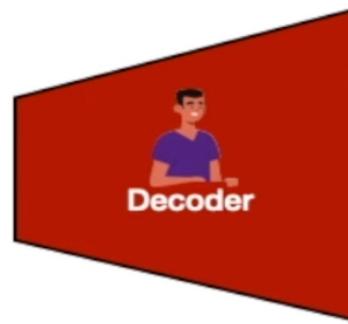
Variational autoencoders



Latent space



Latent
space



Source: Jan Hendrik Metzen
<https://jmetzen.github.io/2015-11-27/vae.html>

Images generated by a VAE



Source: Tom White

Dimensionality reduction

Dataset of images



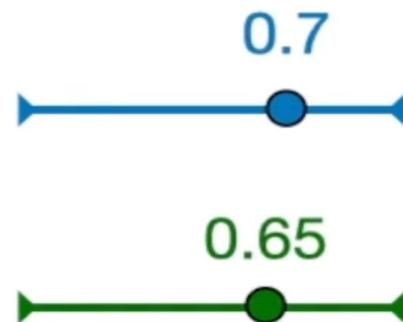
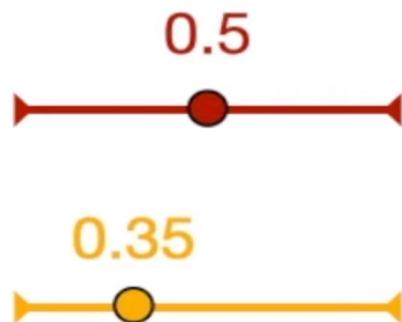
Dataset of images



Dataset of images



Dataset of images

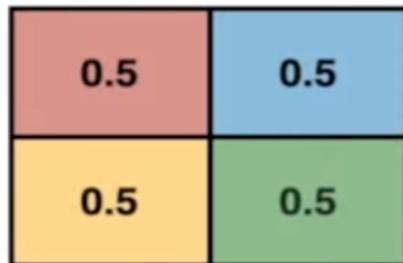
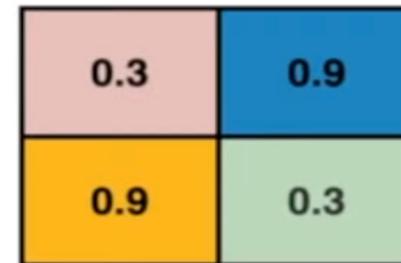
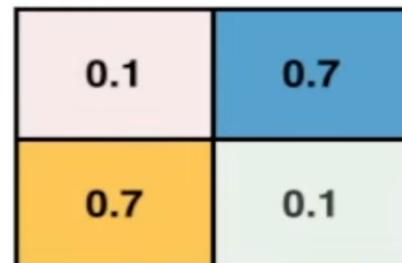


Dataset of images

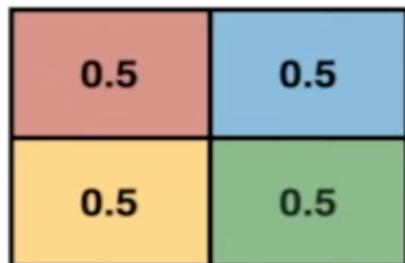
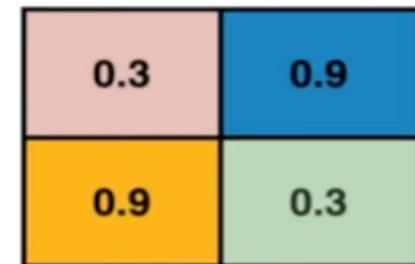
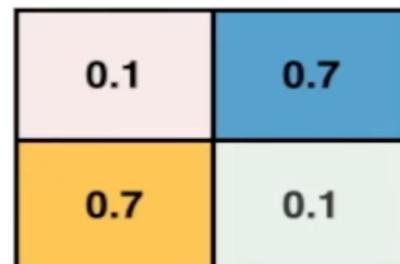
4-dimensional



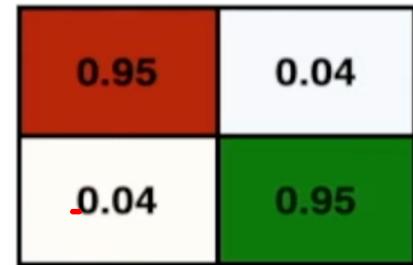
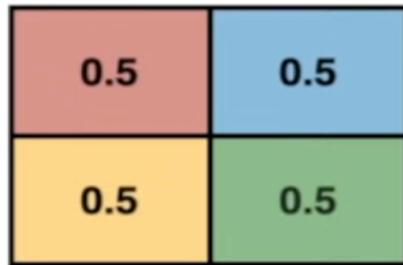
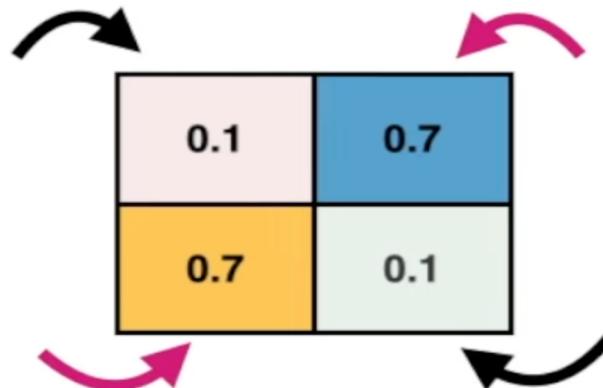
Dataset of images



Dataset of images

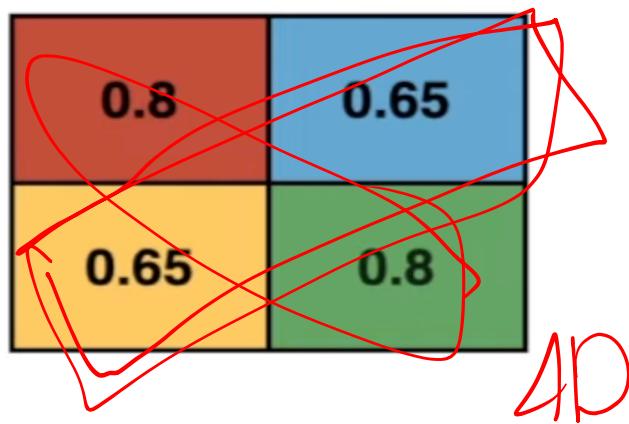


Dataset of images



Dataset of images

[0.8, 0.65]



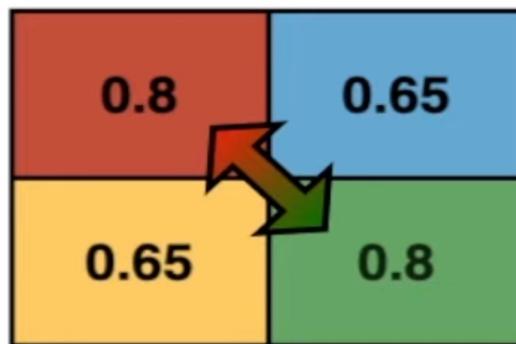
2P

4D

Dataset of images



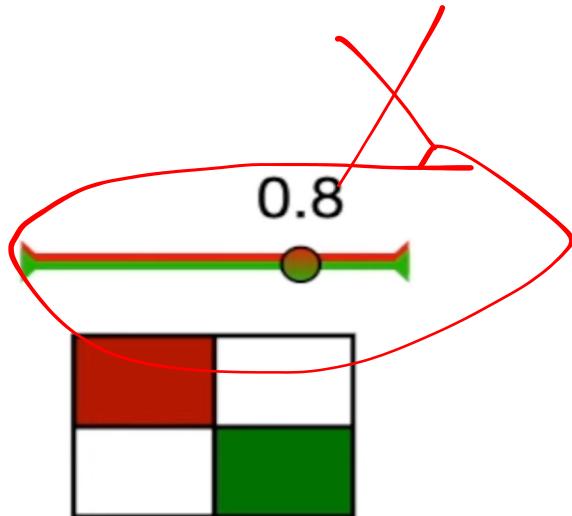
Dataset of images



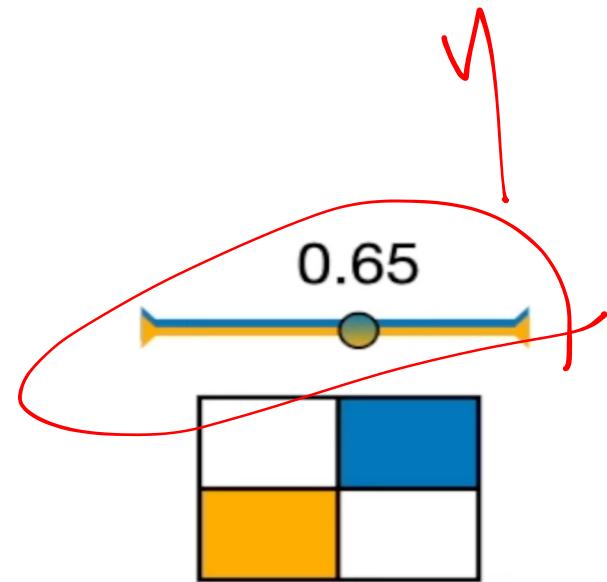
Dataset of images



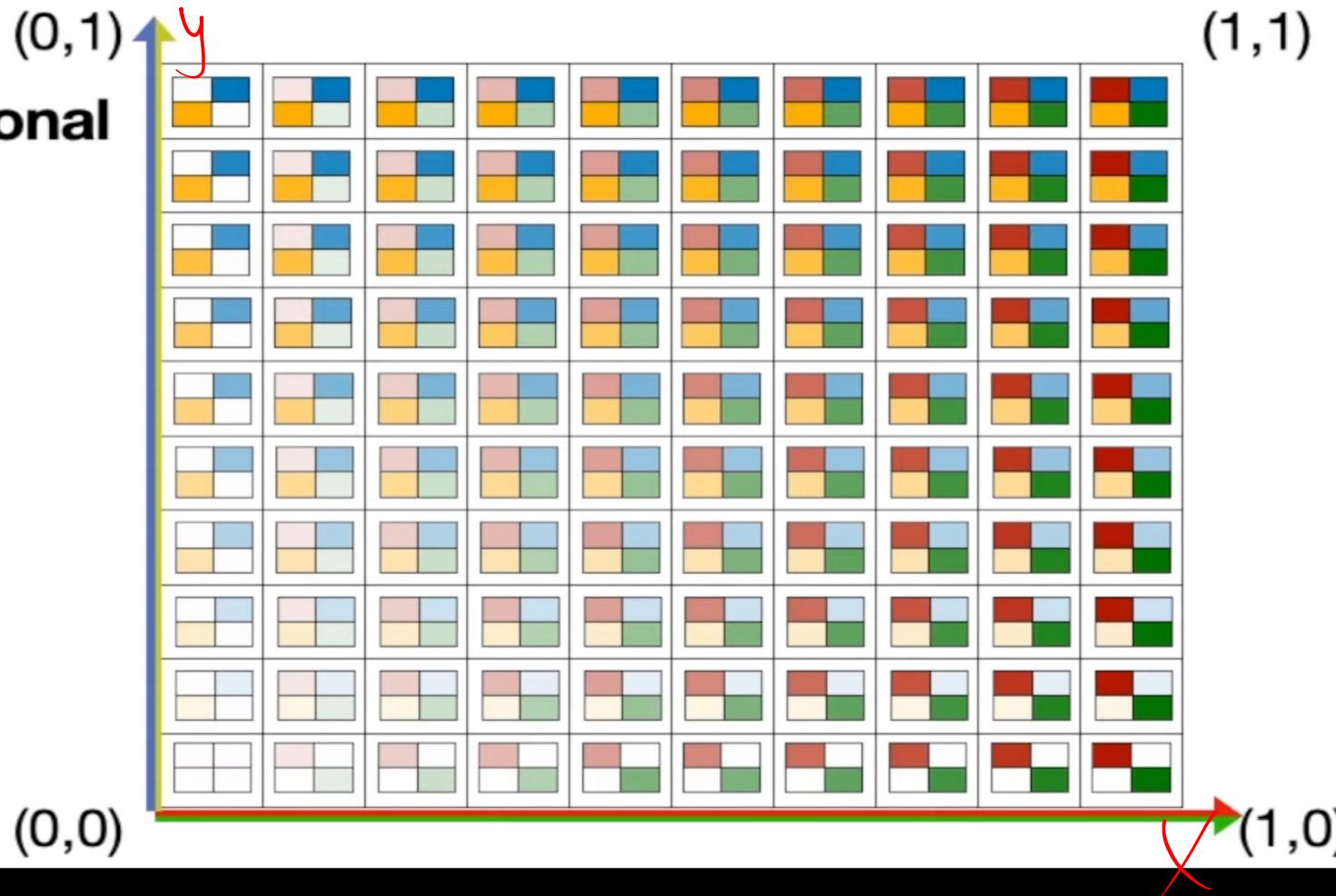
Dataset of images



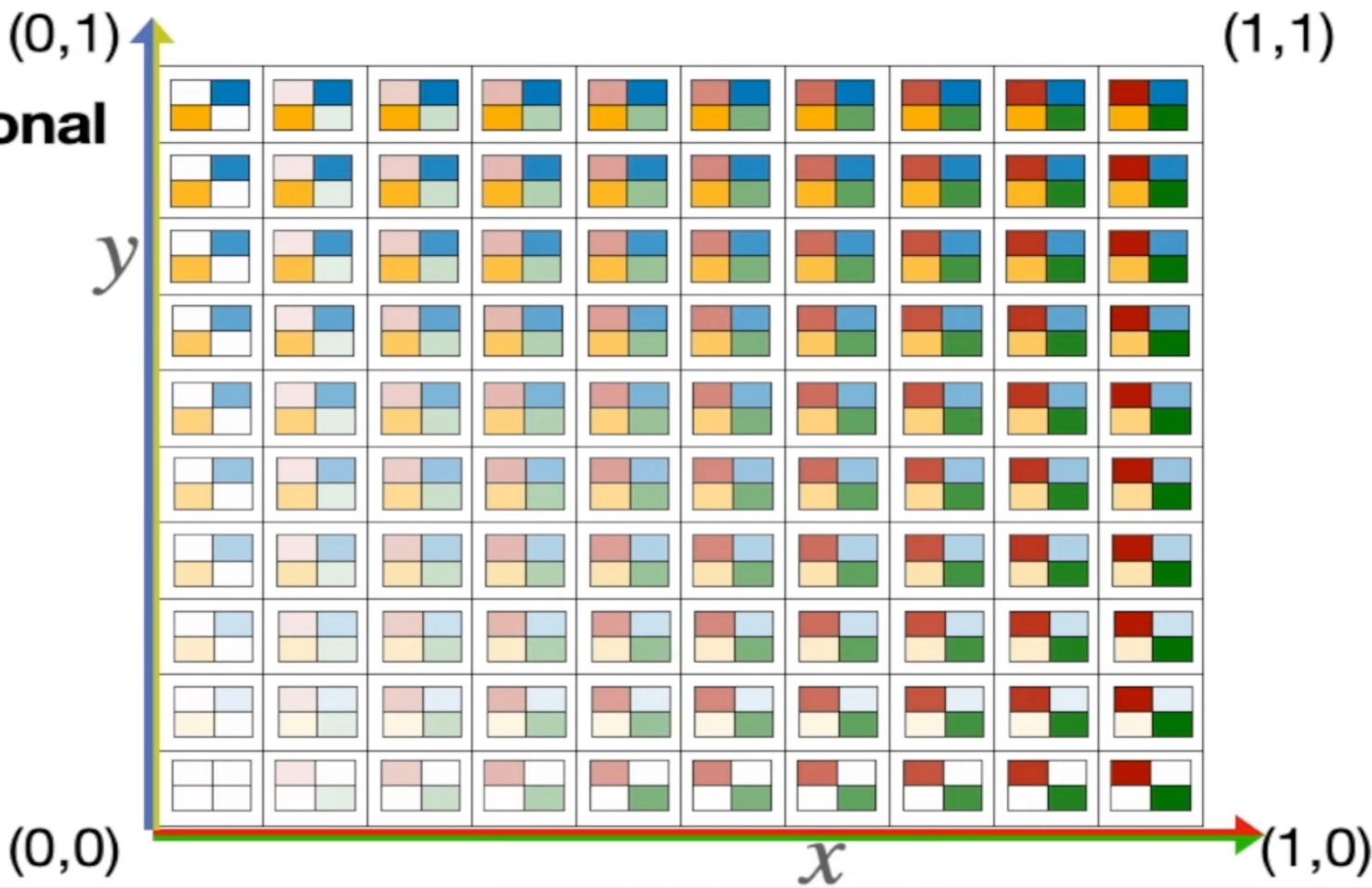
2-dimensional



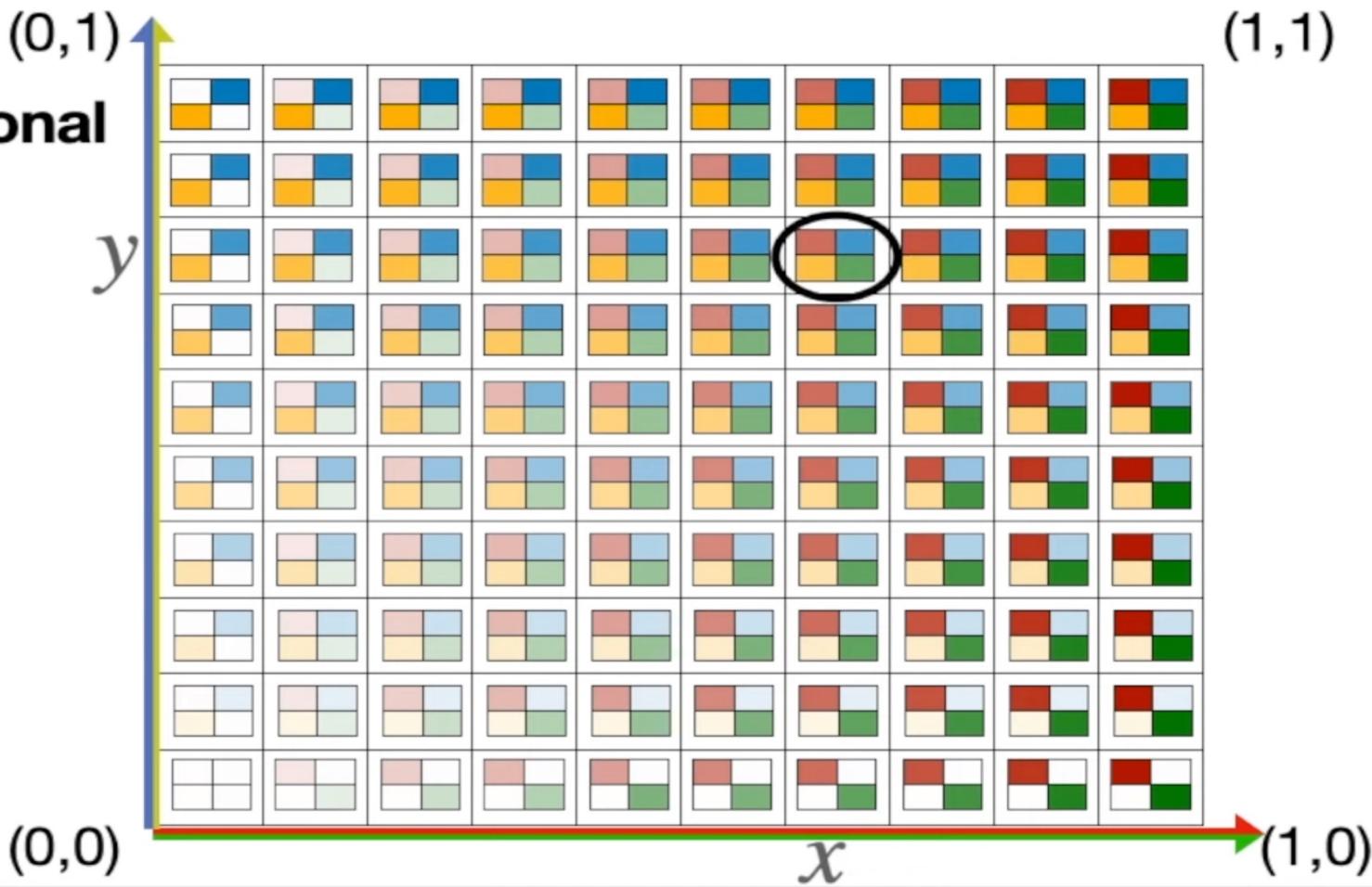
2-dimensional dataset



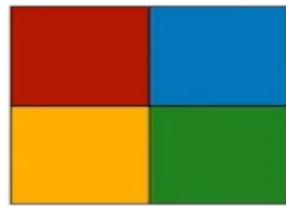
2-dimensional dataset



2-dimensional dataset



2-dimensional dataset



Autoencoder

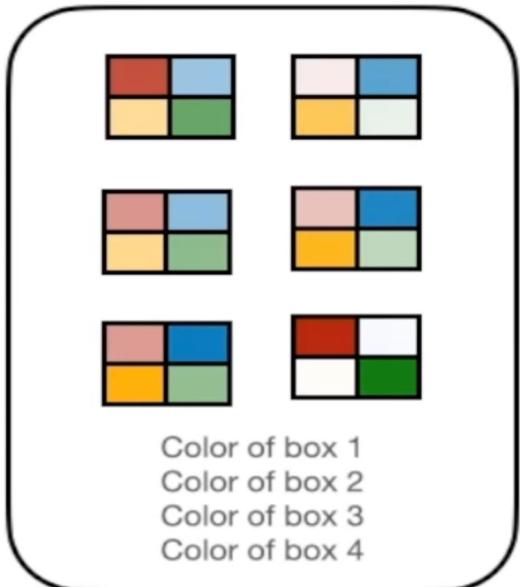


Encoder

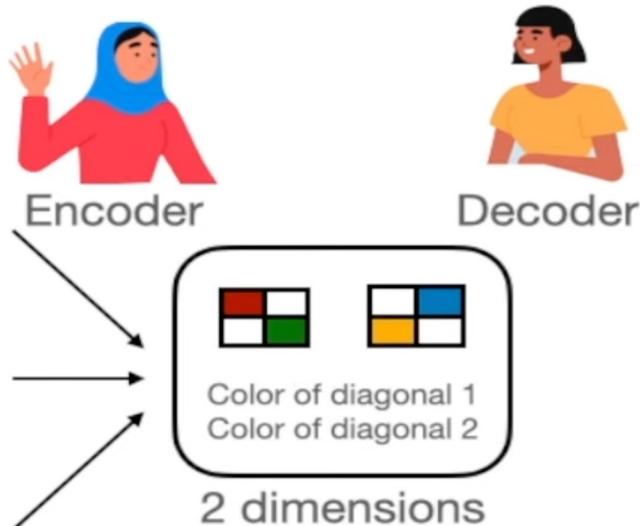
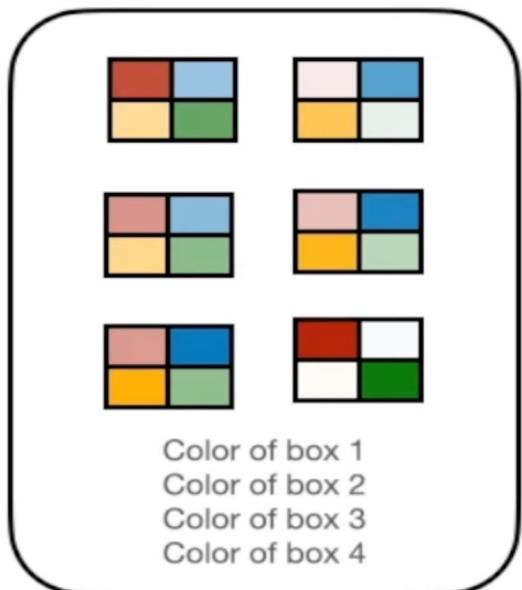


Decoder

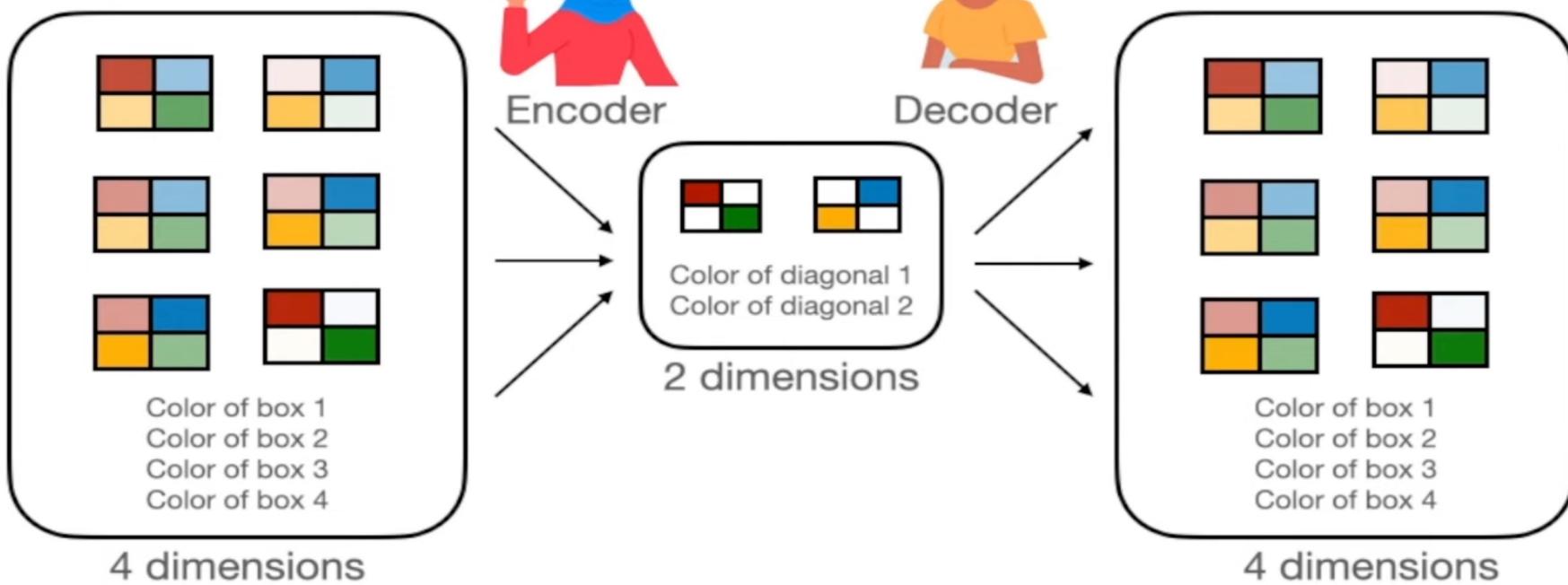
Autoencoder



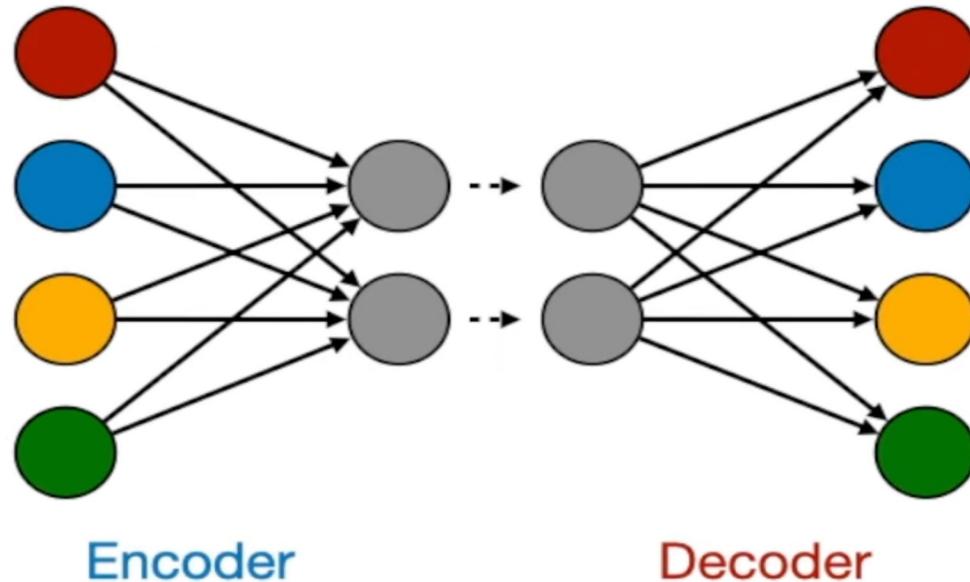
Autoencoder



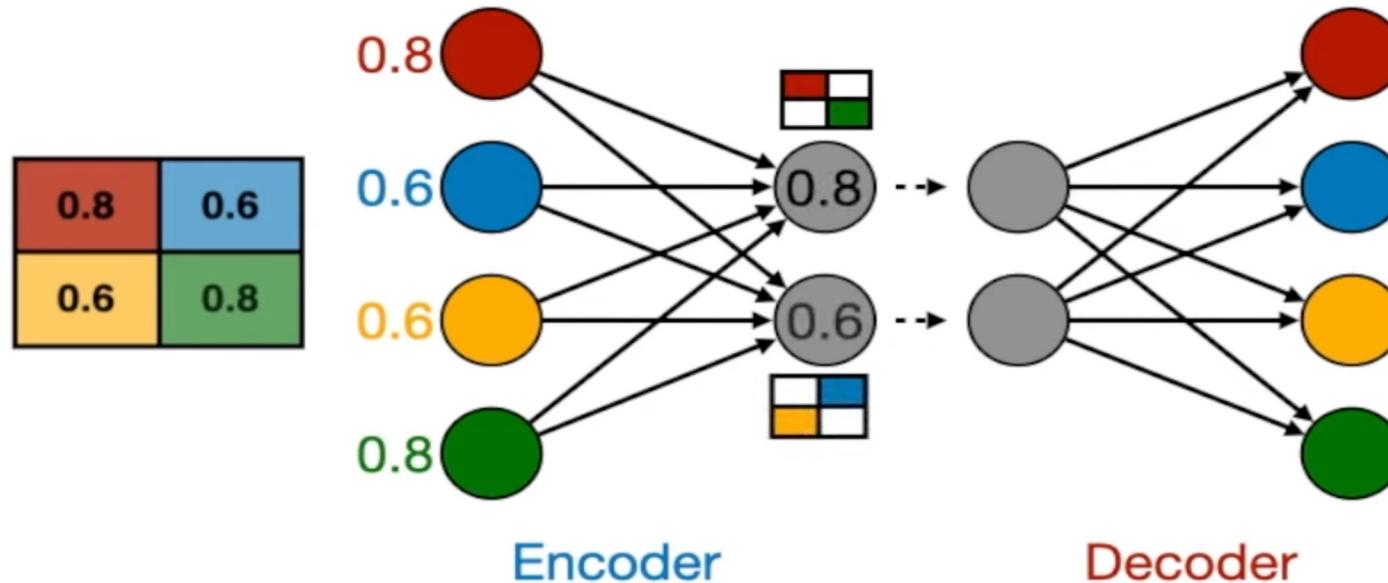
Autoencoder



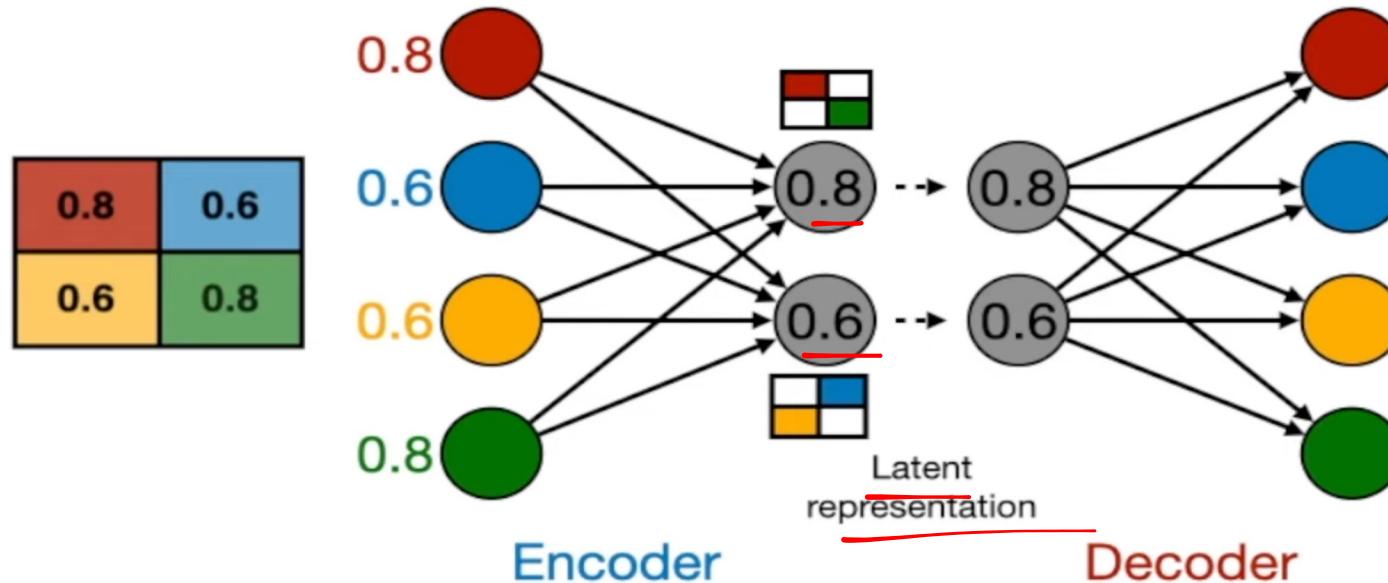
Autoencoder



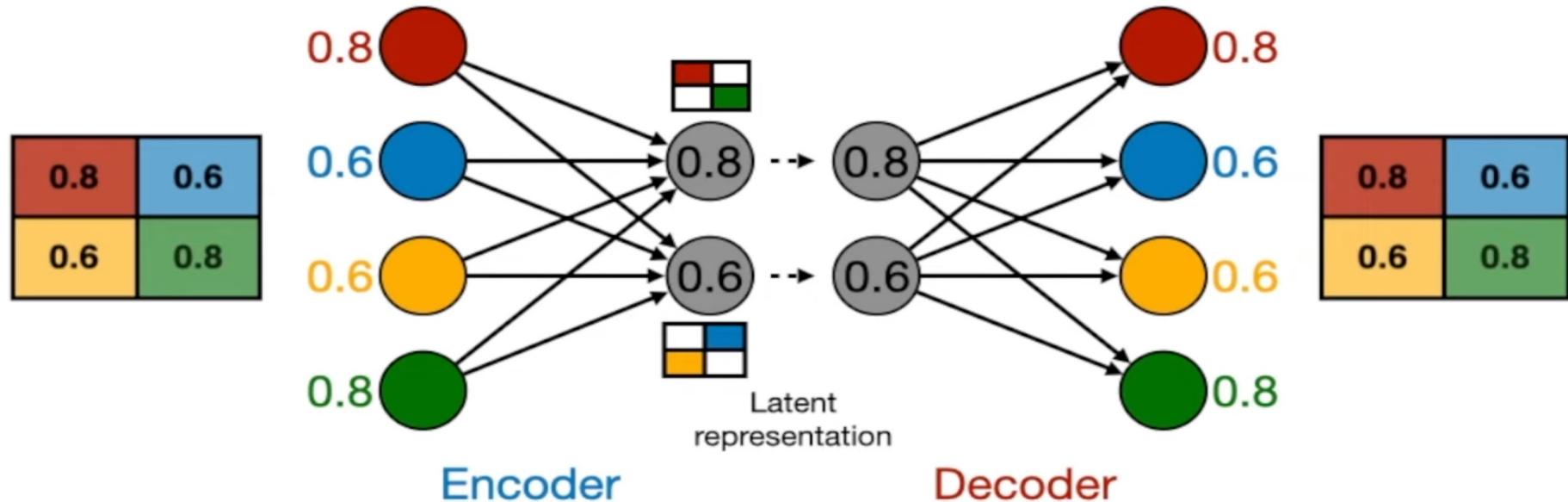
Autoencoder



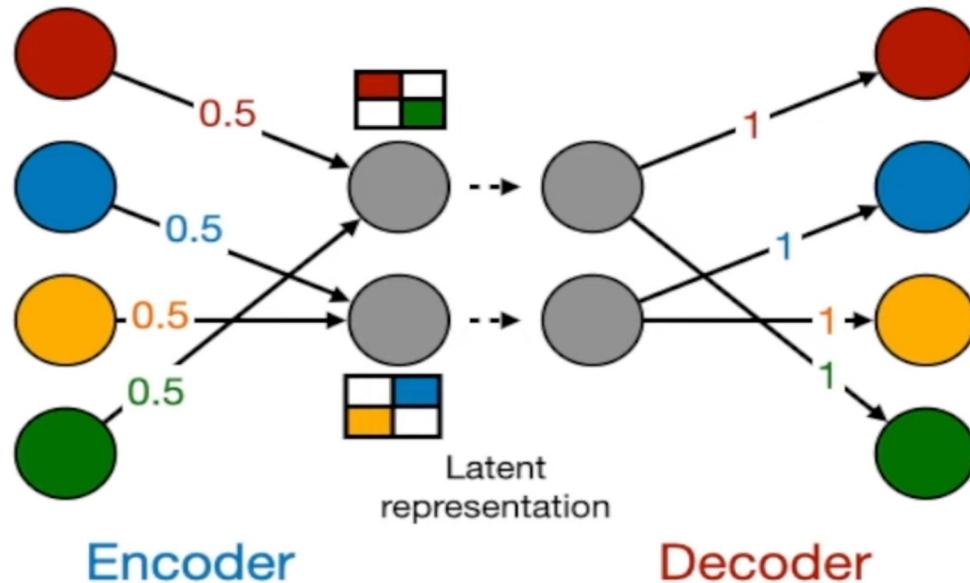
Autoencoder



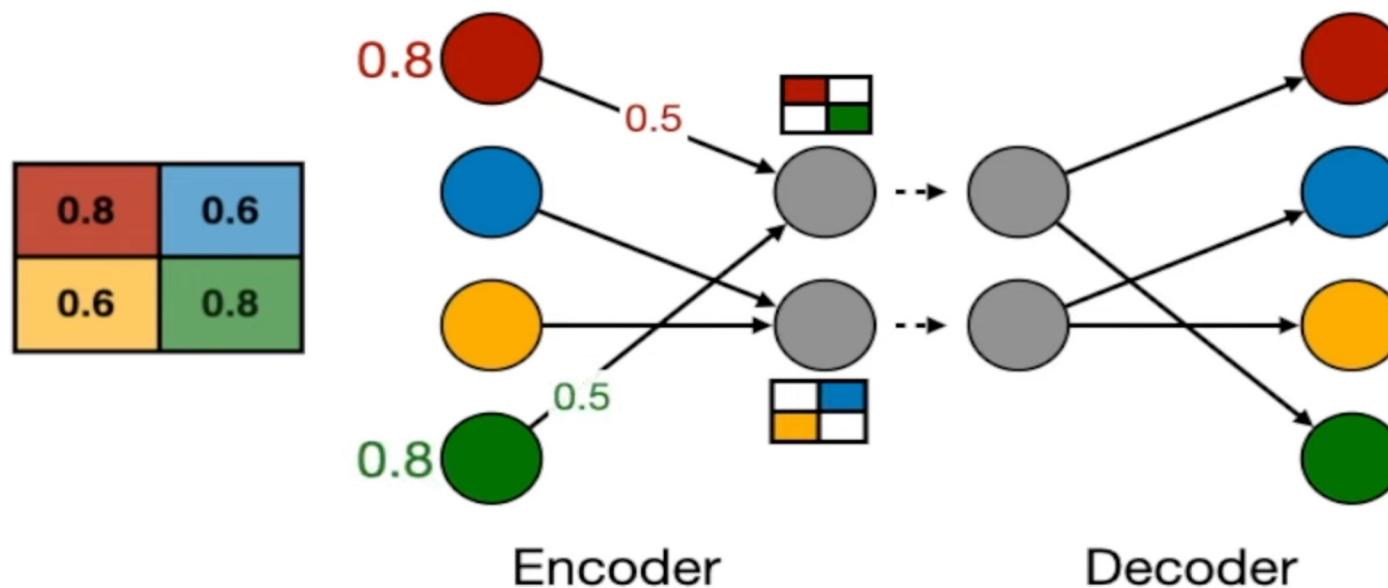
Autoencoder



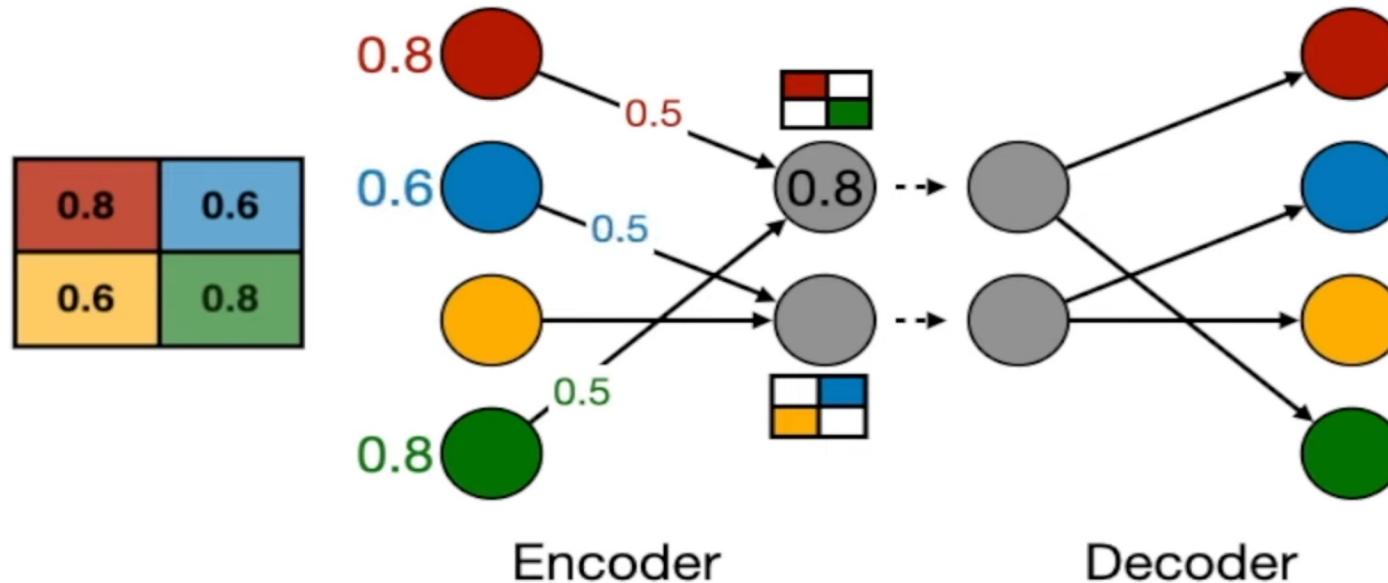
Autoencoder



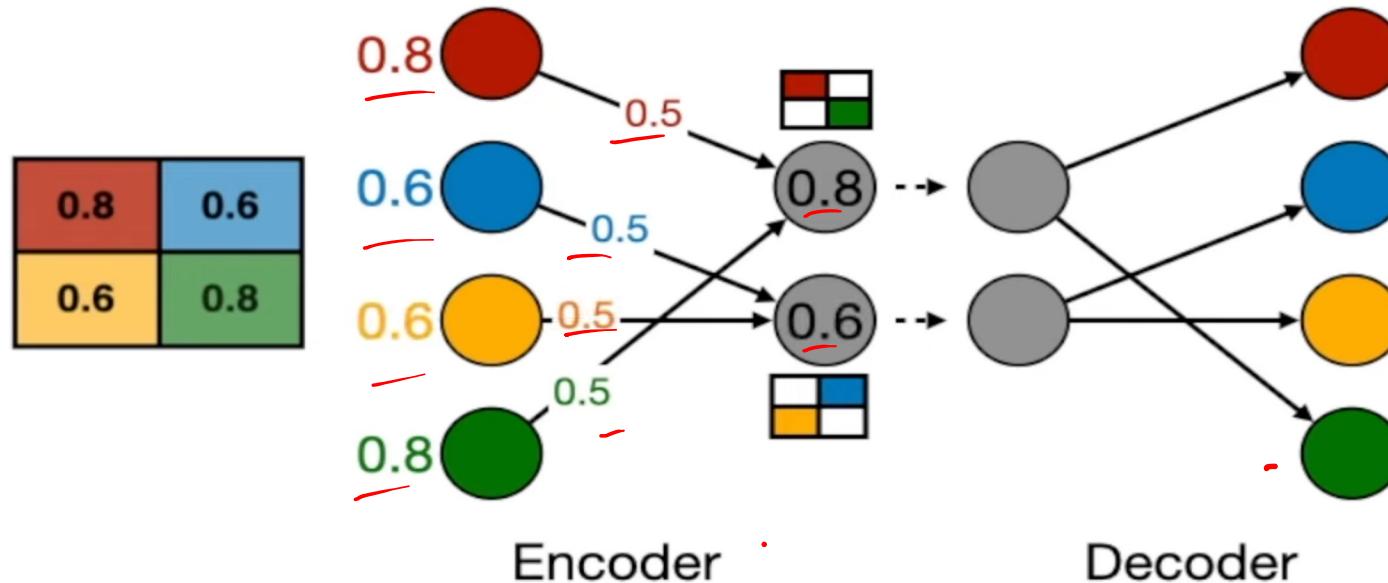
Autoencoder



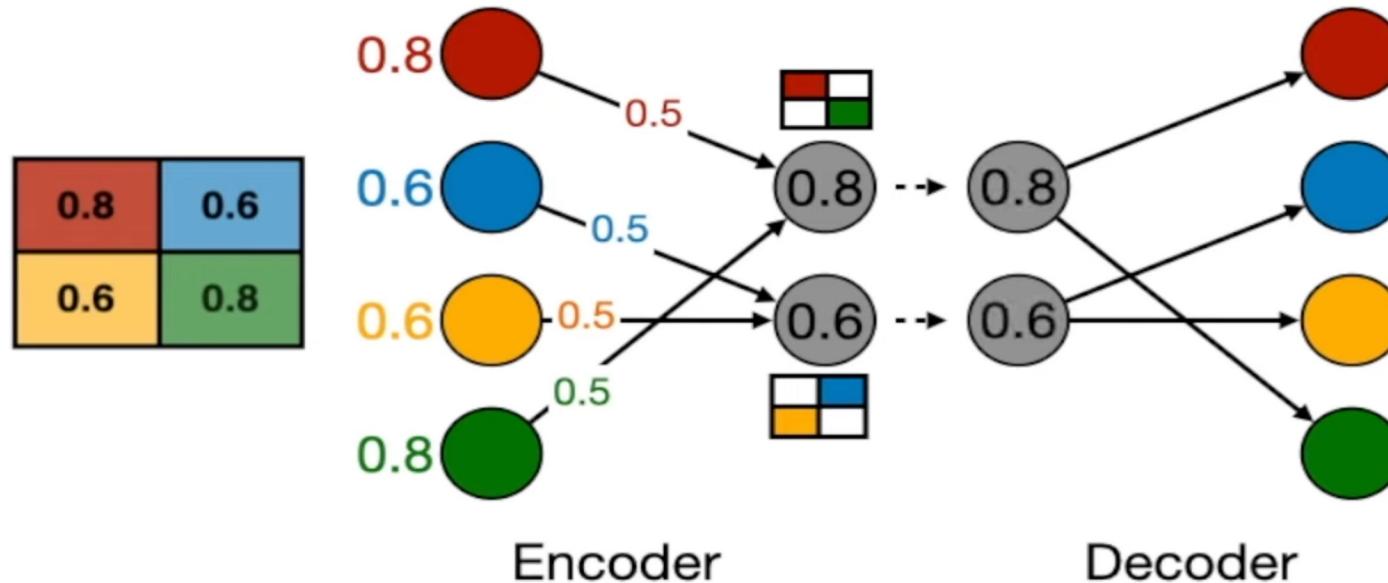
Autoencoder



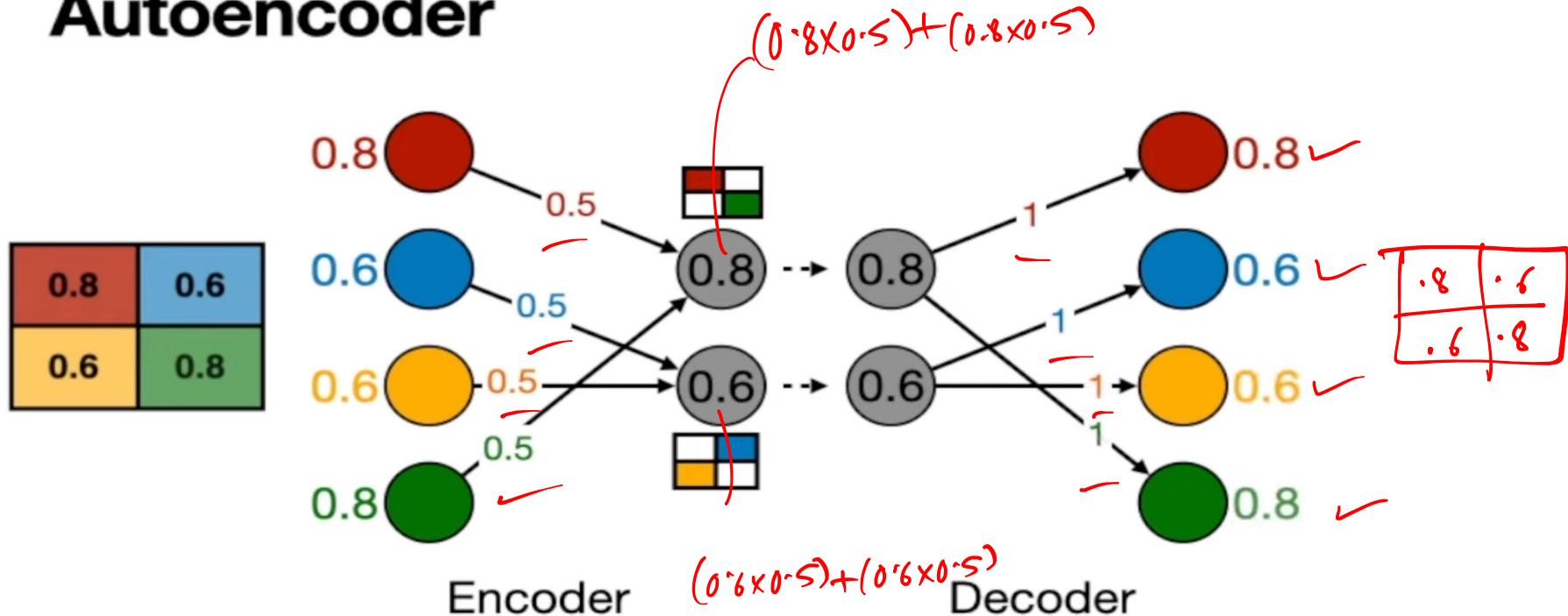
Autoencoder



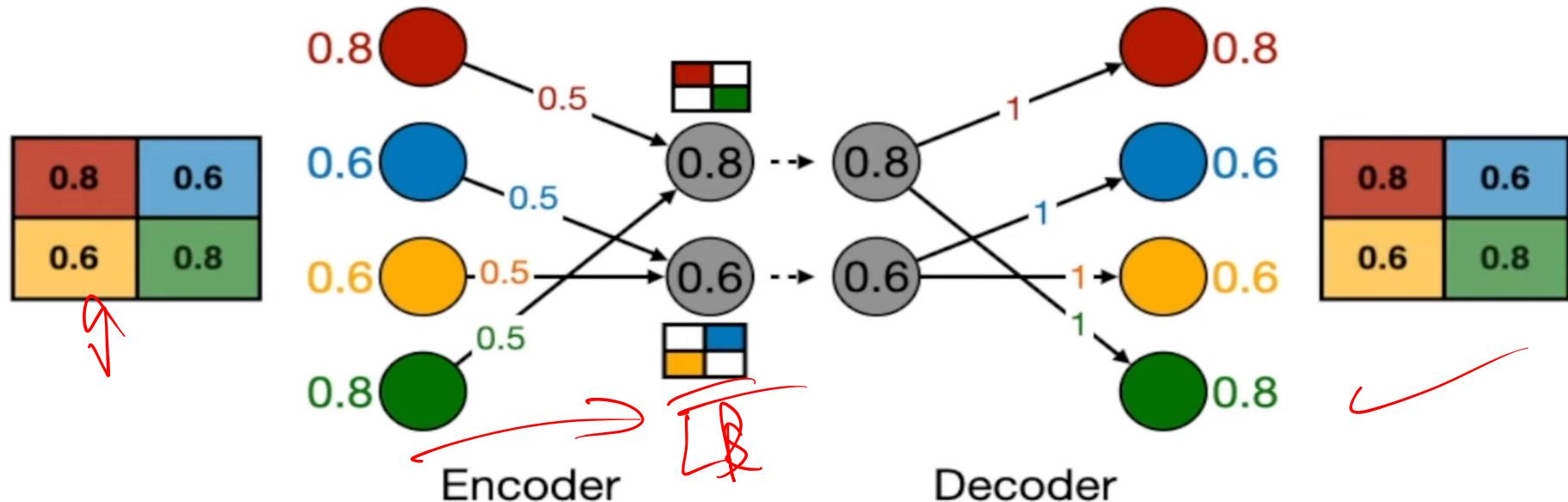
Autoencoder



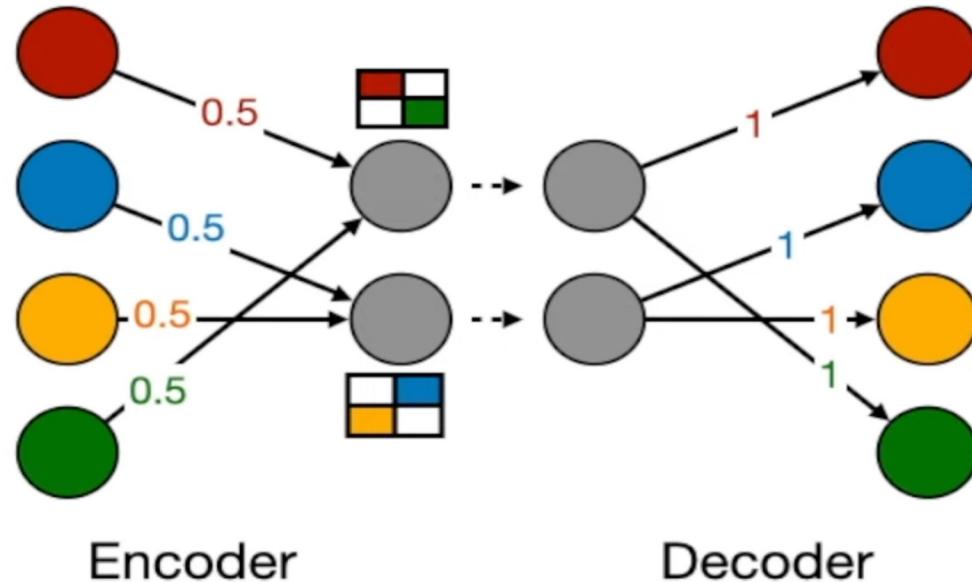
Autoencoder



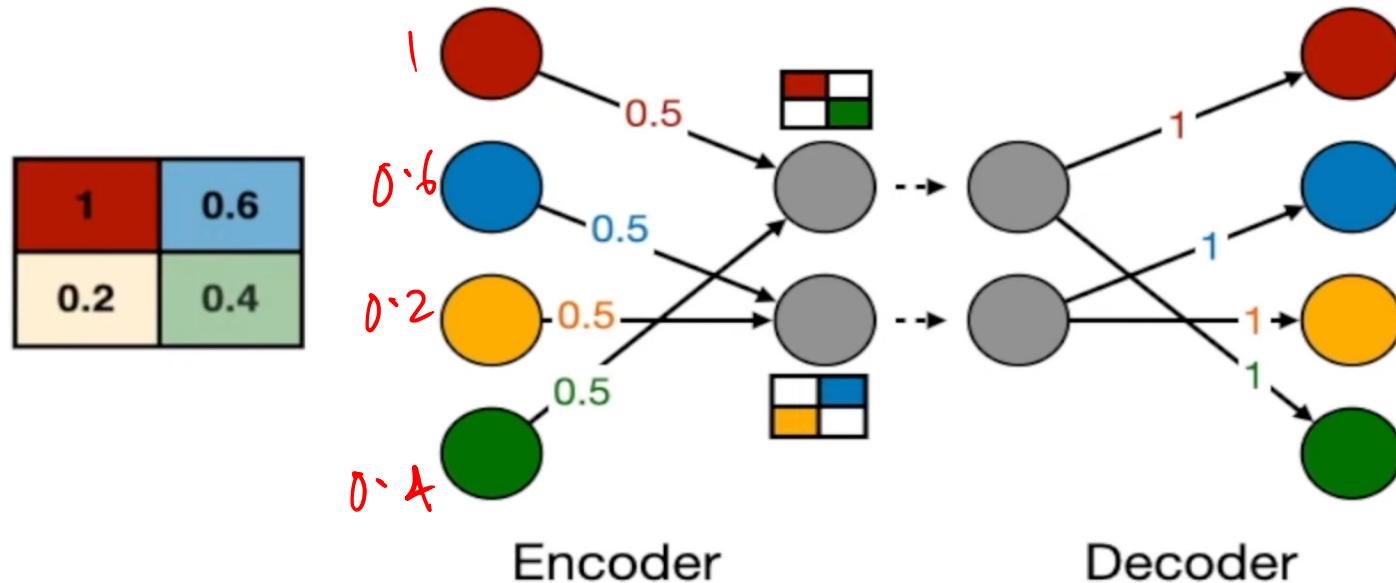
Autoencoder



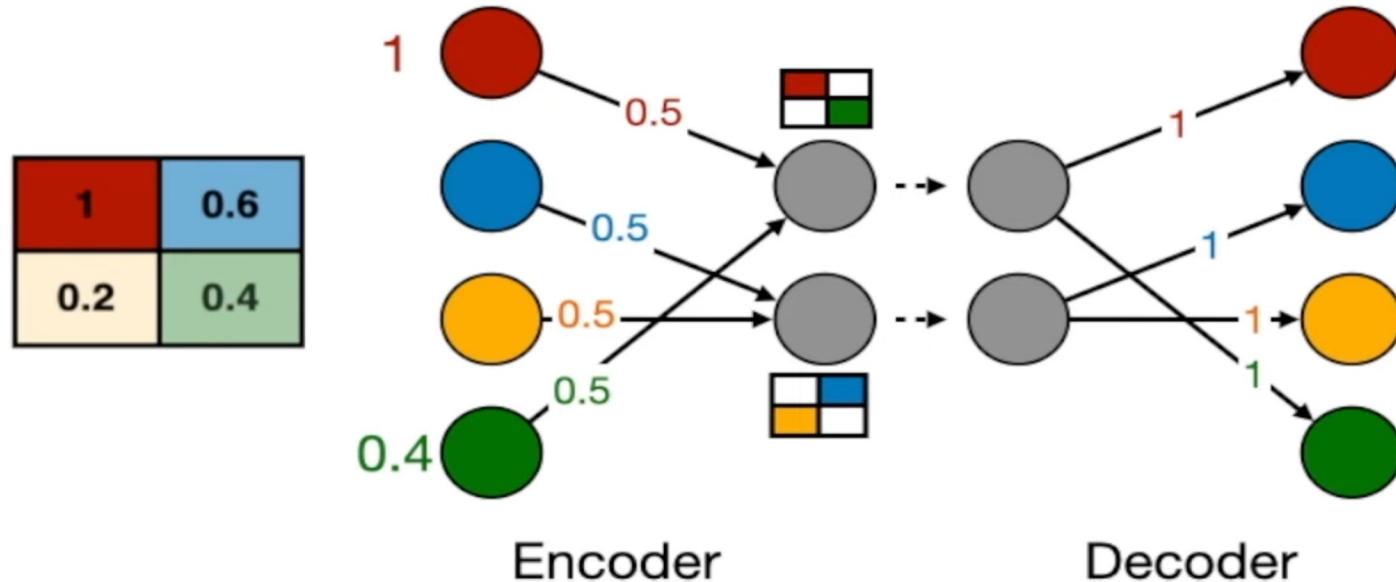
Denoising autoencoder



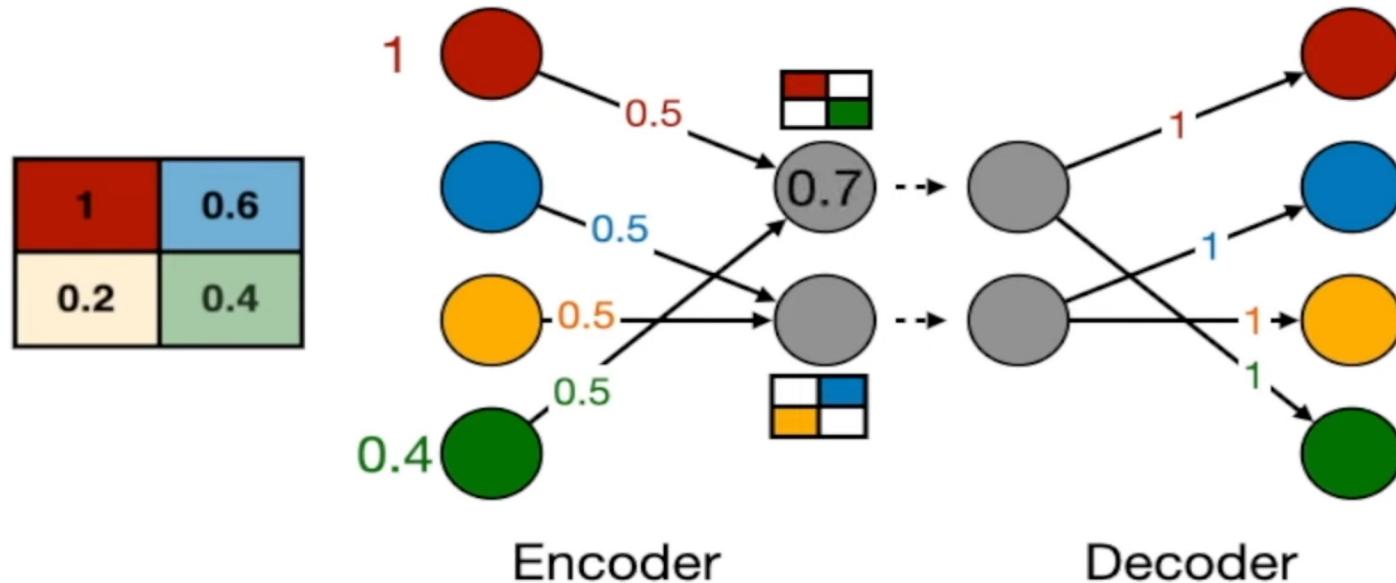
Denoising autoencoder



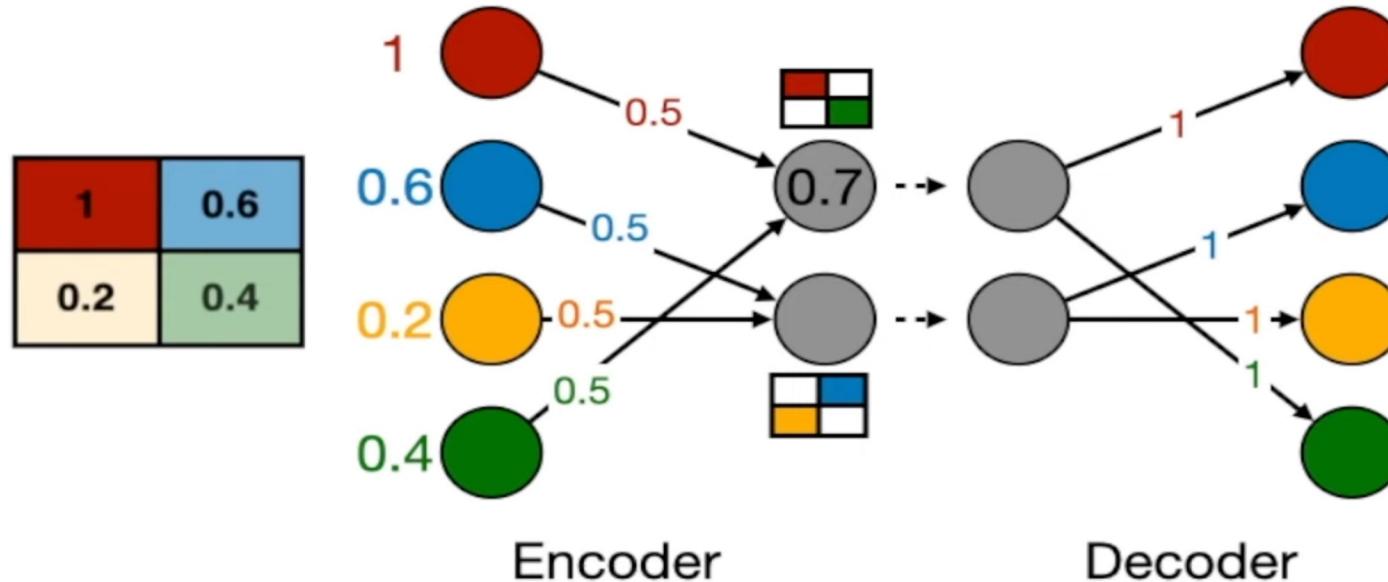
Denoising autoencoder



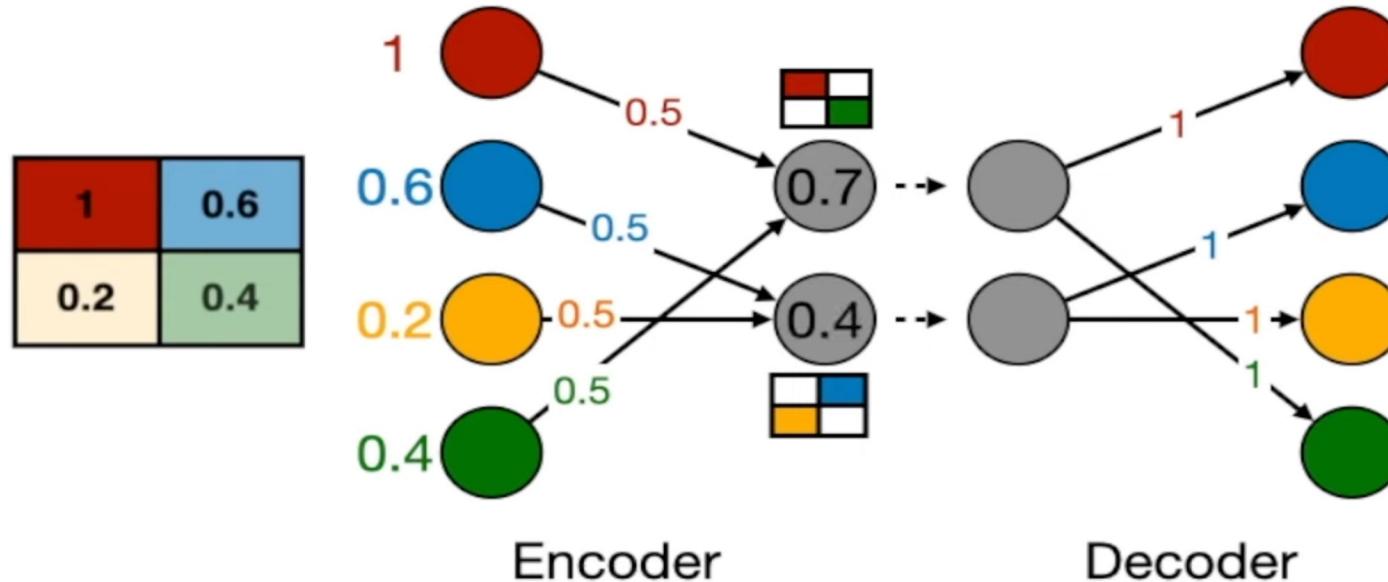
Denoising autoencoder



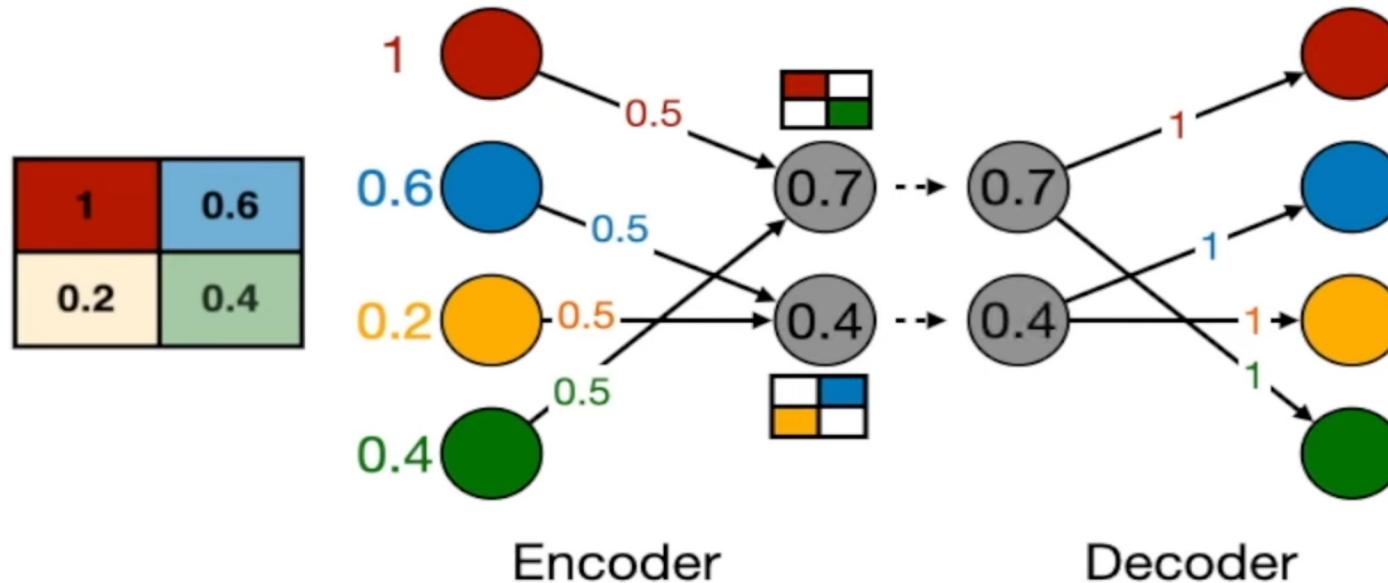
Denoising autoencoder



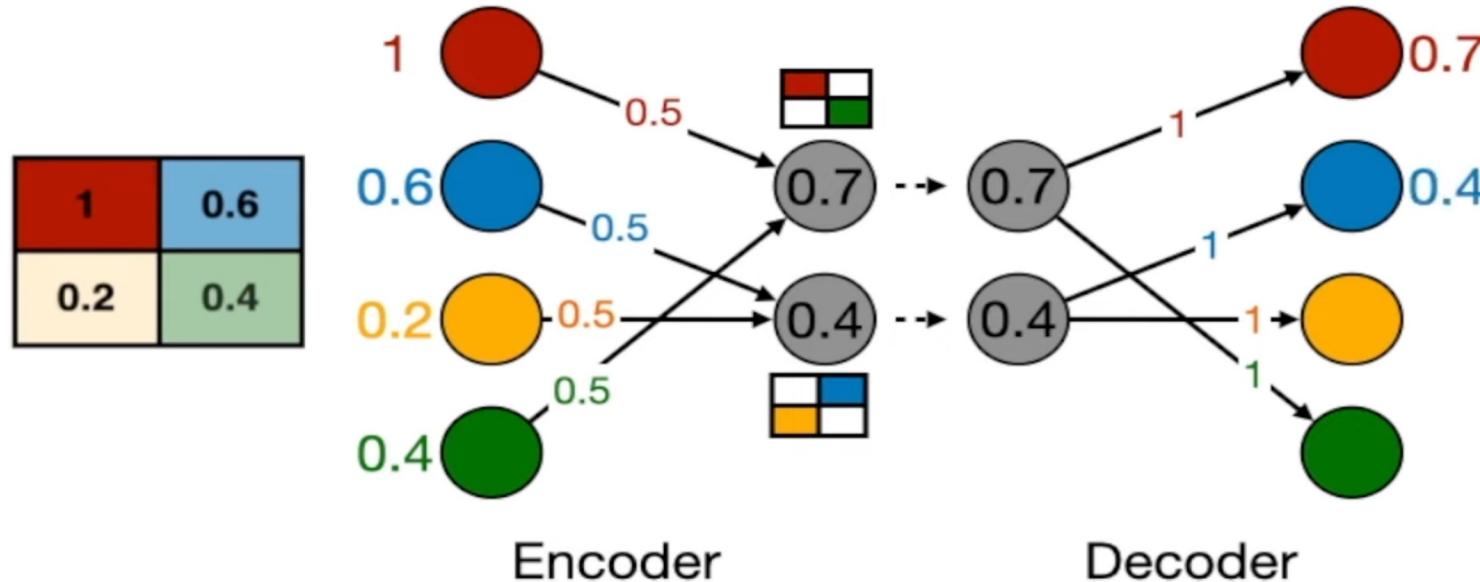
Denoising autoencoder



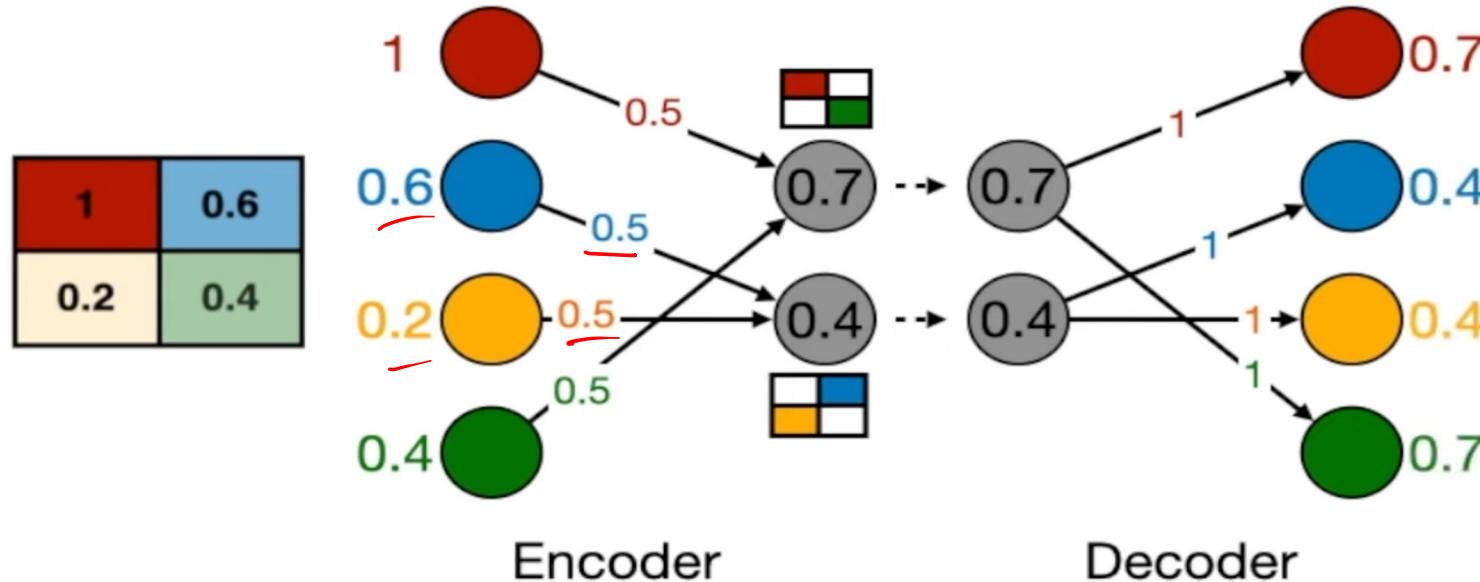
Denoising autoencoder



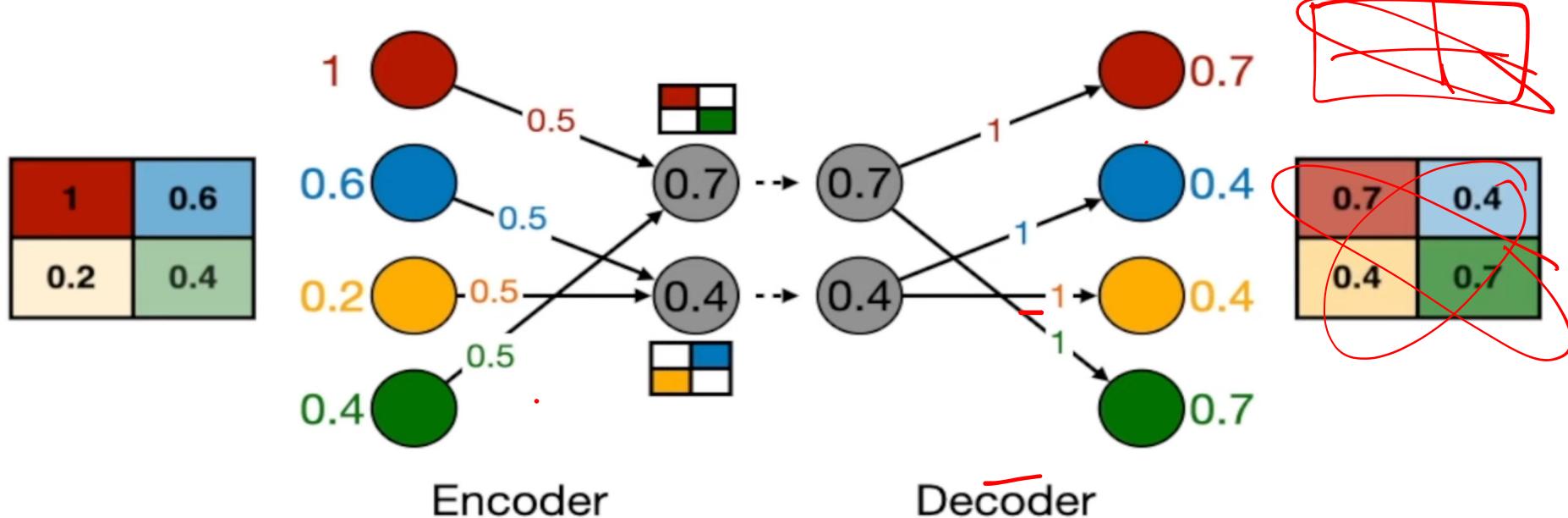
Denoising autoencoder



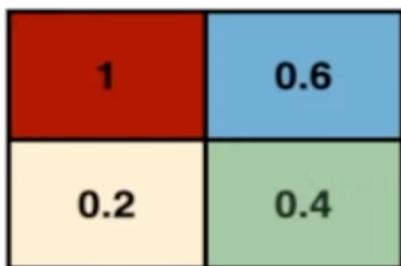
Denoising autoencoder



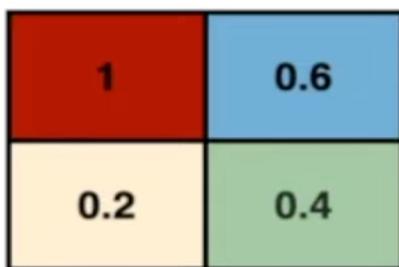
Denoising autoencoder



Correct noise



Correct noise



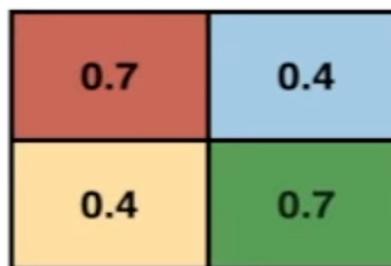
Denoise



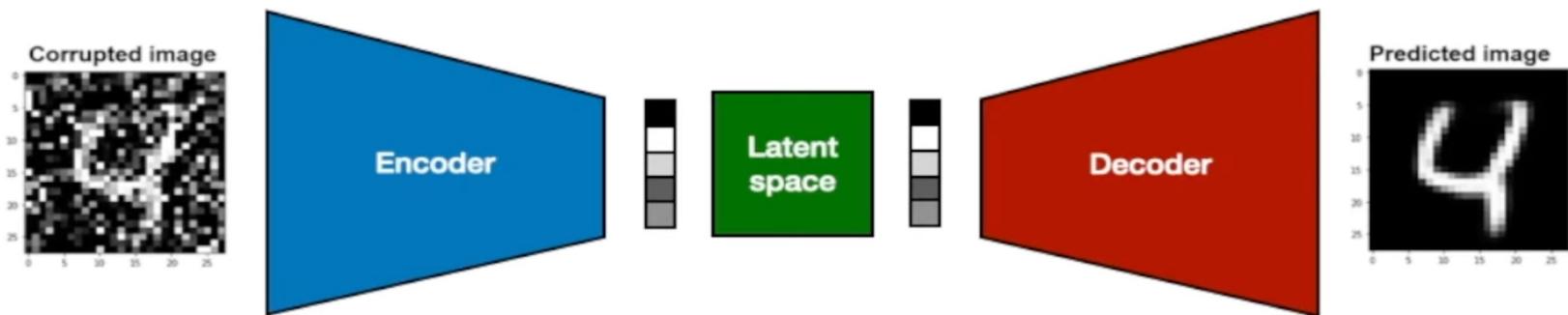
Correct noise



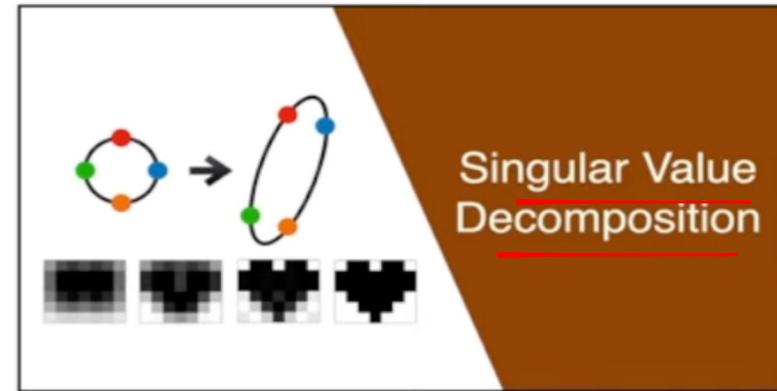
Denoise
----->



Denoising autoencoders

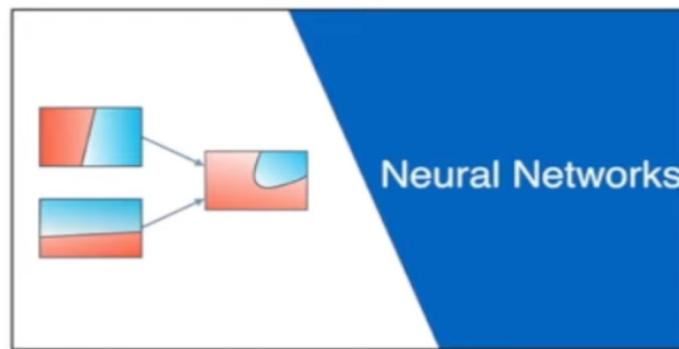
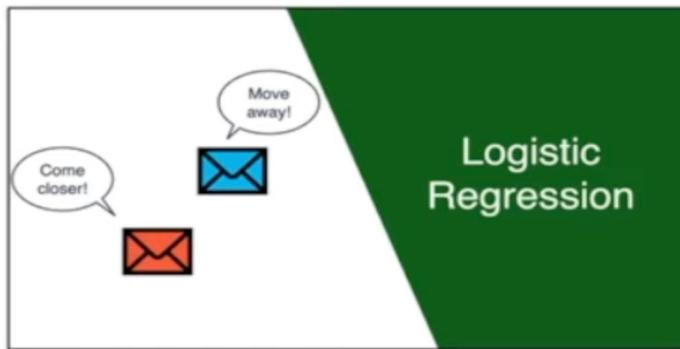


Linear methods

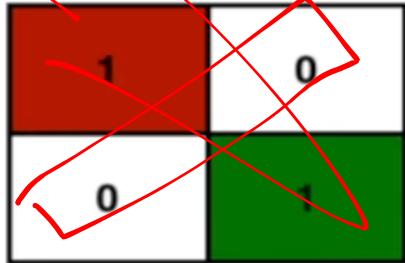


Denoising autoencoders

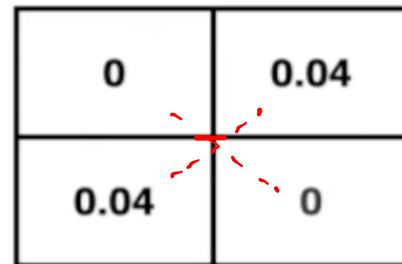
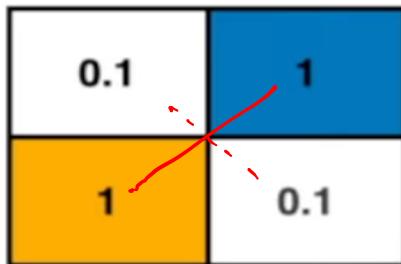
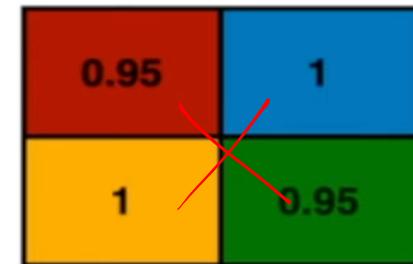
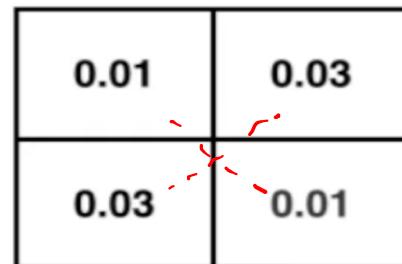
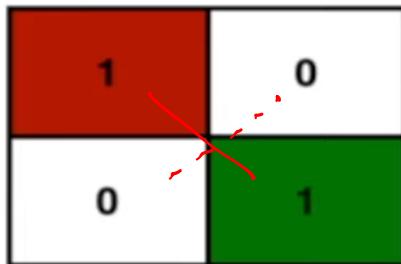
A friendly introduction to deep learning and neural networks



Dataset of images

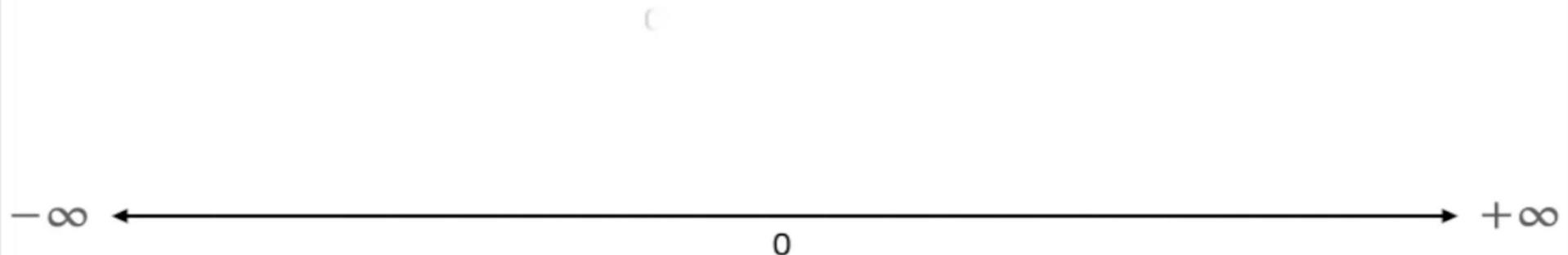


Dataset of images

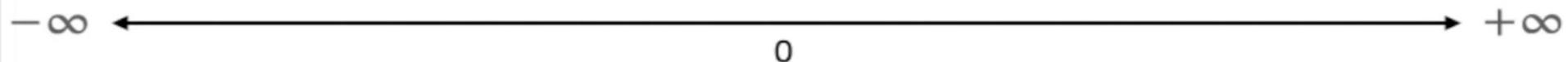
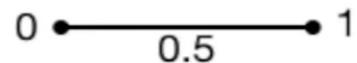


Mapping the real numbers to the interval $(0,1)$

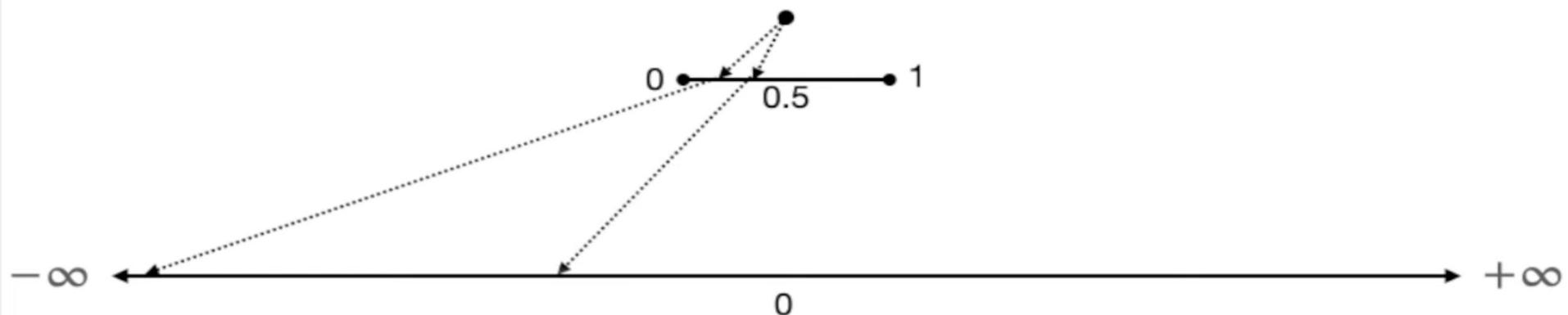
Mapping the real numbers to the interval $(0,1)$



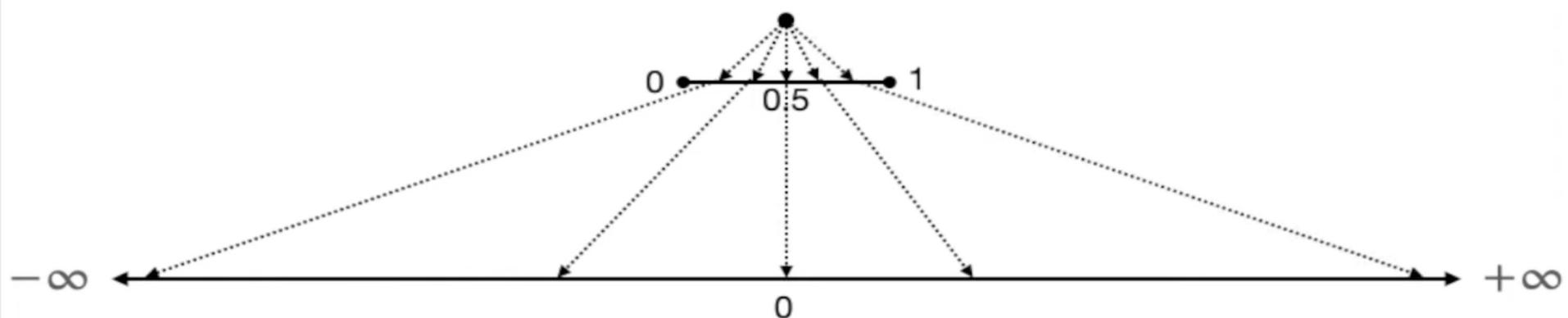
Mapping the real numbers to the interval $(0,1)$



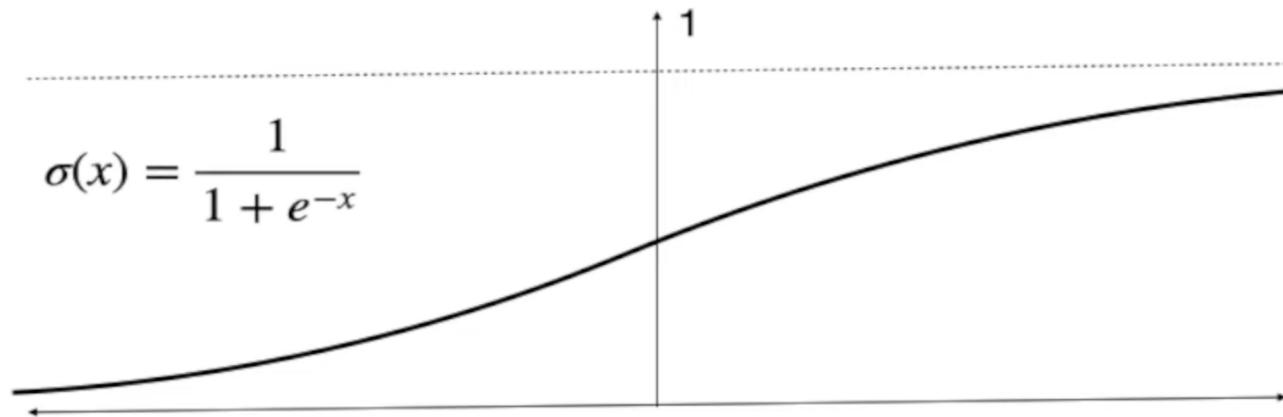
Mapping the real numbers to the interval $(0,1)$



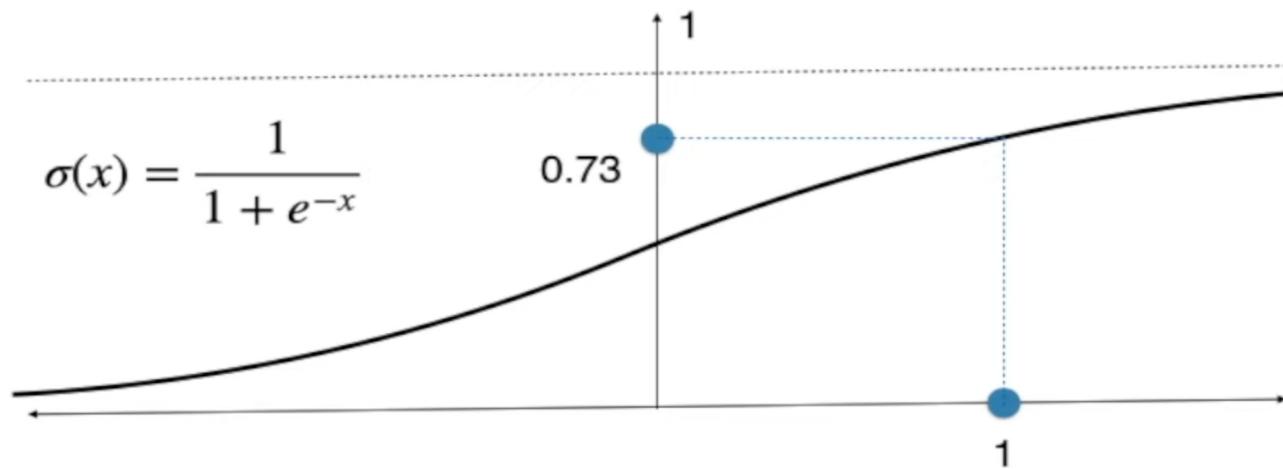
Mapping the real numbers to the interval $(0,1)$



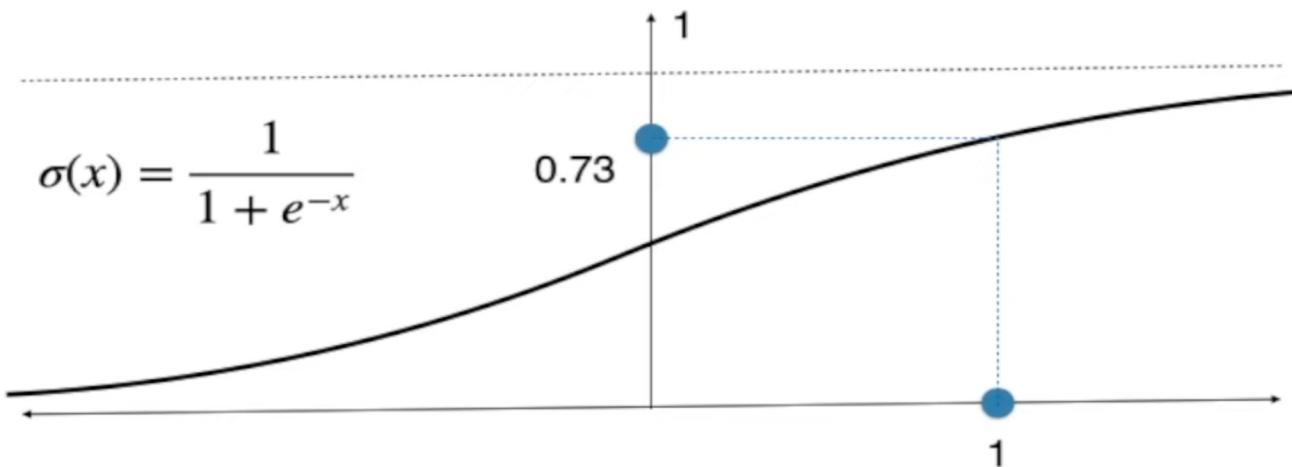
Sigmoid function



Sigmoid function

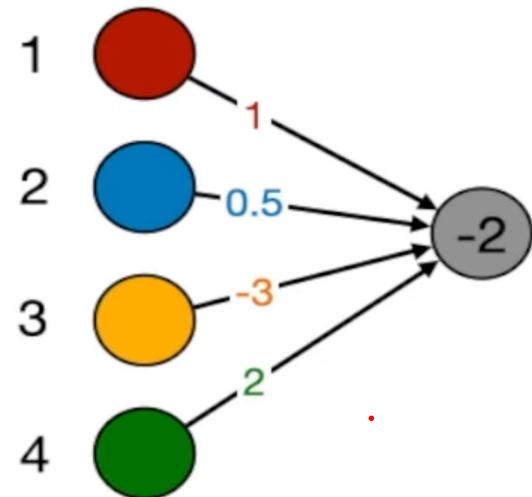


Sigmoid function

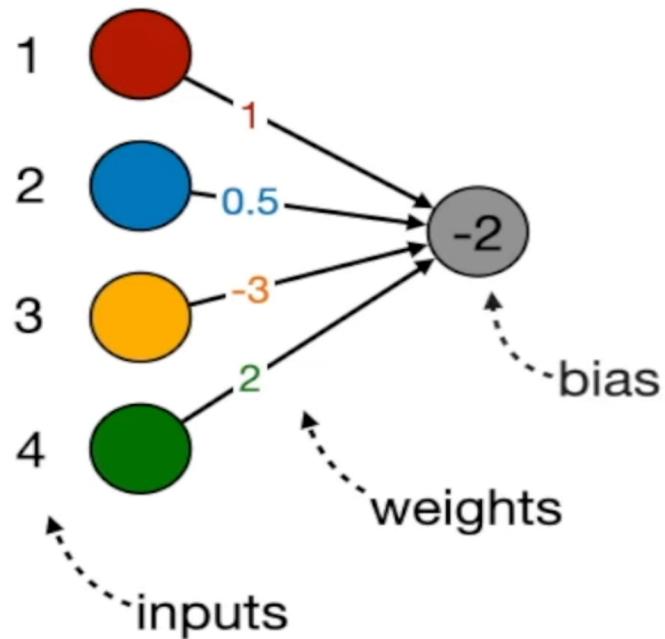


$$\begin{aligned}\sigma(-5) &= 0.01 \\ \sigma(-2) &= 0.12 \\ \sigma(-1) &= 0.27 \\ \sigma(0) &= 0.5 \\ \sigma(1) &= 0.73 \\ \sigma(2) &= 0.88 \\ \sigma(5) &= 0.99\end{aligned}$$

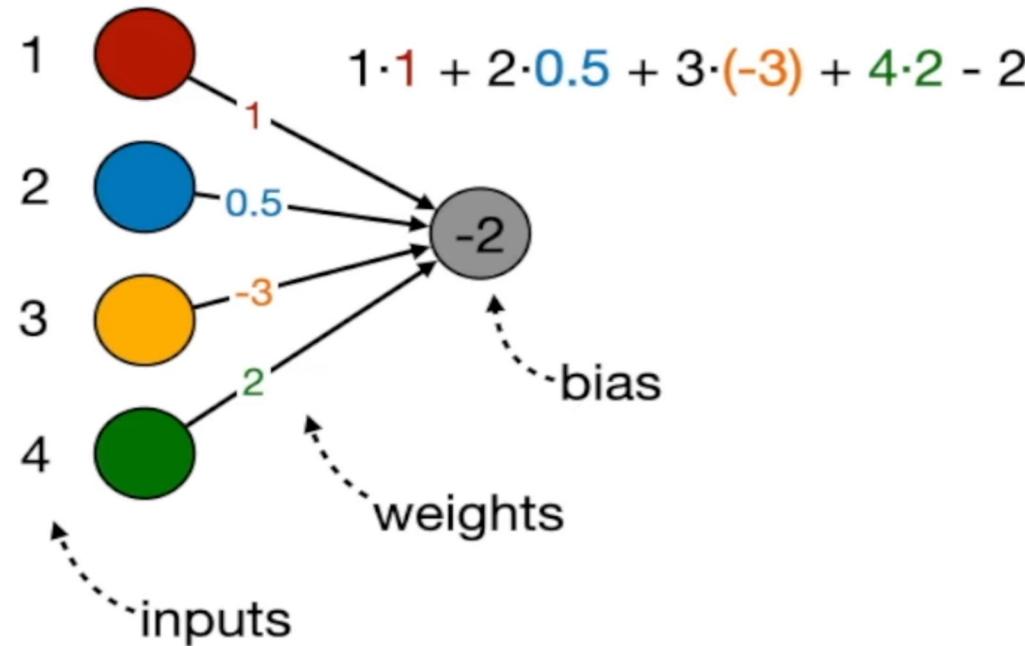
Perceptron



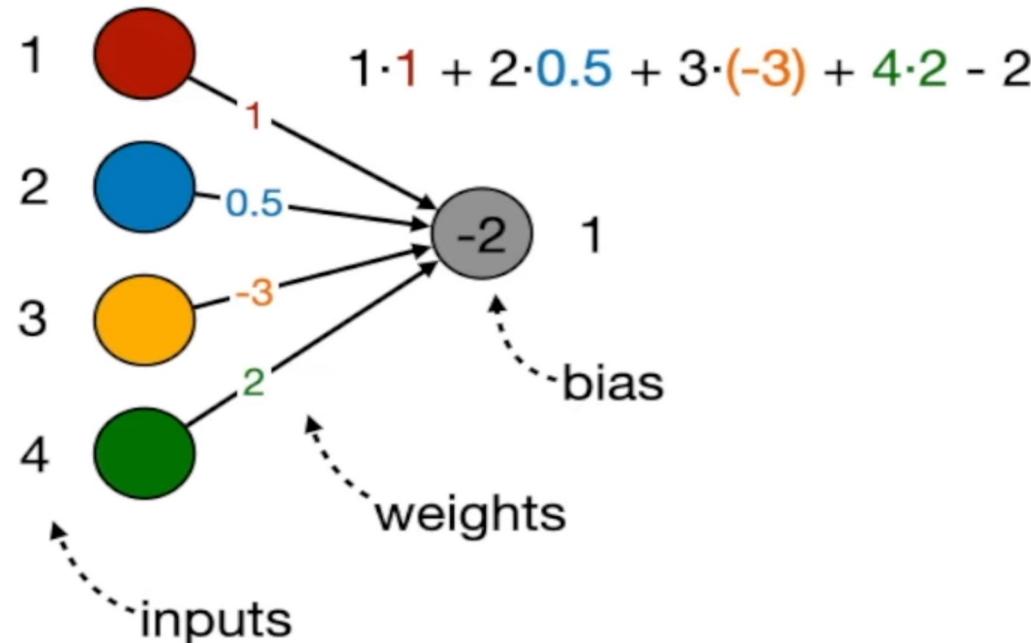
Perceptron



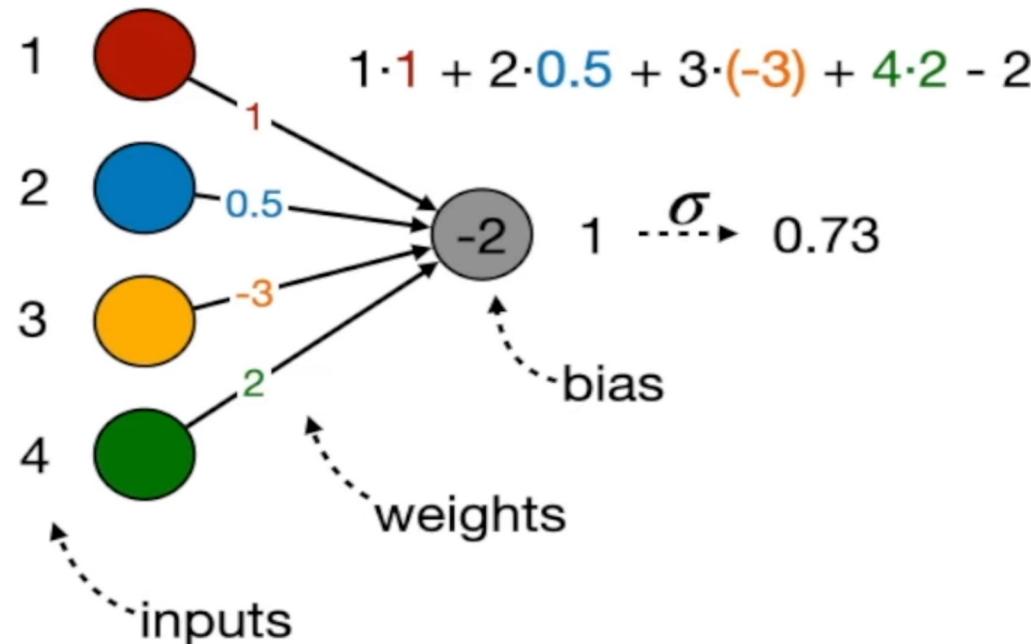
Perceptron



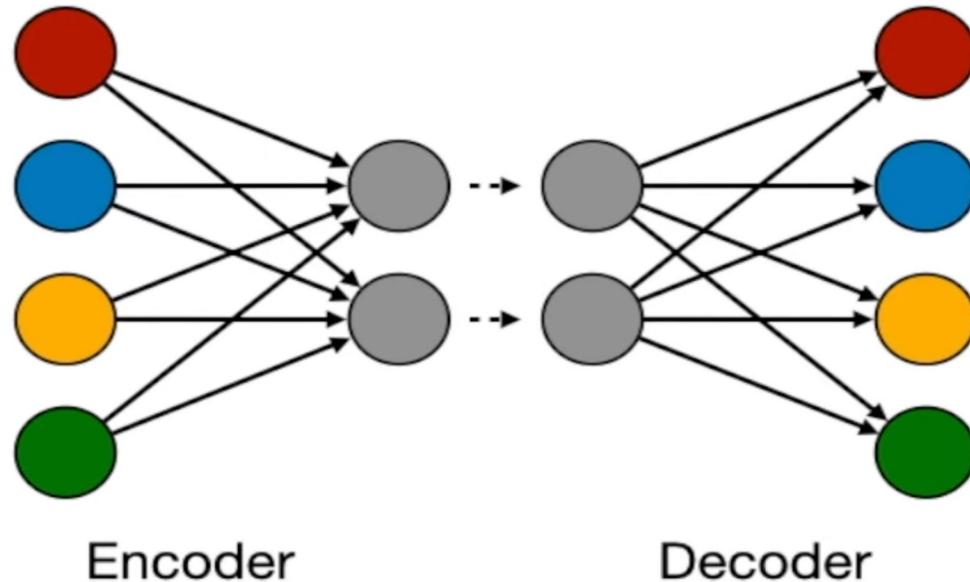
Perceptron



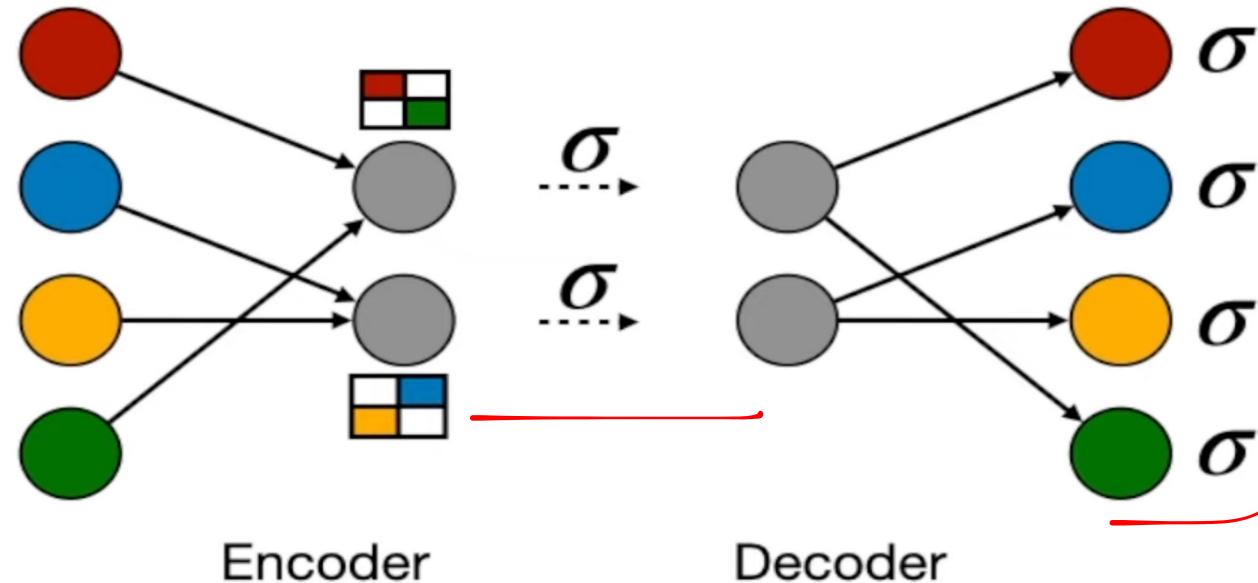
Perceptron



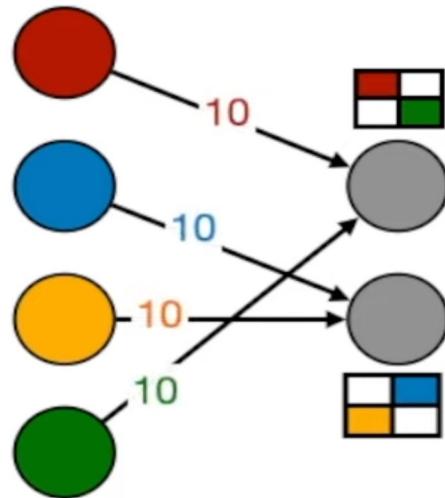
Autoencoder



Autoencoder

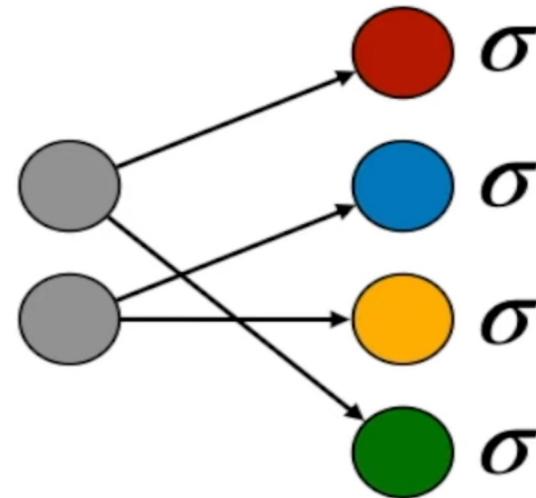


Autoencoder



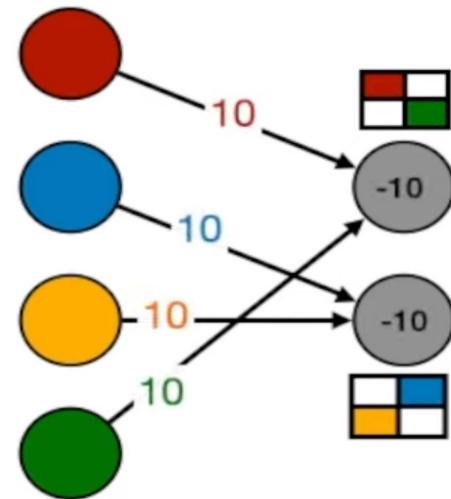
Encoder

σ σ



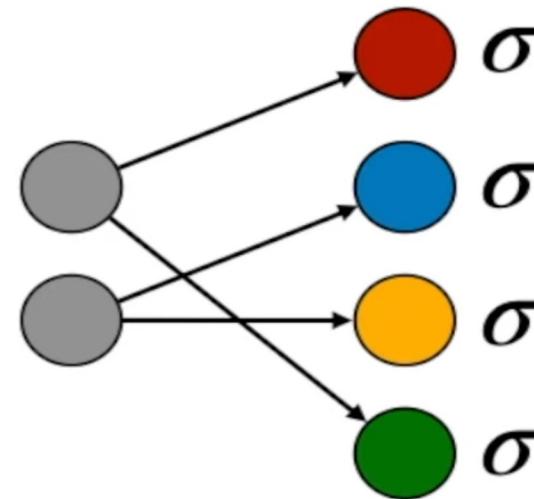
Decoder

Autoencoder



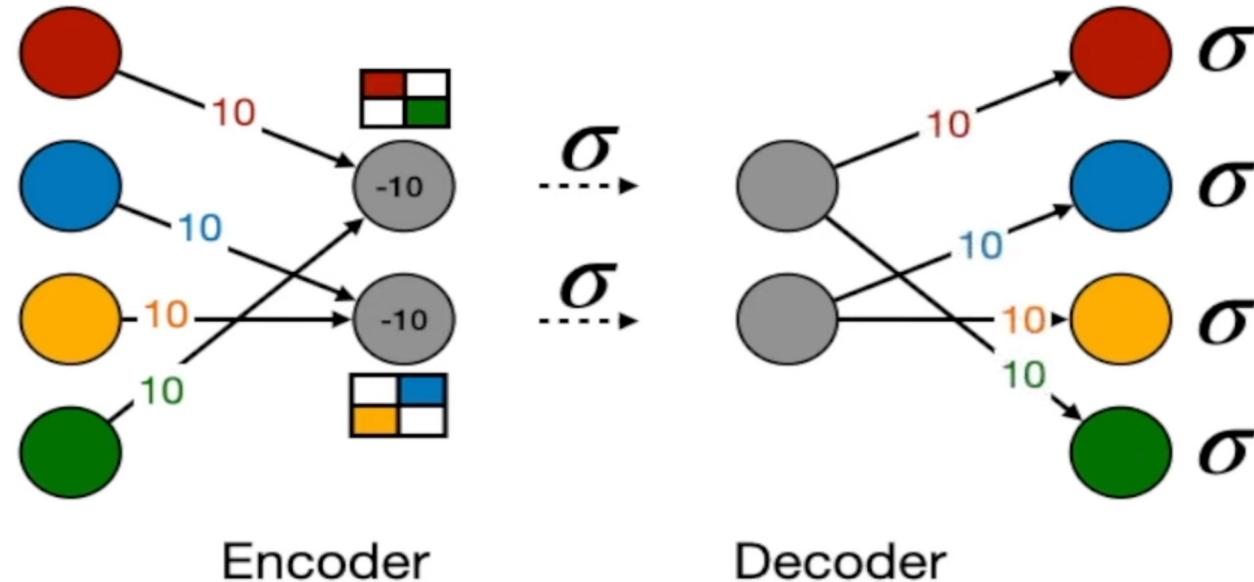
Encoder

σ σ

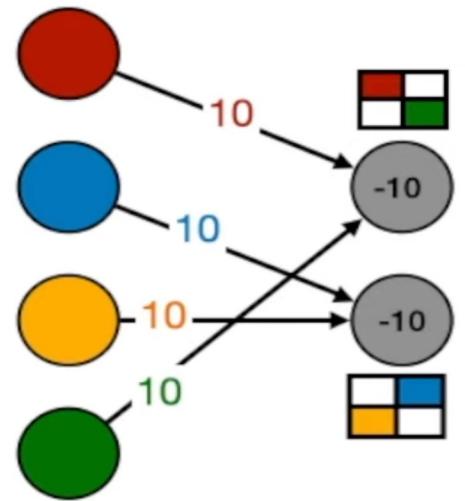


Decoder

Autoencoder

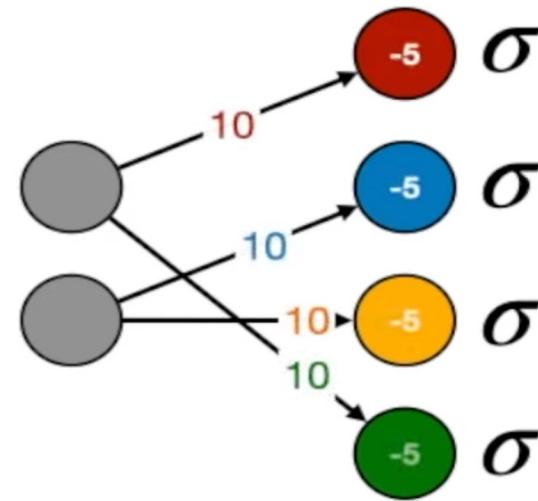


Autoencoder



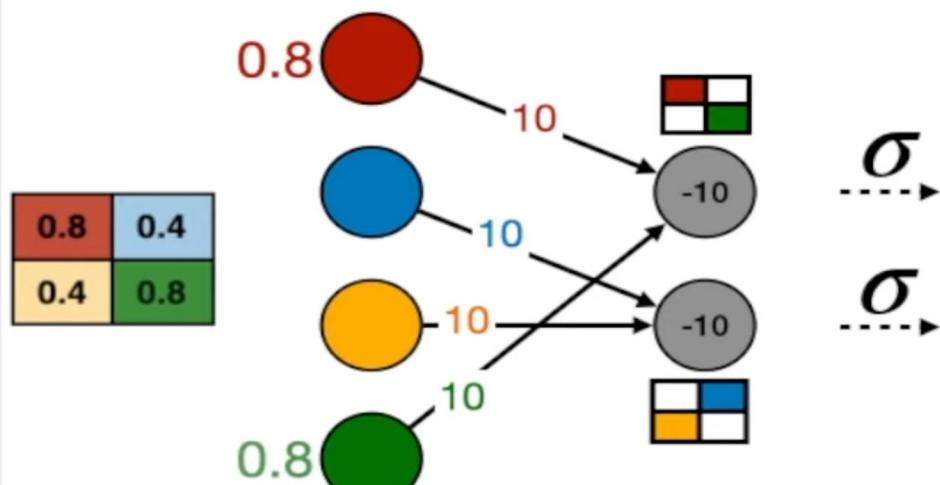
Encoder

σ σ

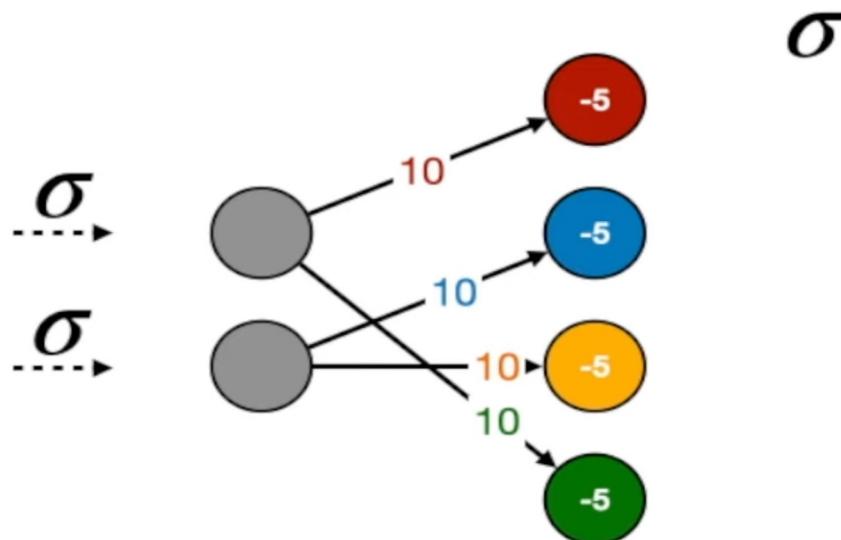


Decoder

Autoencoder

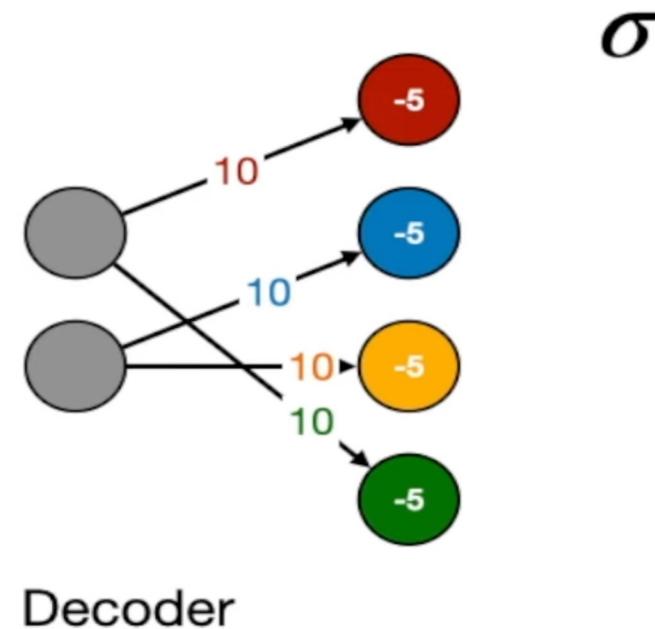
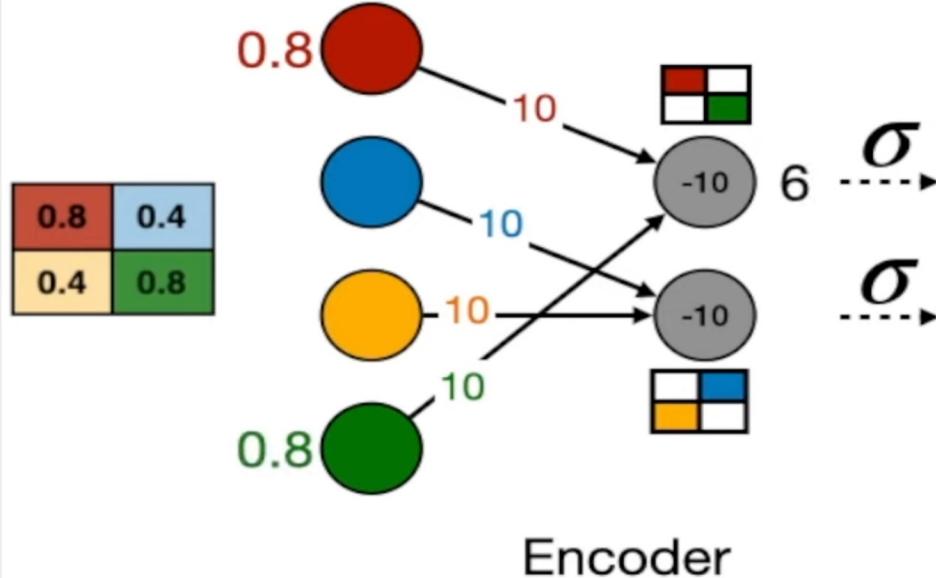


Encoder

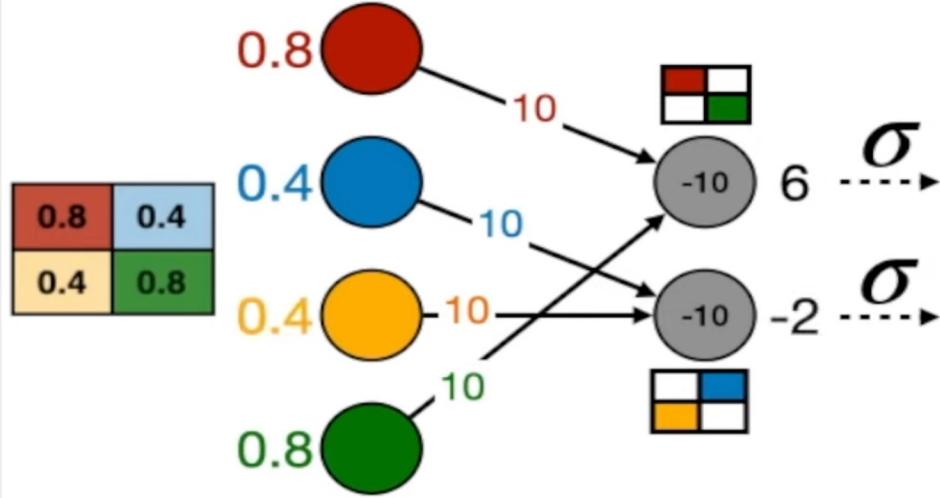


Decoder

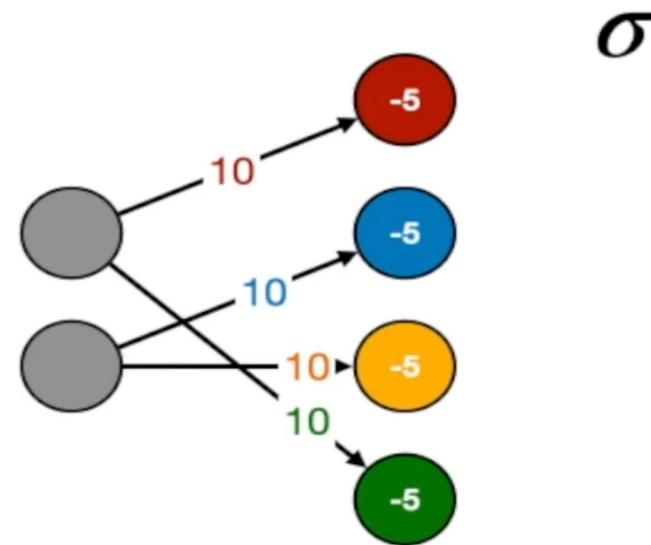
Autoencoder



Autoencoder

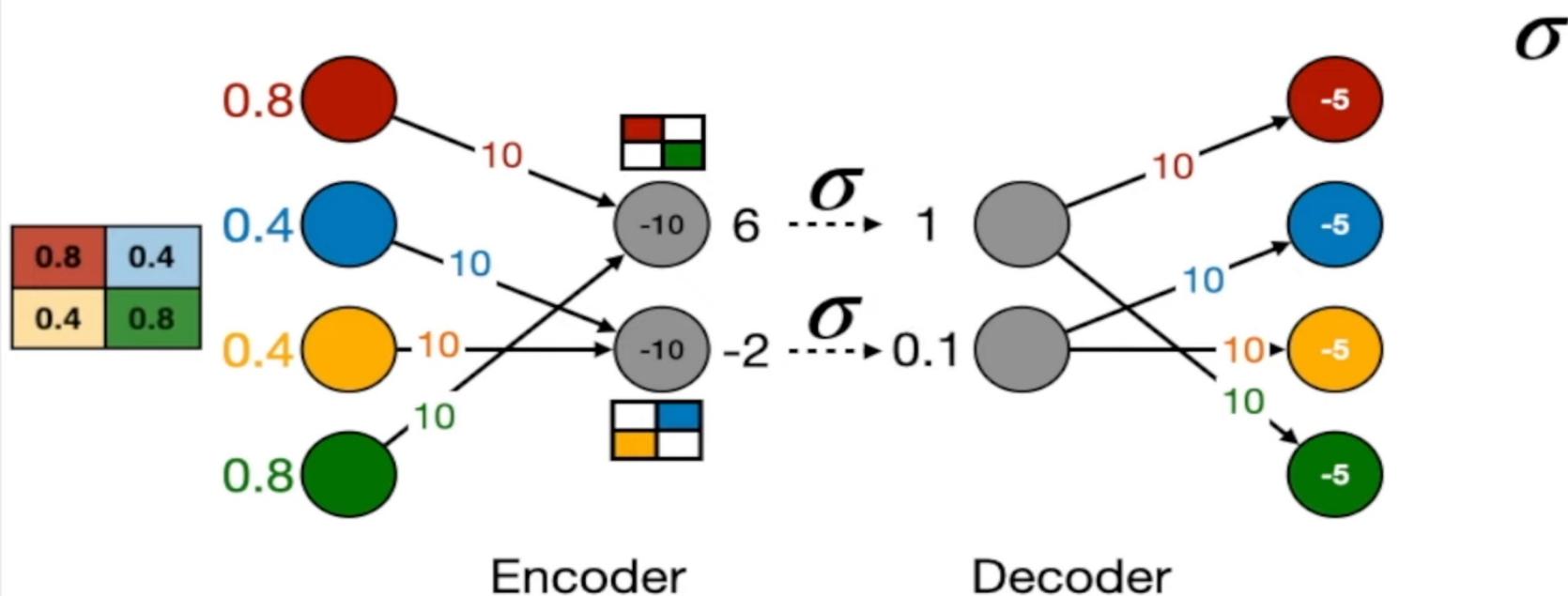


Encoder

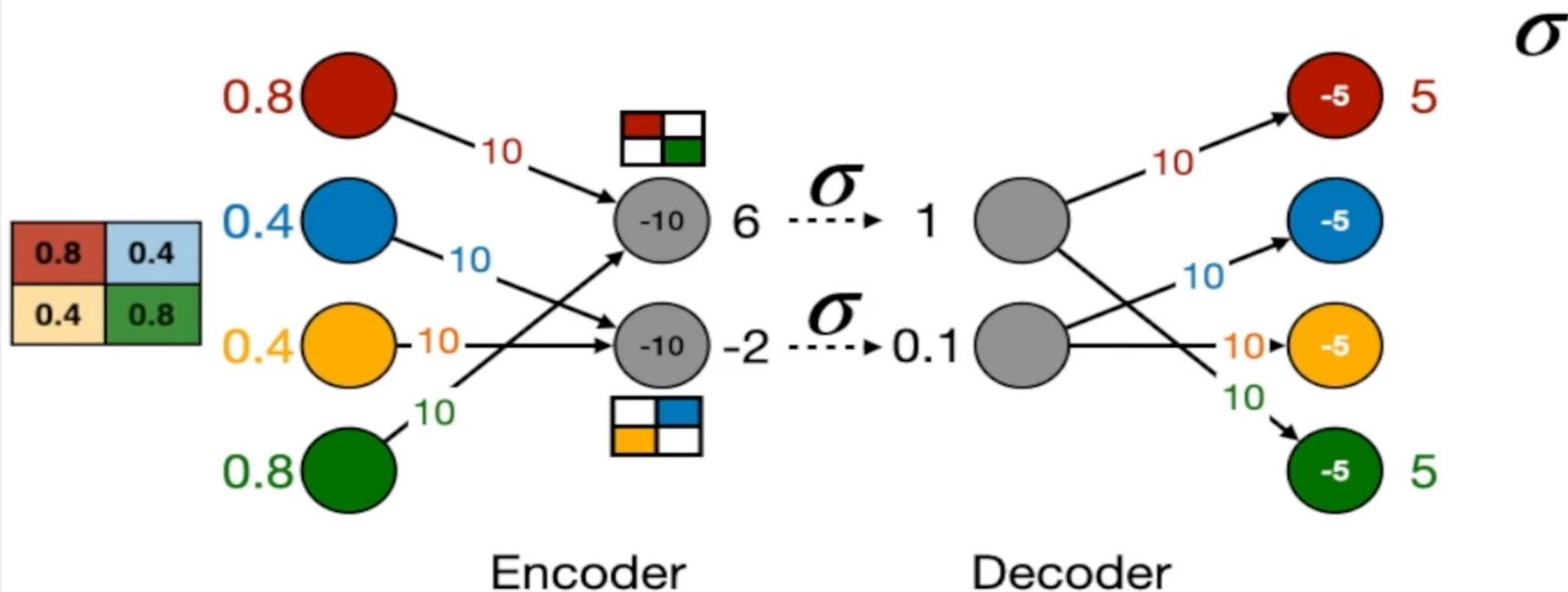


Decoder

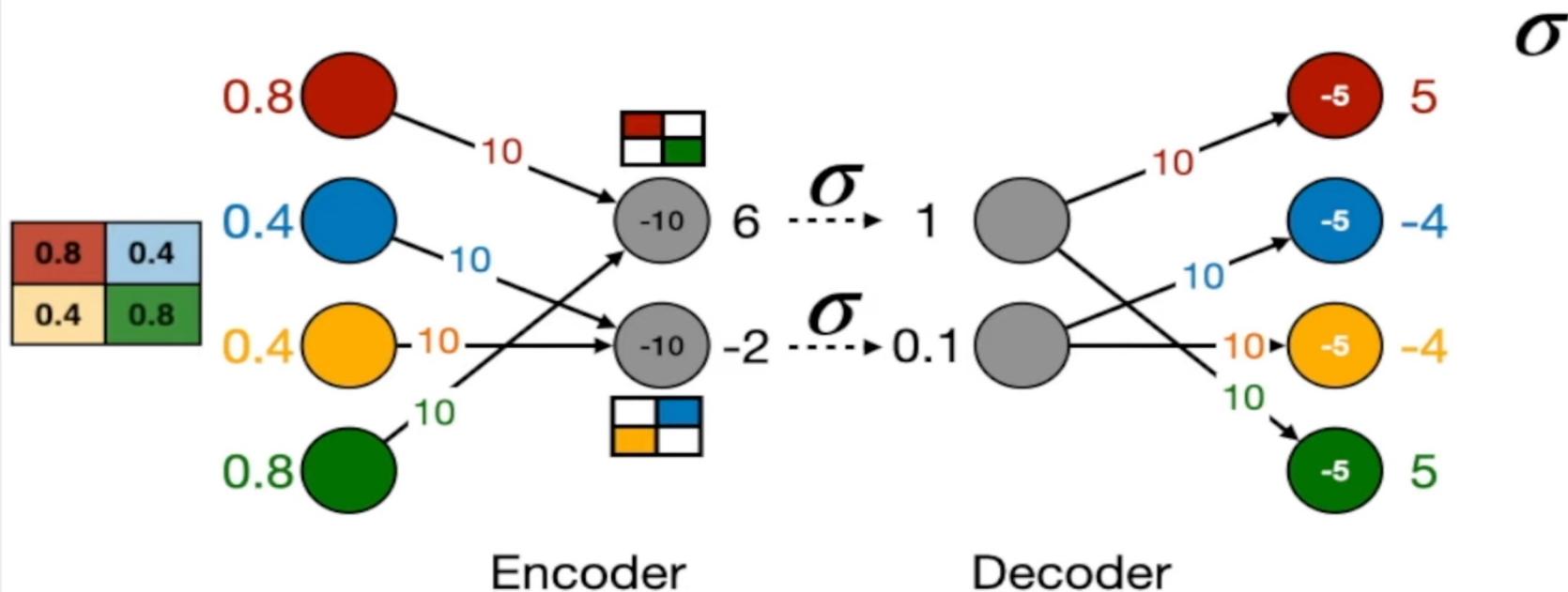
Autoencoder



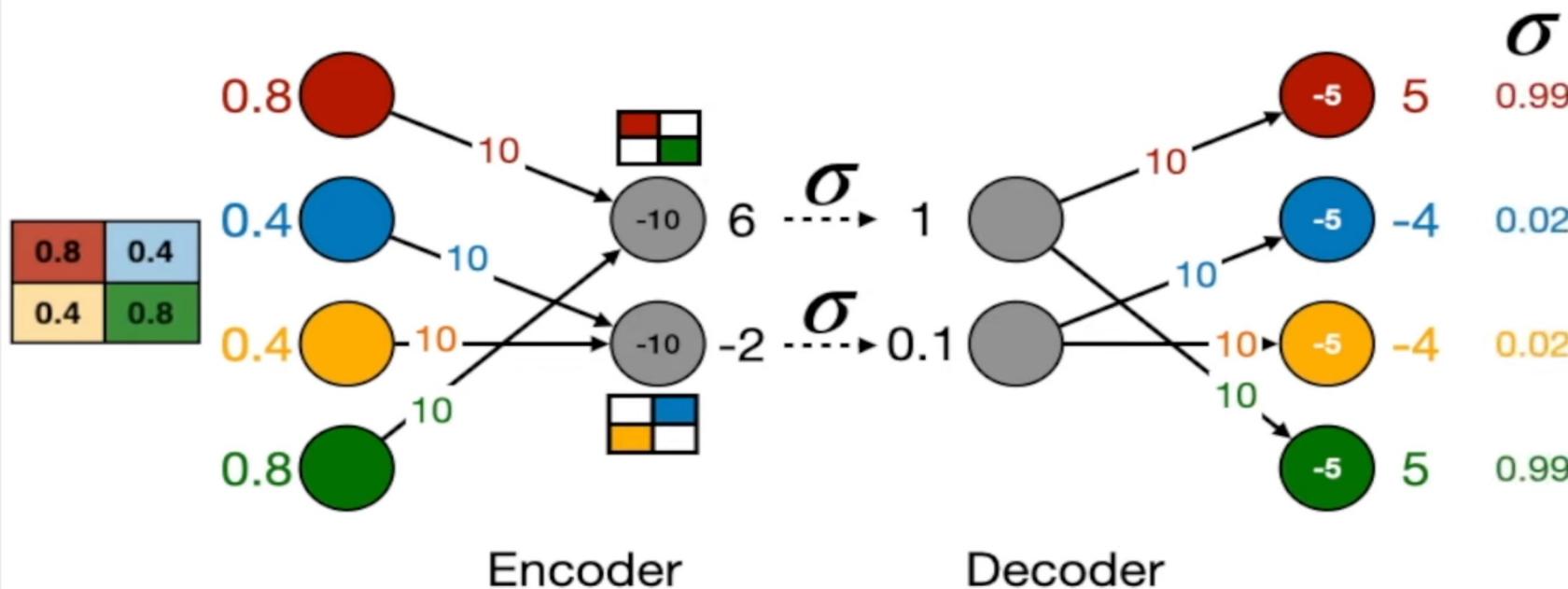
Autoencoder



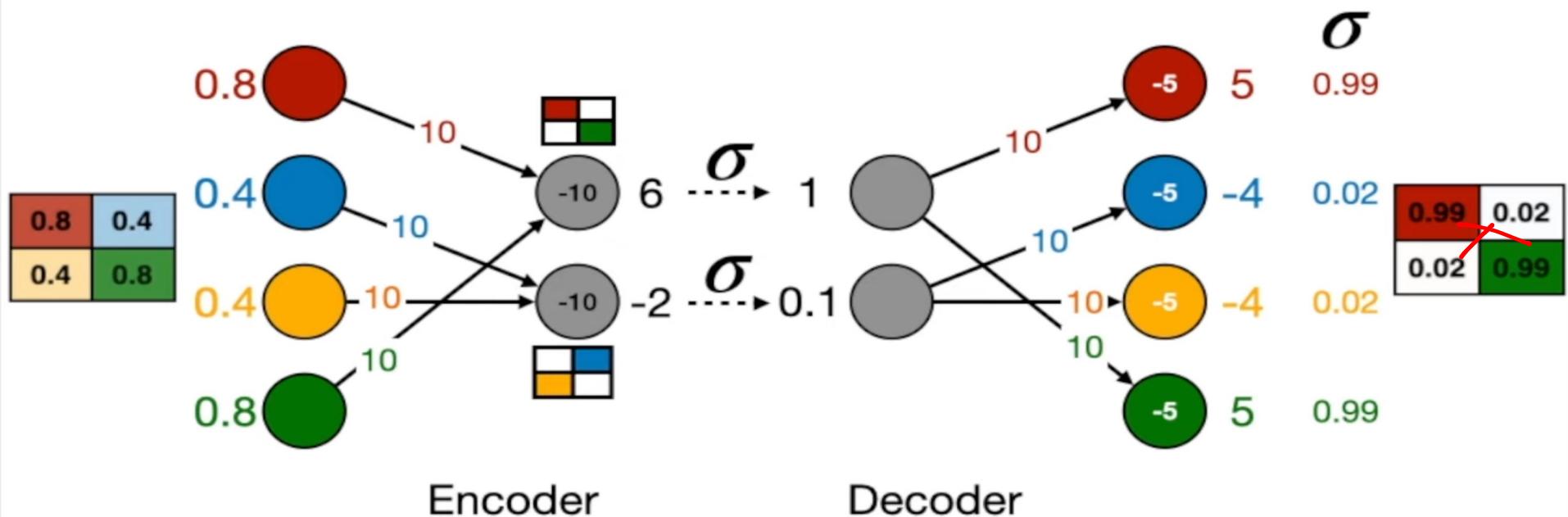
Autoencoder



Autoencoder



Autoencoder



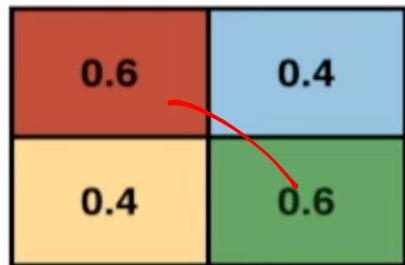
Correct noise



Correct noise



Correct noise



Denoise
→



Correct noise



Denoise
→



Correct noise



Denoise
→



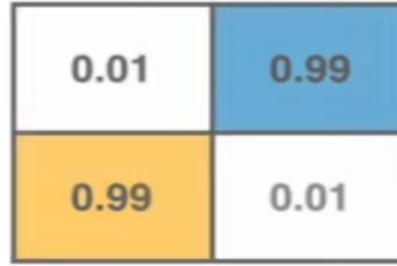
Correct noise



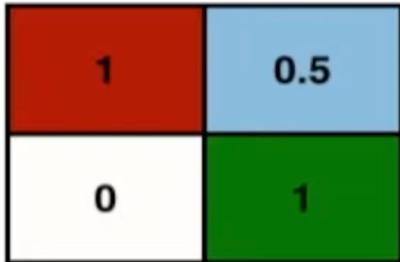
Denoise
→



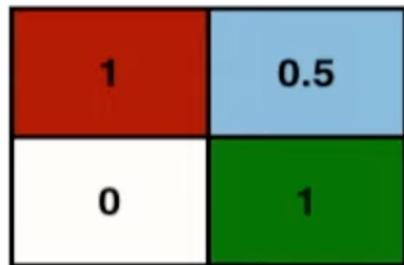
Denoise
→



Correct noise



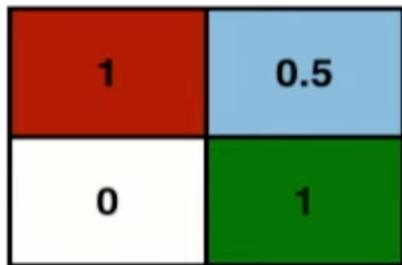
Correct noise



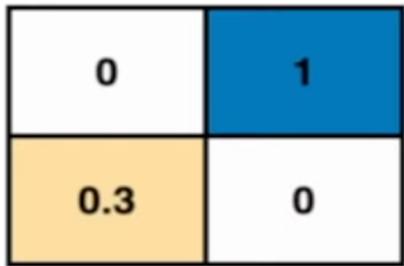
Denoise →



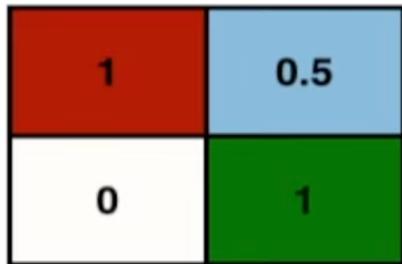
Correct noise



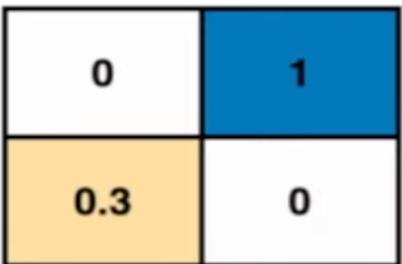
Denoise
→



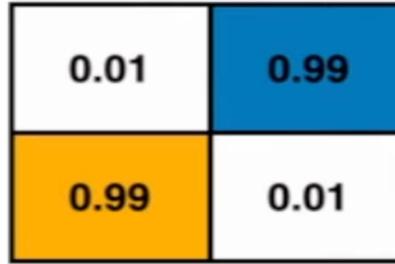
Correct noise



Denoise →



Denoise →



Correct noise



Correct noise



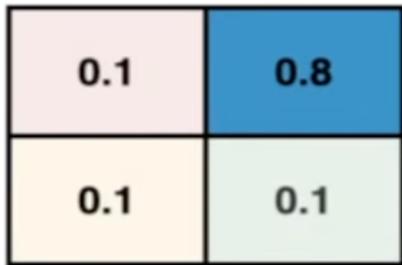
Denoise
→



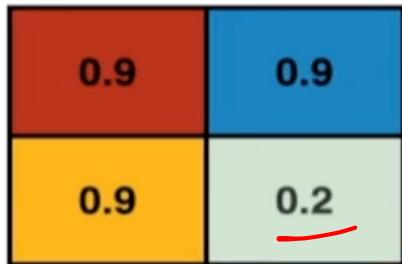
Correct noise



Denoise
→



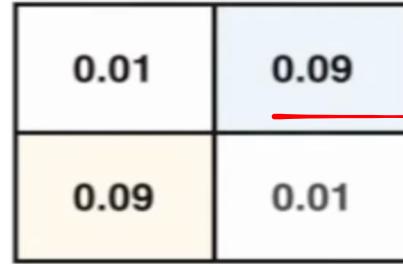
Correct noise



Denoise
→



Denoise
→



More layers

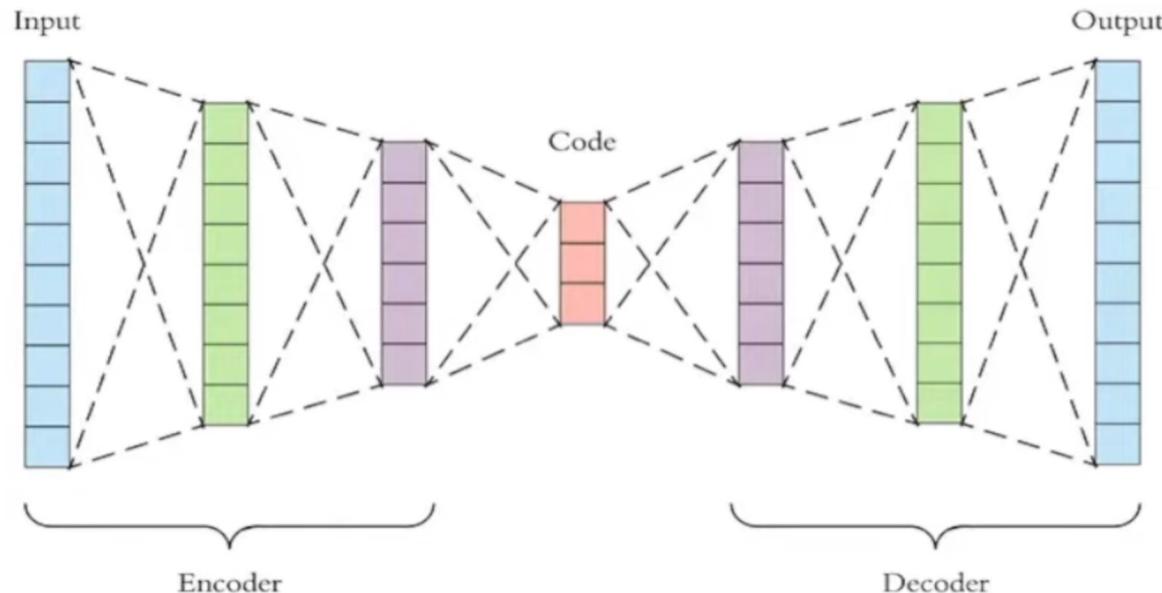


Photo credit: Applied Deep Learning. Arden Dertat

More layers

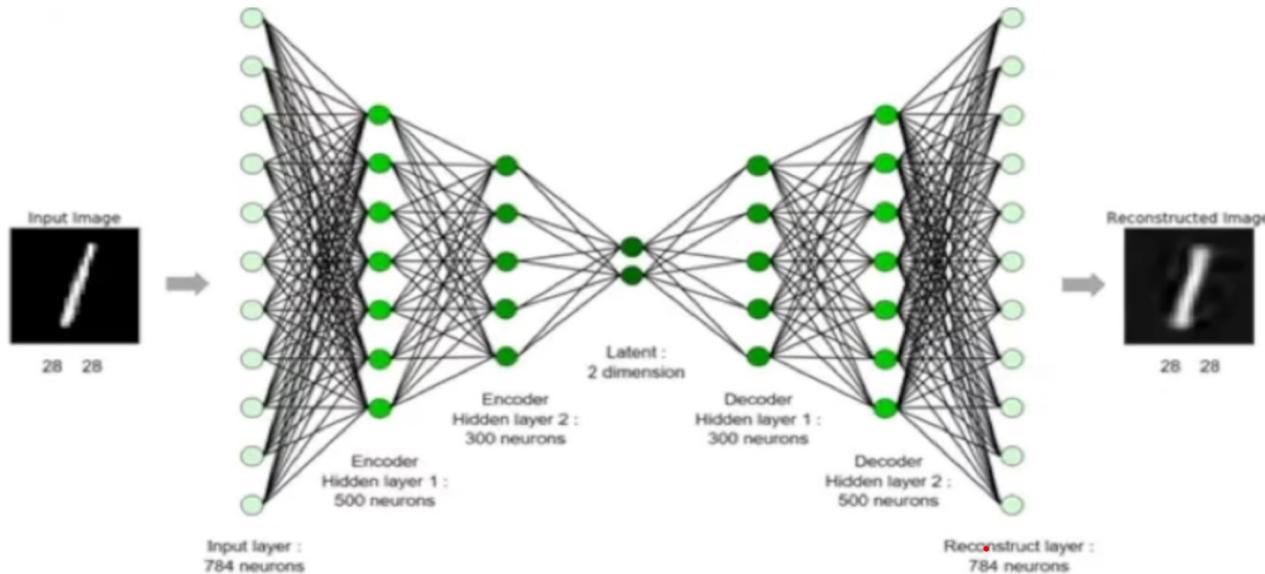
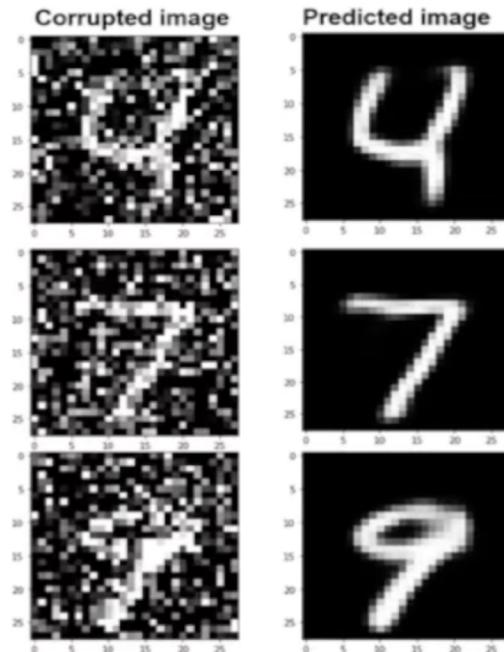


Photo credit: Encode Box: Autoencoder in biology — review, and perspectives.

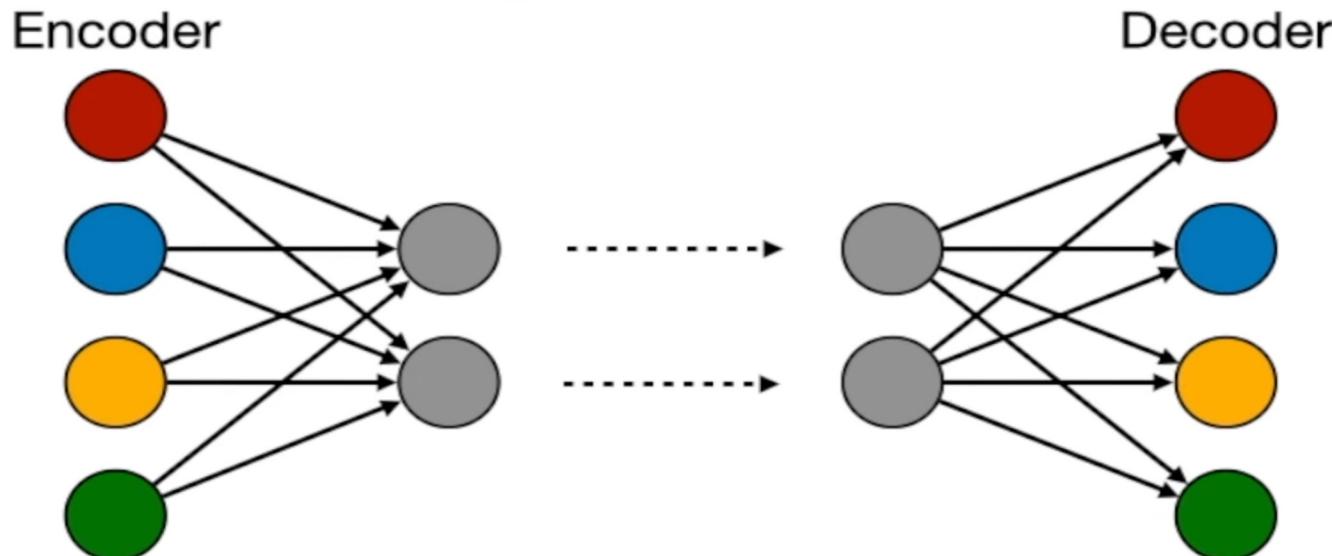
Denoising autoencoders



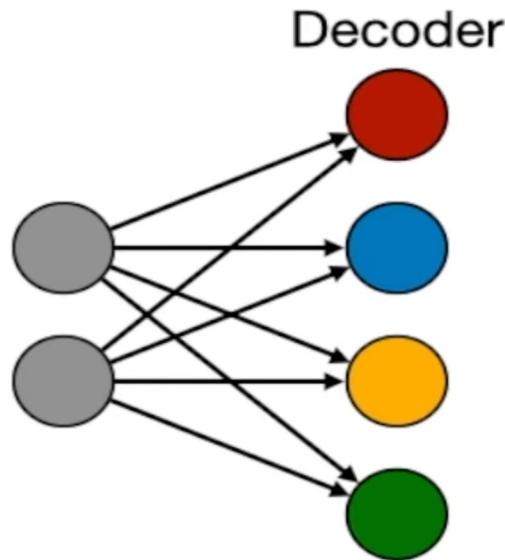
Denoising MNIST digits – Source: Ormdena

Variational autoencoders

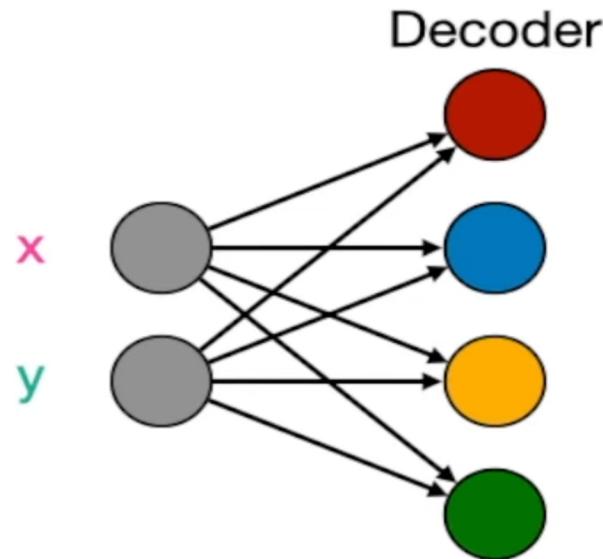
Autoencoders as generators



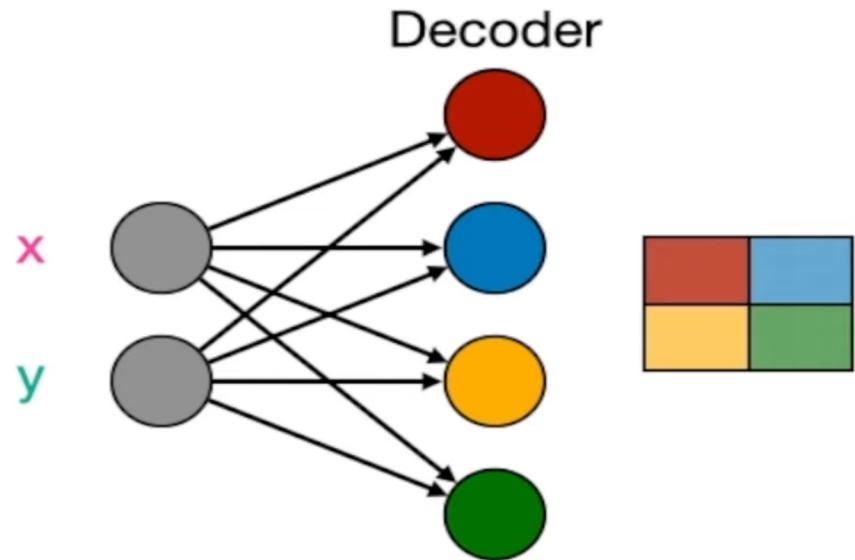
Autoencoders as generators



Autoencoders as generators

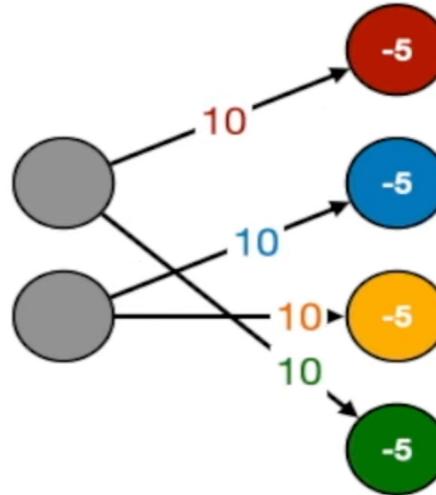


Autoencoders as generators



Autoencoders as generators

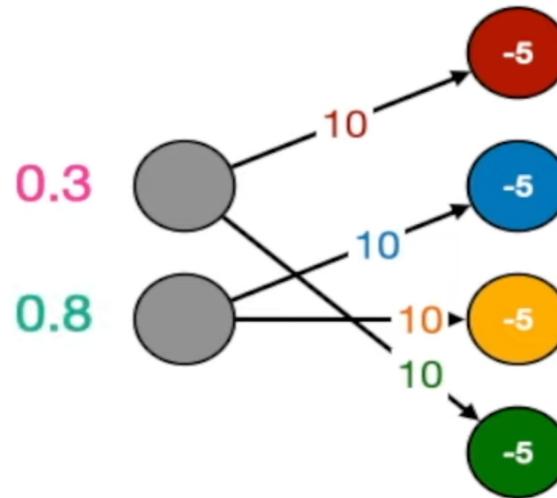
Latent space



Decoder

Autoencoders as generators

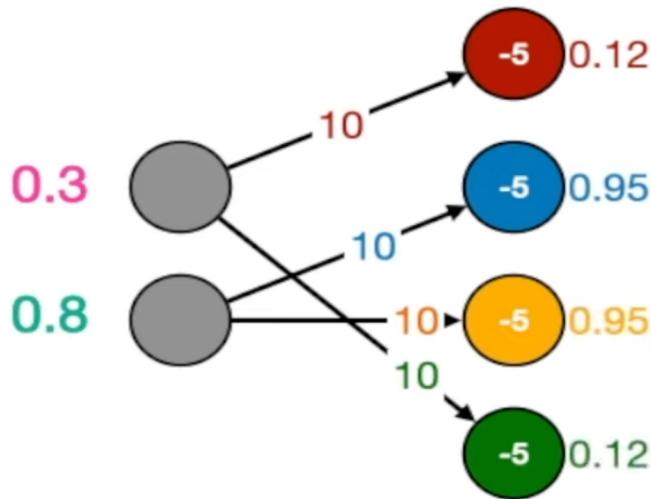
Latent space



Decoder

Autoencoders as generators

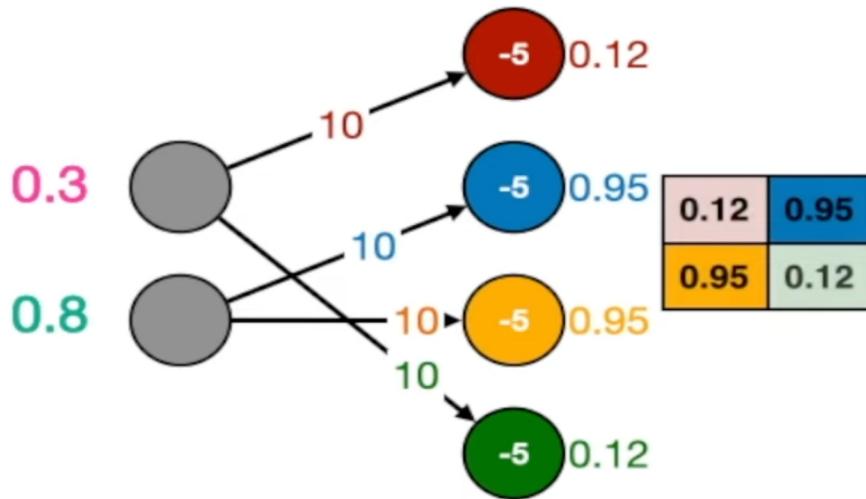
Latent space



Decoder

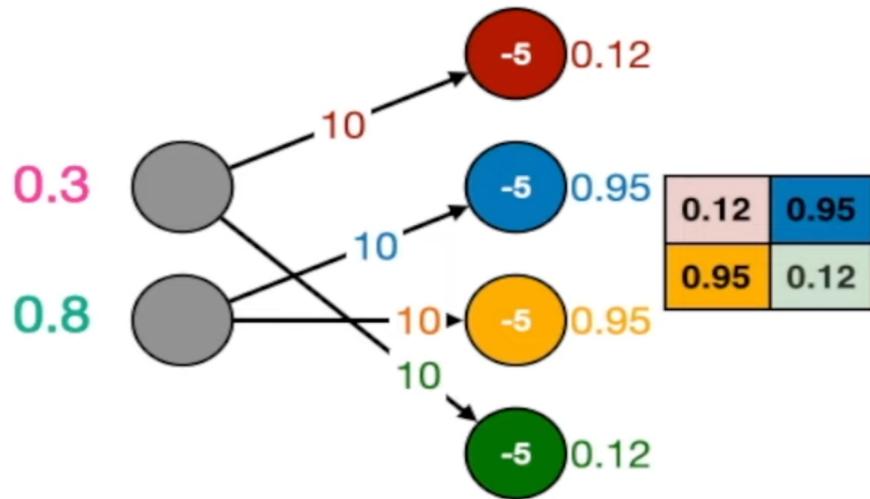
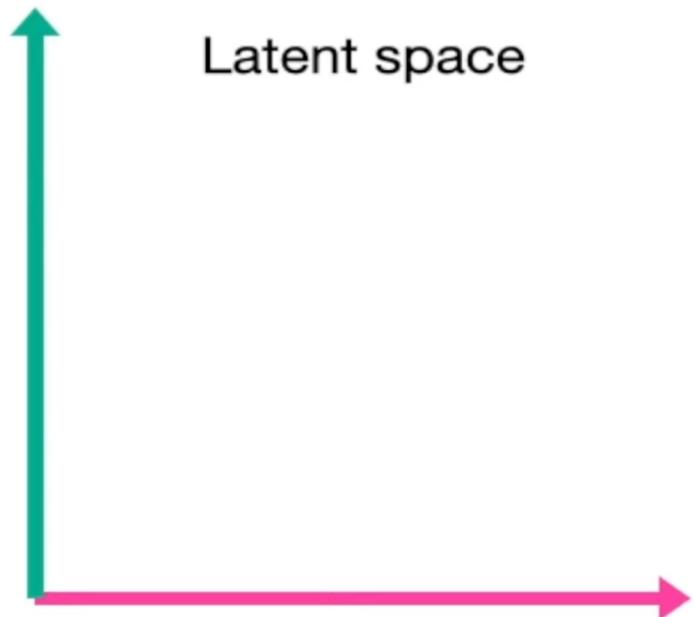
Autoencoders as generators

Latent space



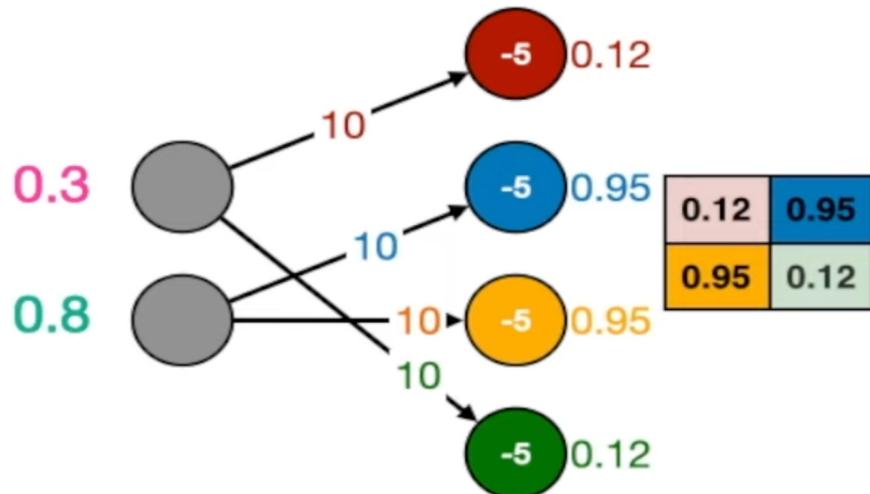
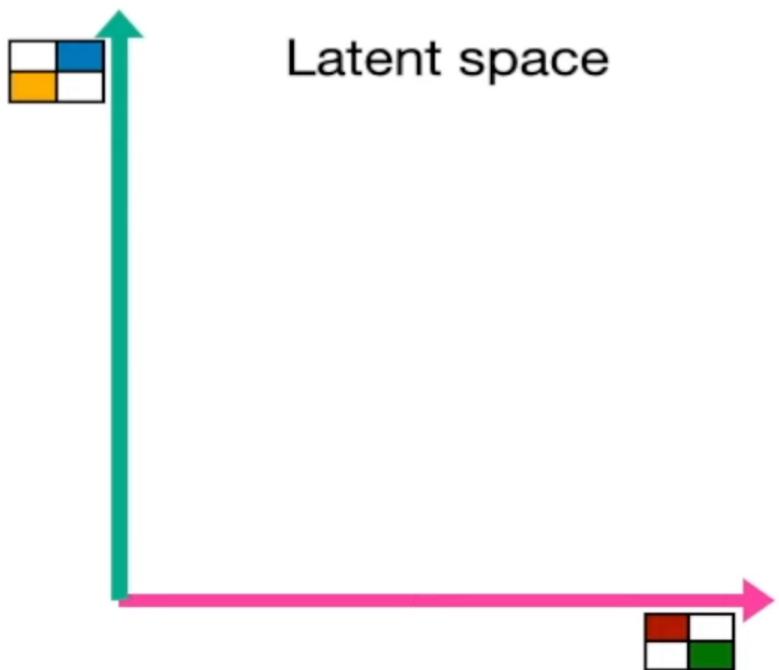
Decoder

Autoencoders as generators



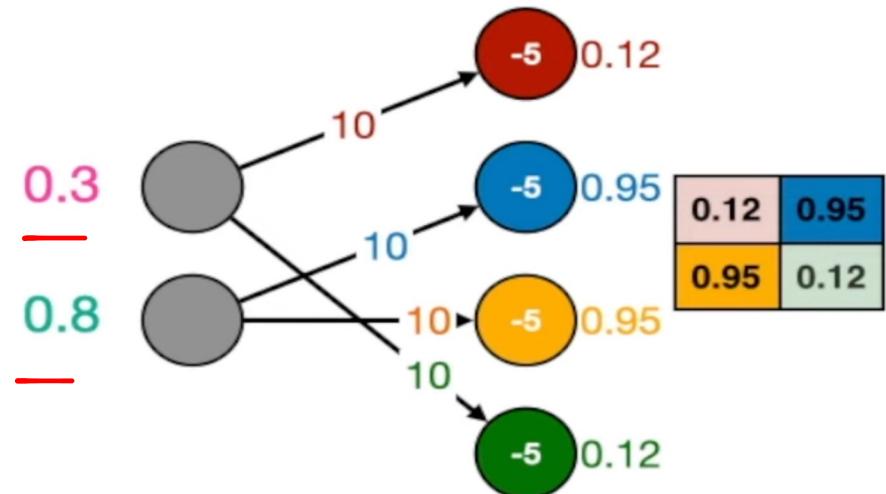
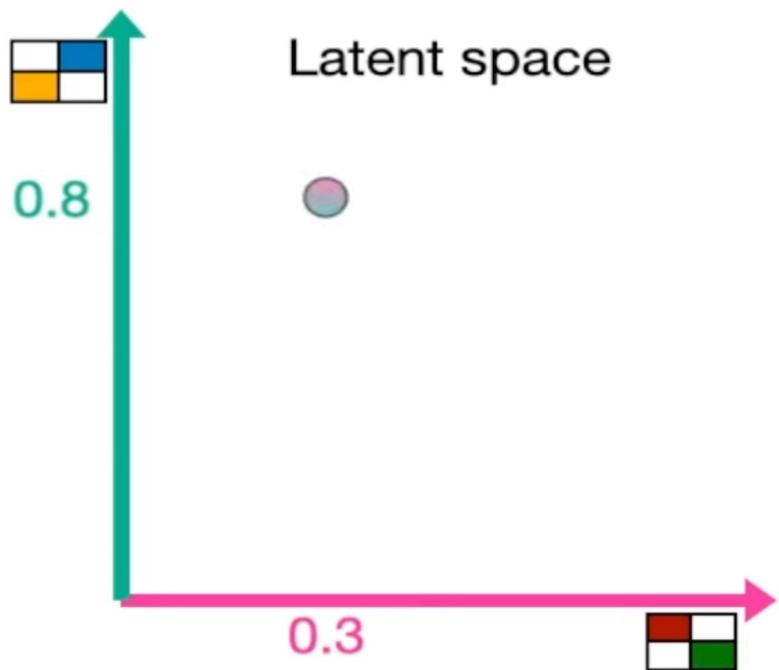
Decoder

Autoencoders as generators

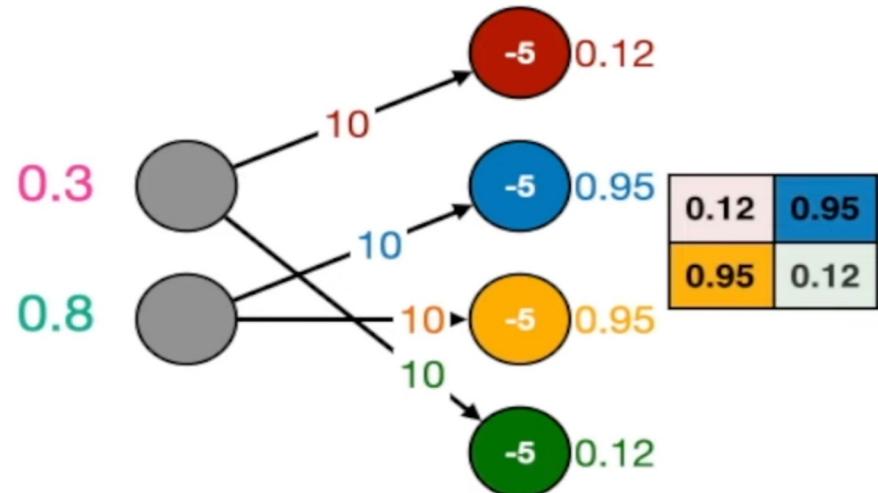
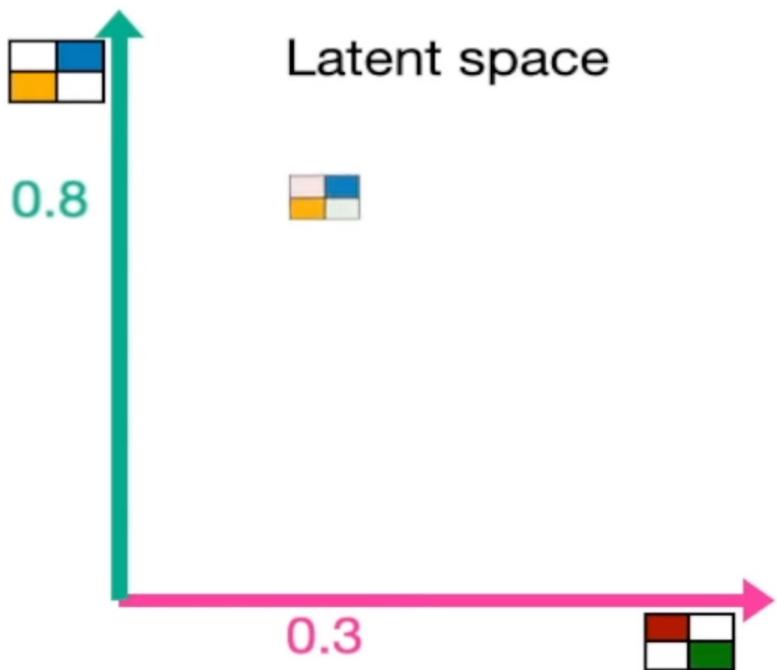


Decoder

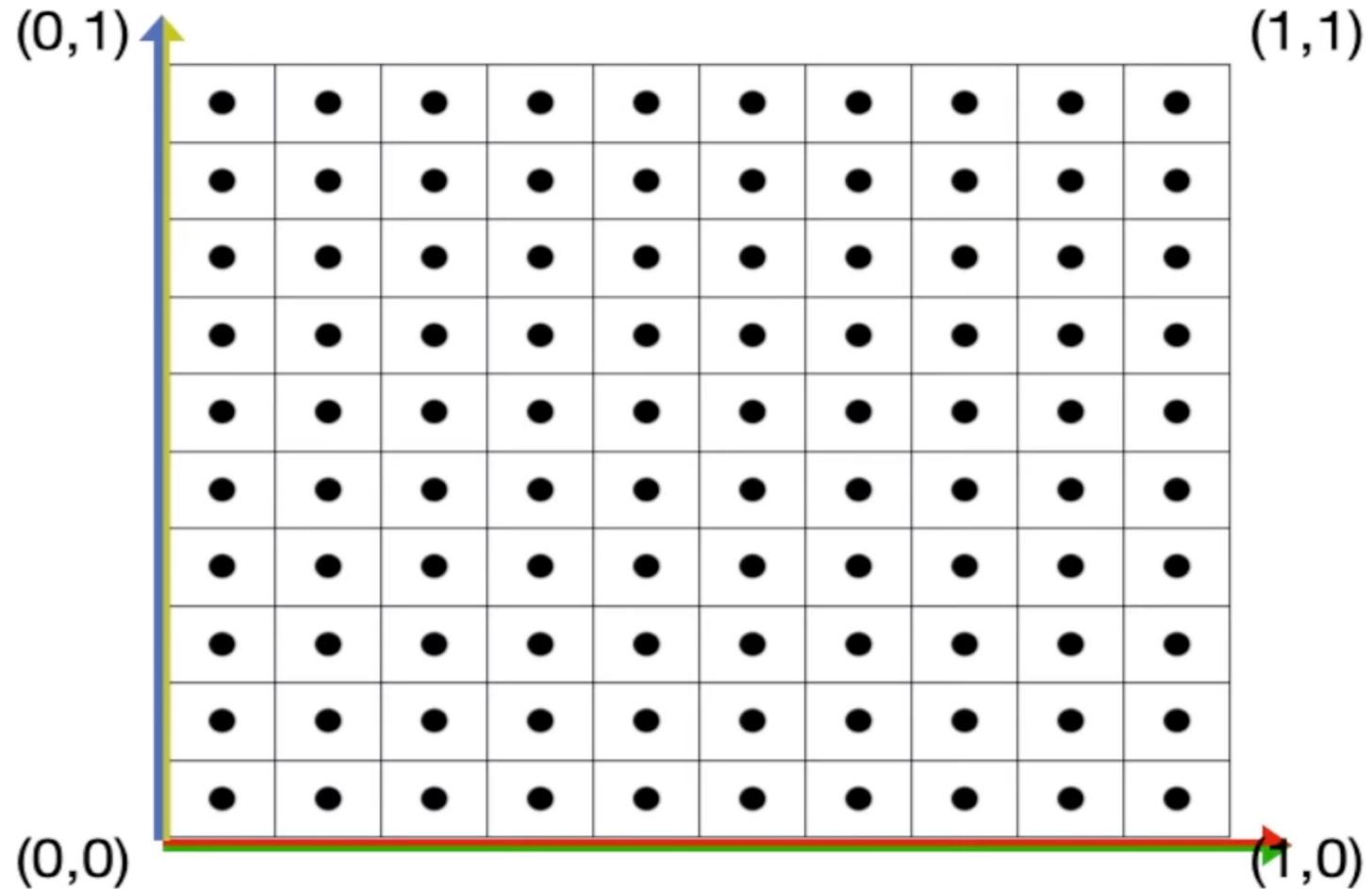
Autoencoders as generators



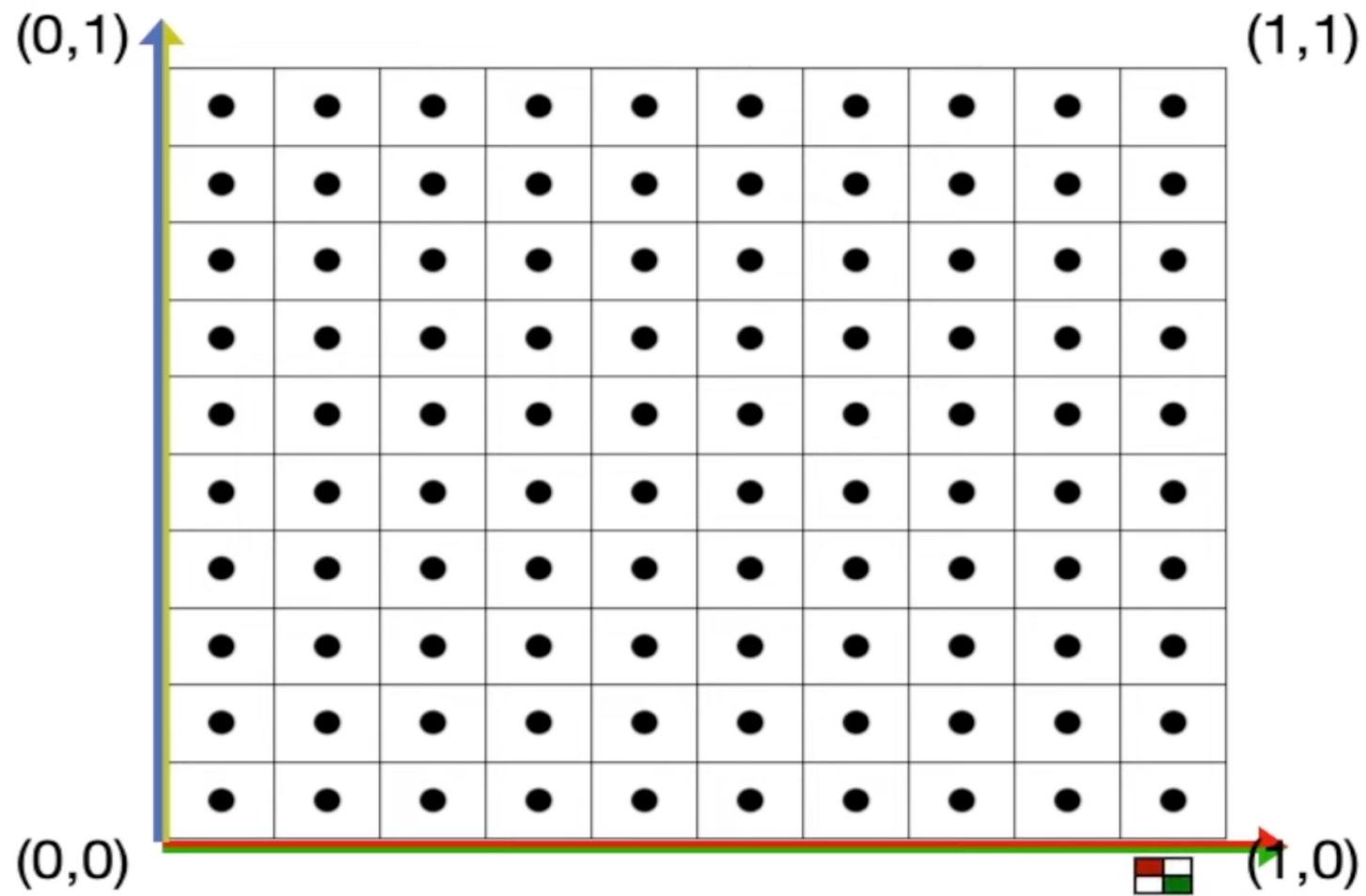
Autoencoders as generators



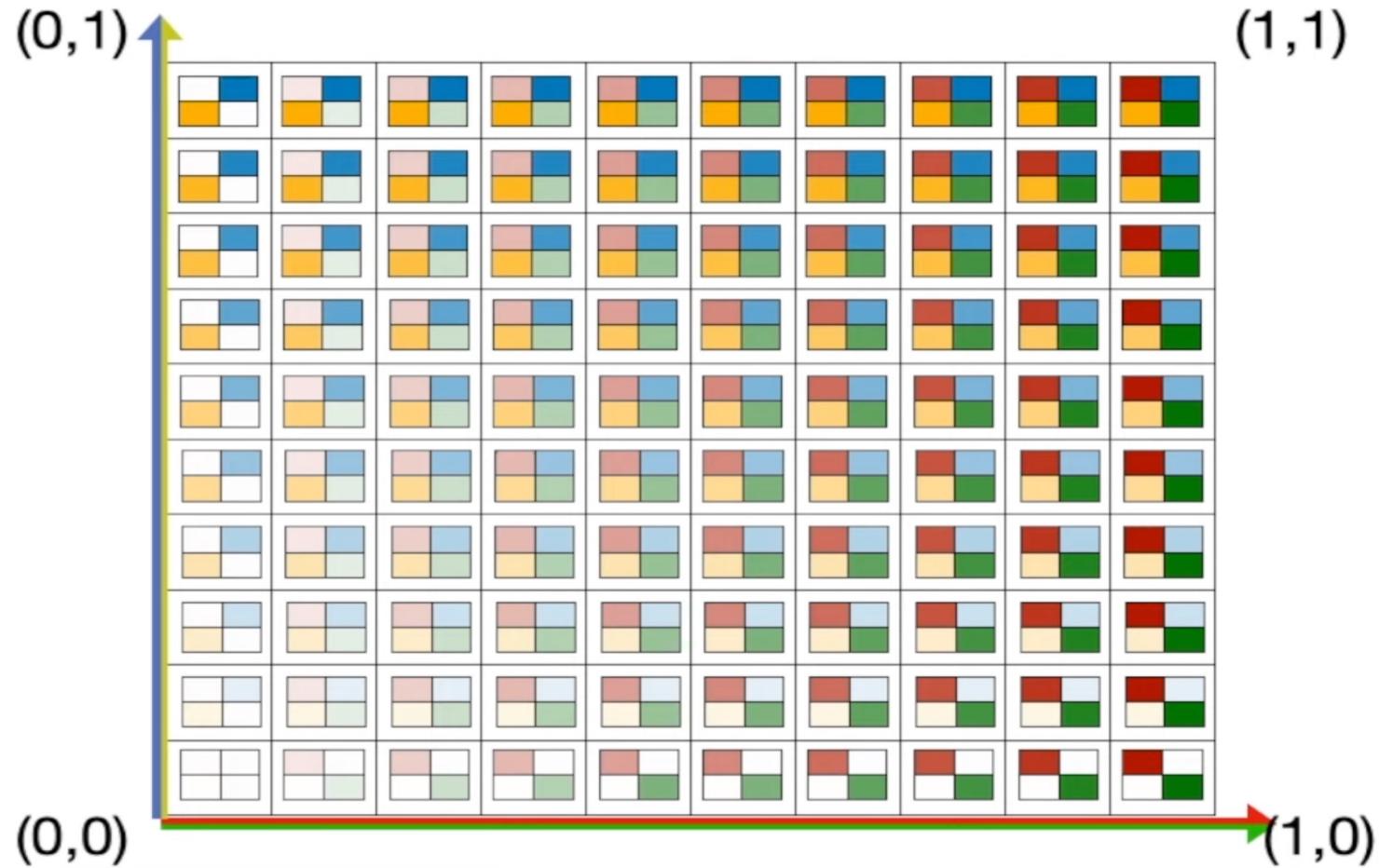
Latent space



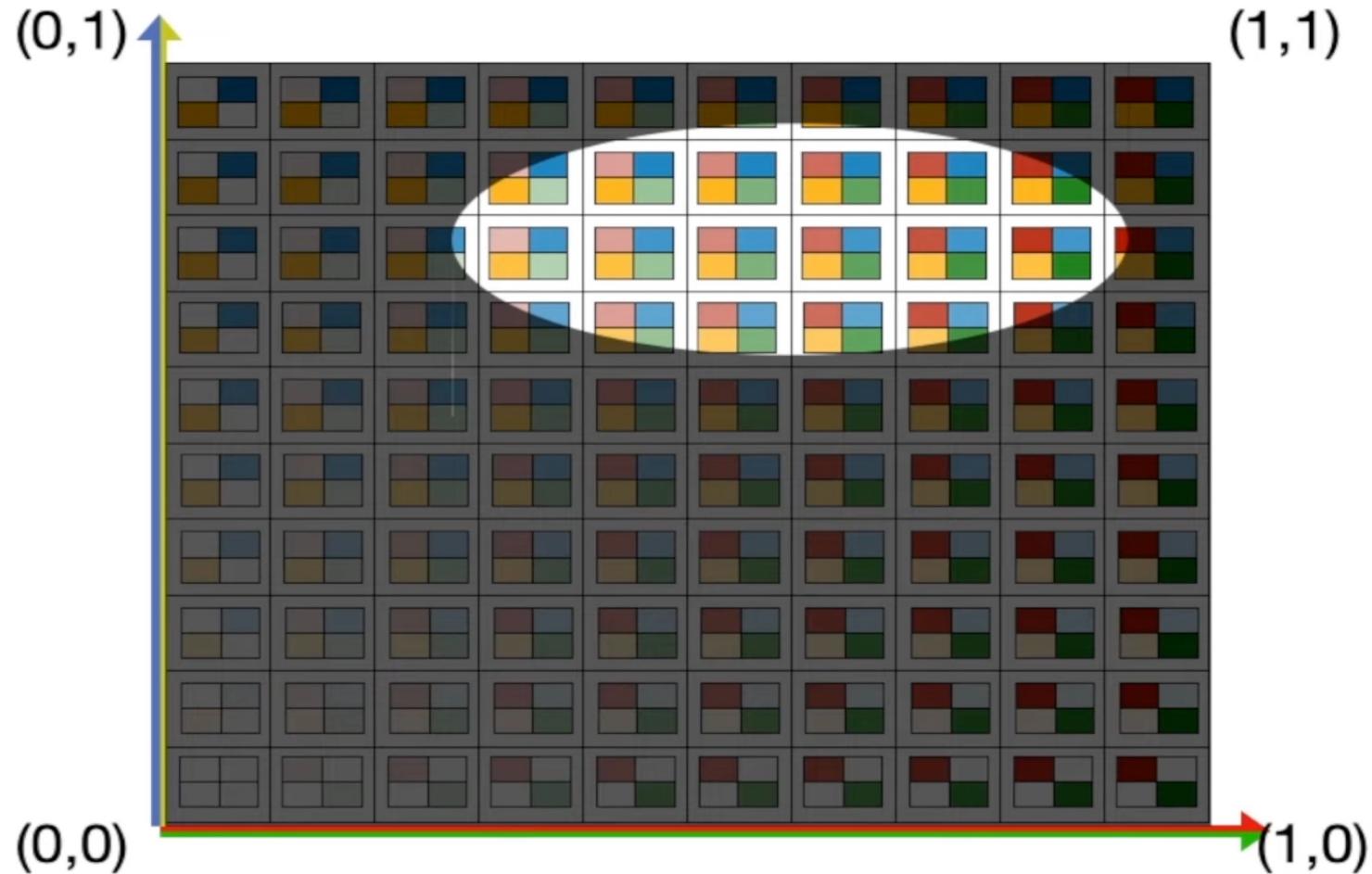
Latent space



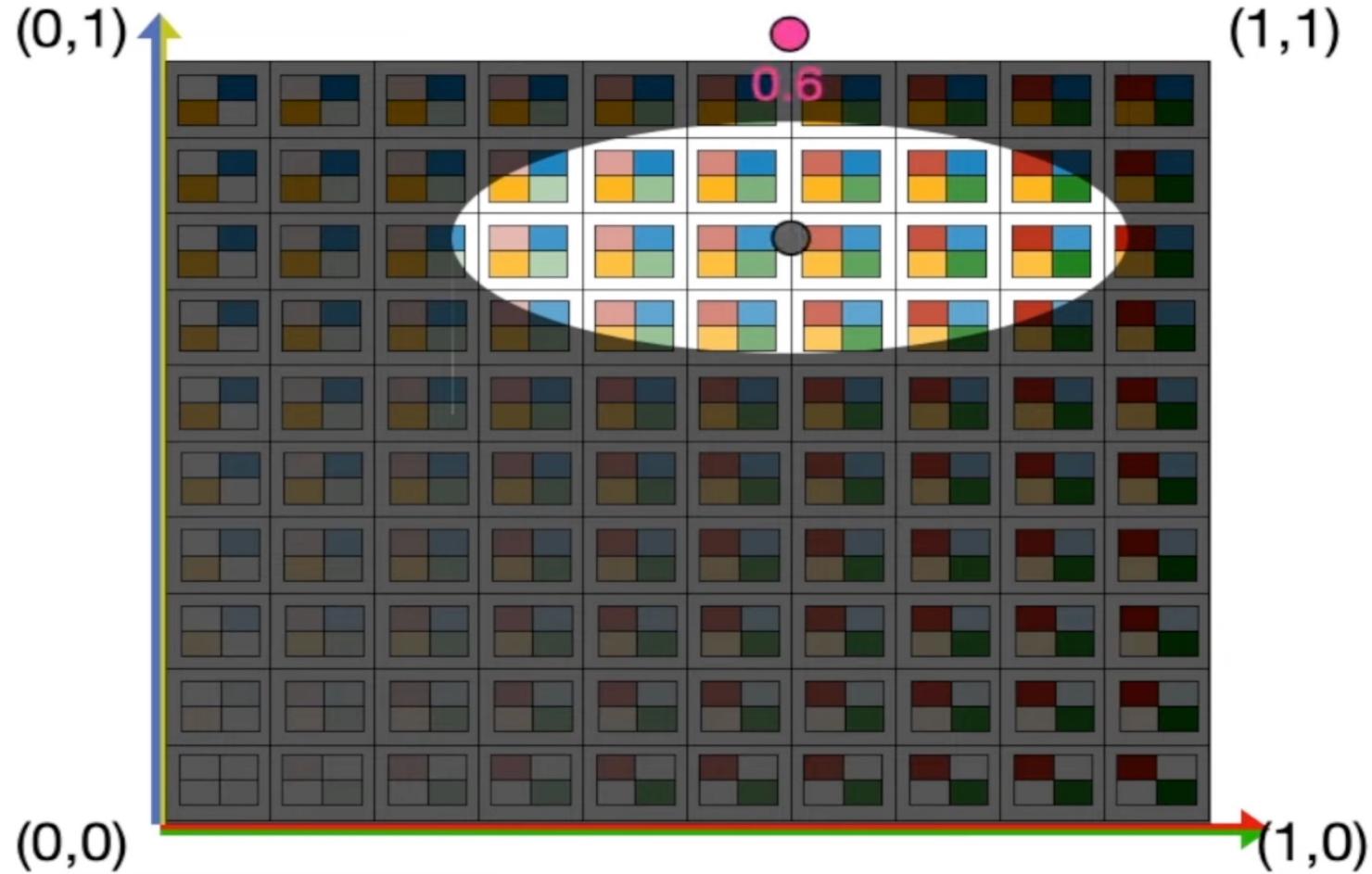
Latent space



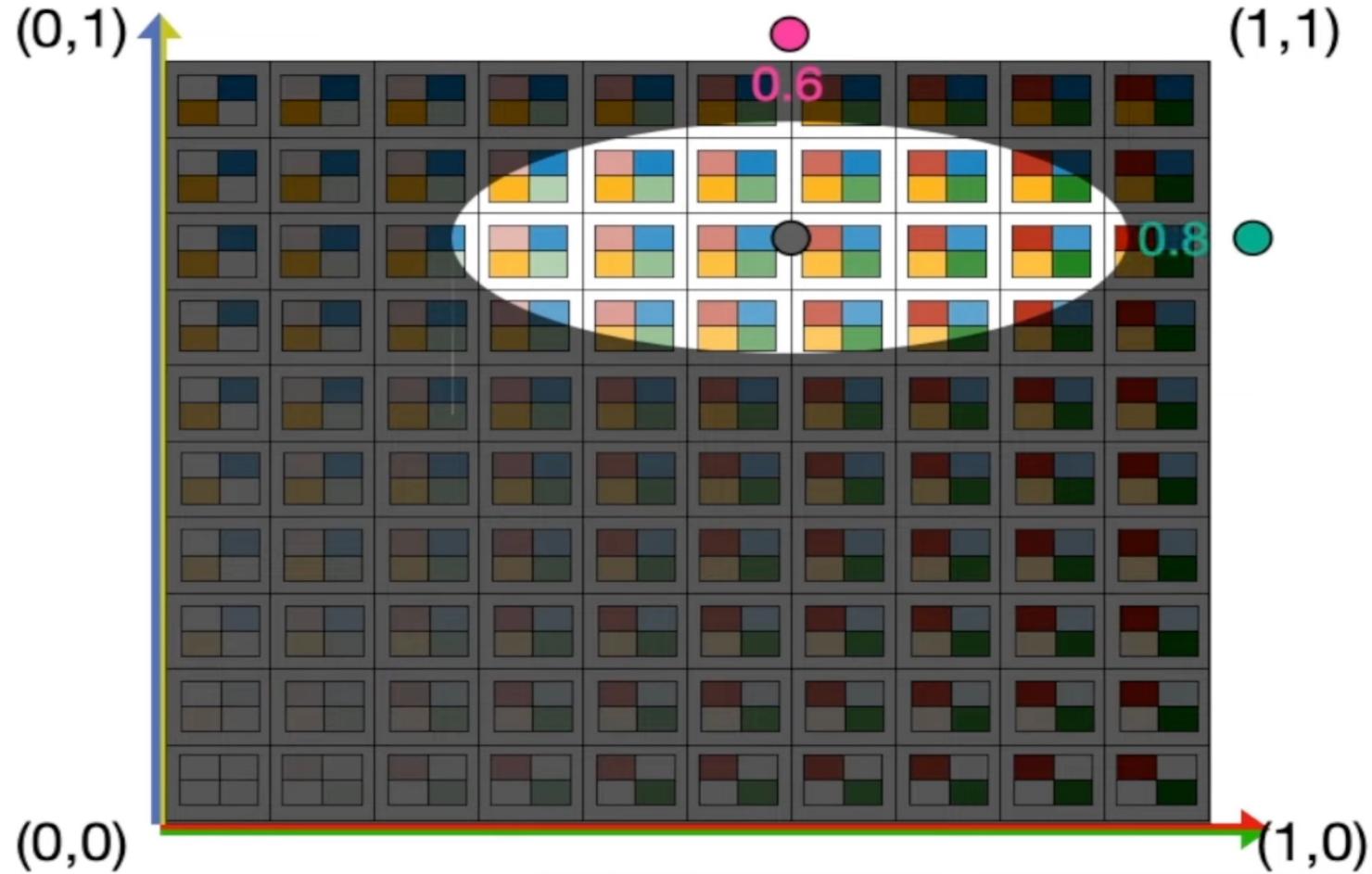
Latent space



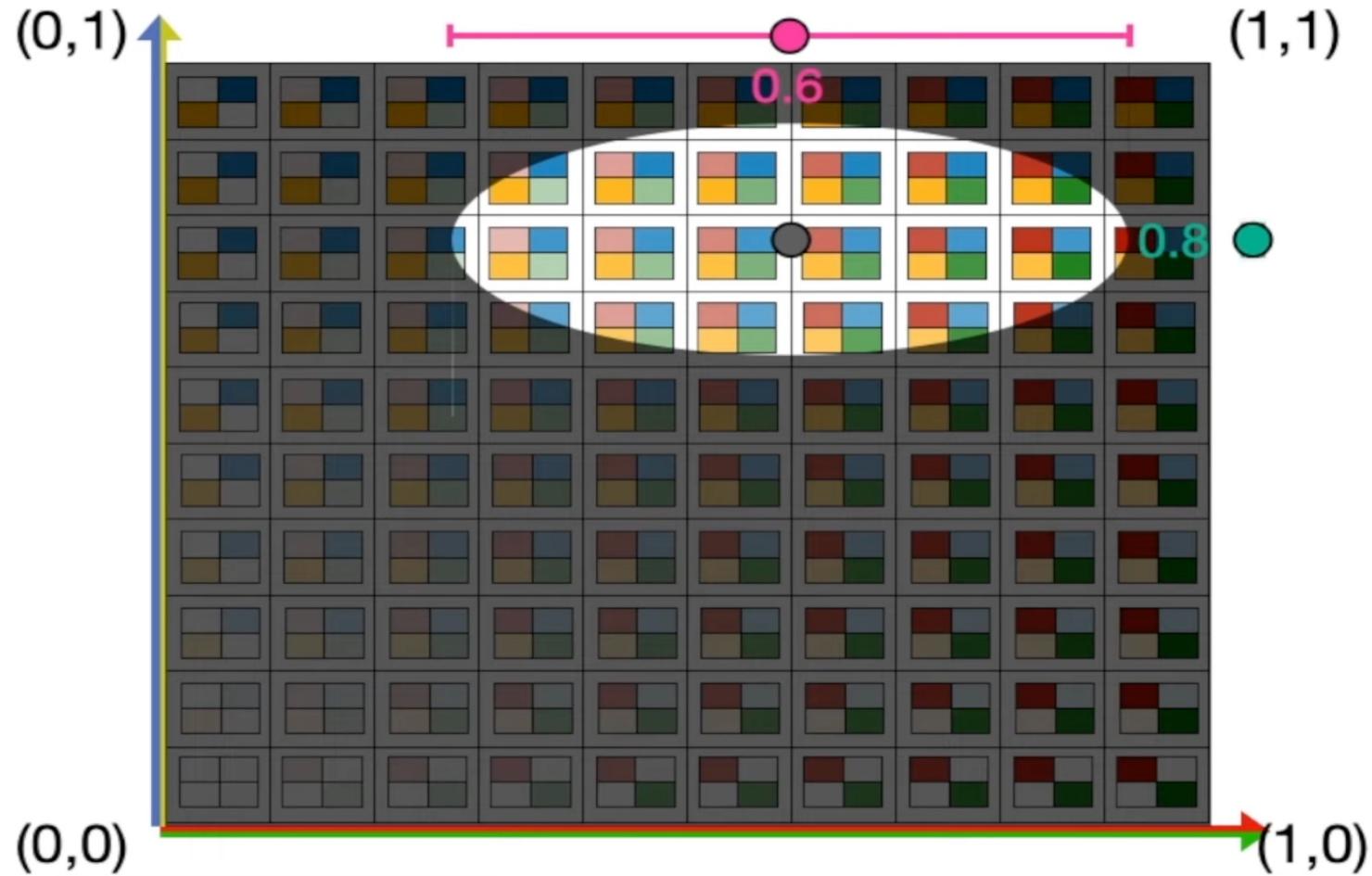
Latent space



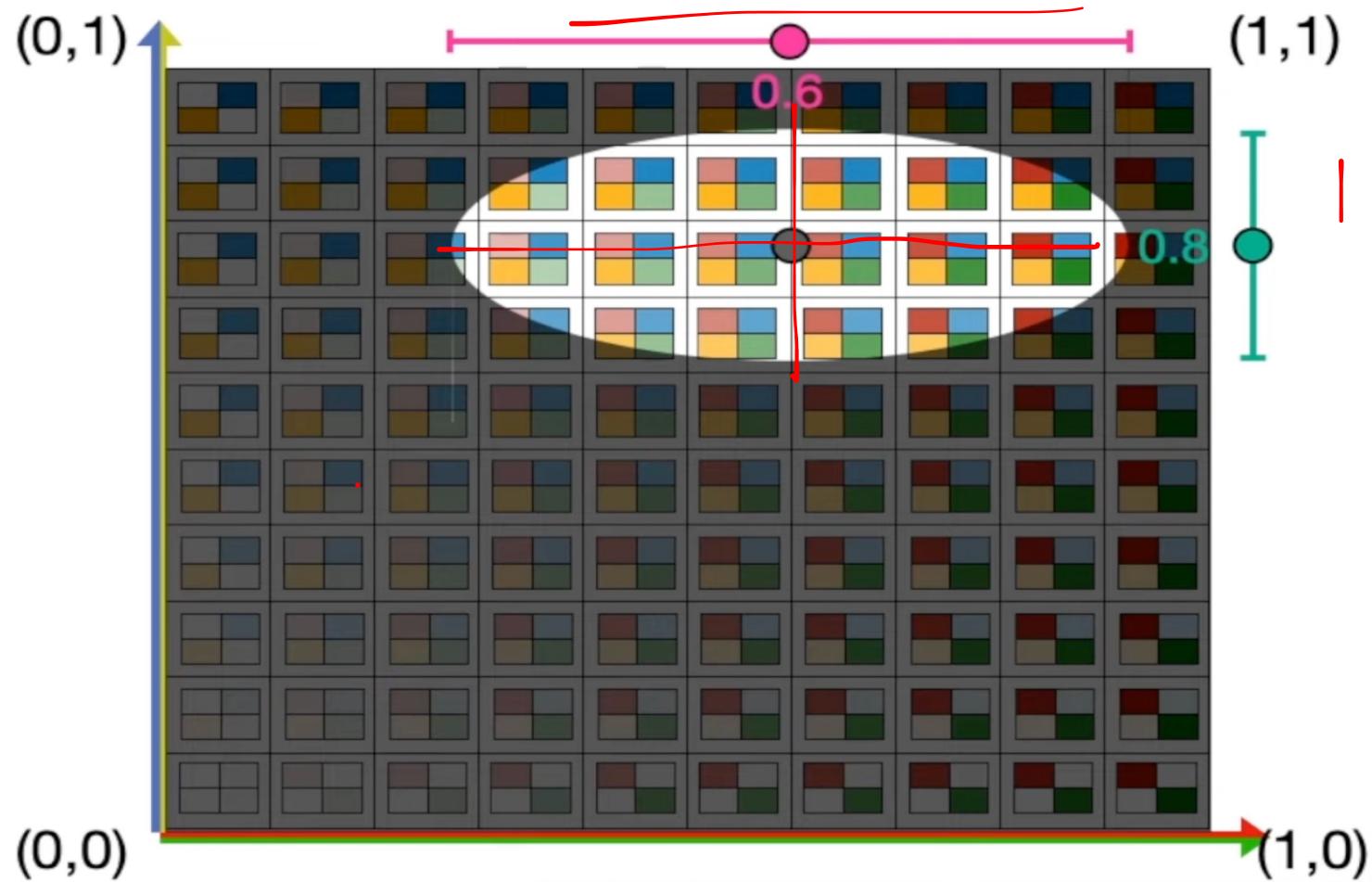
Latent space



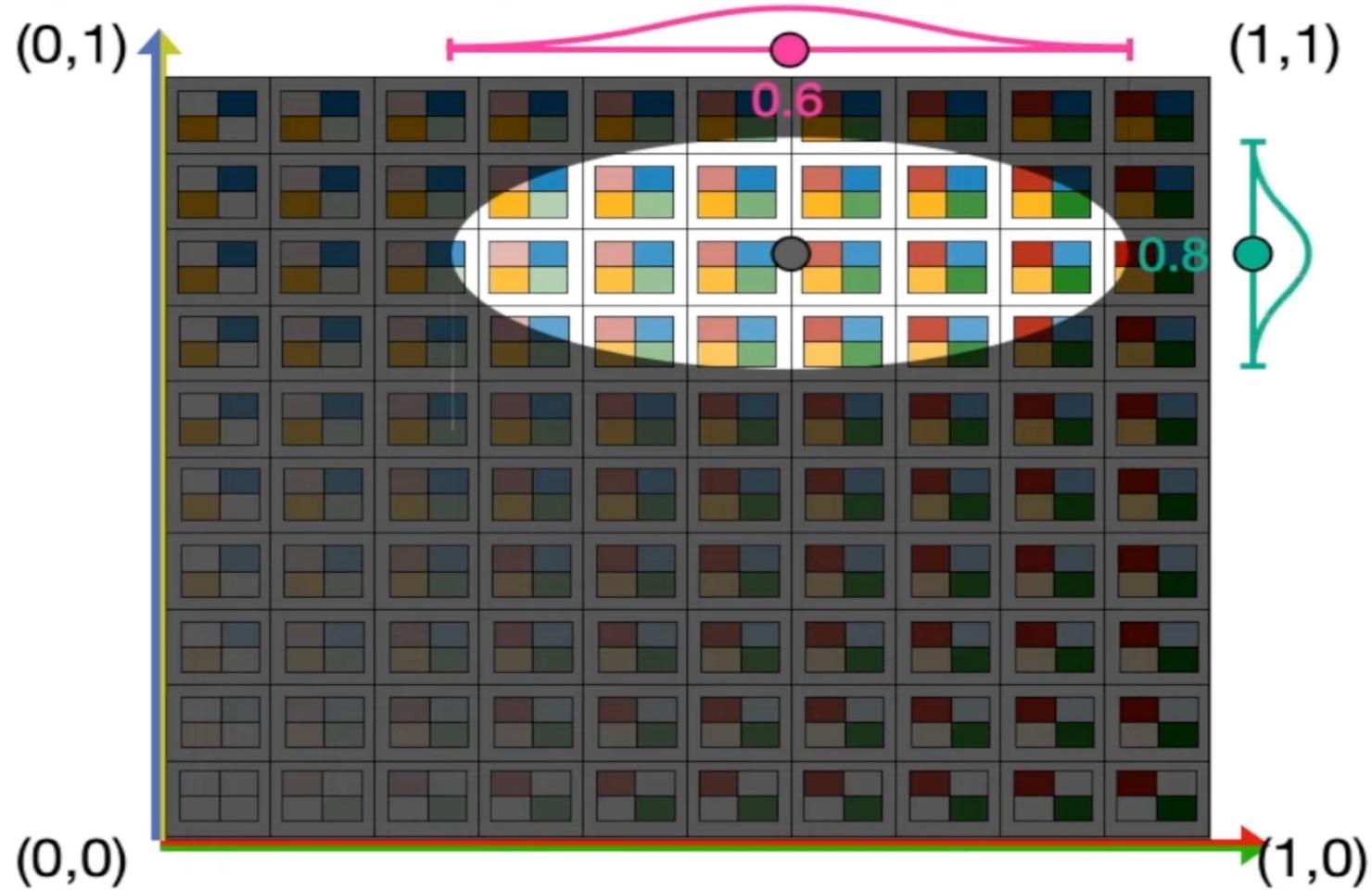
Latent space



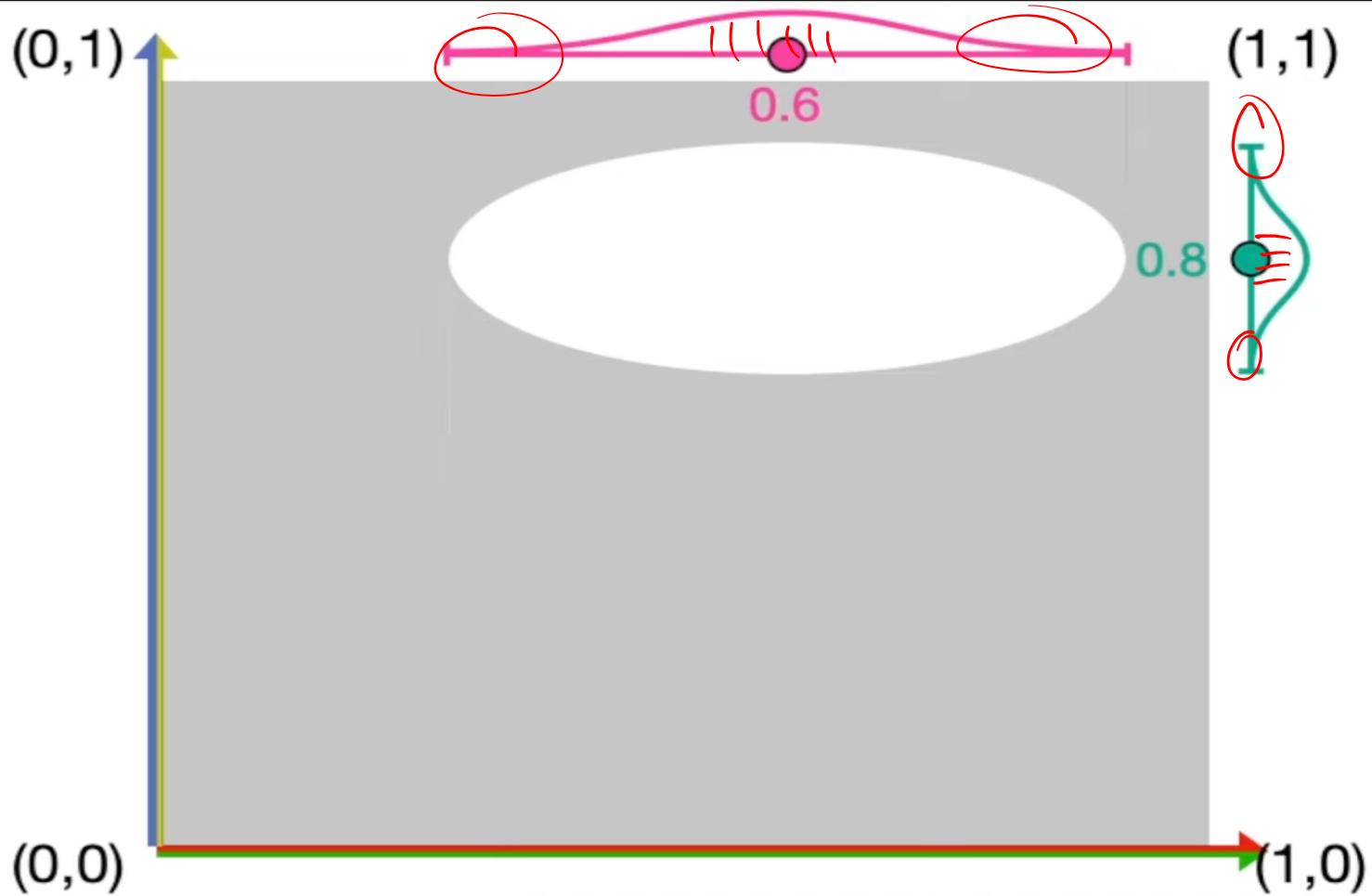
Latent space



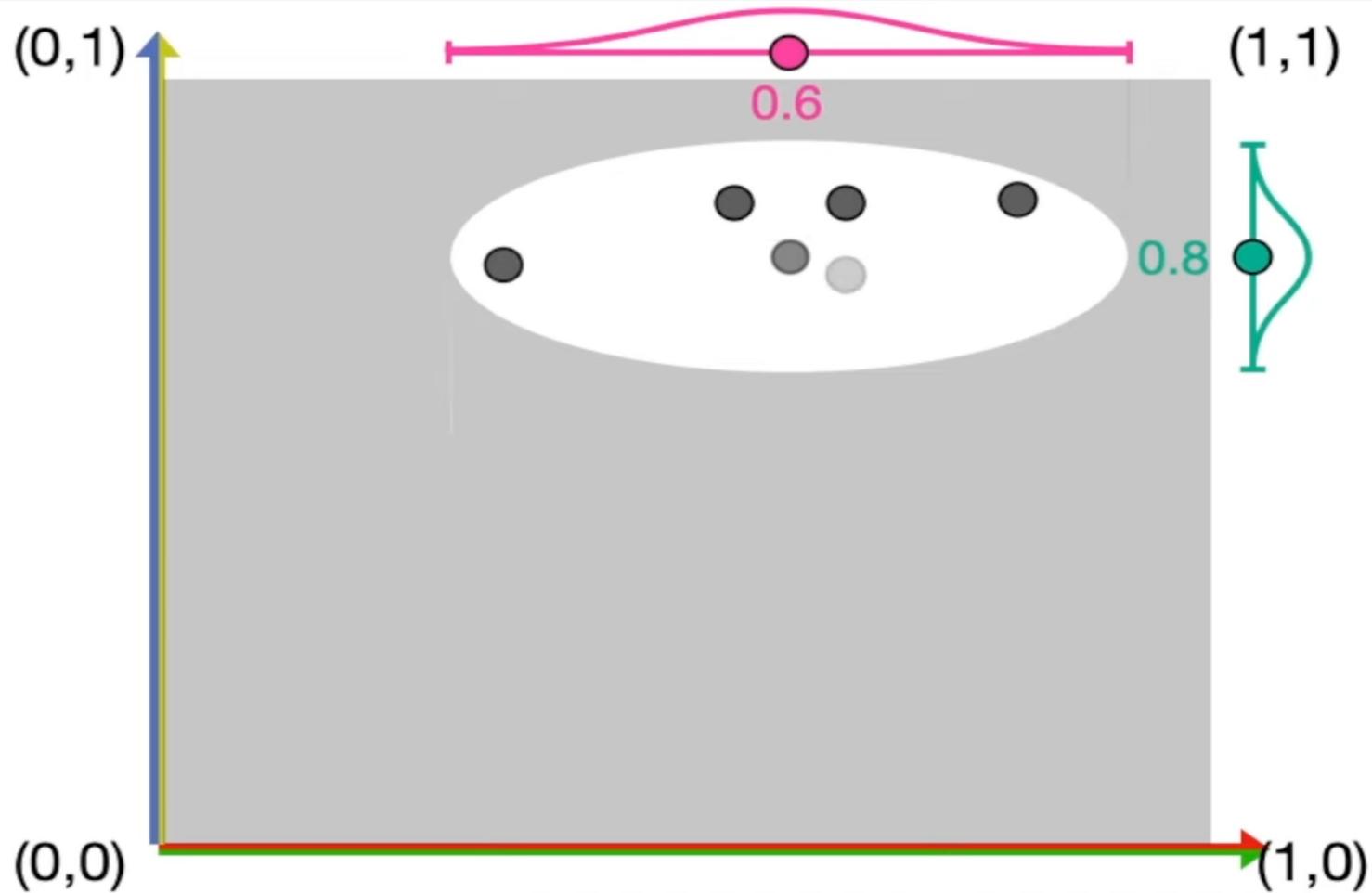
Latent space



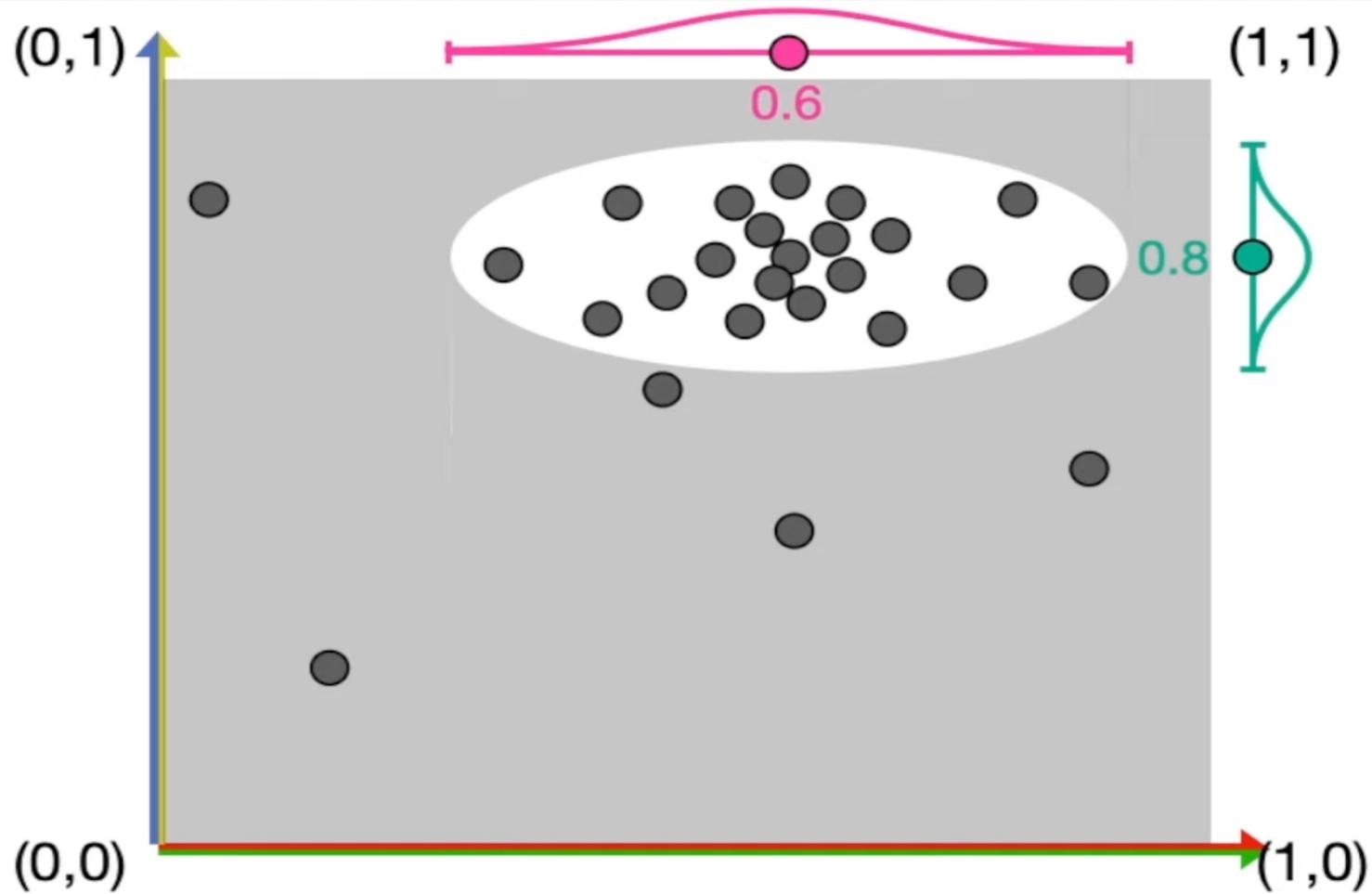
Latent space



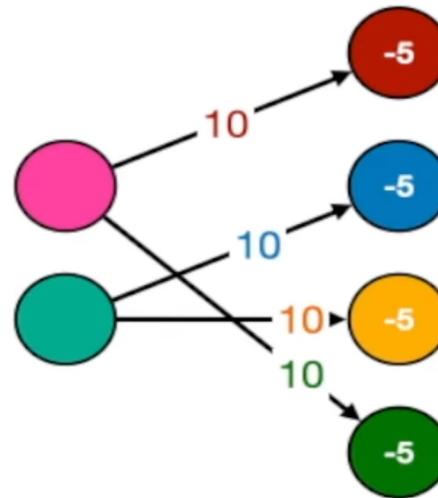
Latent space



Latent space

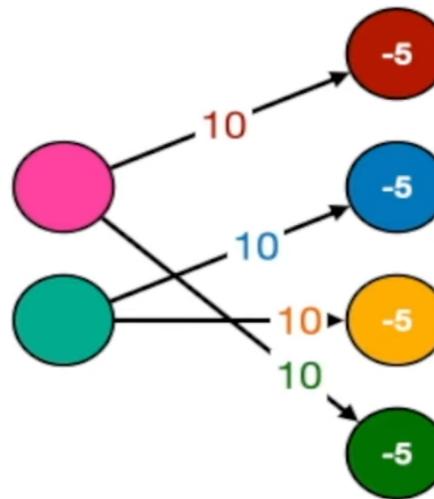
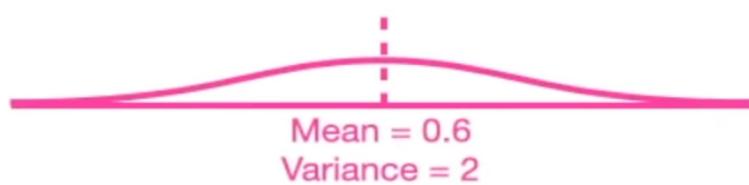


Variational autoencoder



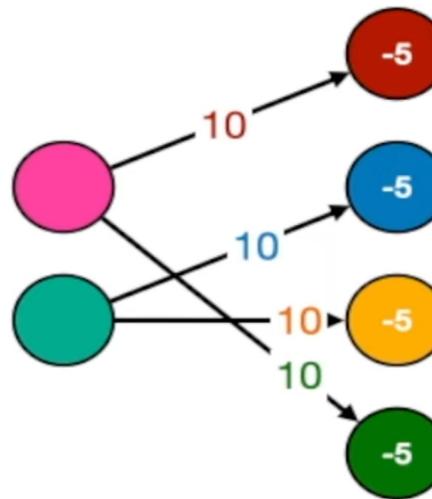
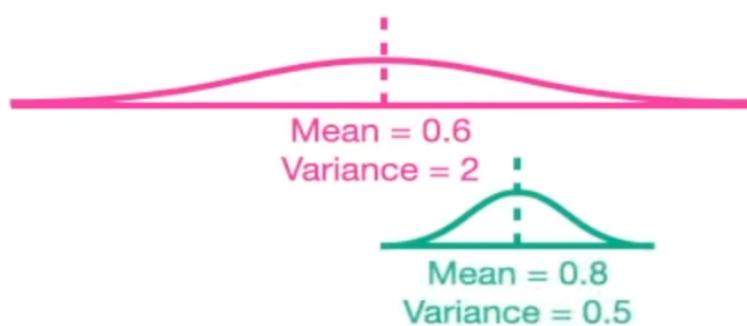
Decoder

Variational autoencoder



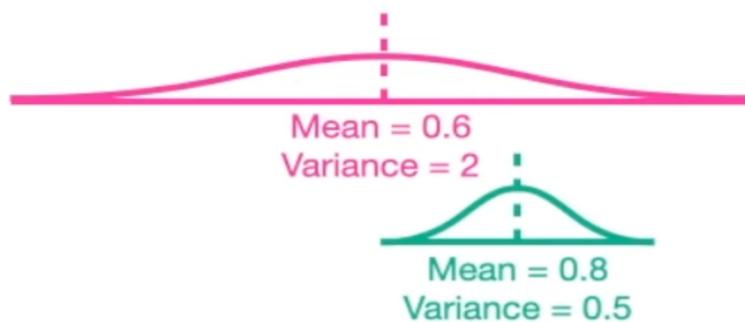
Decoder

Variational autoencoder

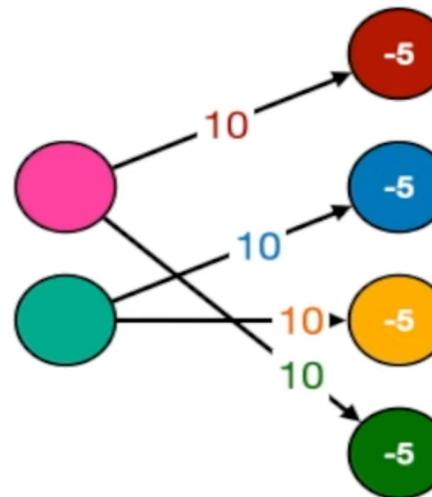


Decoder

Variational autoencoder

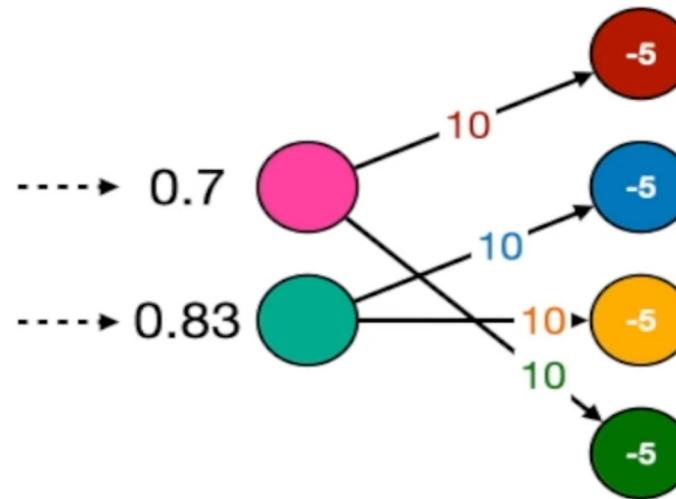
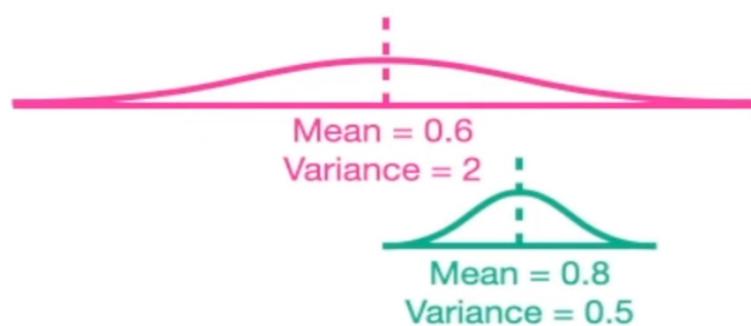


0.7

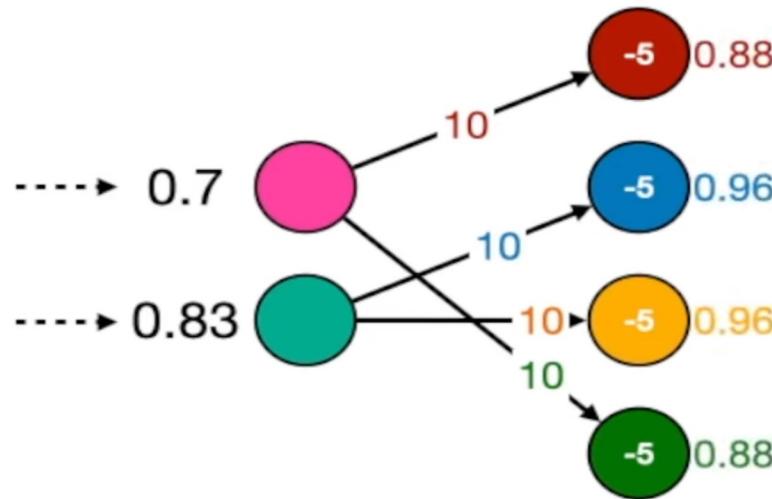
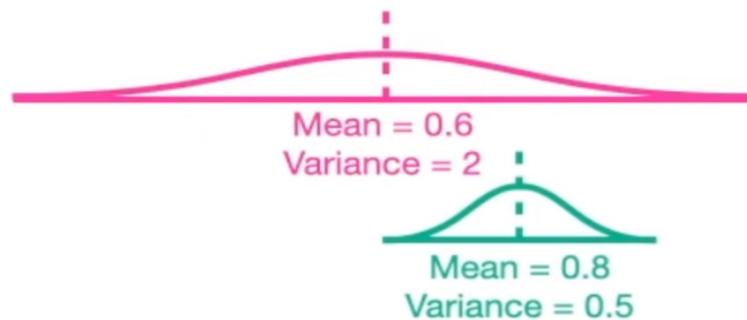


Decoder

Variational autoencoder

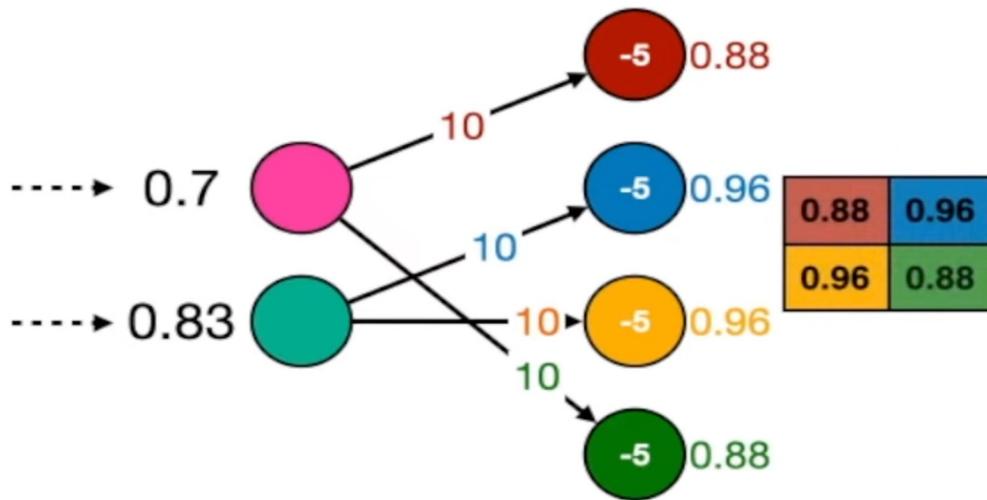
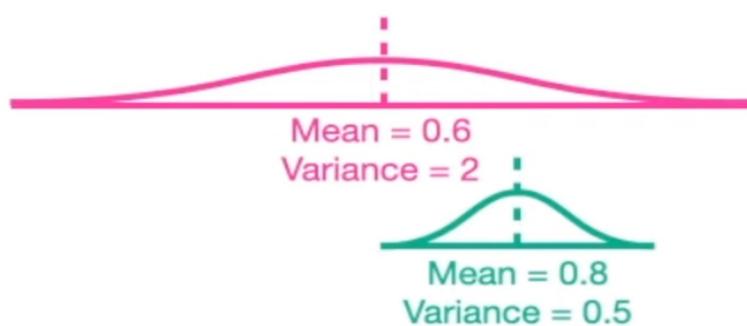


Variational autoencoder



Decoder

Variational autoencoder

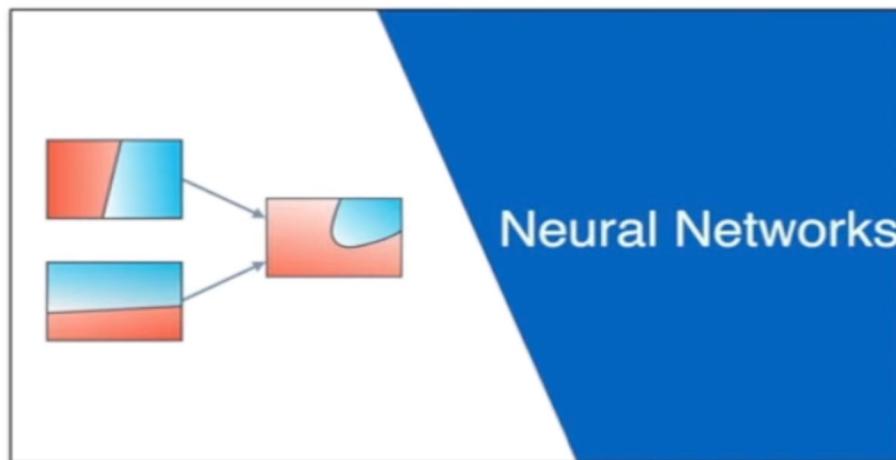


Decoder

Training autoencoders

Training a neural network - loss function

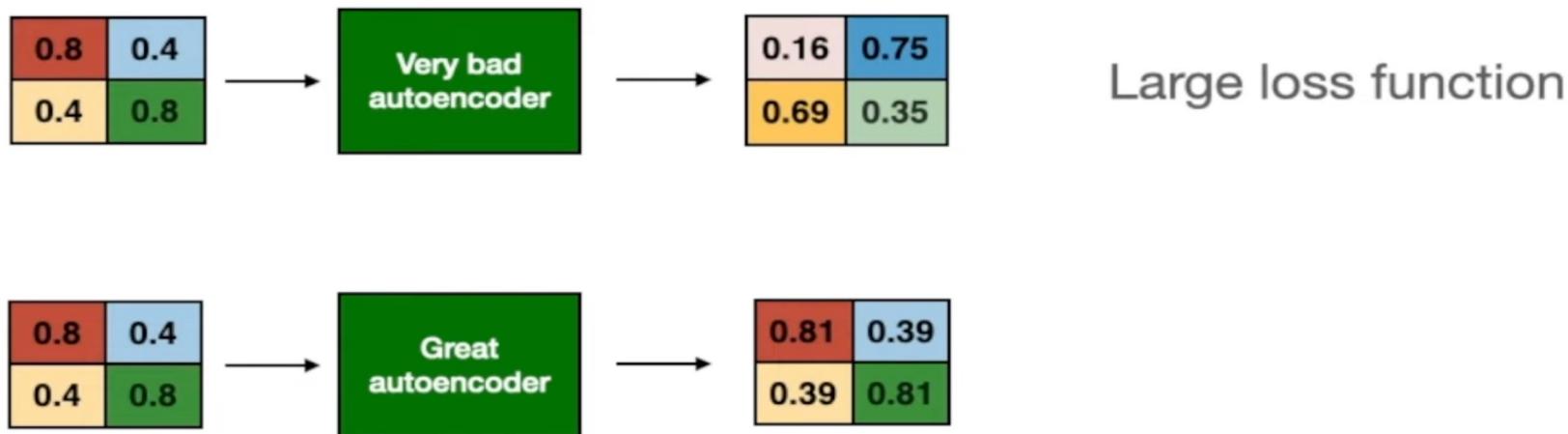
Training a neural network - loss function



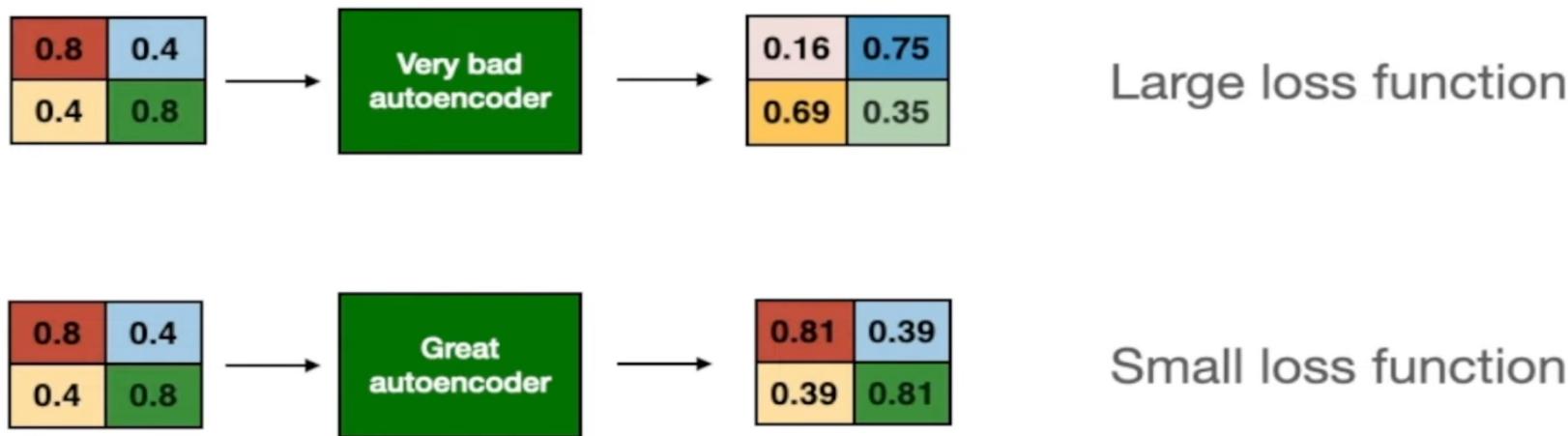
Training a neural network - loss function



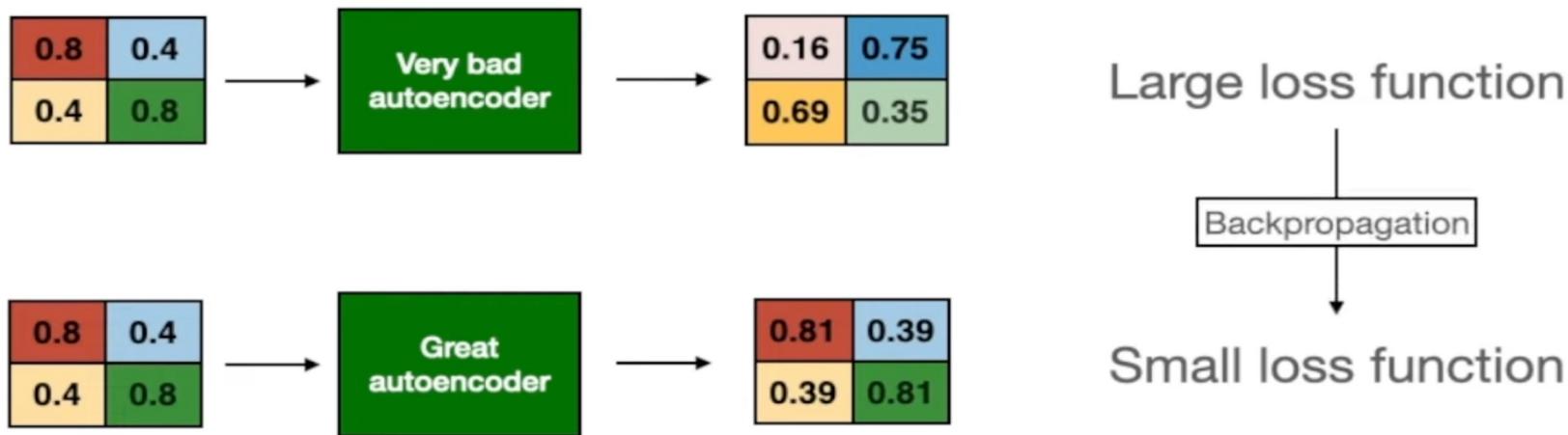
Training a neural network - loss function



Training a neural network - loss function

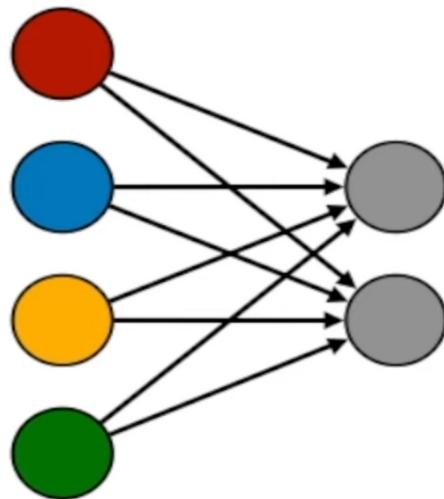


Training a neural network - loss function



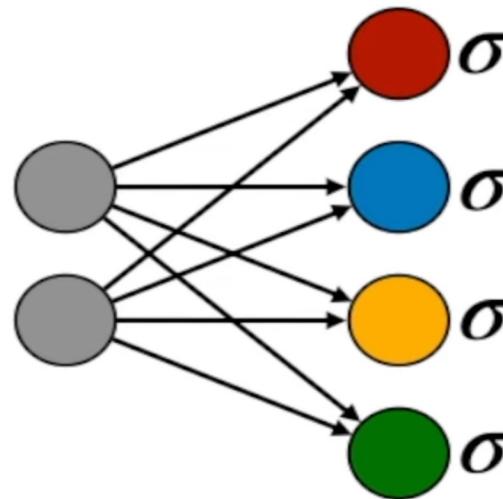
Training an autoencoder

Encoder

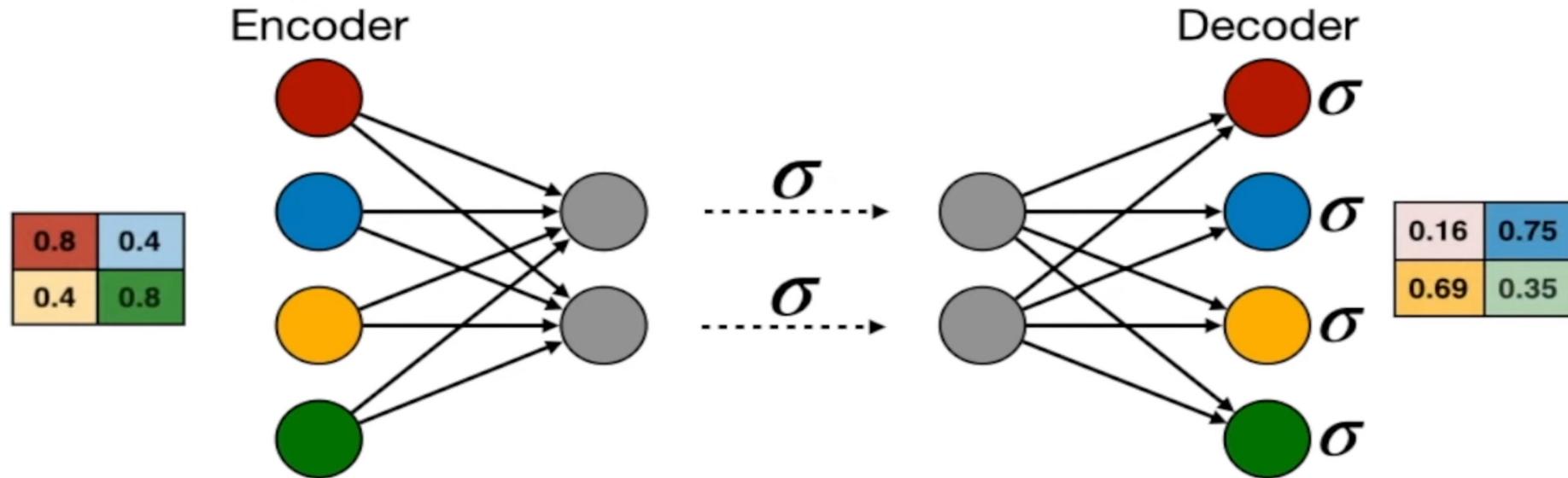


σ
 σ

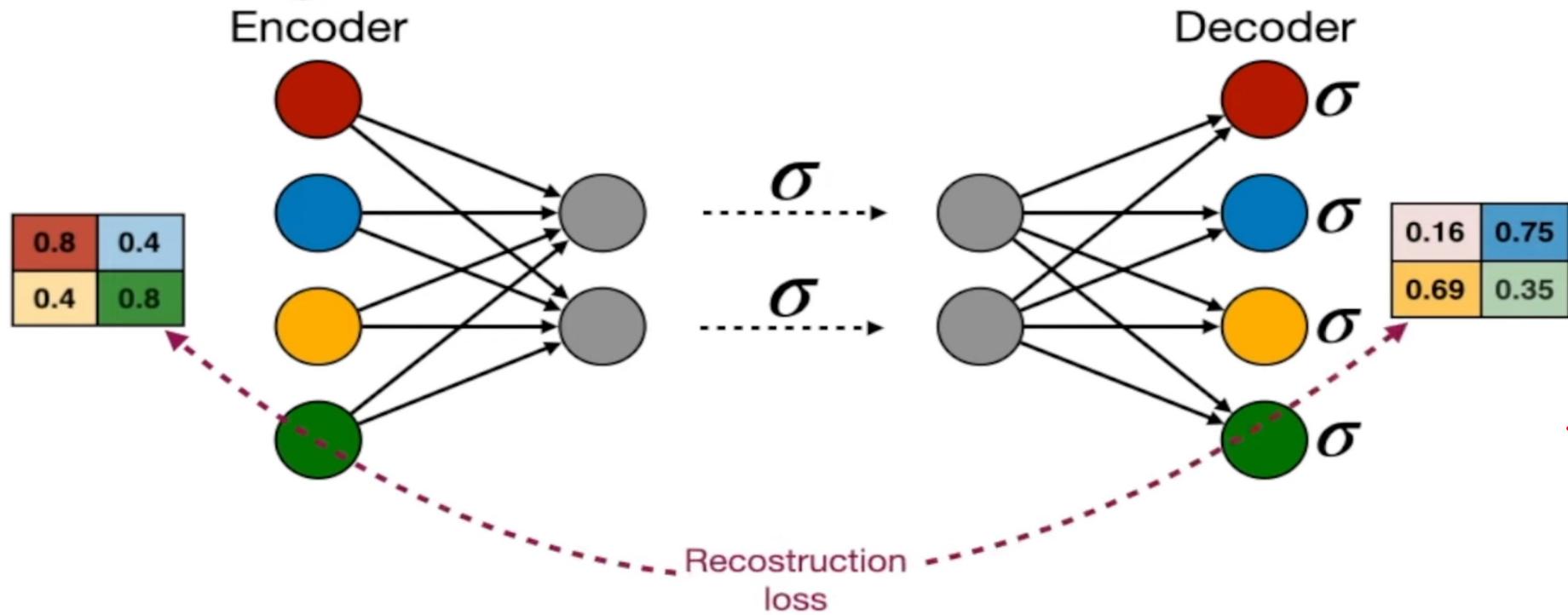
Decoder



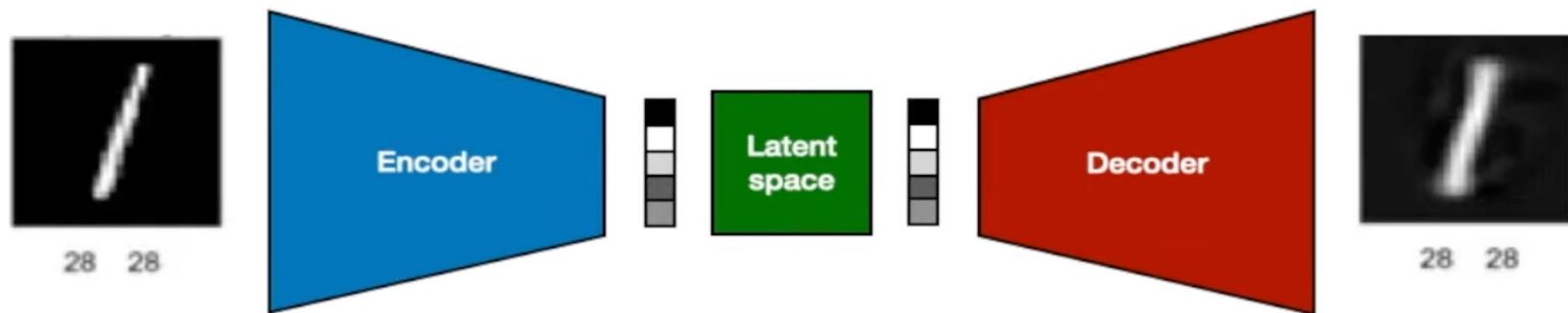
Training an autoencoder



Training an autoencoder

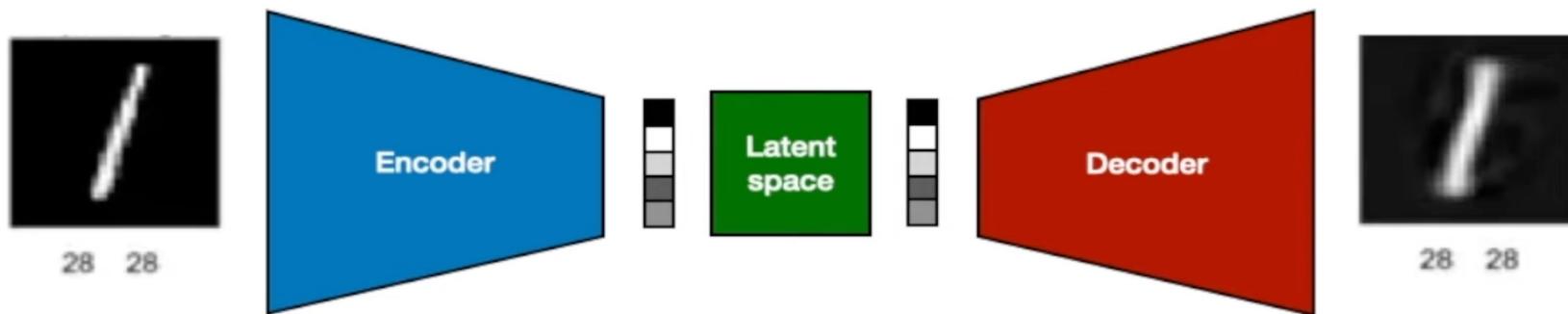


Training autoencoders



Training autoencoders

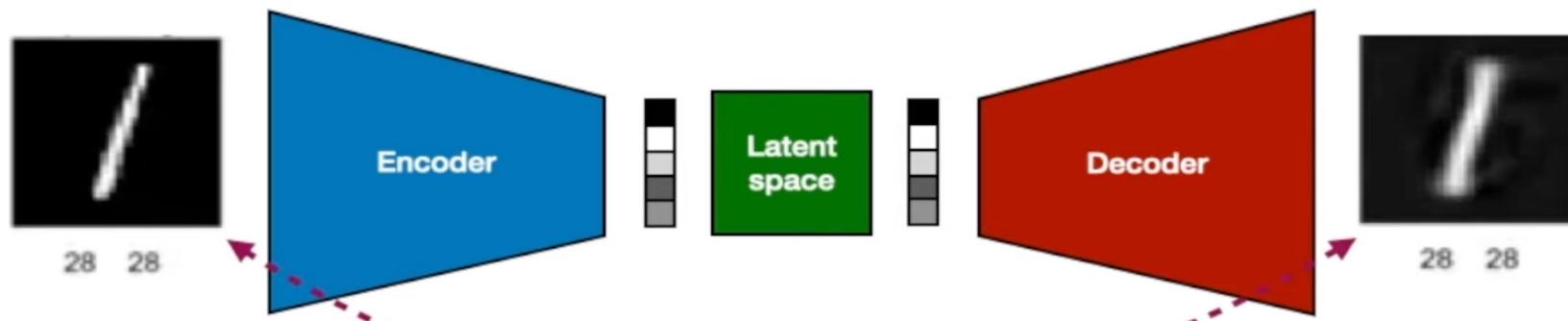
Initialize encoder and decoder
with random weights



Error: reconstruction loss

Training autoencoders

Initialize encoder and decoder
with random weights



Error: reconstruction loss

Do backpropagation
and update weights

Reconstruction loss (Mean squared error)

Reconstruction loss (Mean squared error)

$$MSE(\mathbf{x}, \mathbf{y}) = \sum_i (\mathbf{x}_i - \mathbf{y}_i)^2$$

Reconstruction loss (Mean squared error)

x	
0.6	0.4
0.4	0.6

y	
0.65	0.42
0.36	0.59

$$MSE(\mathbf{x}, \mathbf{y}) = \sum_i (x_i - y_i)^2$$

Reconstruction loss (Mean squared error)

x	
	y
0.6	0.4
0.4	0.6

x	
	y
0.65	0.42
0.36	0.59

$$MSE(\mathbf{x}, \mathbf{y}) = \sum_i (x_i - y_i)^2$$

$$(0.6 - 0.65)^2 + (0.4 - 0.42)^2 + (0.4 - 0.36)^2 + (0.6 - 0.59)^2 \\ = 0.0046$$

Reconstruction loss (Mean squared error)

x	
	y
0.6	0.4
0.4	0.6

x	
	y
0.65	0.42
0.36	0.59

$$MSE(\mathbf{x}, \mathbf{y}) = \sum_i (x_i - y_i)^2$$

$$(0.6 - 0.65)^2 + (0.4 - 0.42)^2 + (0.4 - 0.36)^2 + (0.6 - 0.59)^2 \\ = 0.0046$$

x	
	y
0.6	0.4
0.4	0.6

x	
	y
0.1	0.9
0.85	0.2

Reconstruction loss (Mean squared error)

x	
0.6	0.4
0.4	0.6

y	
0.65	0.42
0.36	0.59

$$MSE(\mathbf{x}, \mathbf{y}) = \sum_i (x_i - y_i)^2$$

$$(0.6 - 0.65)^2 + (0.4 - 0.42)^2 + (0.4 - 0.36)^2 + (0.6 - 0.59)^2 \\ = 0.0046$$

x	
0.6	0.4
0.4	0.6

y	
0.1	0.9
0.85	0.2

$$(0.6 - 0.1)^2 + (0.4 - 0.9)^2 + (0.4 - 0.85)^2 + (0.6 - 0.2)^2 \\ = 0.8625$$

Reconstruction loss (log-loss)

Reconstruction loss (log-loss)

$$LL(\textcolor{blue}{x}, \textcolor{red}{y}) = \sum_i -\textcolor{blue}{x}_i \ln \textcolor{red}{y}_i - (1-\textcolor{blue}{x}_i) \ln (1-\textcolor{red}{y}_i)$$

Reconstruction loss (log-loss)

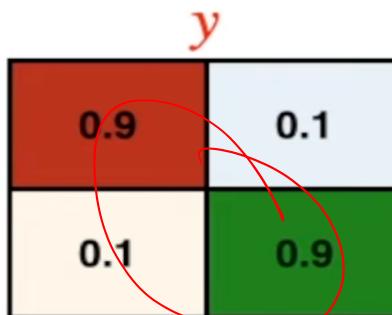
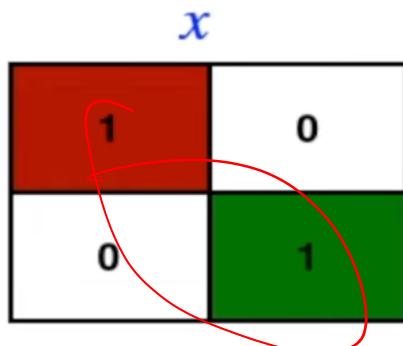
x	y
1	0
0	1

x	y
0.9	0.1
0.1	0.9

$$LL(x, y) = \sum_i -x_i \ln y_i - (1-x_i) \ln(1-y_i)$$

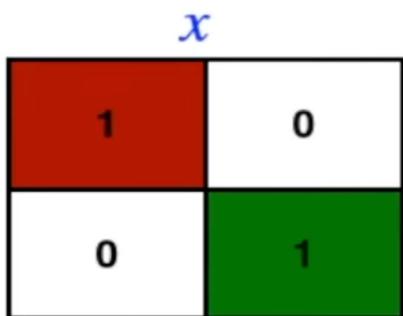
$$-\ln(0.9) - \ln(1 - 0.1) - \ln(0.9) - \ln(1 - 0.1) \\ = 0.421$$

Reconstruction loss (log-loss)



$$LL(x, y) = \sum_i -x_i \ln y_i - (1-x_i) \ln(1-y_i)$$

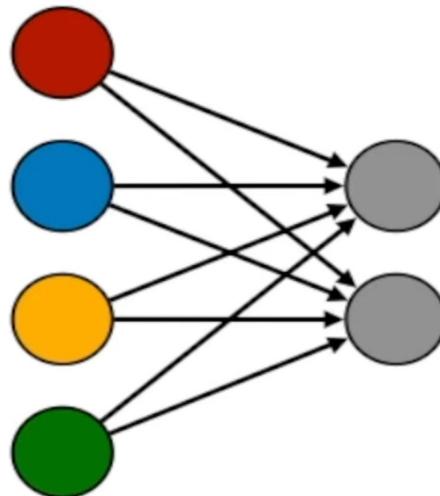
$$\begin{aligned} & -\ln(0.9) - \ln(1 - 0.1) - \ln(0.9) - \ln(1 - 0.1) \\ & = 0.421 \end{aligned}$$



$$\begin{aligned} & -\ln(0.2) - \ln(1 - 0.8) - \ln(0.2) - \ln(1 - 0.8) \\ & = 6.438 \end{aligned}$$

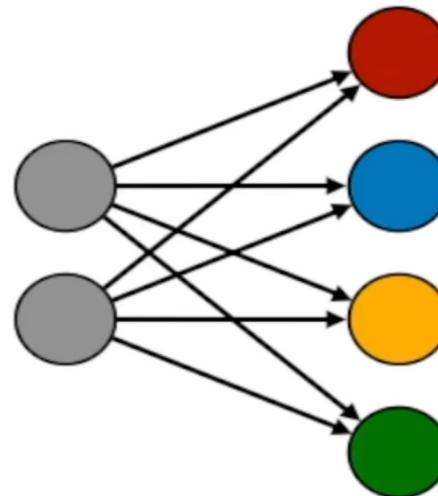
Training an variational autoencoder

Encoder

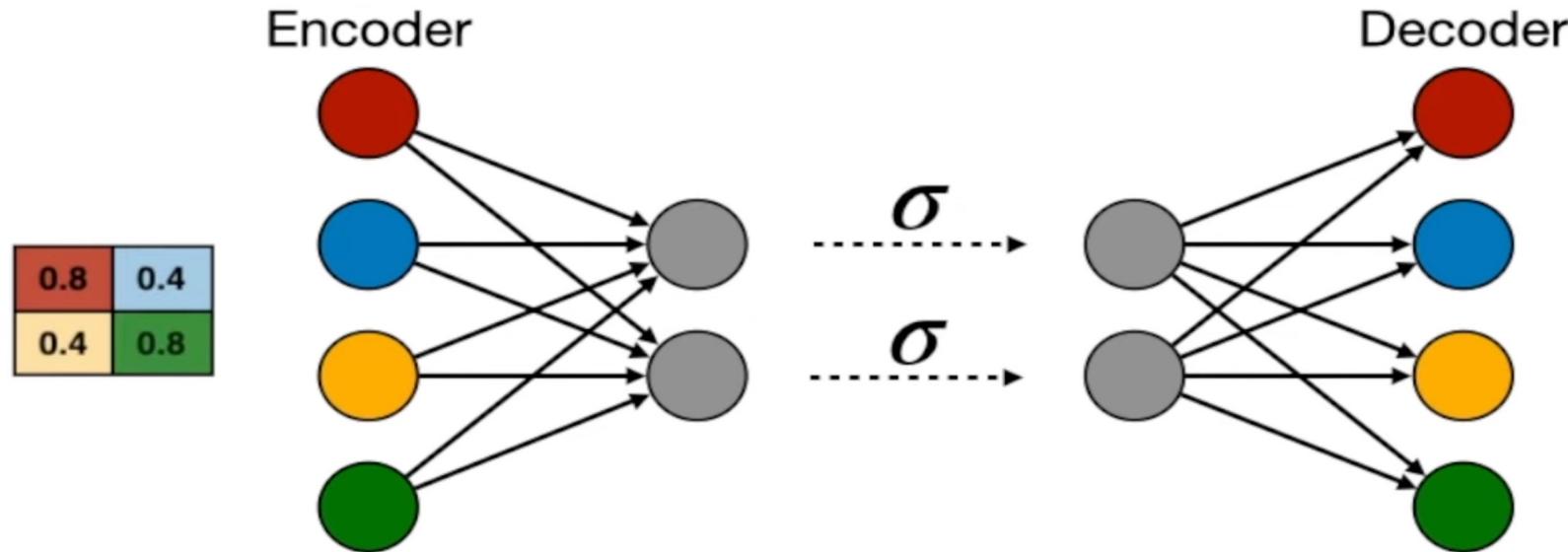


σ
 σ

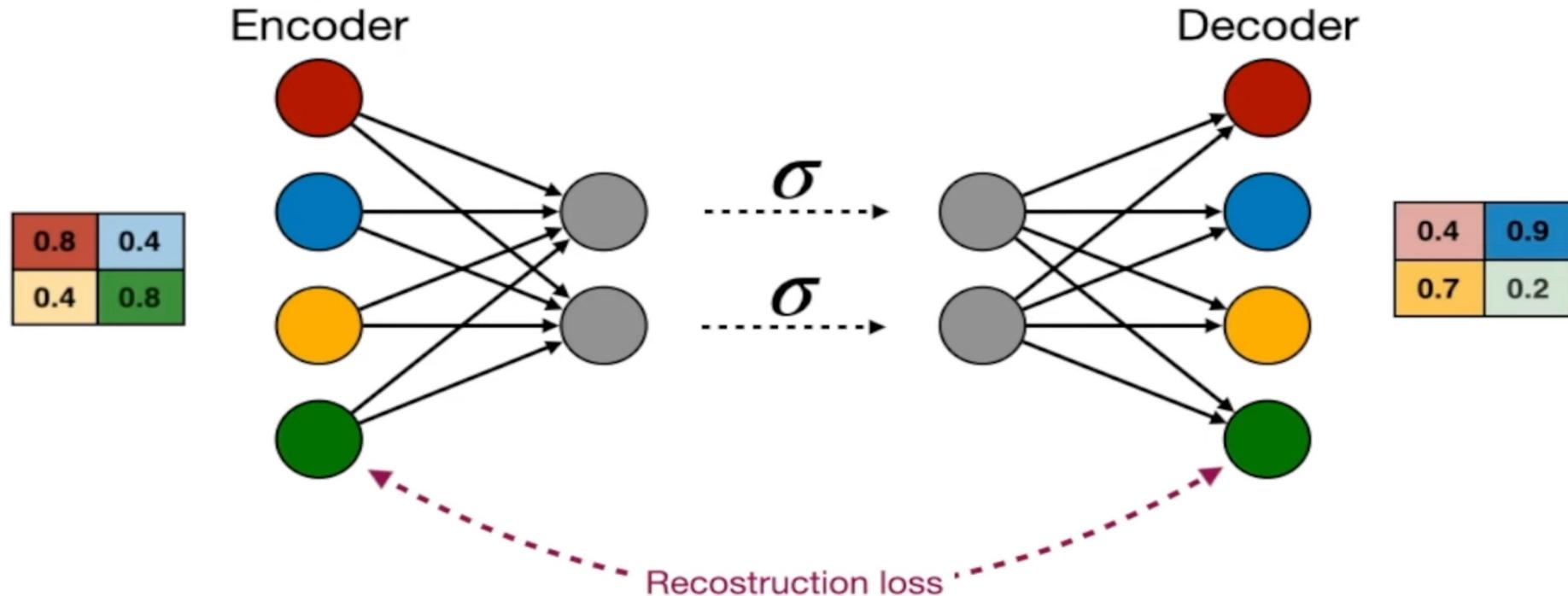
Decoder



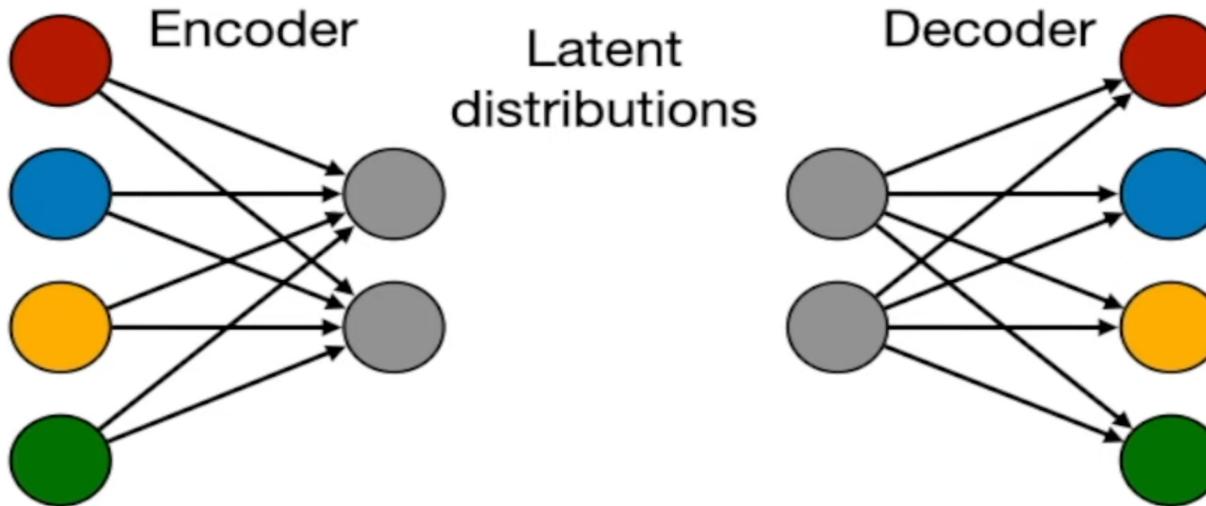
Training an variational autoencoder



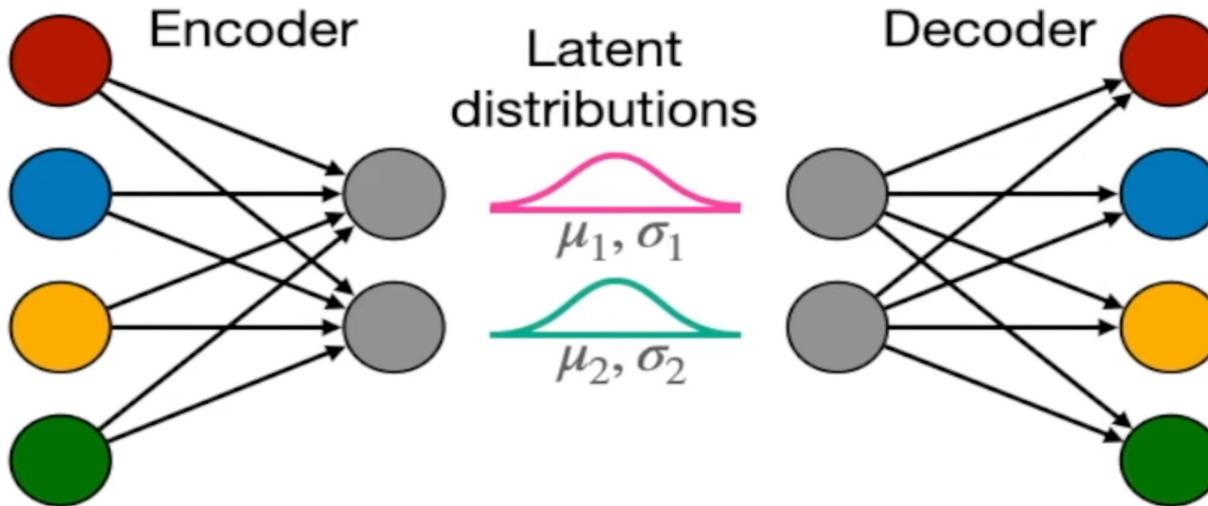
Training an variational autoencoder



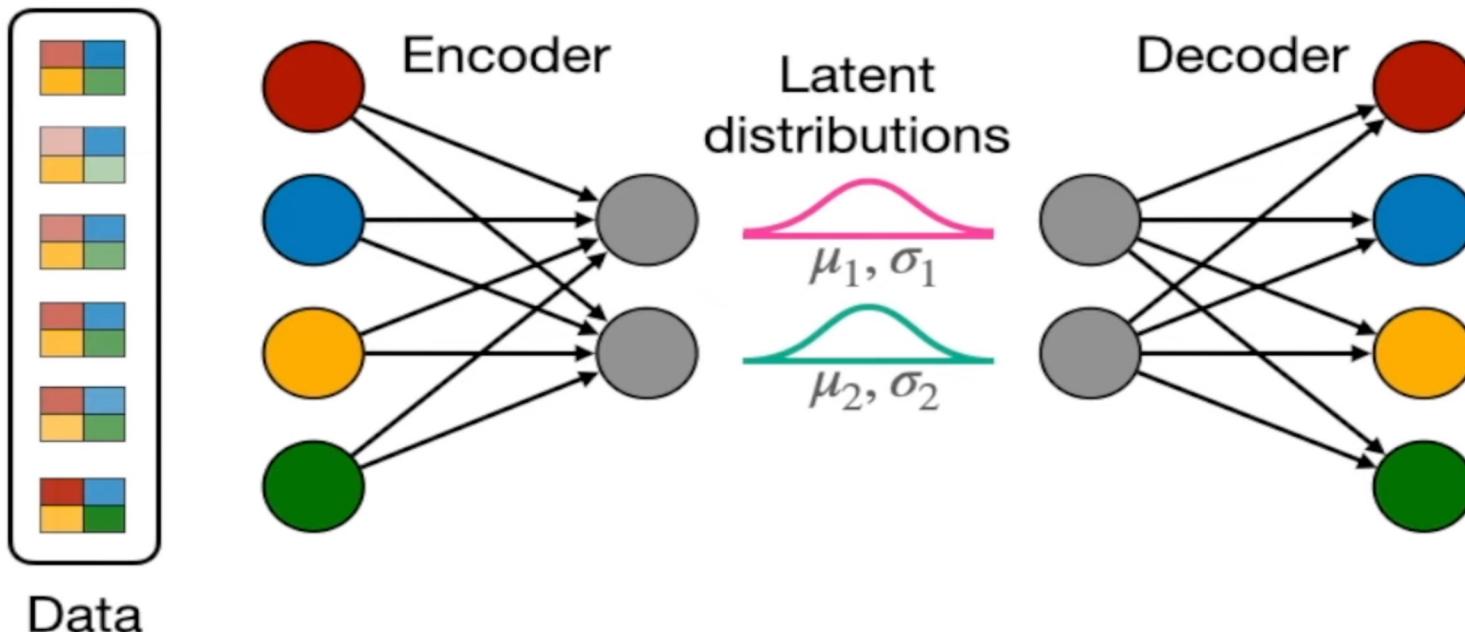
Training a variational auto encoder



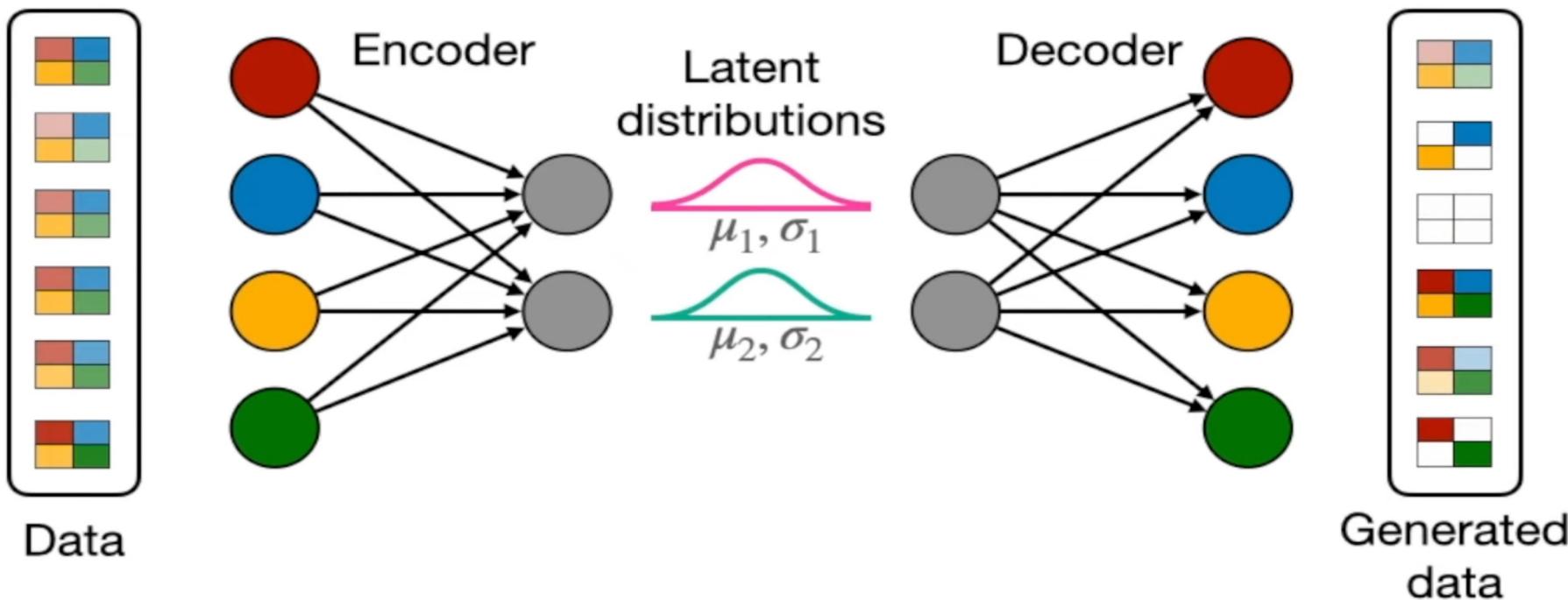
Training a variational auto encoder



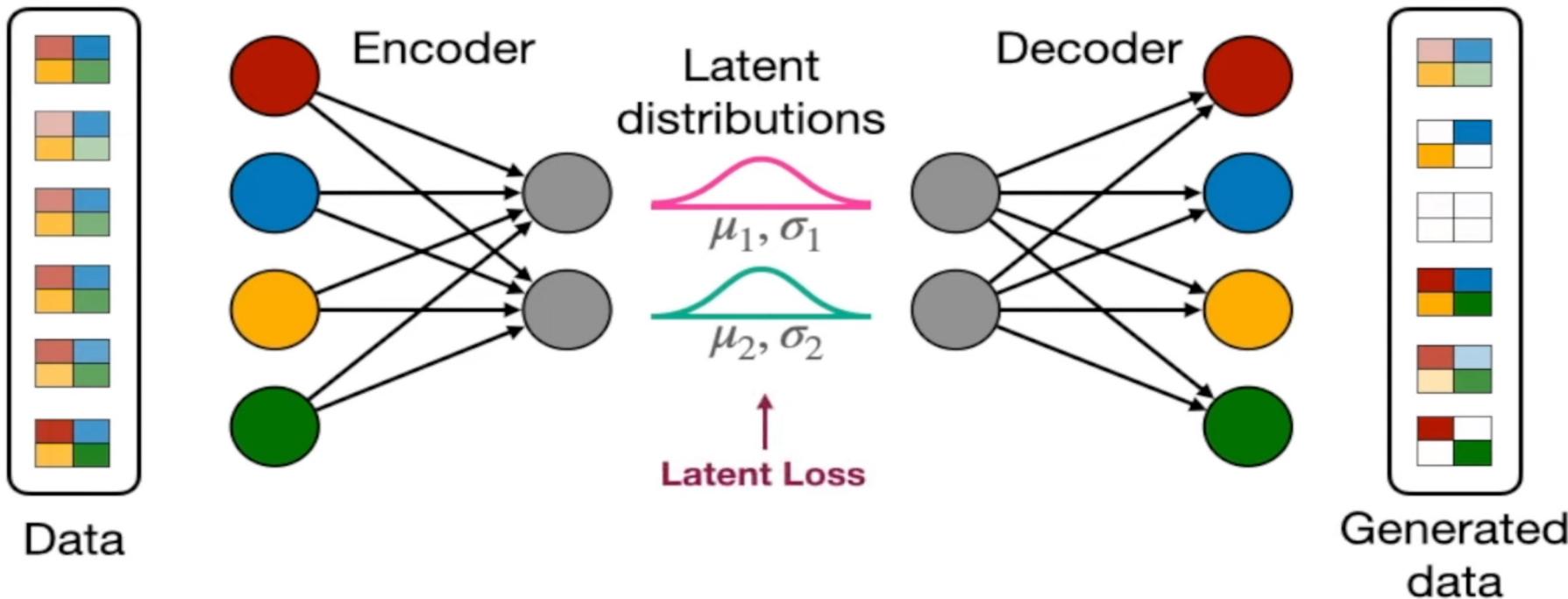
Training a variational auto encoder



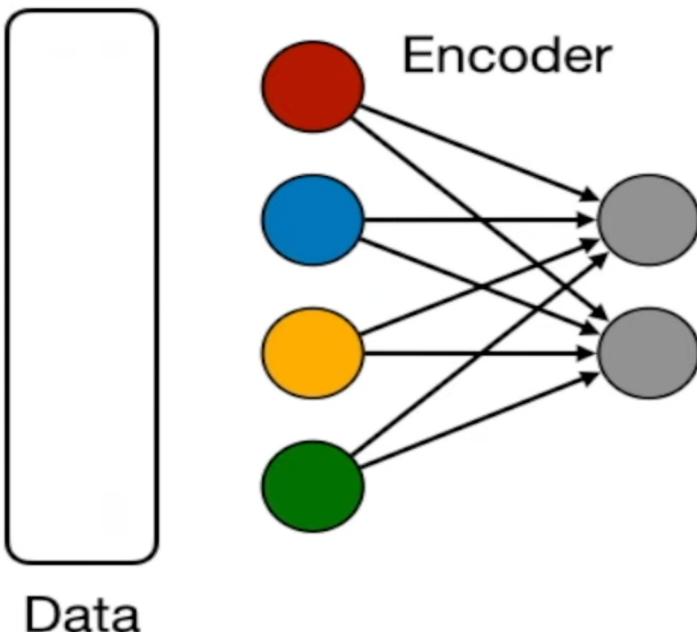
Training a variational auto encoder



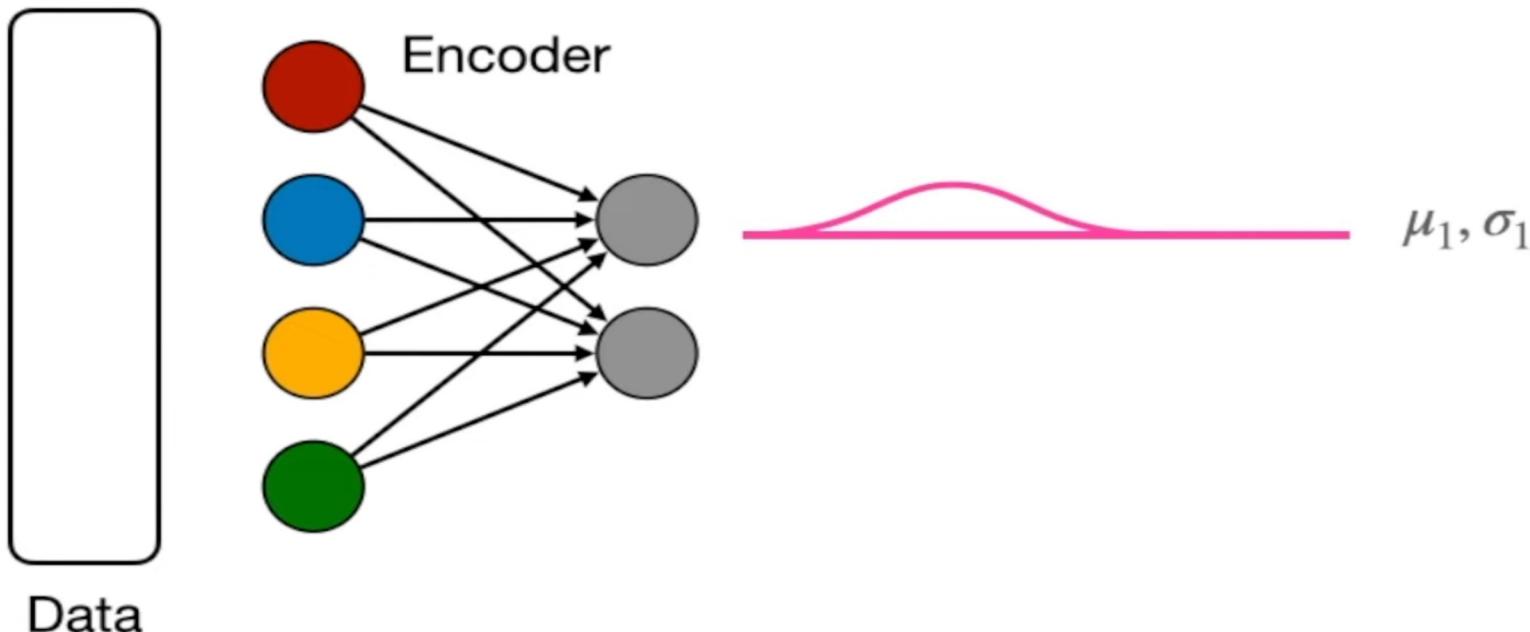
Training a variational auto encoder



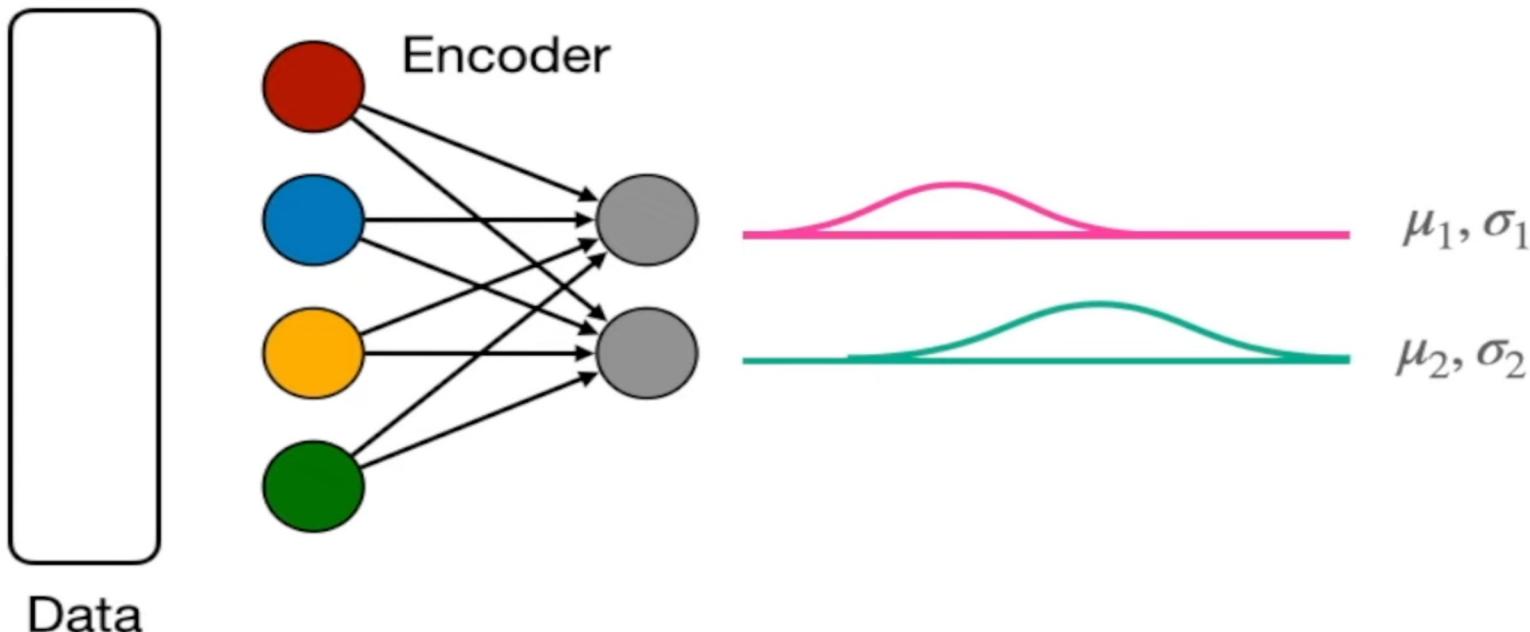
Training a variational auto encoder



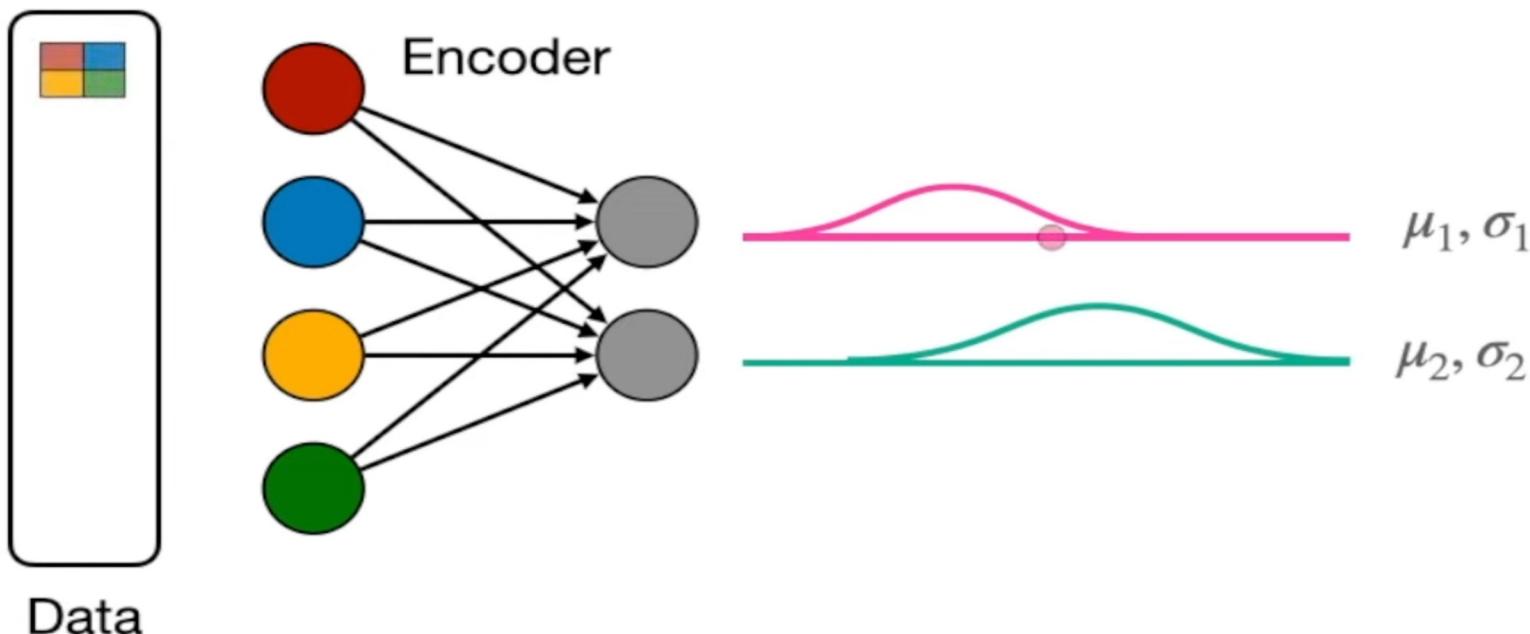
Training a variational auto encoder



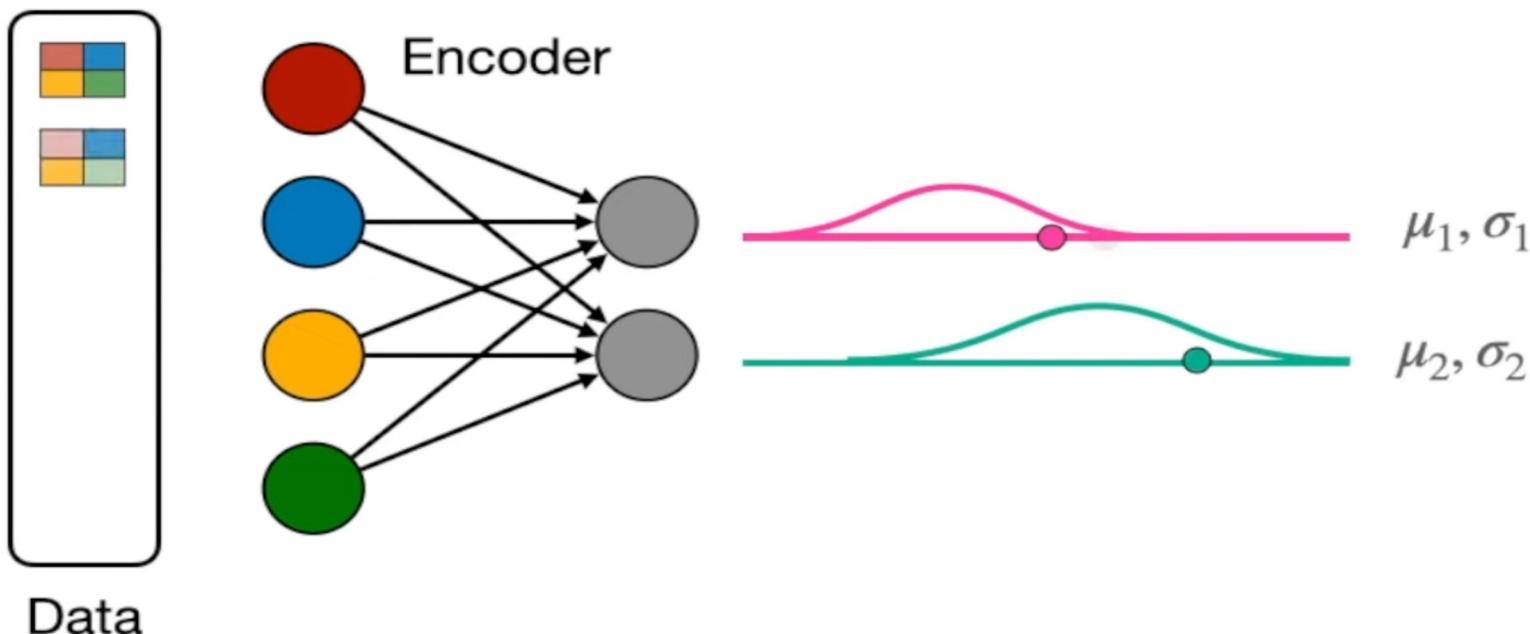
Training a variational auto encoder



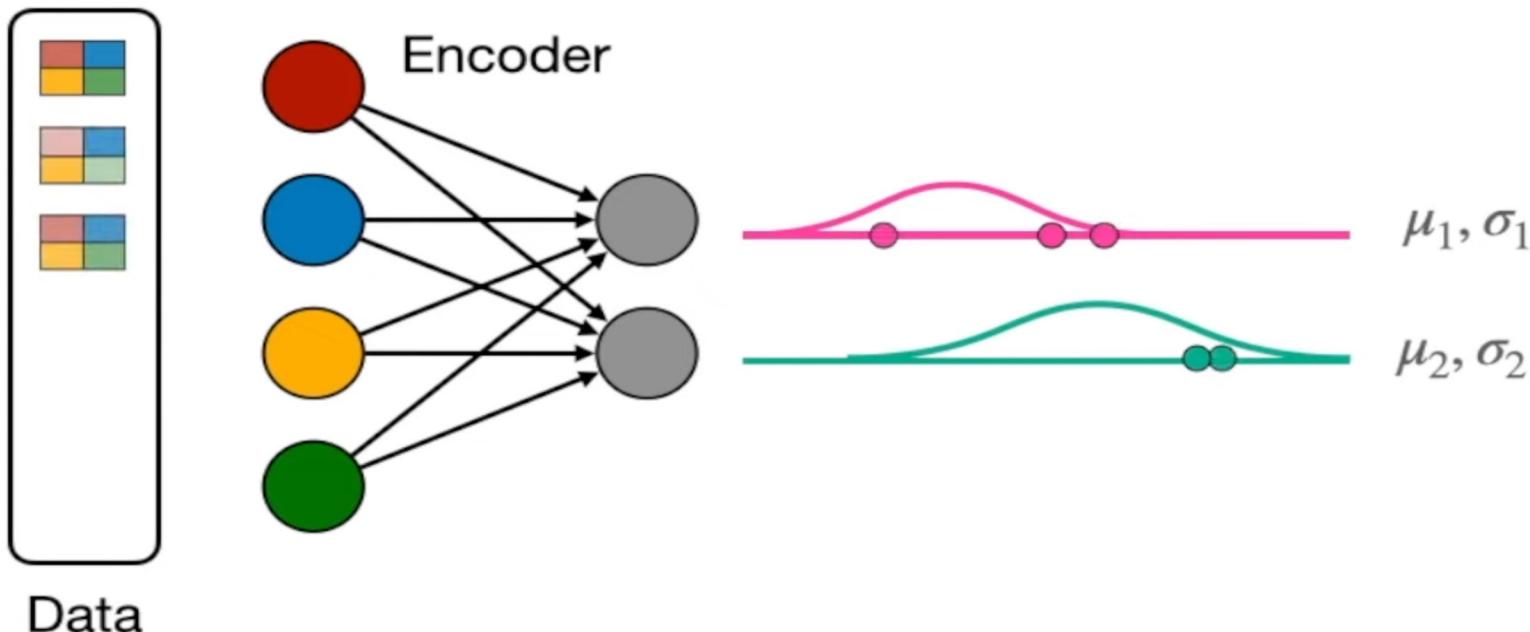
Training a variational auto encoder



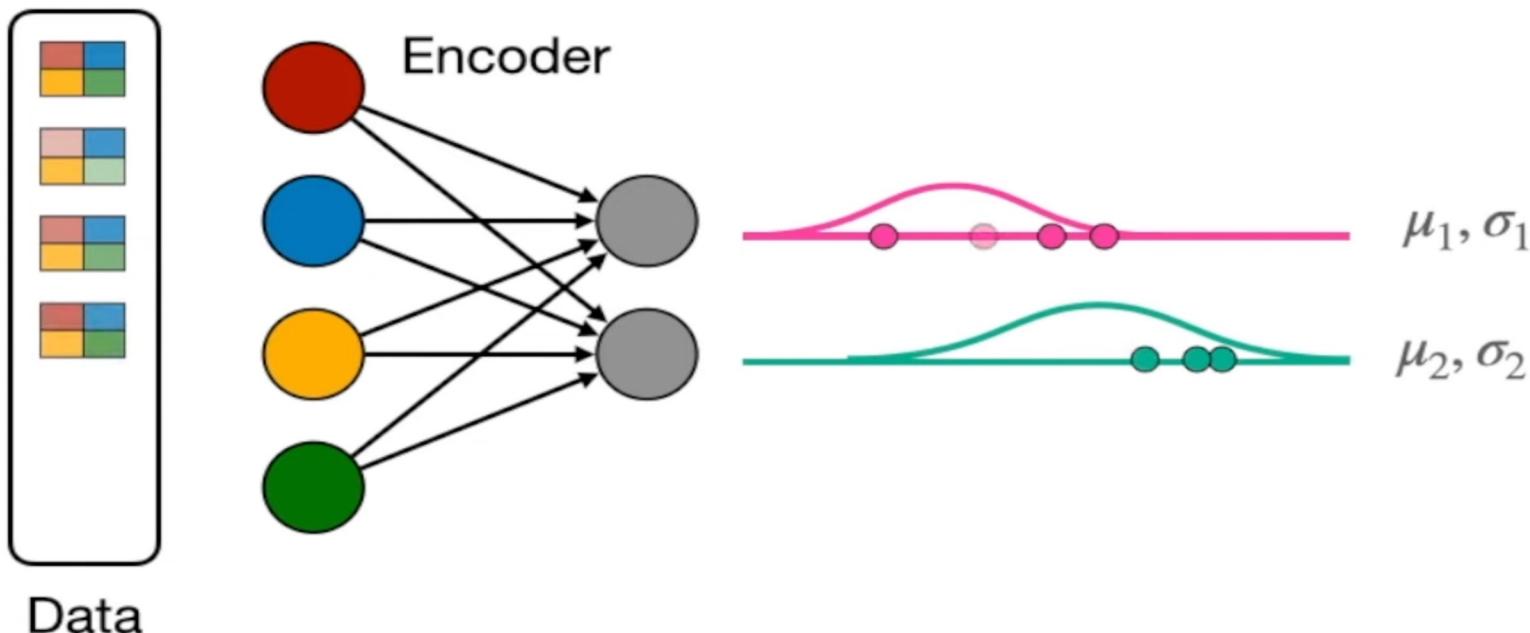
Training a variational auto encoder



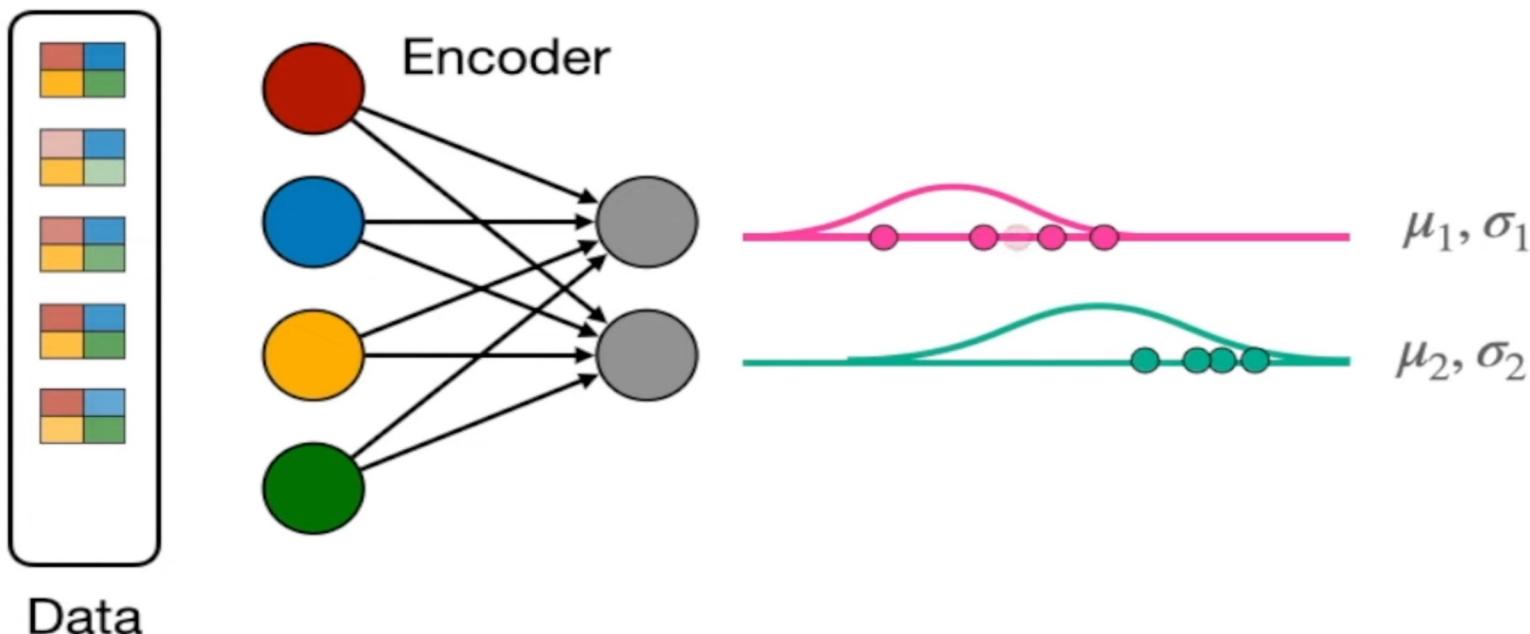
Training a variational auto encoder



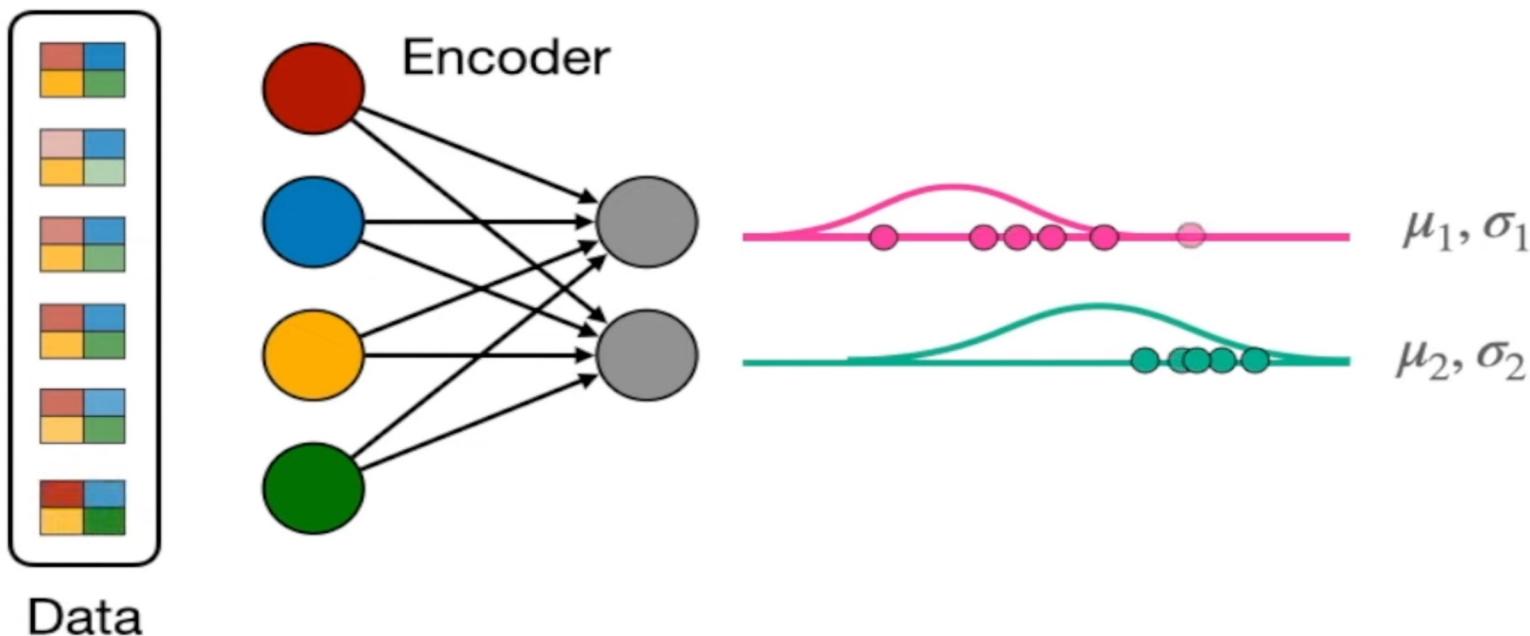
Training a variational auto encoder



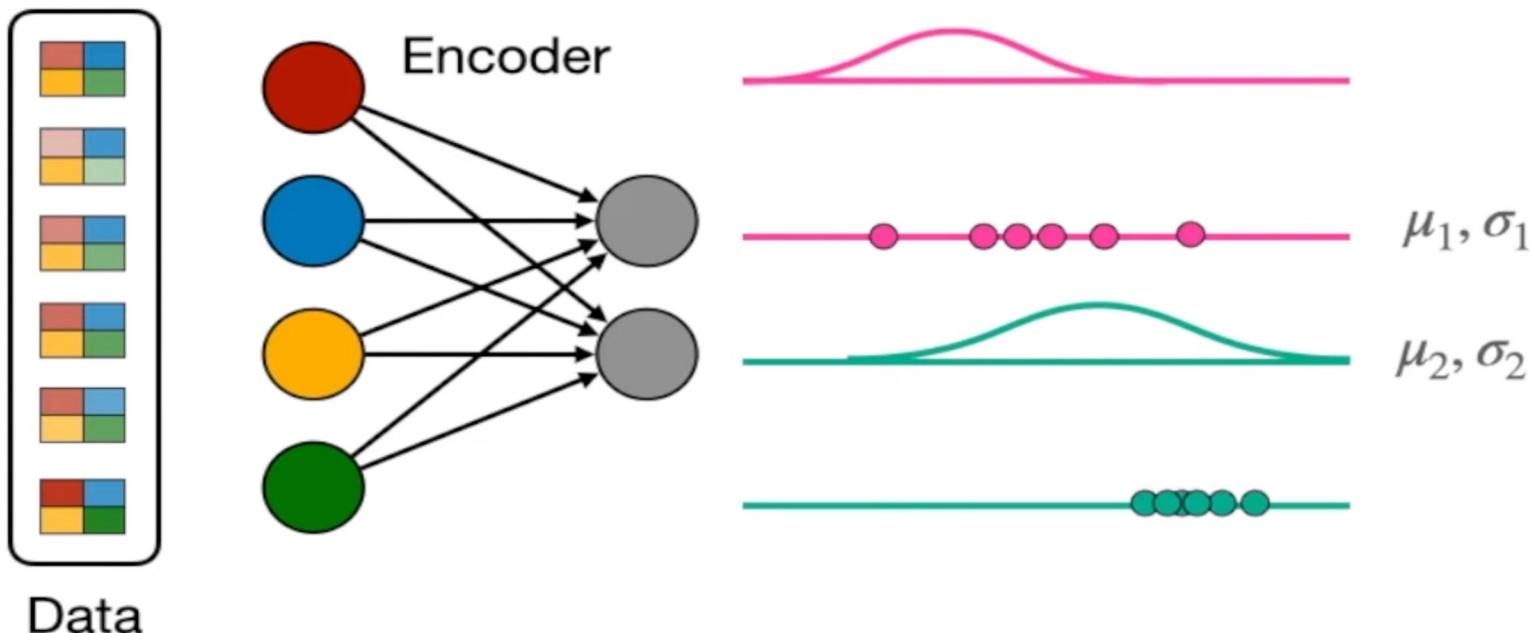
Training a variational auto encoder



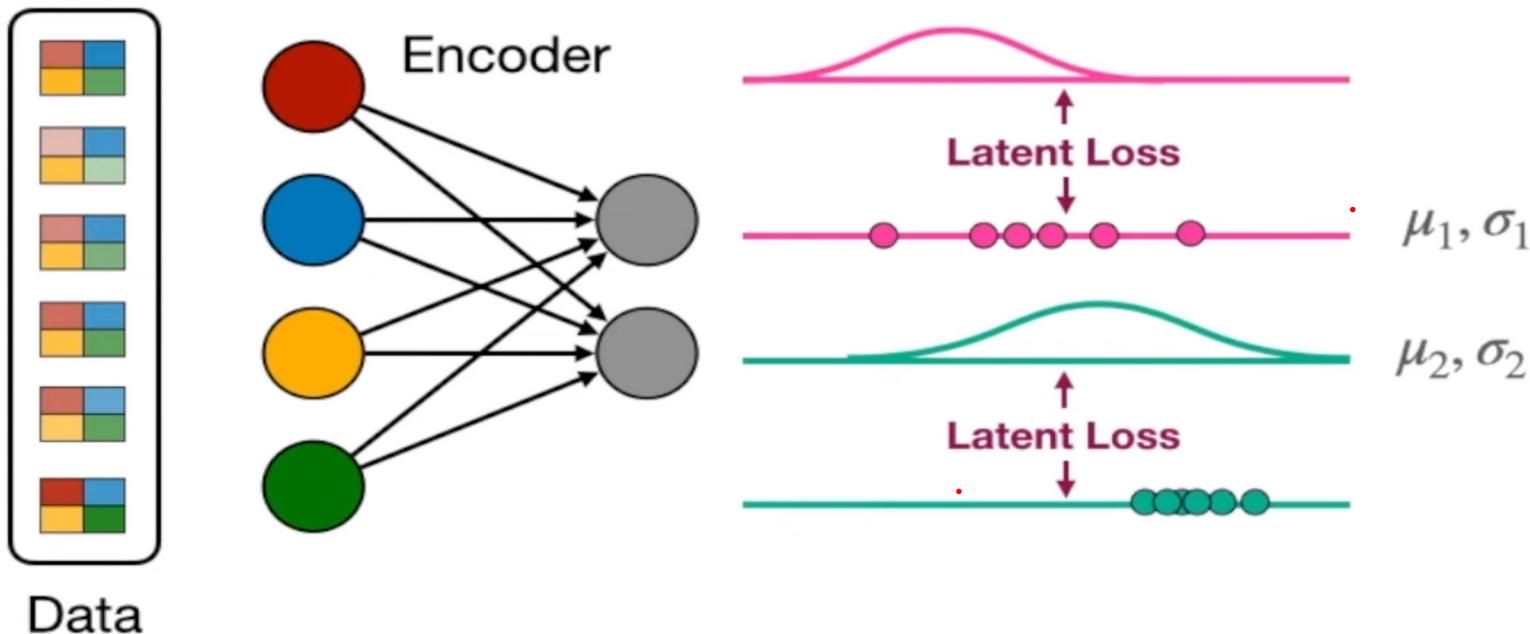
Training a variational auto encoder



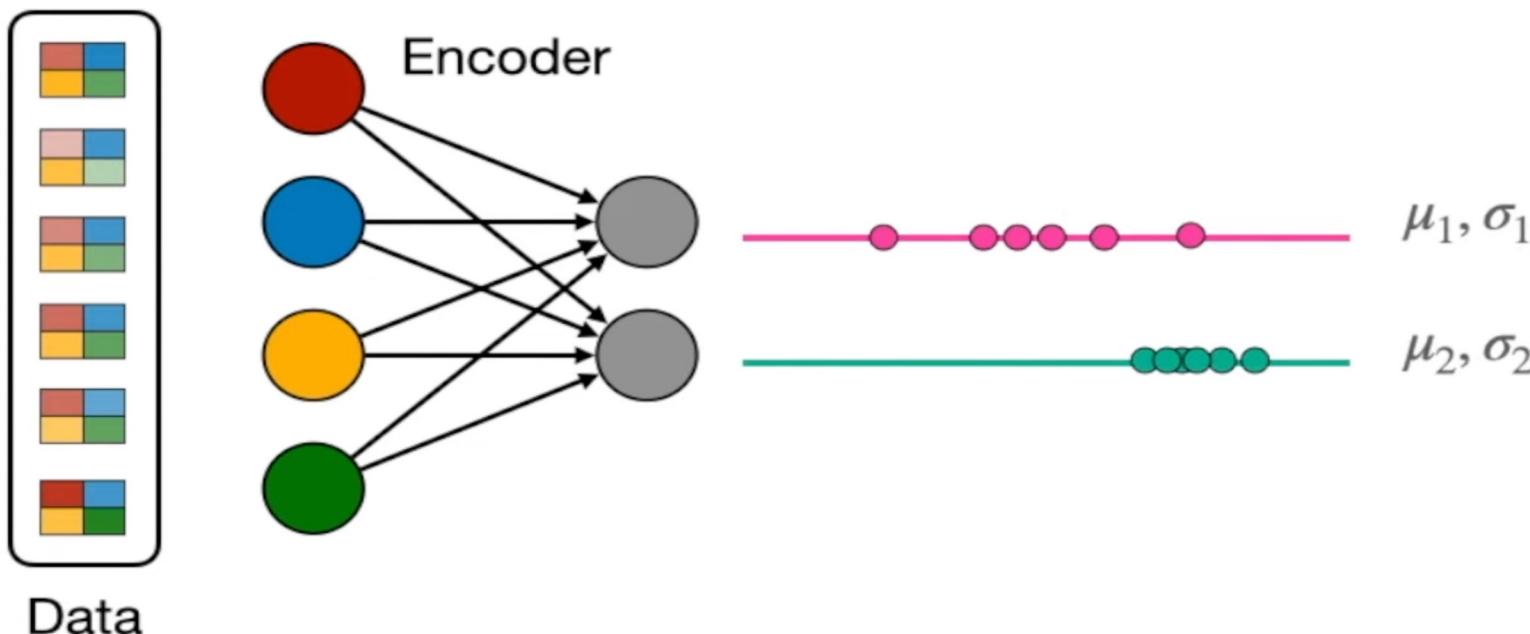
Training a variational auto encoder



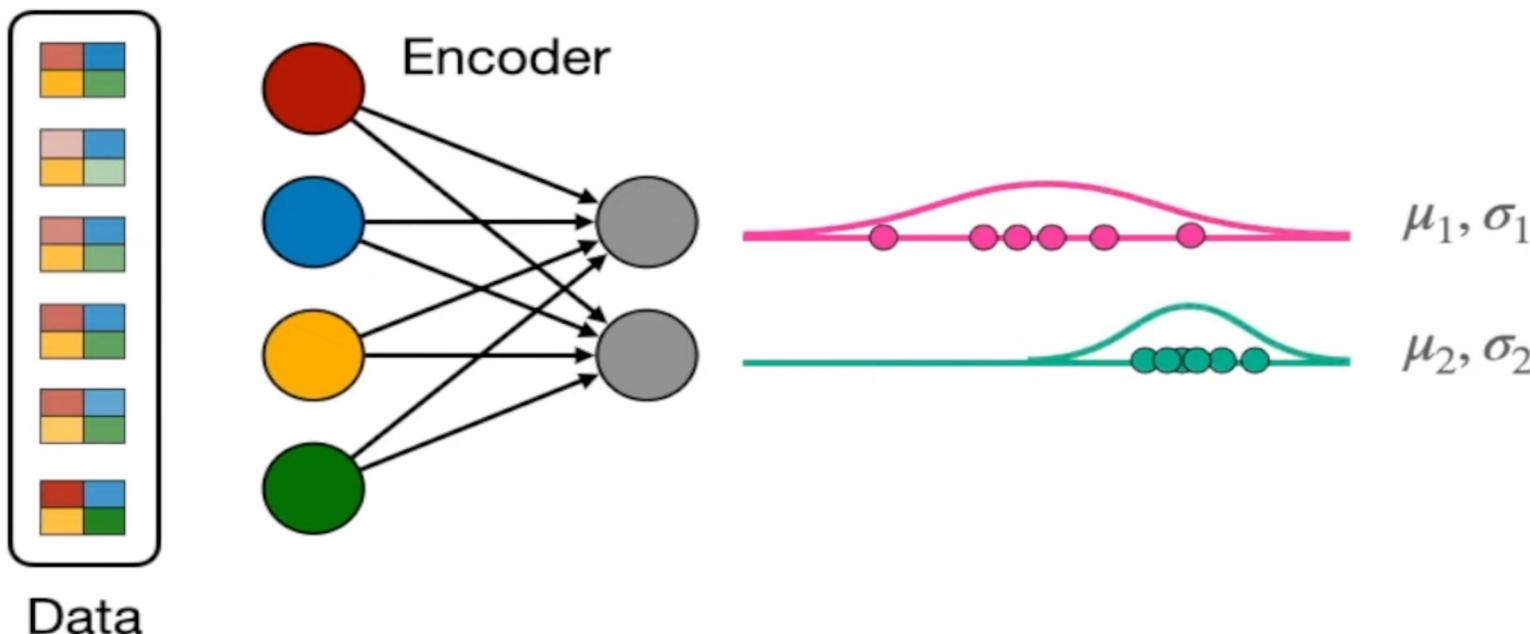
Training a variational auto encoder



Training a variational auto encoder



Training a variational auto encoder



Latent loss (KL-divergence)

Latent loss (KL-divergence)

$$KL(p, q) = \sum_i p_i \ln \left(\frac{p_i}{q_i} \right)$$

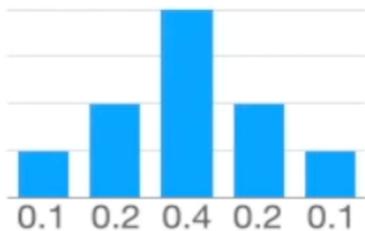
Latent loss (KL-divergence)

$$KL(p, q) = \sum_i p_i \ln \left(\frac{p_i}{q_i} \right)$$



Latent loss (KL-divergence)

p

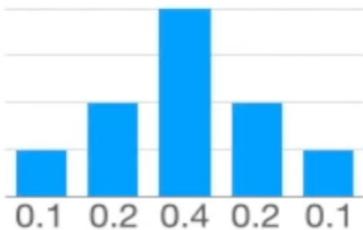


$$KL(p, q) = \sum_i p_i \ln \left(\frac{p_i}{q_i} \right)$$

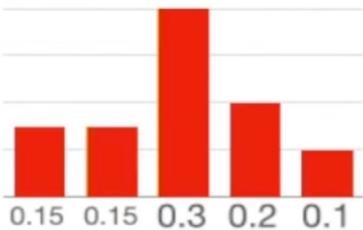
Latent loss (KL-divergence)

$$KL(\textcolor{blue}{p}, \textcolor{red}{q}) = \sum_i p_i \ln \left(\frac{p_i}{q_i} \right)$$

p



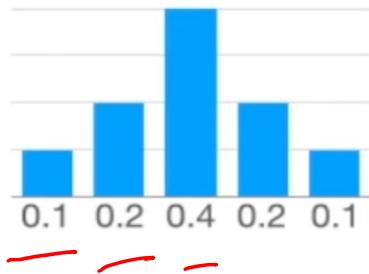
q



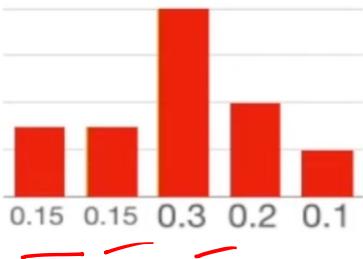
Latent loss (KL-divergence)

$$KL(p, q) = \sum_i p_i \ln \left(\frac{p_i}{q_i} \right)$$

p



q

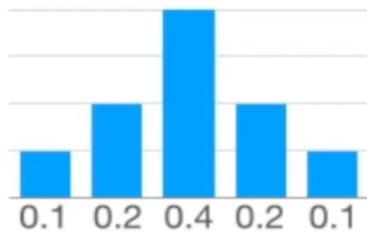


$$\begin{aligned} & 0.1 \ln \frac{0.1}{0.15} + 0.2 \ln \frac{0.2}{0.15} + 0.4 \ln \frac{0.4}{0.4} + 0.2 \ln \frac{0.2}{0.2} + 0.1 \ln \frac{0.1}{0.1} \\ & = 0.132 \end{aligned}$$

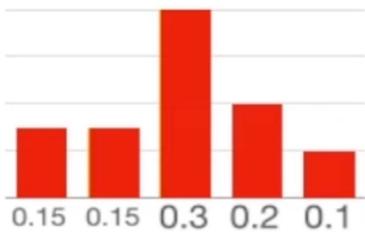
Latent loss (KL-divergence)

$$KL(p, q) = \sum_i p_i \ln \left(\frac{p_i}{q_i} \right)$$

p

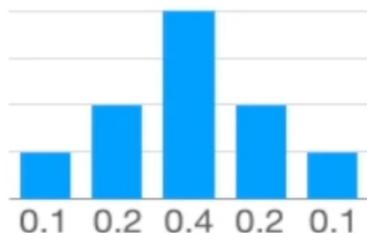


q

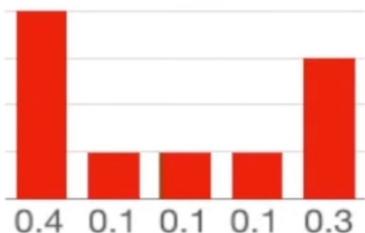


$$0.1 \ln \frac{0.1}{0.15} + 0.2 \ln \frac{0.2}{0.15} + 0.4 \ln \frac{0.4}{0.4} + 0.2 \ln \frac{0.2}{0.2} + 0.1 \ln \frac{0.1}{0.1}$$
$$= 0.132$$

p



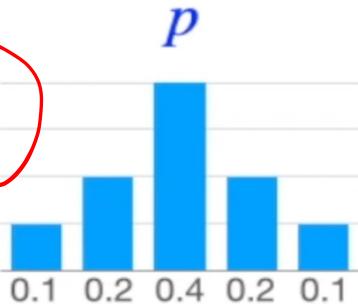
q



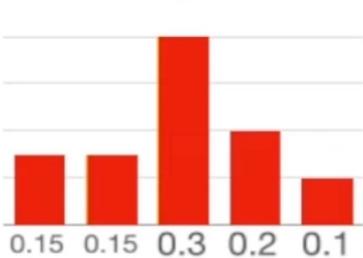
Latent loss (KL-divergence)

$$KL(p, q) = \sum_i p_i \ln \left(\frac{p_i}{q_i} \right)$$

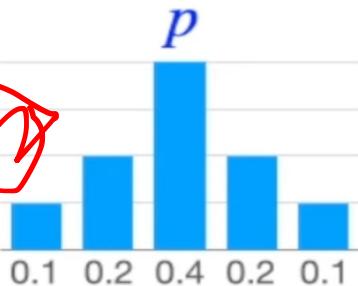
p



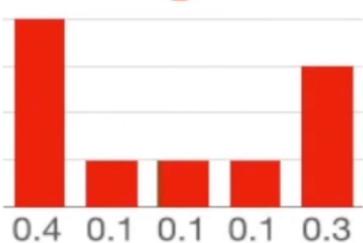
q



p



q

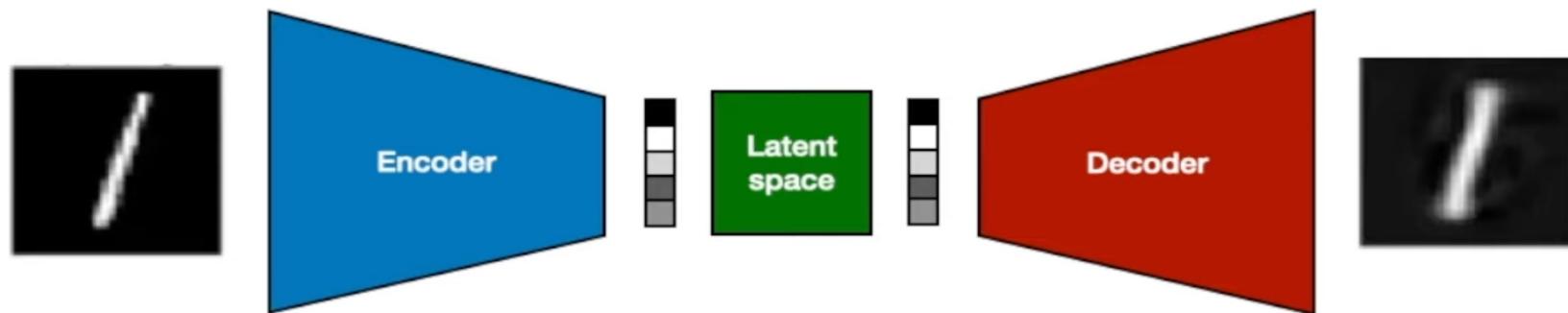


$$0.1 \ln \frac{0.1}{0.15} + 0.2 \ln \frac{0.2}{0.15} + 0.4 \ln \frac{0.4}{0.4} + 0.2 \ln \frac{0.2}{0.2} + 0.1 \ln \frac{0.1}{0.1}$$
$$= 0.132$$

$$0.1 \ln \frac{0.1}{0.4} + 0.2 \ln \frac{0.2}{0.1} + 0.4 \ln \frac{0.4}{0.1} + 0.2 \ln \frac{0.2}{0.1} + 0.1 \ln \frac{0.1}{0.3}$$
$$= 0.583$$

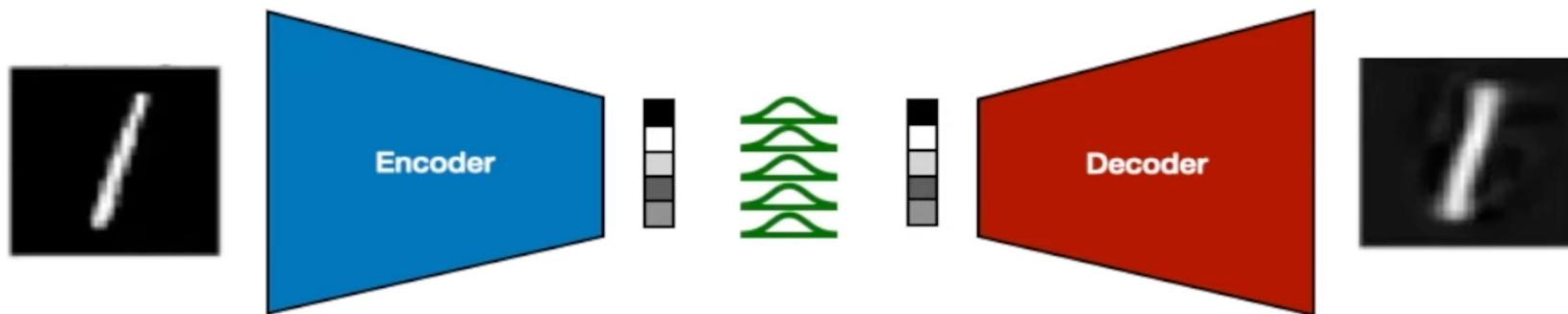
?

Training autoencoders



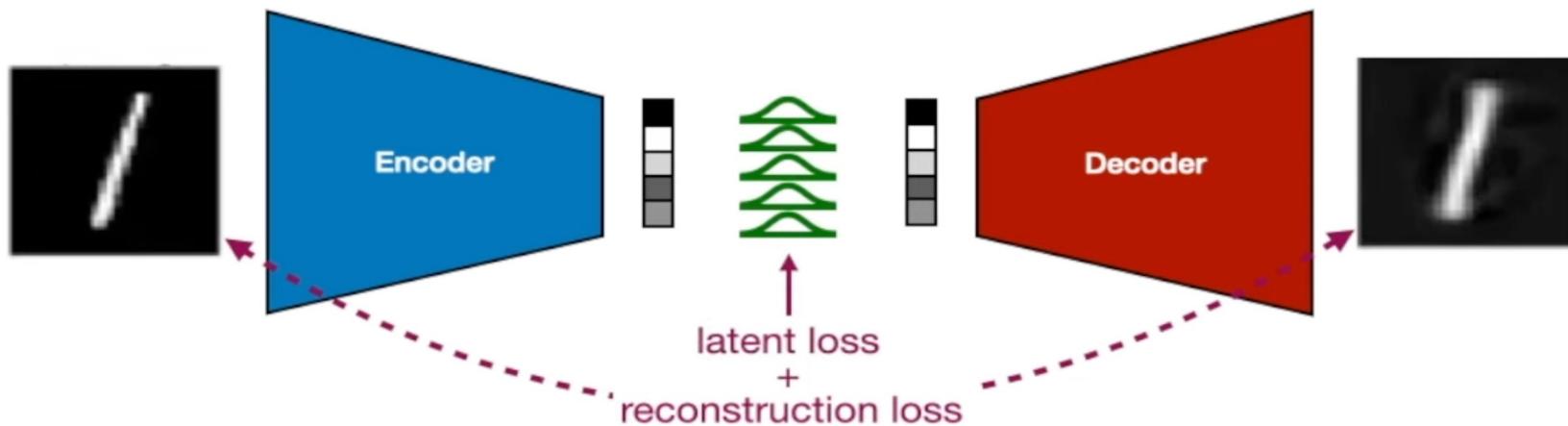
Training autoencoders

Initialize encoder, decoder
and latent distributions
with random weights



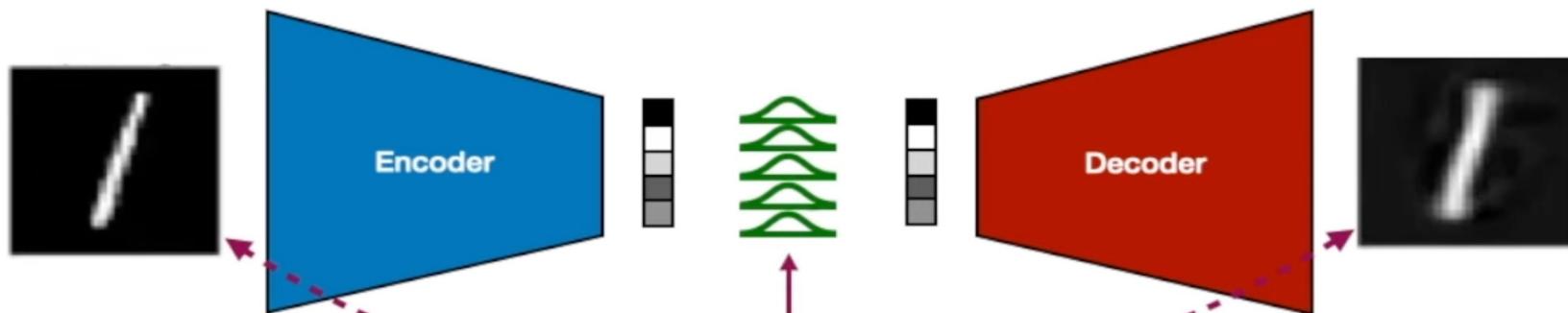
Training autoencoders

Initialize encoder, decoder
and latent distributions
with random weights



Training autoencoders

Initialize encoder, decoder
and latent distributions
with random weights



Do backpropagation
and update weights

Images generated by a VAE



Source: Tom White