UMAP AND T-SNE ARE ALSO TWO USEFUL TECHNIQUES TO VISUALIZE LARGE DIMENSIONAL SPACES

INSTEAD OF MATRIX FACTORIZATION SUCH AS PCA, THEY ARE BASED ON GRAPH LAYOUT

1. WE BUILD A GRAPH OF OUR HIGH-DIMENSIONALITY DATA

2. WE OPTIMIZE TO GET THE MOST SIMILAR GRAPH IN TWO DIMENSIONS

https://lvdmaaten.github.io/tsne/

https://umap-learn.readthedocs.io/en/latest/how_umap_works.html

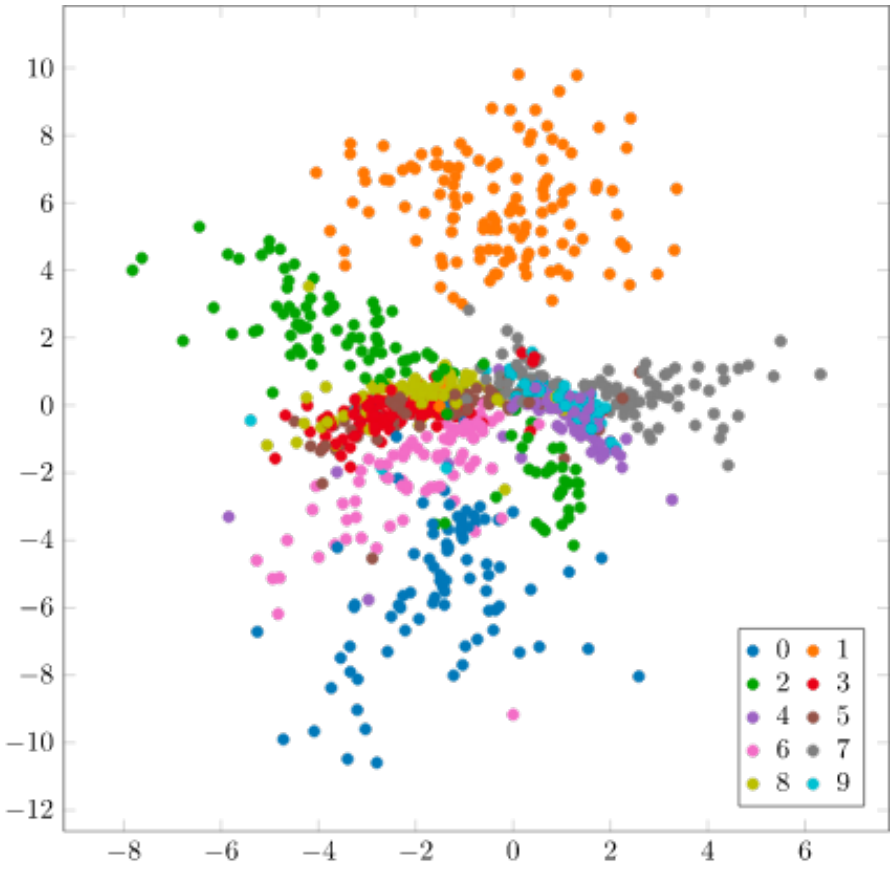
UMAP EXAMPLES AND EXPLANATIONS;

https://pair-code.github.io/understanding-umap/#:~:text=In%20the%20simplest%20sense%2C%20UMAP,behind%20them%20is%20remarkably%20simple.

HOW MIGHT YOU SOLVE THESE RELATED PROBLEMS ONCE YOU HAVE THE LATENT SPACE COORDINATES FOR YOUR TRAINING SAMPLE?

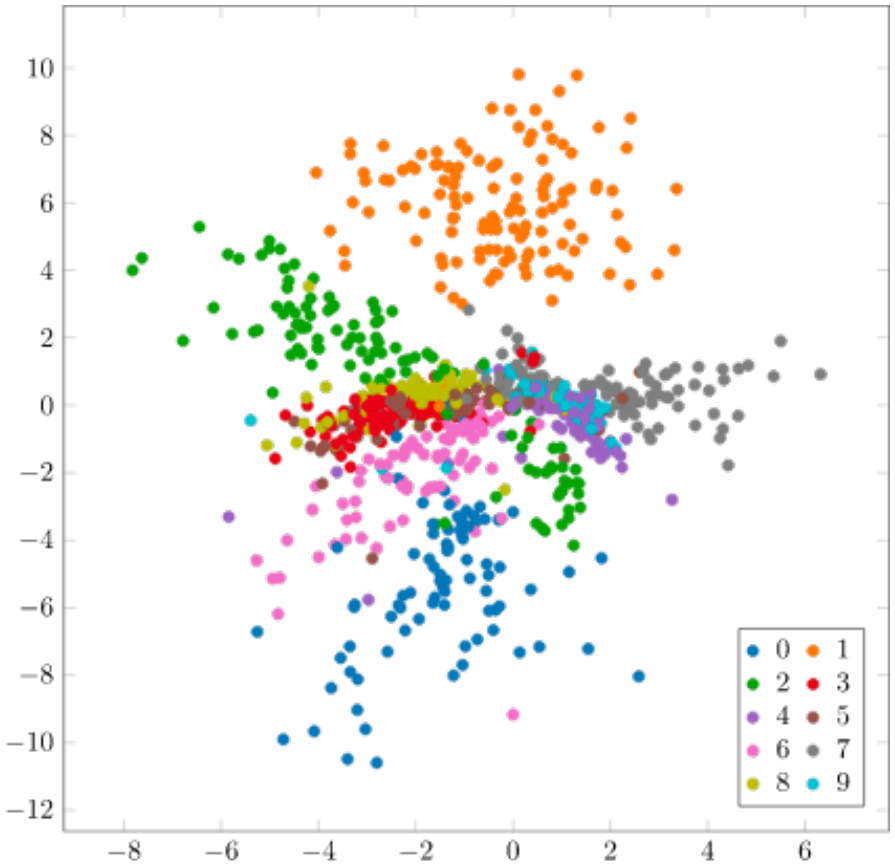
GENERATE A RANDOM SAMPLE DRAWN FROM THE INPUT DISTRIBUTION ("**GENERATIVE MODEL**")

ESTIMATE THE PROBABILITY DENSITY OF AN ARBITRARY INPUT, RELATIVE TO THE INPUT DISTRIBUTION ("**PROBABILISTIC MODEL**")



HOW CAN I ESTIMATE P(X)?

(P(z|x))

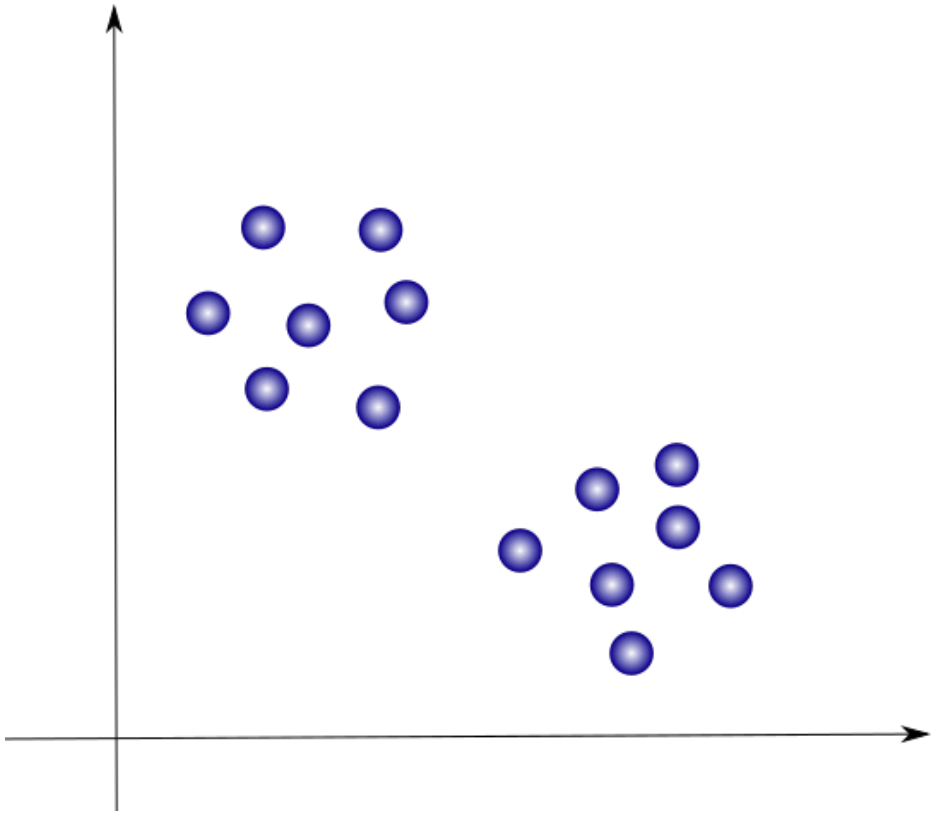


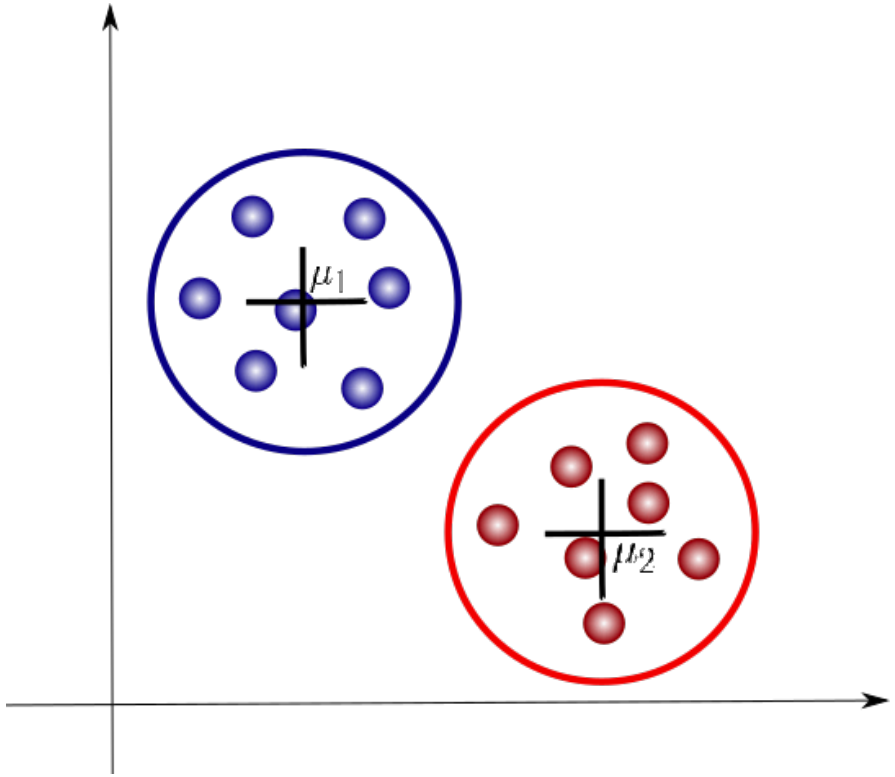
HOW CAN I ESTIMATE P(X)?

(P(z|x))

WHEN YOU DON'T KNOW, ASSUME IT IS GAUSSIAN….

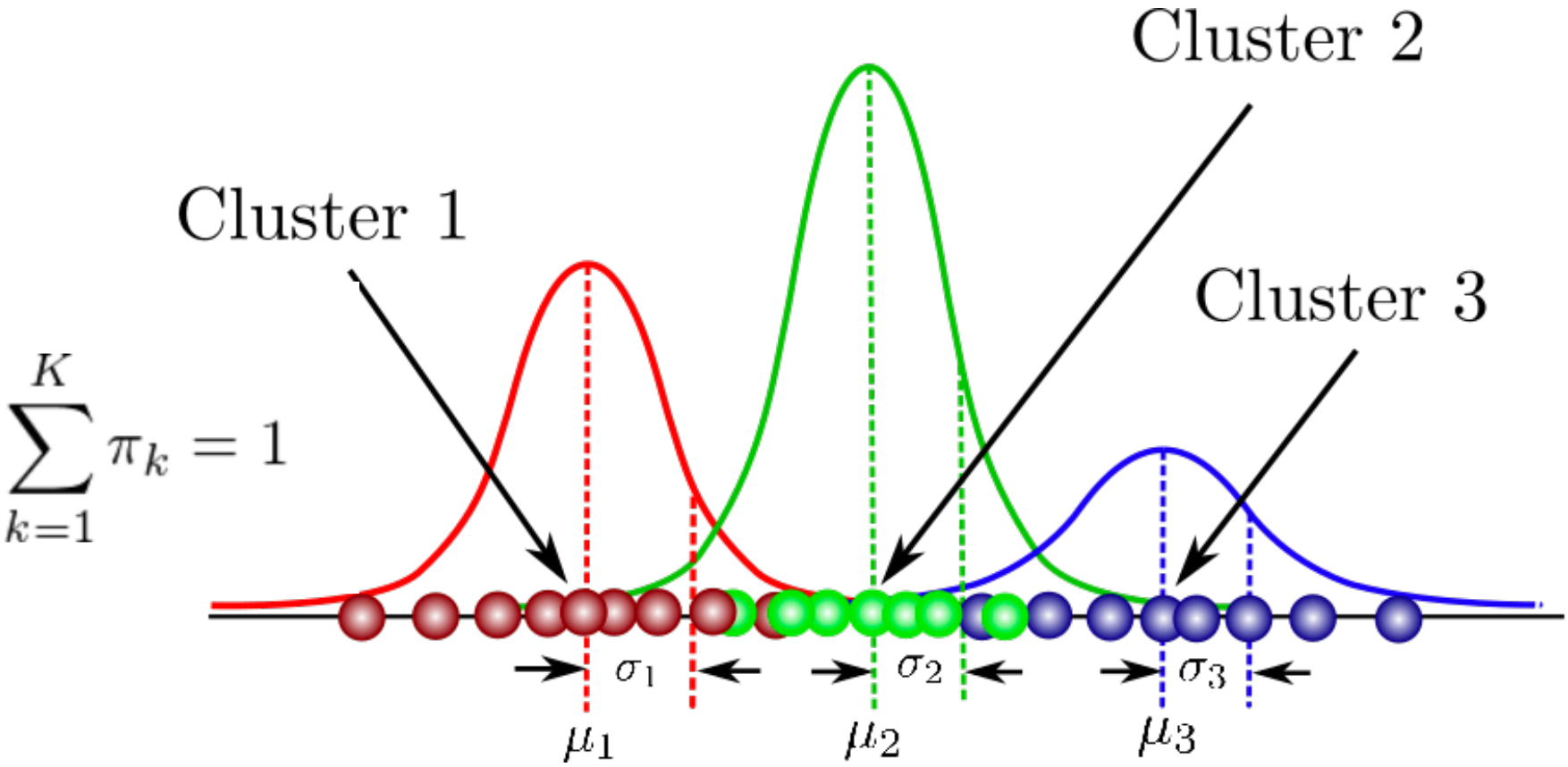
GAUSSIAN MIXTURE MODELS (GMMs) ARE DENSITY ESTIMATOR METHODS THAT FIT MULTIPLE GAUSSIANS TO THE REPRESENTATION





https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95

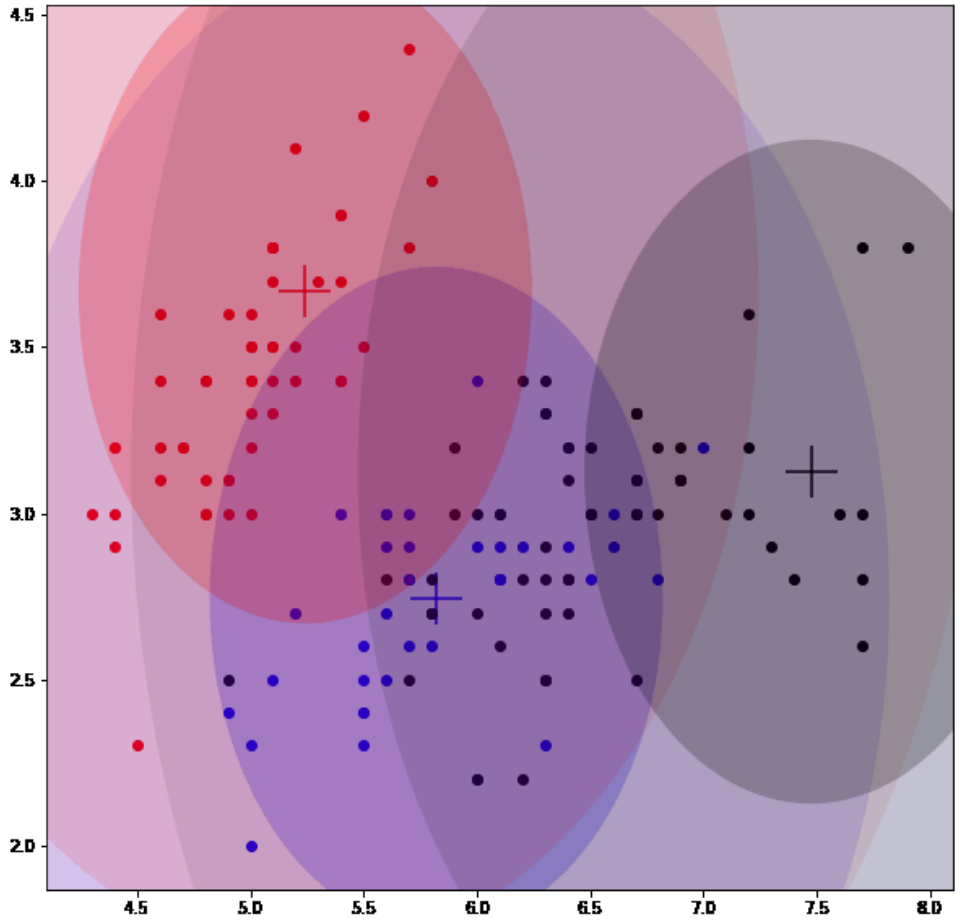
GAUSSIAN MIXTURE MODELS (GMMs) ARE DENSITY ESTIMATOR METHODS THAT FIT MULTIPLE GAUSSIANS TO THE REPRESENTATION



means, sigmas and scale factors of each gaussian are free parameters

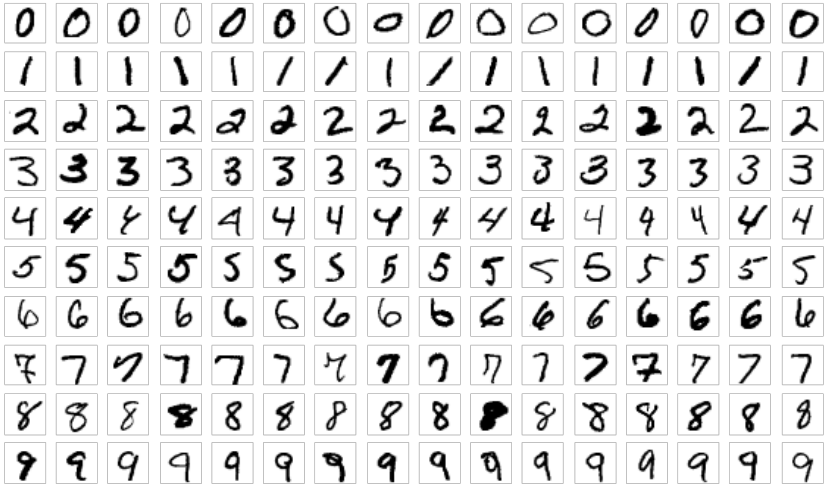
https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95

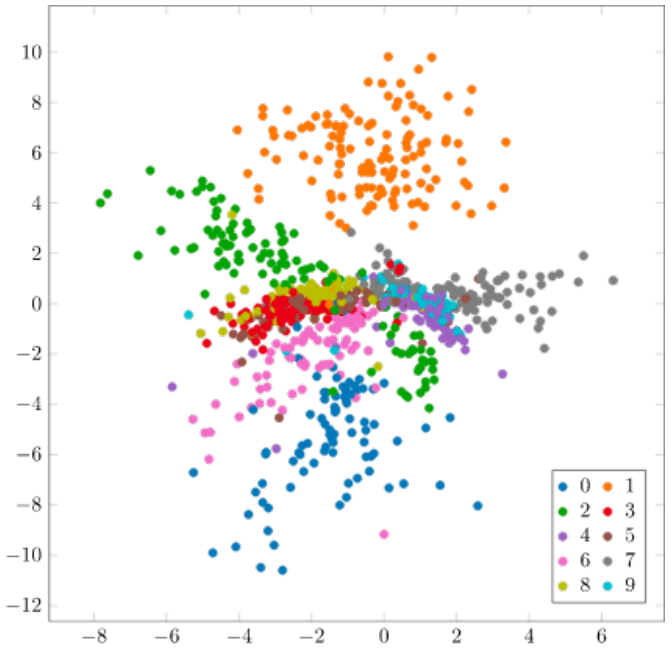
GAUSSIAN MIXTURE MODELS (GMMs) ARE DENSITY ESTIMATOR METHODS THAT FIT MULTIPLE GAUSSIANS TO THE REPRESENTATION



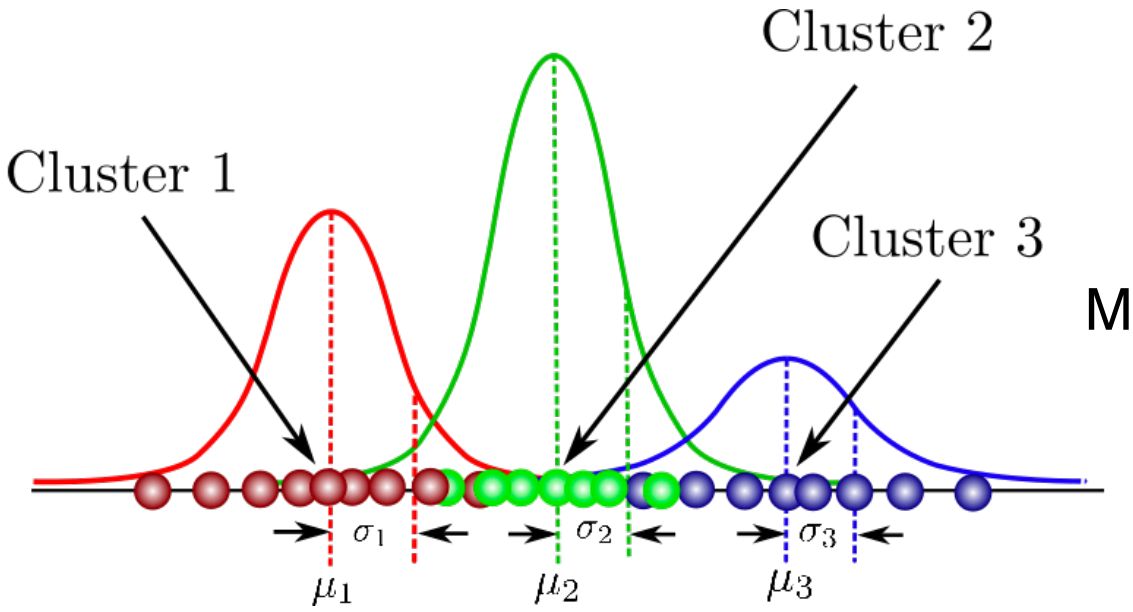
https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95

DATA



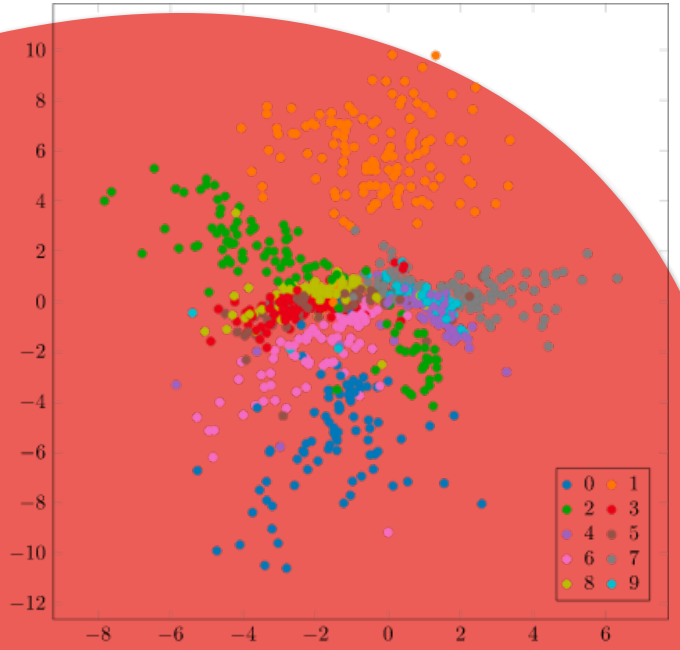


REPRESENTATION
(AUTOENCODER)



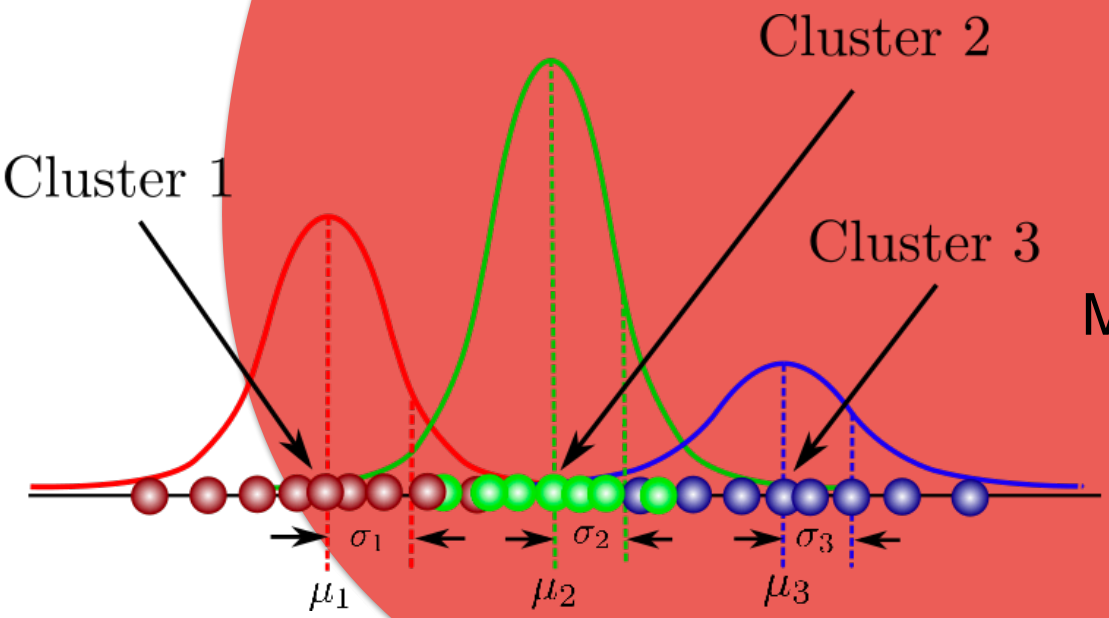
MODELING OF P(X) WITH GMMs

DATA



REPRESENTATION
(AUTOENCODER)

WE COMBINE THESE 2 STEPS?



MODELING OF P(X) WITH GMMs

Much of the recent progress in unsupervised deep learning has been to invent network architectures that are capable of solving either or both of these related problems directly, without resorting to any auxiliary methods

VAE

(VARIATIONAL AUTOENCODER)

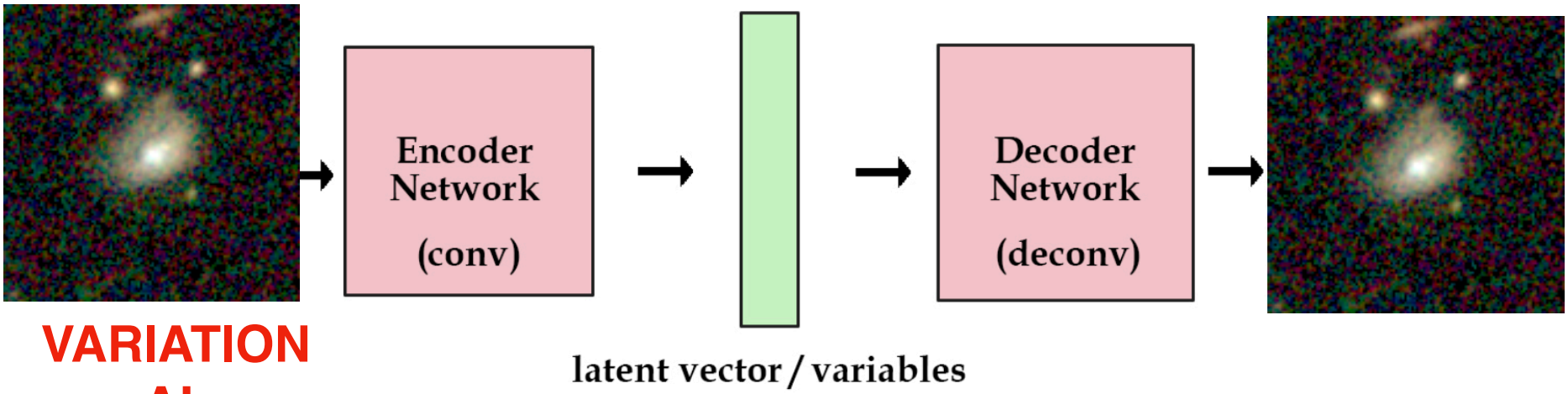
GAN

(GENERATIVE ADVERSARIAL NETOWRK)

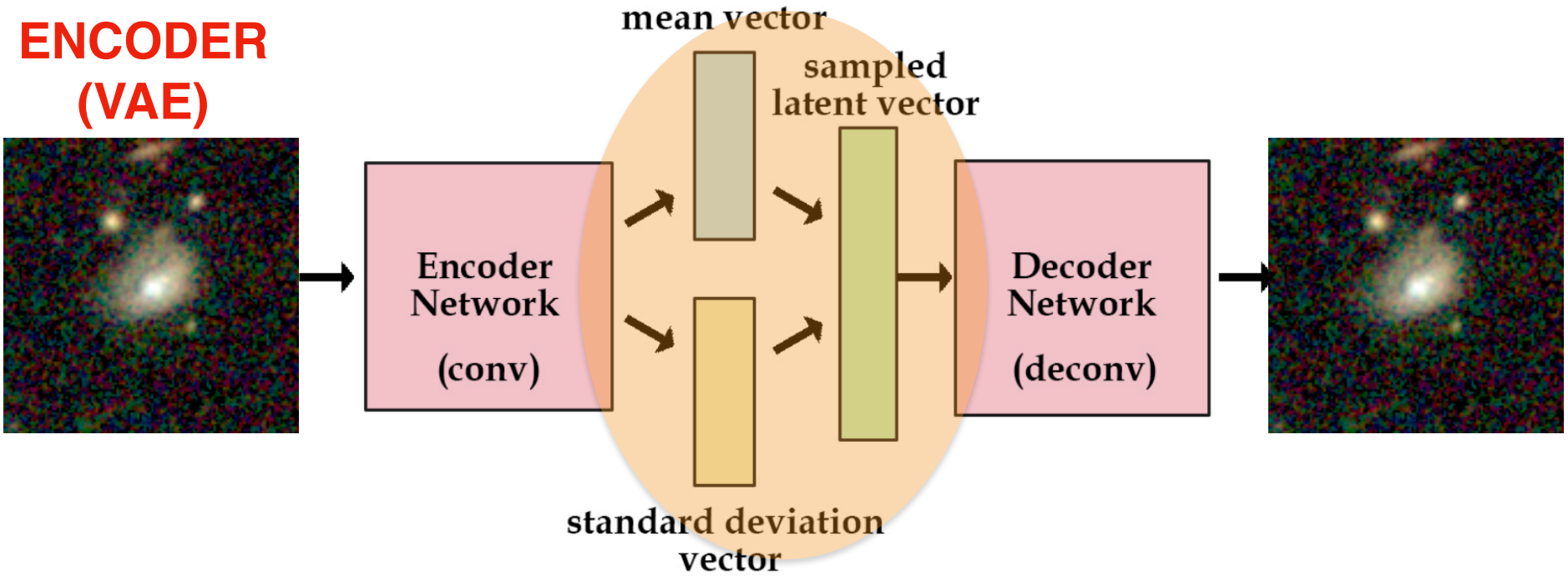
NF-ARF

(NORMALIZING FLOWS, AUTOREGRESSIVE FLOWS)

AUTO-ENCODER

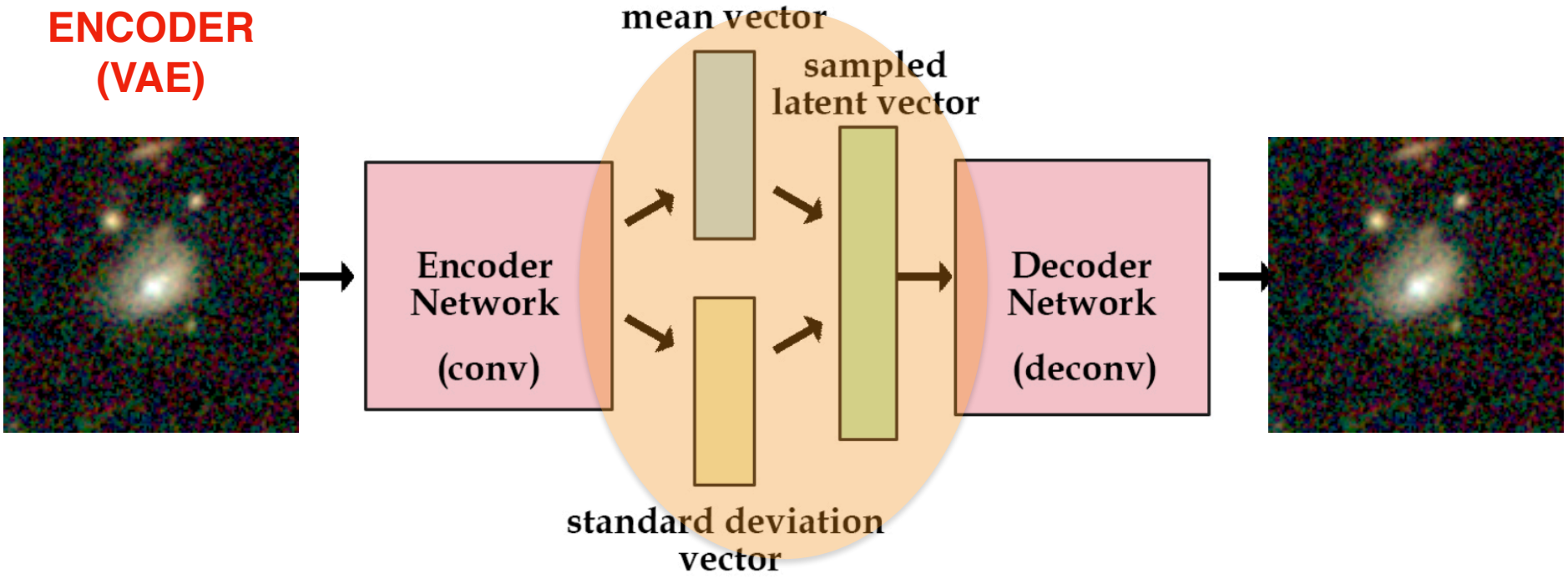


VARIATIONAL AUTO-ENCODER (VAE)



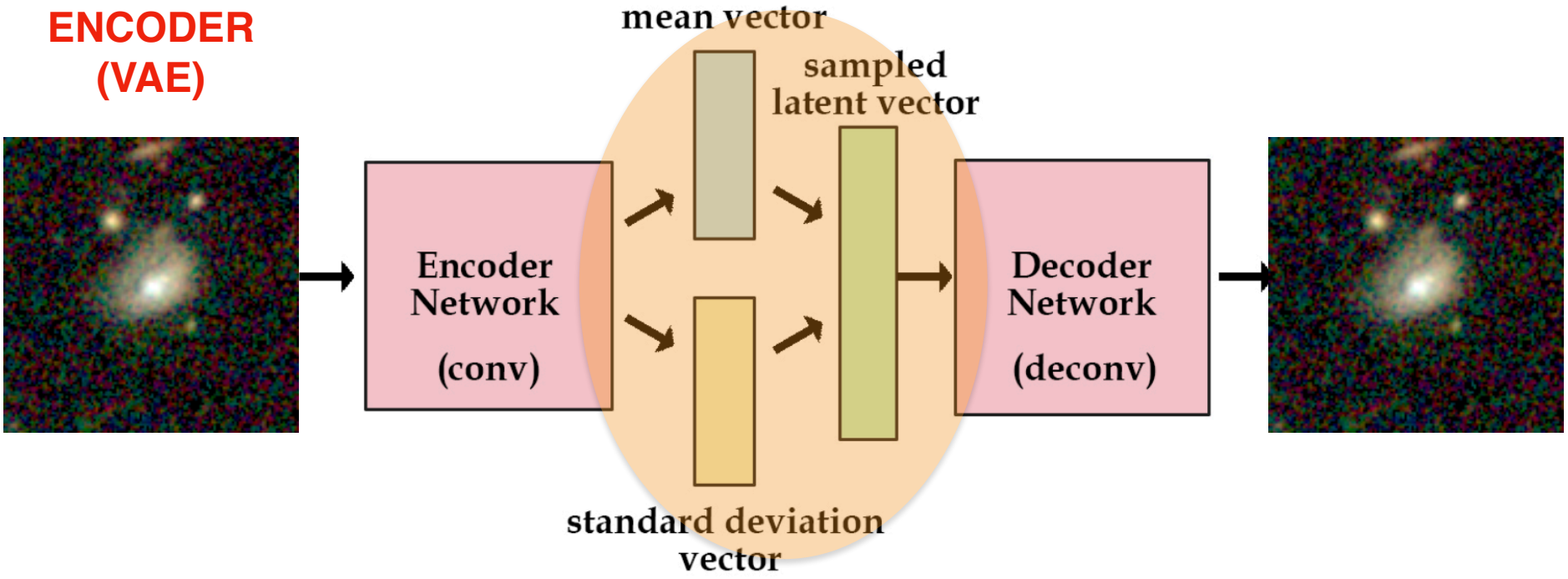
LET'S MODEL THE LATENT SPACE WITH A MIXUTRE OF GAUSSIANS

VARIATIONAL AUTO-ENCODER (VAE)



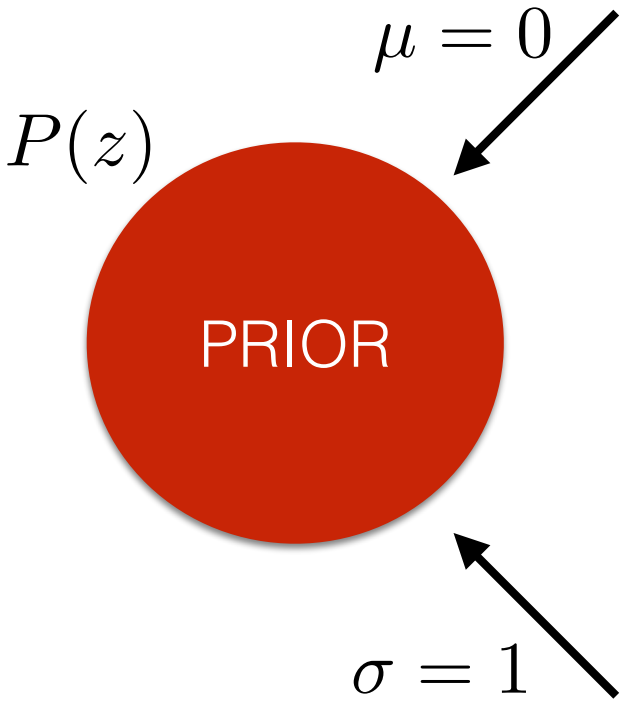
HOWEVER, NOTHING GUARANTEES US THAT THE LATENT SPACE CAN BE MODELLED BY A MIXTURE OF GAUSSIANS….

VARIATIONAL AUTO-ENCODER (VAE)

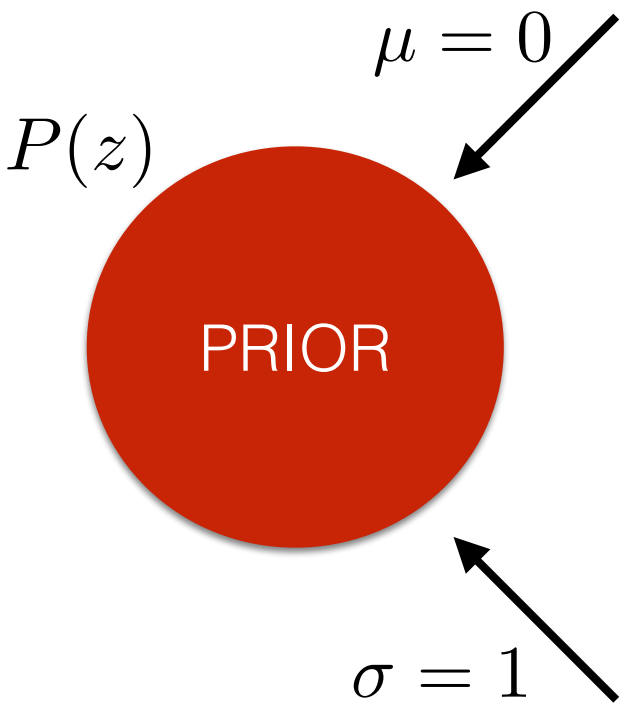


HOWEVER, NOTHING GUARANTEES US THAT THE LATENT SPACE CAN BE MODELLED BY A MIXTURE OF GAUSSIANS….

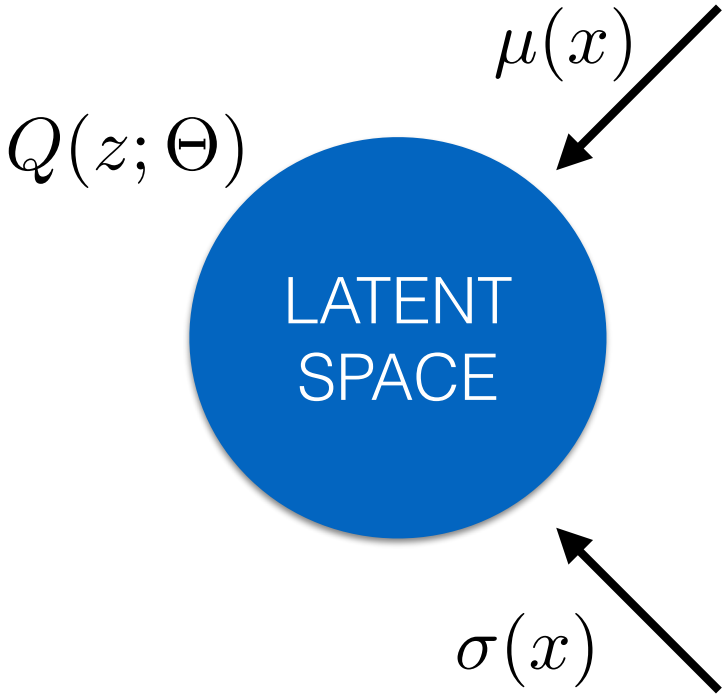
… LET'S FORCE IT TO BE GAUSSIAN LIKE!



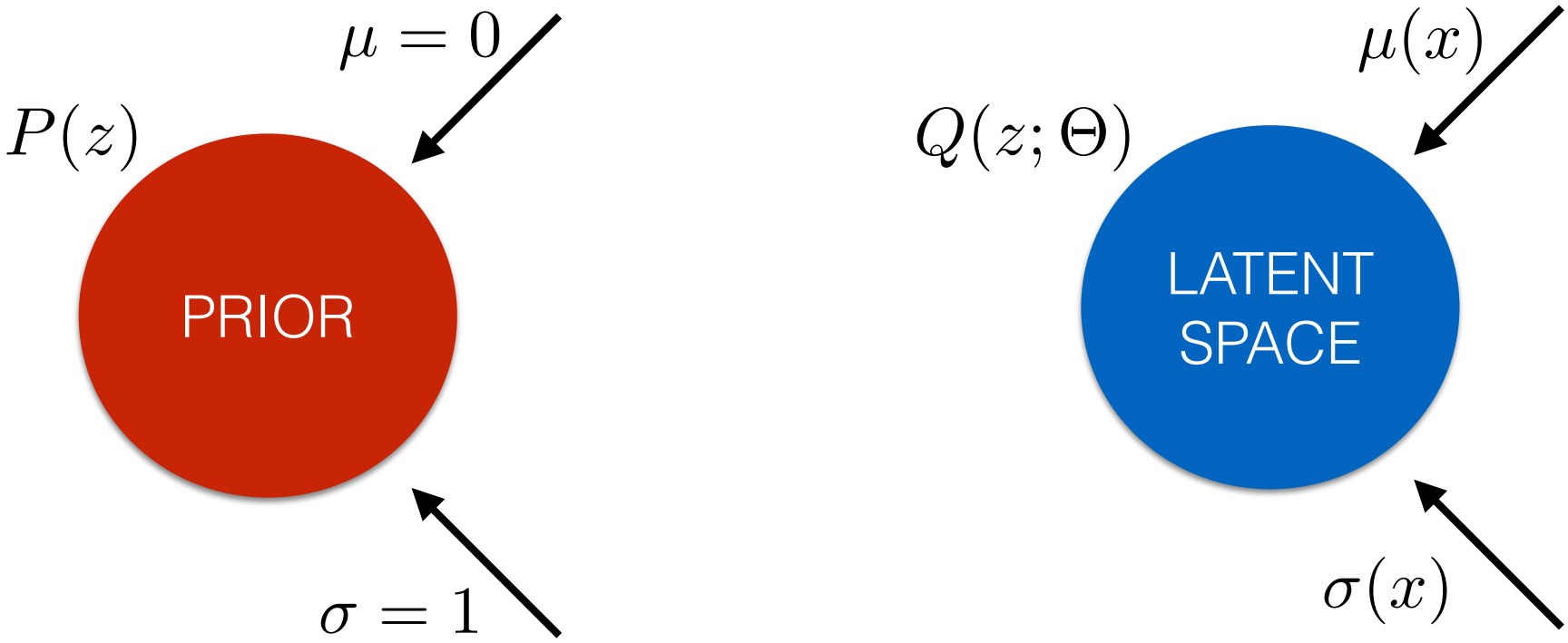
WE ASSUME A SIMPLE PRIOR



WE ASSUME A SIMPLE PRIOR

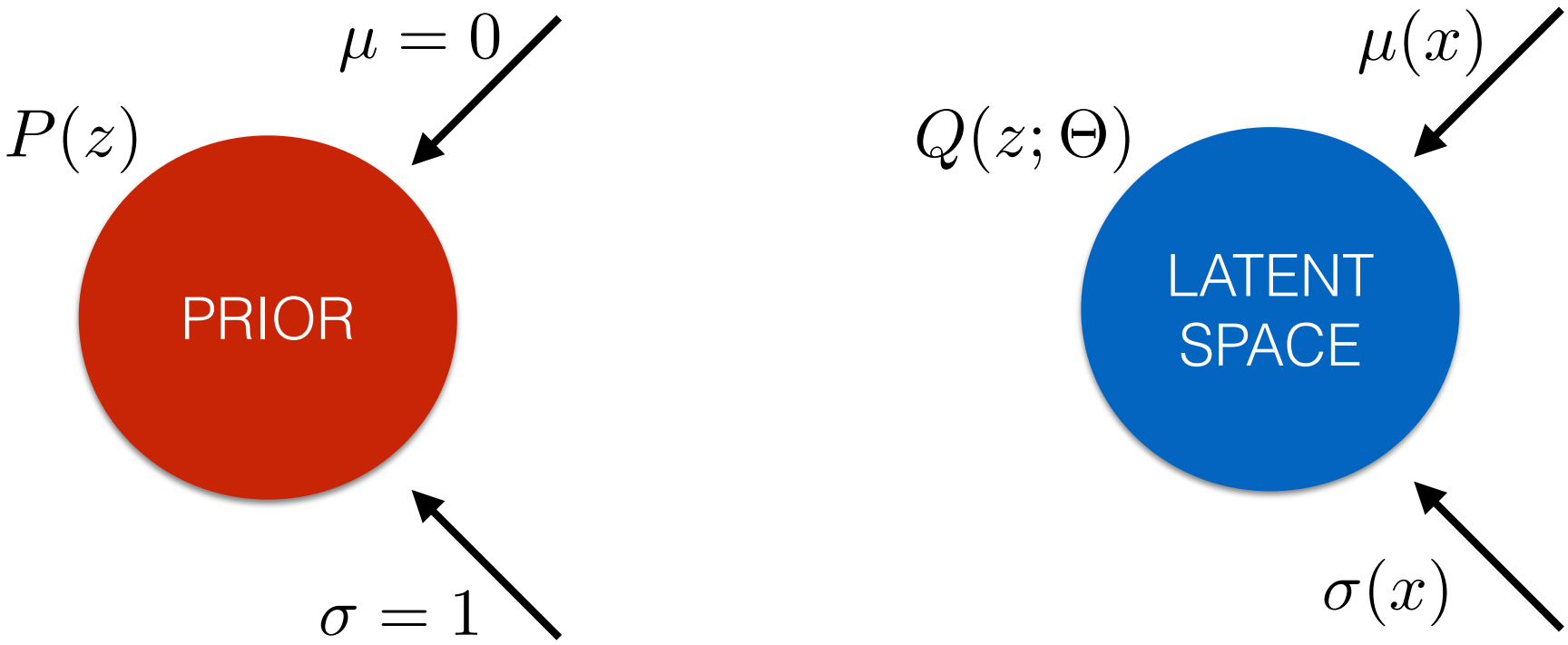


LATENT SPACE MODELING



WE ASSUME A SIMPLE PRIOR ⟺ LATENT SPACE MODELLING

WE WANT Q TO BE CLOSE TO THE PRIOR…



WE ASSUME A SIMPLE PRIOR ⟺ LATENT SPACE MODELLING

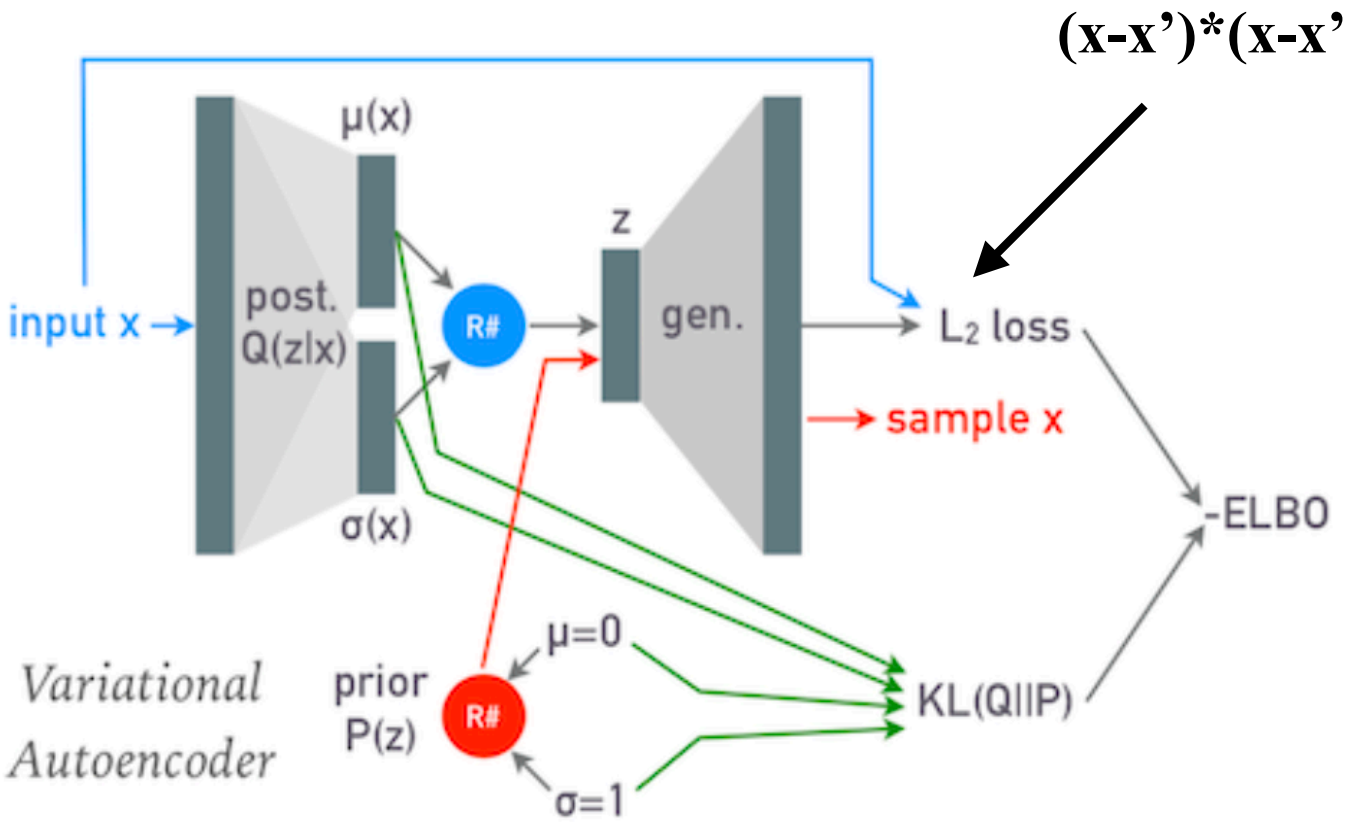
WE WANT Q TO BE CLOSE TO THE PRIOR…

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right).$$

$$D_{\mathrm{KL}}(P \parallel Q) = \int_{\mathcal{X}} \log\left(\frac{dP}{dQ}\right) \frac{dP}{dQ}\, dQ,$$

WE MINIMIZE THE K-L DIVERGENCE BETWEEN P AND Q

WHAT WOULD BE THEN THE LOSS FUNCTION OF A VAE?

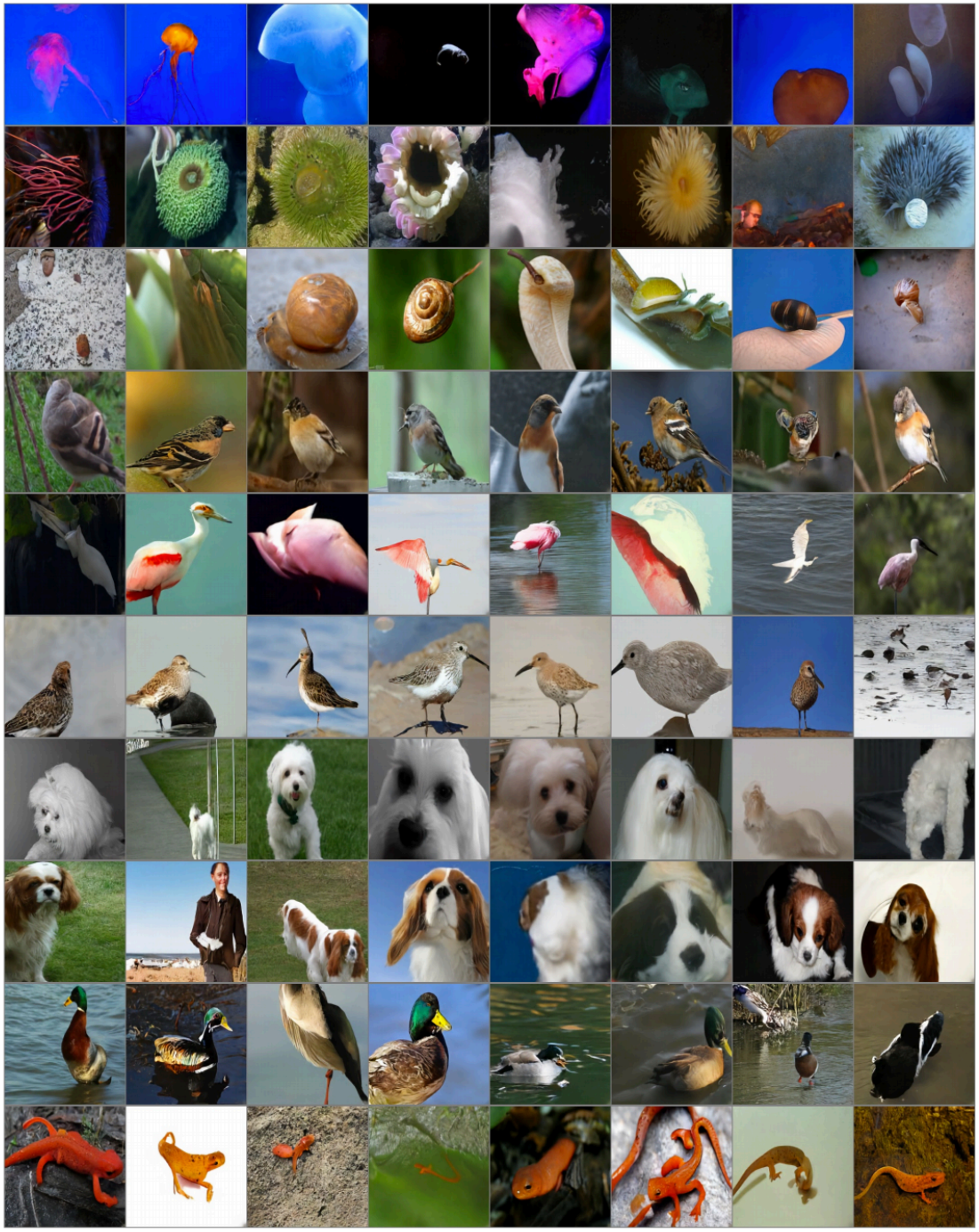


The key insight of VAE is that we are actually performing **variational inference** here, which then tells us what the loss function should be...

$$-\text{ELBO} = \langle \log P(\mathbf{x} \mid \mathbf{z}) \rangle_{z \sim Q} + \text{KL}\left(Q(\mathbf{z}; \Theta) \parallel P(\mathbf{z})\right) \ ,$$

L2 LOSS

REGULARIZATION TERM

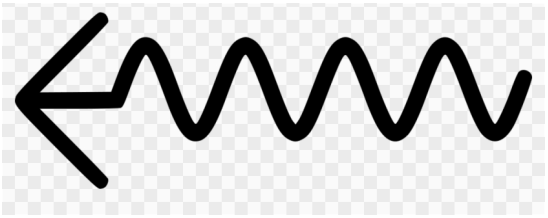


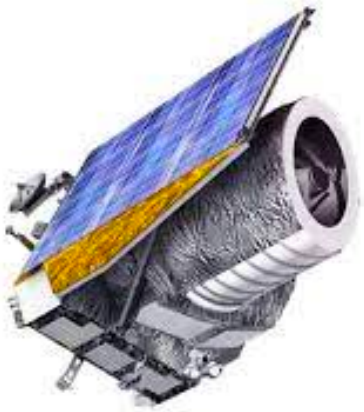
(VQ-VAE)

Razavi+19
(deepmind)

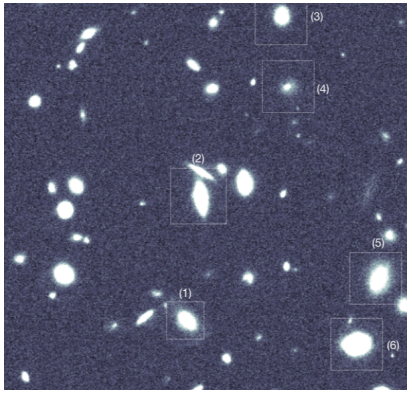
Preparing a space mission requires very detailed simulations

From photons

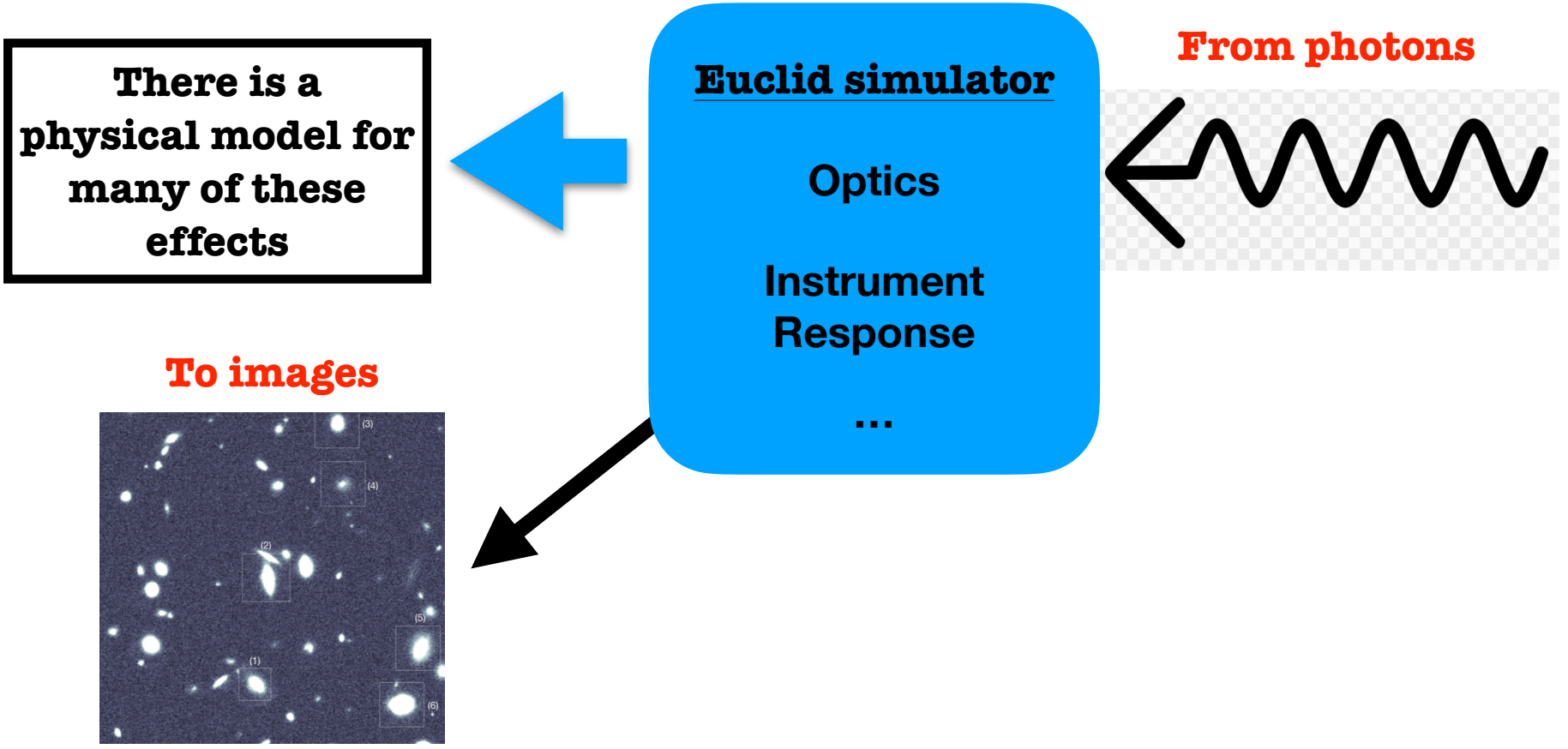




To images



Preparing a space mission requires very detailed simulations



Preparing a space mission requires very detailed simulations

There is a physical model for many of these effects

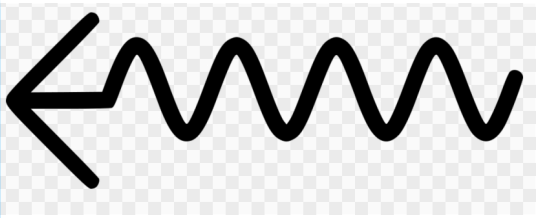
Euclid simulator

Optics

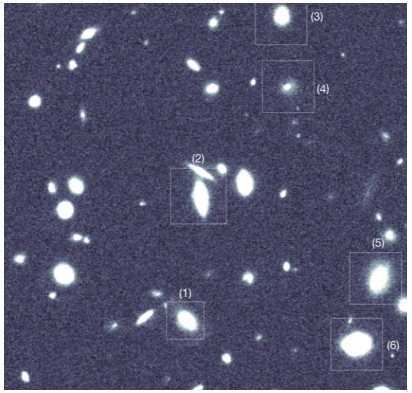
Instrument Response

...

From photons



To images



Others are more difficult to model

Preparing a space mission requires very detailed simulations

There is a physical model for many of these effects

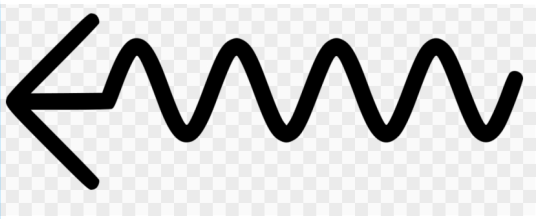
Euclid simulator

Optics

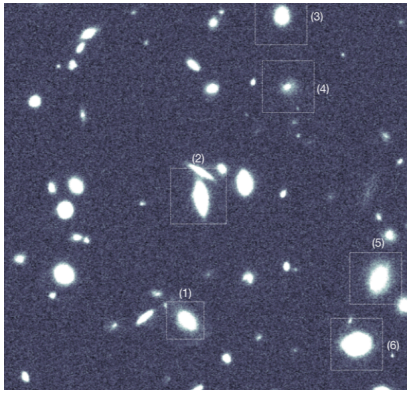
Instrument Response

...

From photons



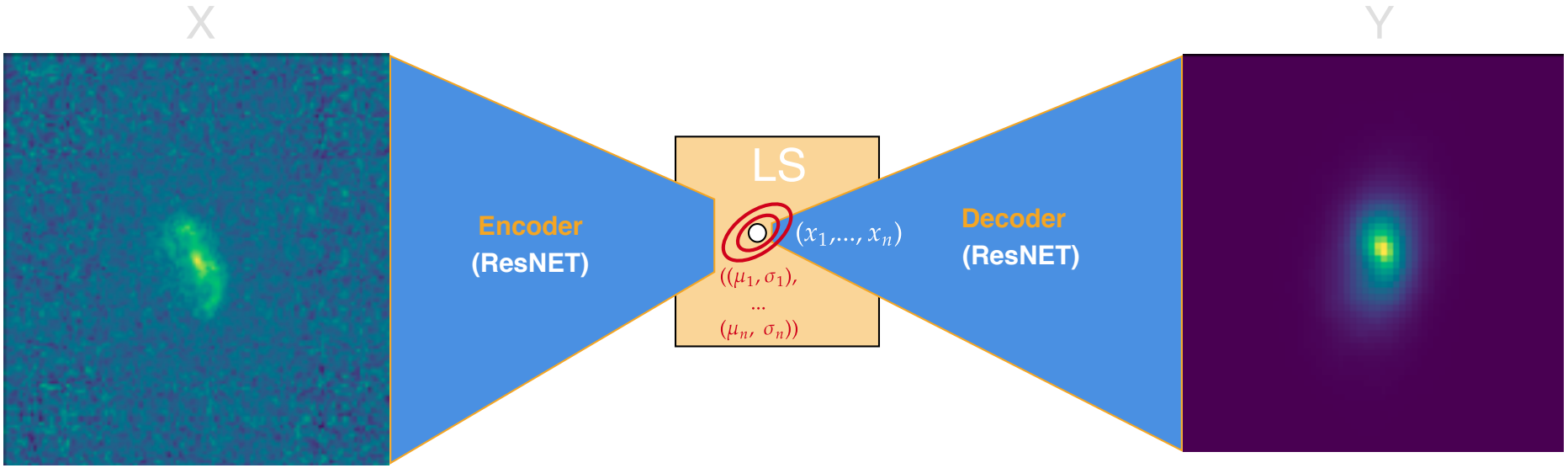
To images



Neural Network

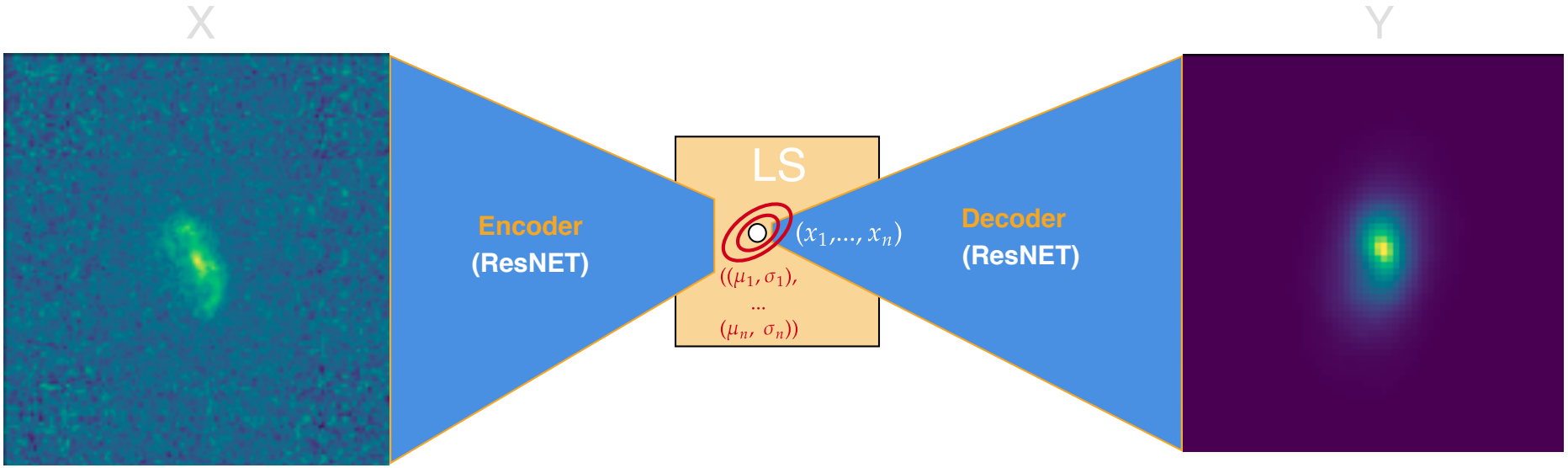
Others are more difficult to model

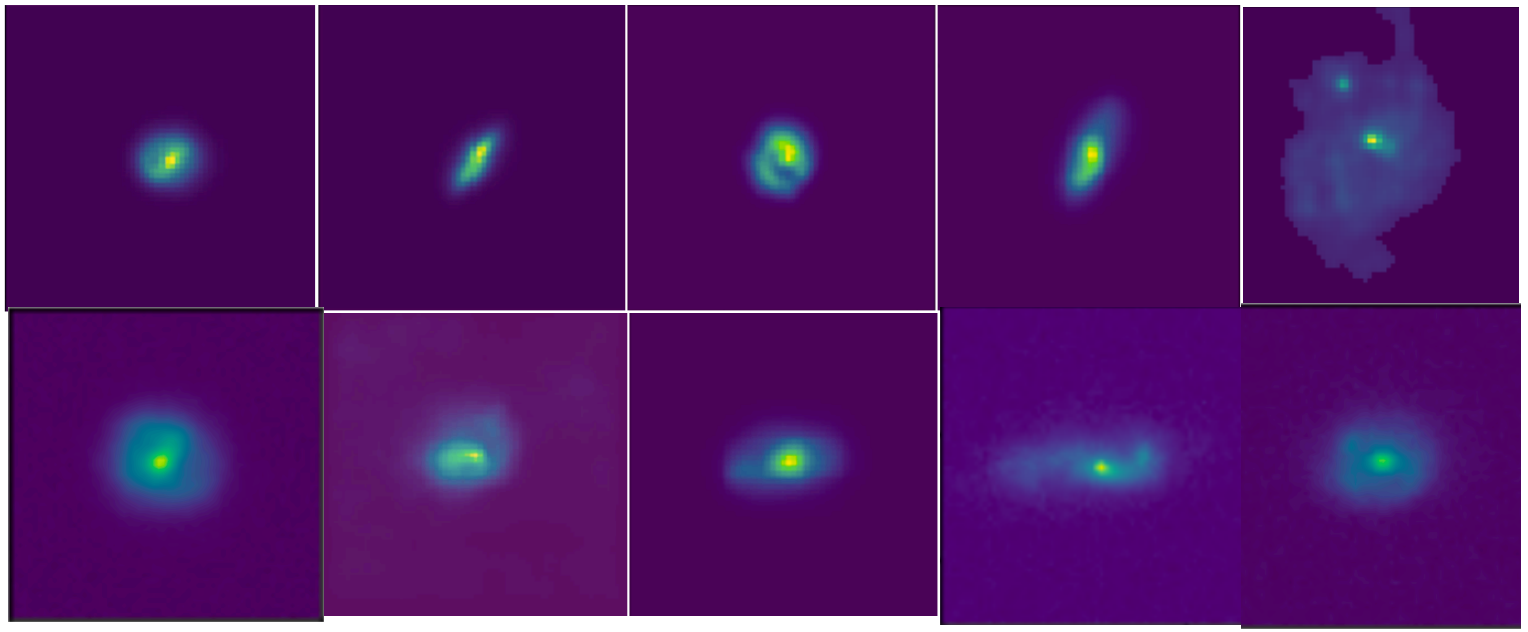
Include a Generative Model for galaxy generation in the Euclid Simulation Pipeline to model process for which we do not have a physical model



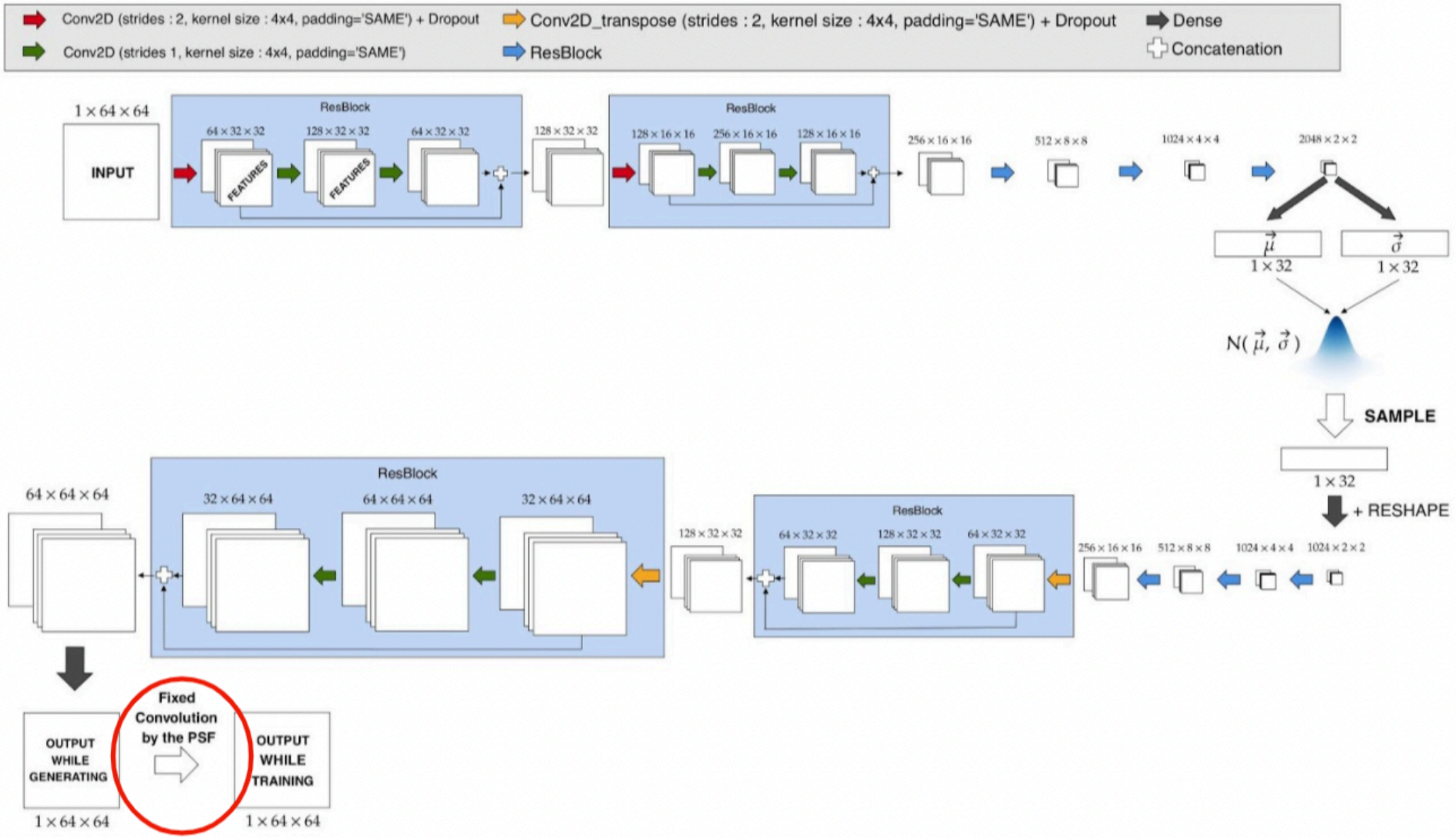
Bretonnière+21

Include a Generative Model for galaxy generation in the Euclid Simulation Pipeline to model process for which we do not have a physical model





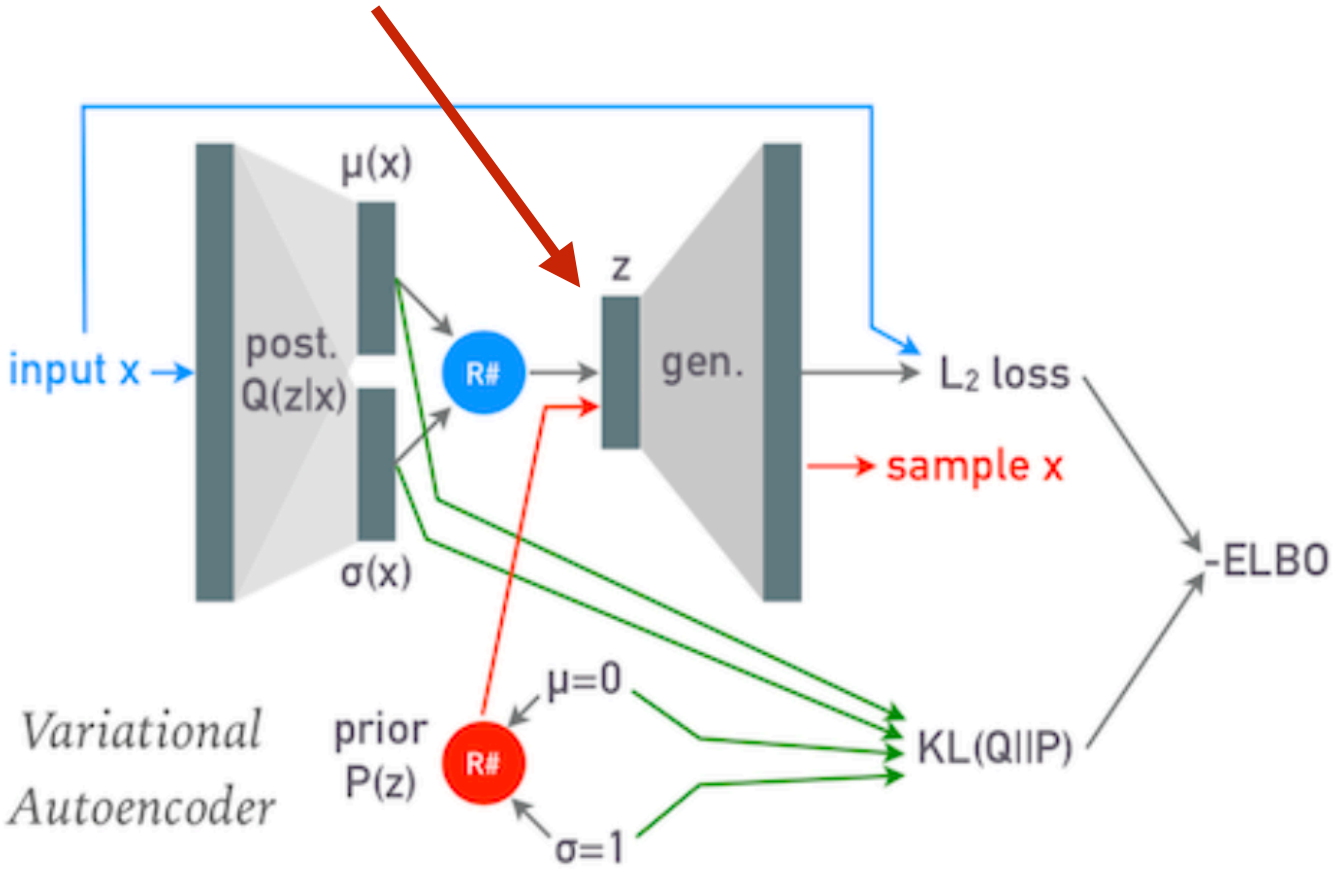
Bretonnière+21

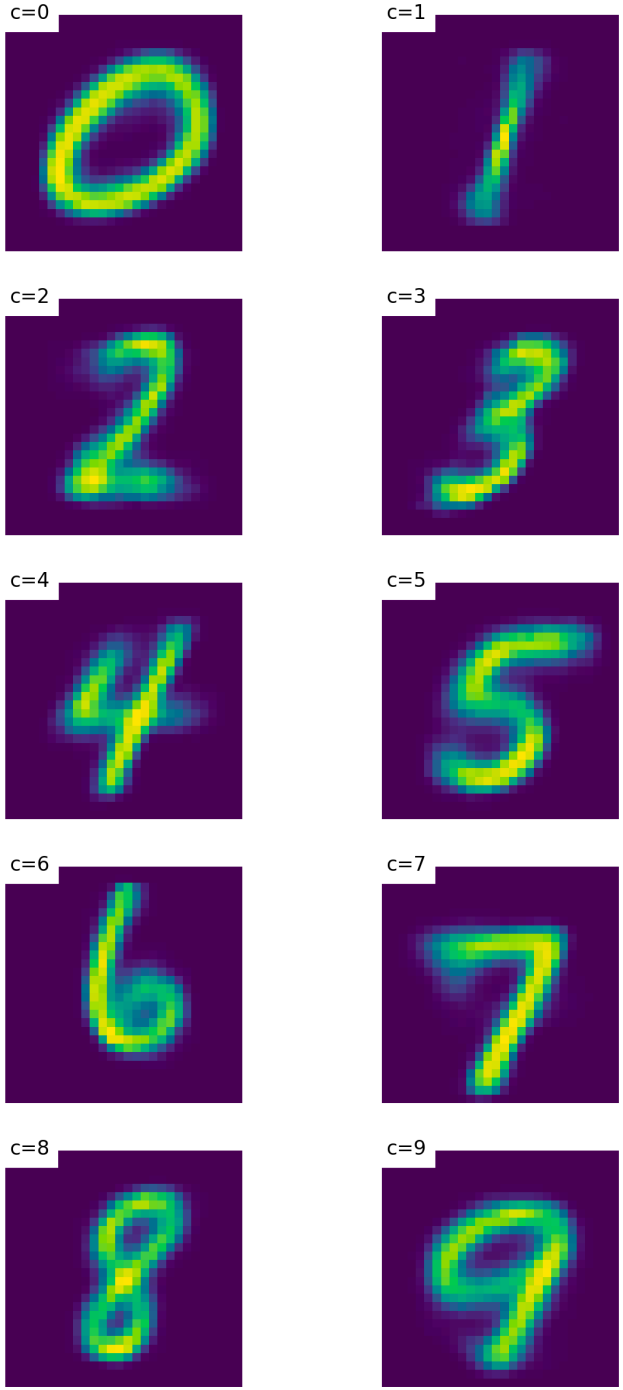


Physical information is included as well within the architecture

VAE's can also be conditioned, to generate a given class

concatenate labels



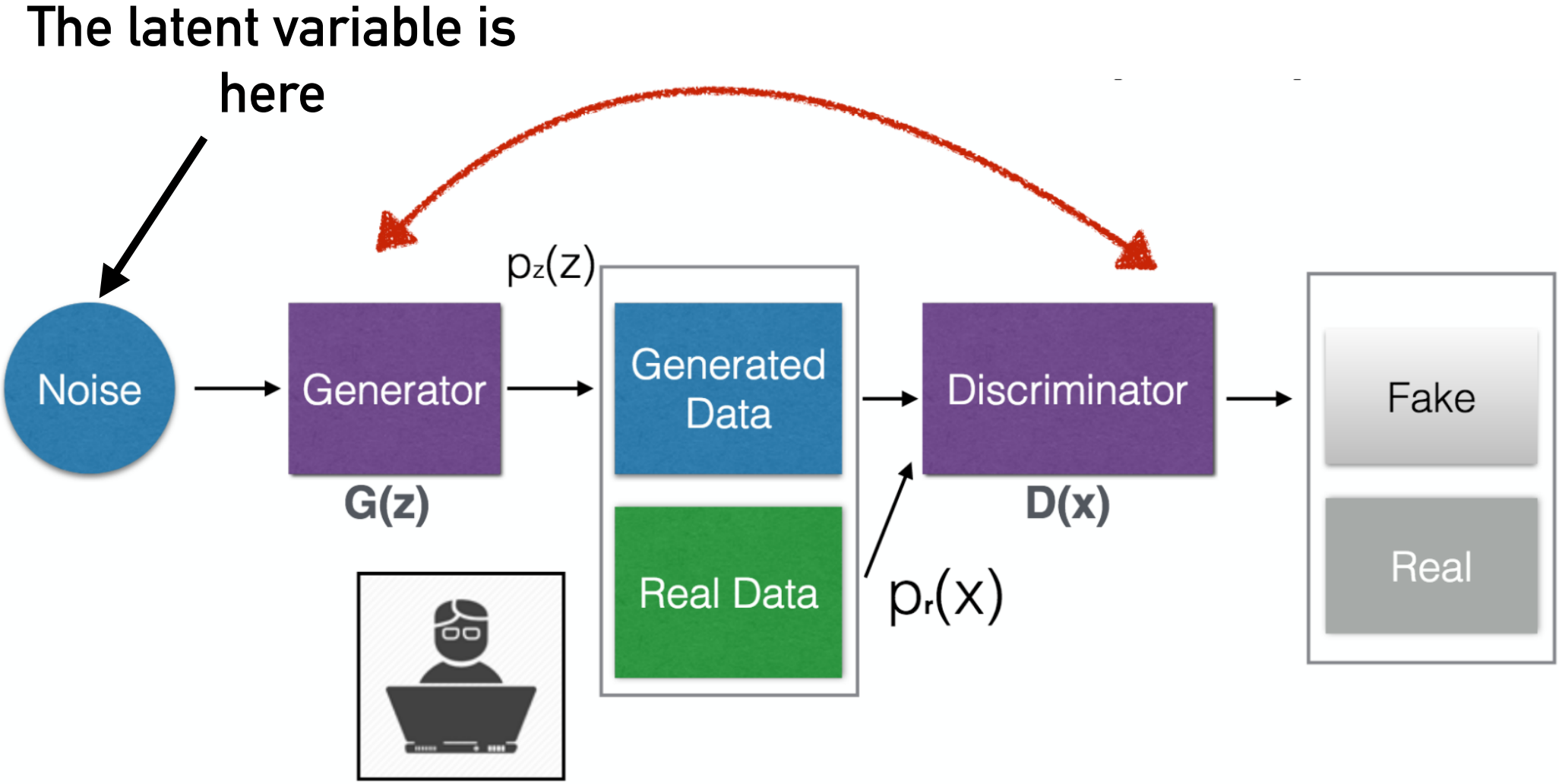

c=0
c=1
c=2
c=3
c=4
c=5
c=6
c=7
c=8
c=9

As for VAEs,
the goal of generative Adversarial Networks (GANs) is to estimate $p(z|x)$

They convert the problem into a “supervised approach” by using two competing neural networks

GENERATIVE ADVERSARIAL NETWORKS

(Goodfellow+14)

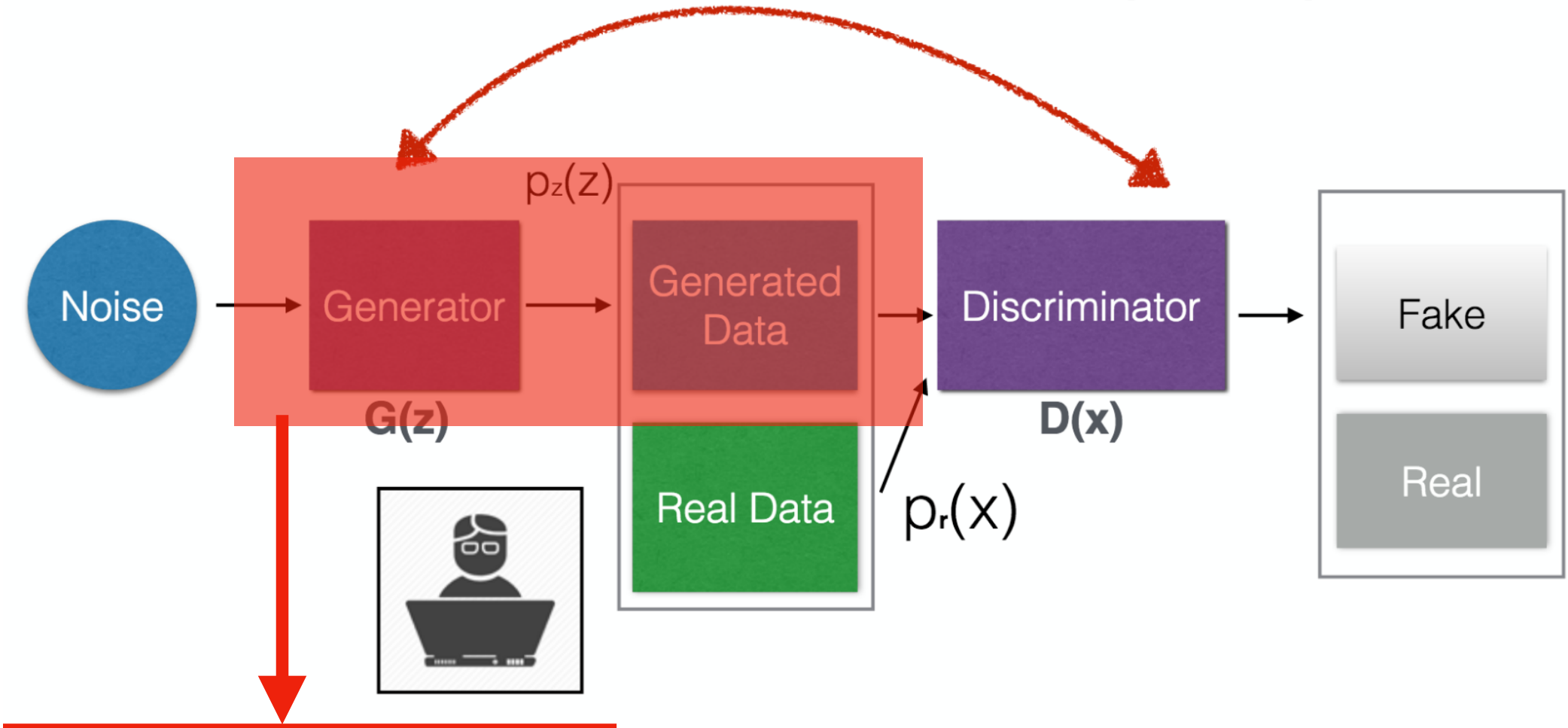


TWO COMPETING NETWORKS

GENERATIVE ADVERSARIAL NETWORKS

(Goodfellow+)

TWO COMPETING NETWORKS

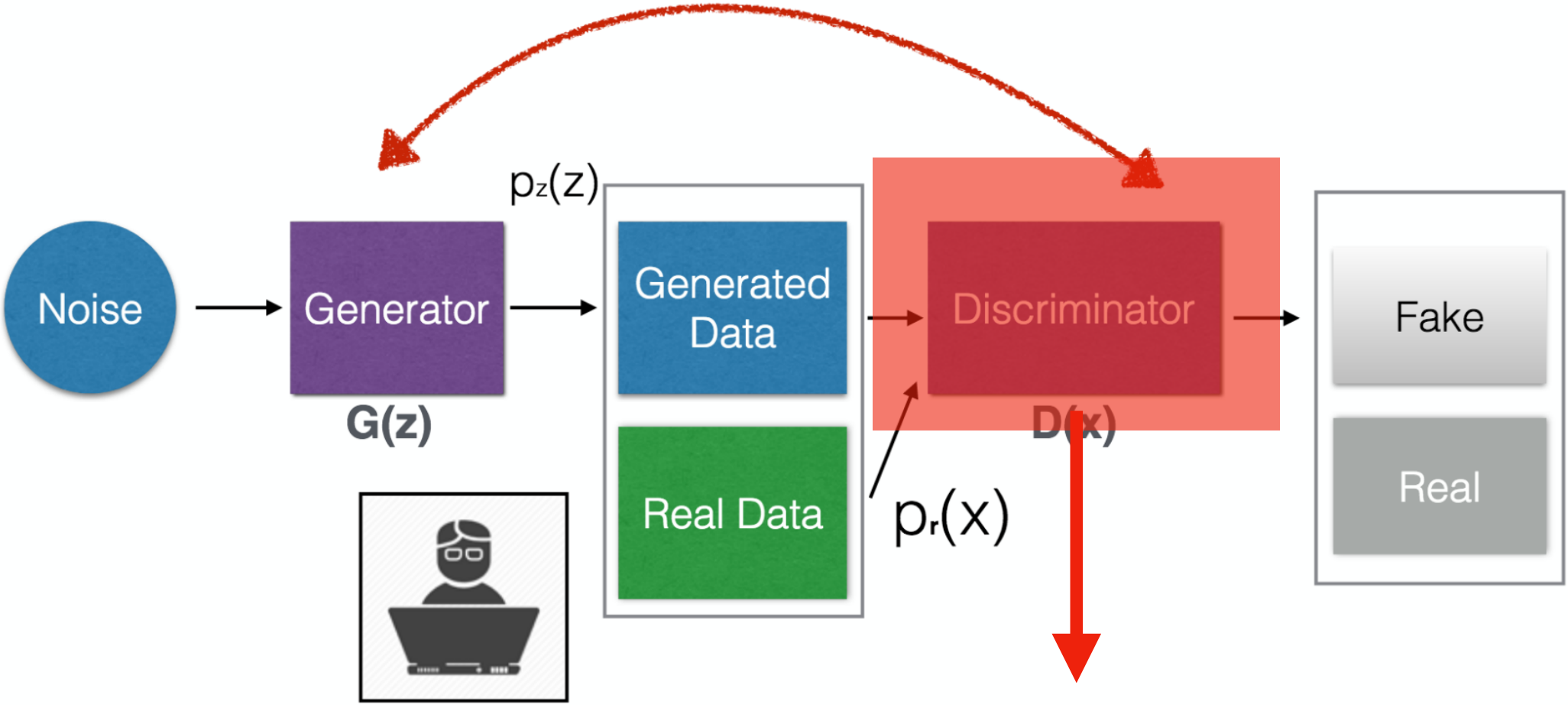


Every N iterations the generator is trained to force the discriminator to classify as real

GENERATIVE ADVERSARIAL NETWORKS

(Goodfellow+)

TWO COMPETING NETWORKS

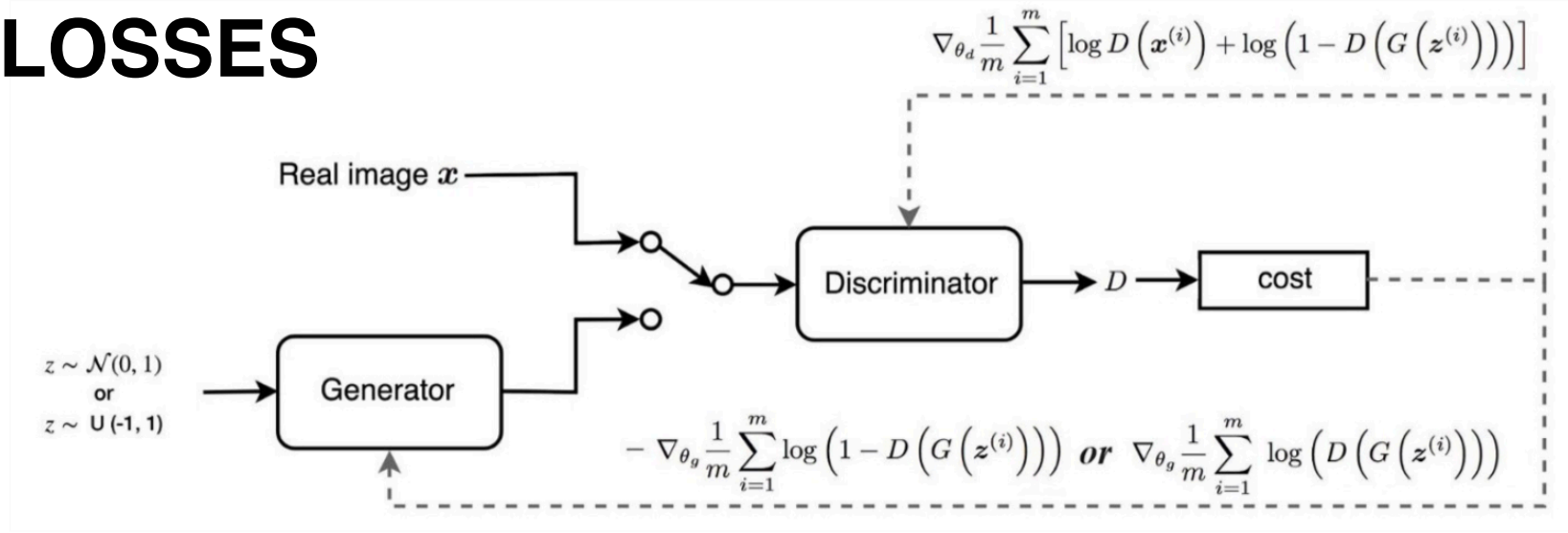


Every N iterations the discriminator is trained to force to distinguish between real and fake

IN PRACTICE

GAN LOSSES

DISCRIMINATOR LOSS (CROSS-ENTROPY)



GENERATOR LOSS

GANs CAN ACHIEVE IMPRESSIVE RESULTS…

Karras+19

THEY ARE ALSO VERY HARD TO TRAIN

1. IT IS HARD TO REACH EQUILIBRIUM AND IT IS ACTUALLY NOT GUARANTEED…

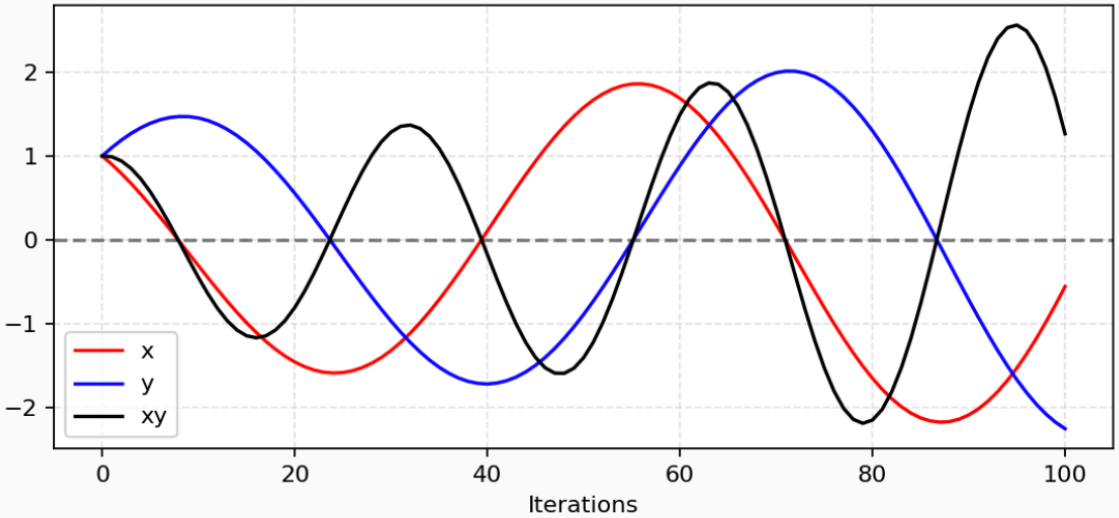


Fig. 3. A simulation of our example for updating x to minimize xy and updating y to minimize $-xy$. The learning rate $\eta = 0.1$. With more iterations, the oscillation grows more and more unstable.

https://lilianweng.github.io/lil-log/2017/08/20/from-GAN-to-WGAN.html

SALIMANS+16

THEY ARE ALSO VERY HARD TO TRAIN

2. LOW DIMENSIONAL SUPPORTS

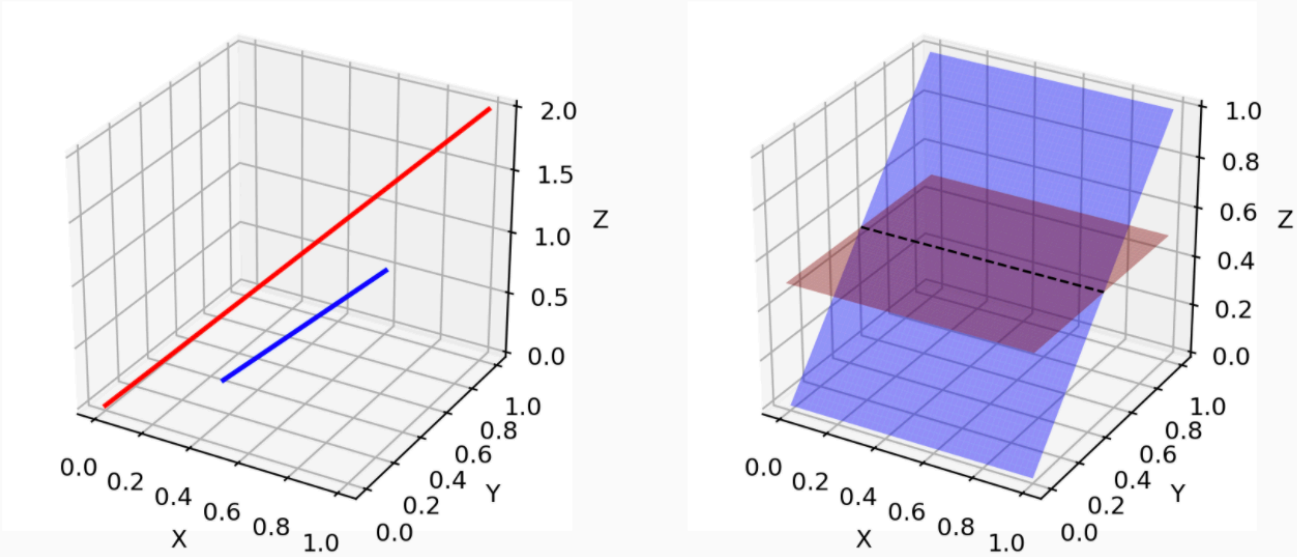


Fig. 4. Low dimensional manifolds in high dimension space can hardly have overlaps. (Left) Two lines in a three-dimension space. (Right) Two surfaces in a three-dimension space.

https://lilianweng.github.io/lil-log/2017/08/20/from-GAN-to-WGAN.html

ARJOVSKI+17

THEY ARE ALSO VERY HARD TO TRAIN

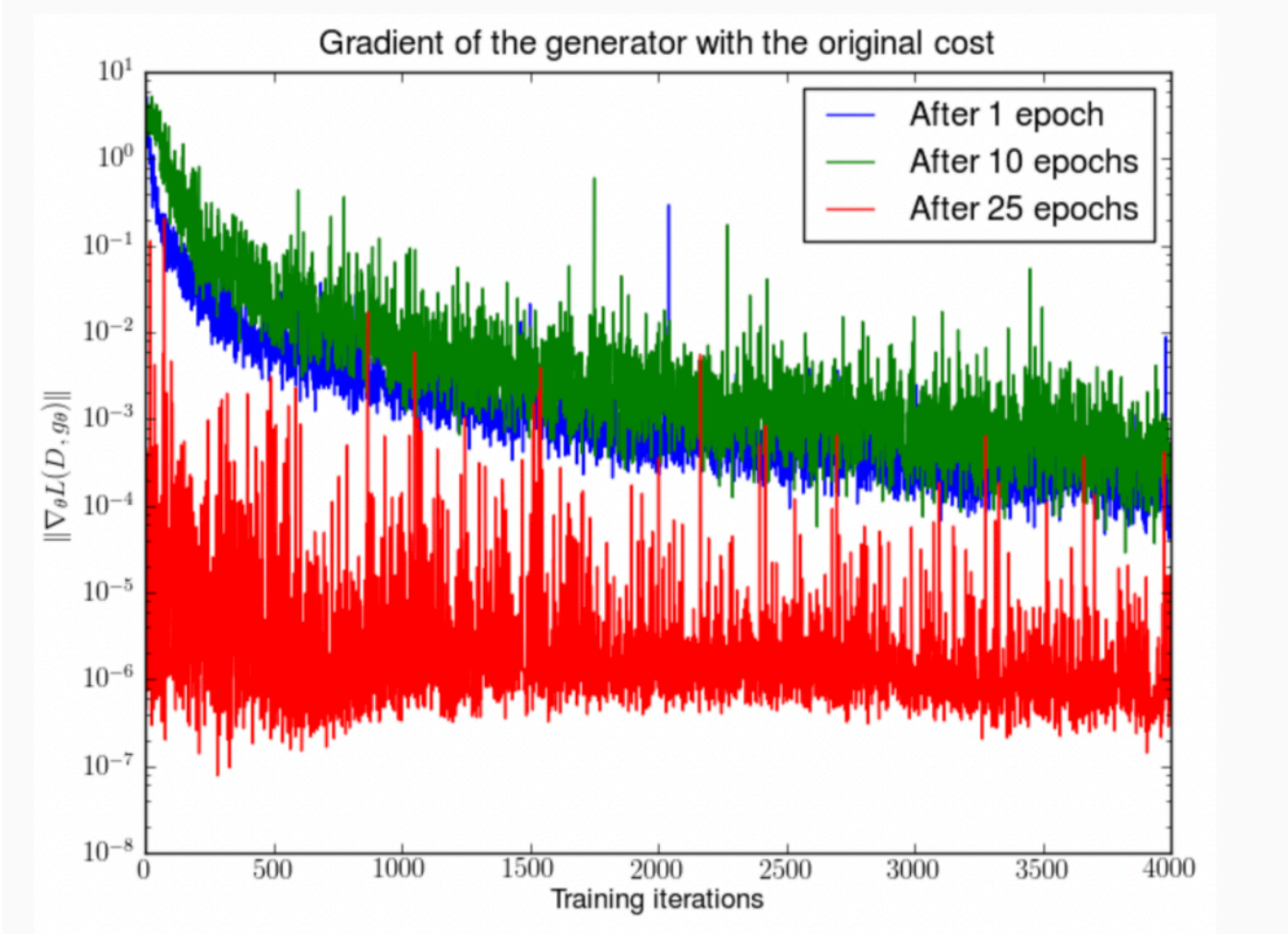
3. MODE COLLAPSE

https://lilianweng.github.io/lil-log/2017/08/20/from-GAN-to-WGAN.html

razavi+19

THEY ARE ALSO VERY HARD TO TRAIN

4. VANISHING GRADIENT

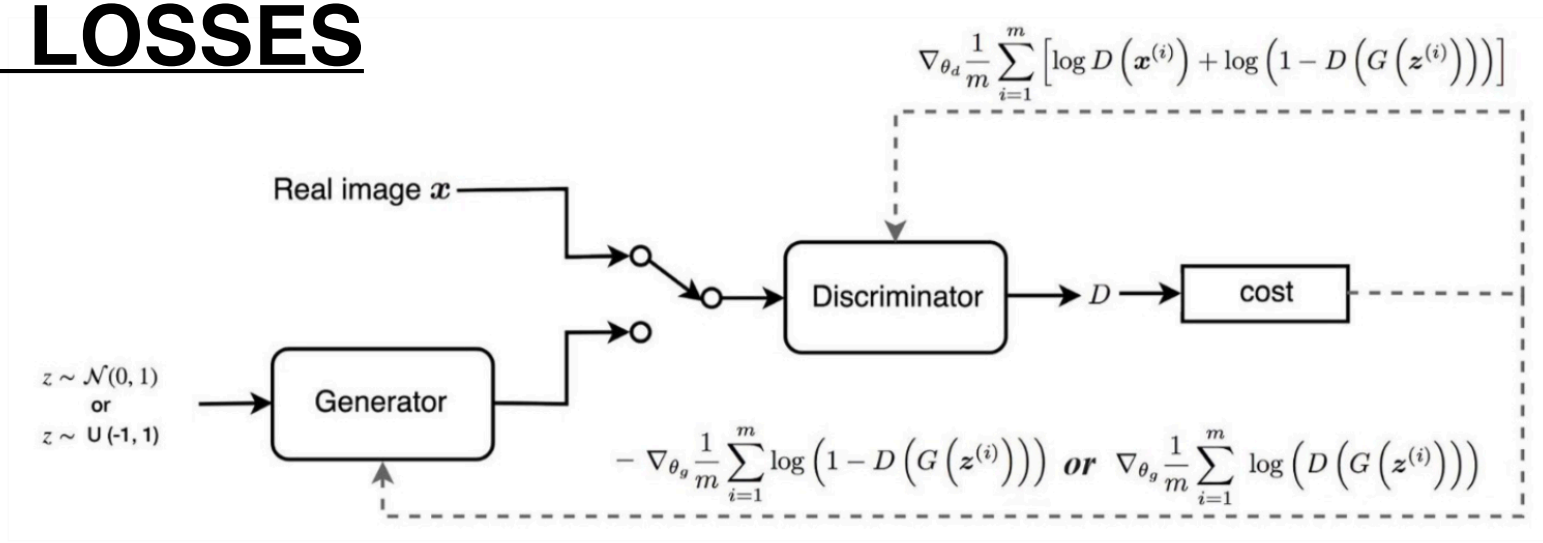


When the discriminator becomes perfect, there are no more gradients to update

IN PRACTICE

GAN LOSSES

DISCRIMINATOR LOSS (CROSS-ENTROPY)



GENERATOR LOSS

WASSERSTEIN GAN LOSSES

Cross entropy is replaced by Wasserstein distance