Generative modeling

Prototyping with Deep Learning

Learning outcomes

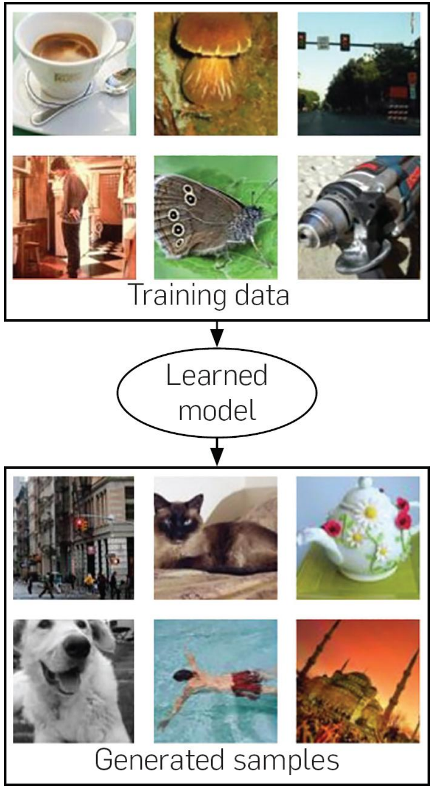
After this lesson you will be able to:

- Identify the need for generative models
- Recognize popular model architectures
- Understand the challenges of generative models

Goal

Learn to generate *new data* from examples

How awesome does it sound?



Recap: Learning paradigms

Supervised learning

f($\boldsymbol{x}$) = y

Challenge: get high-quality labels

Unsupervised learning

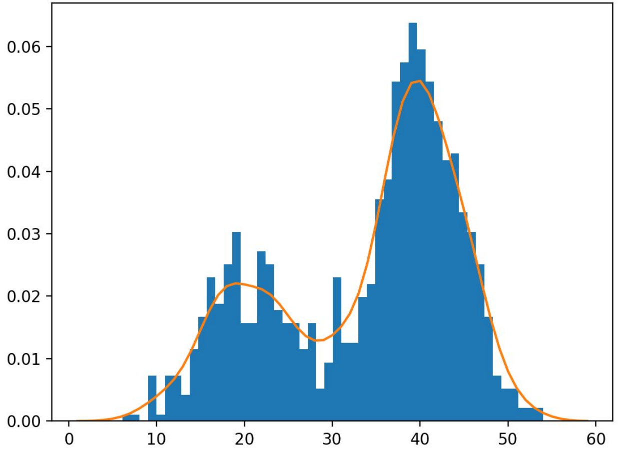
f($\boldsymbol{x}$) = $\boldsymbol{x}'$

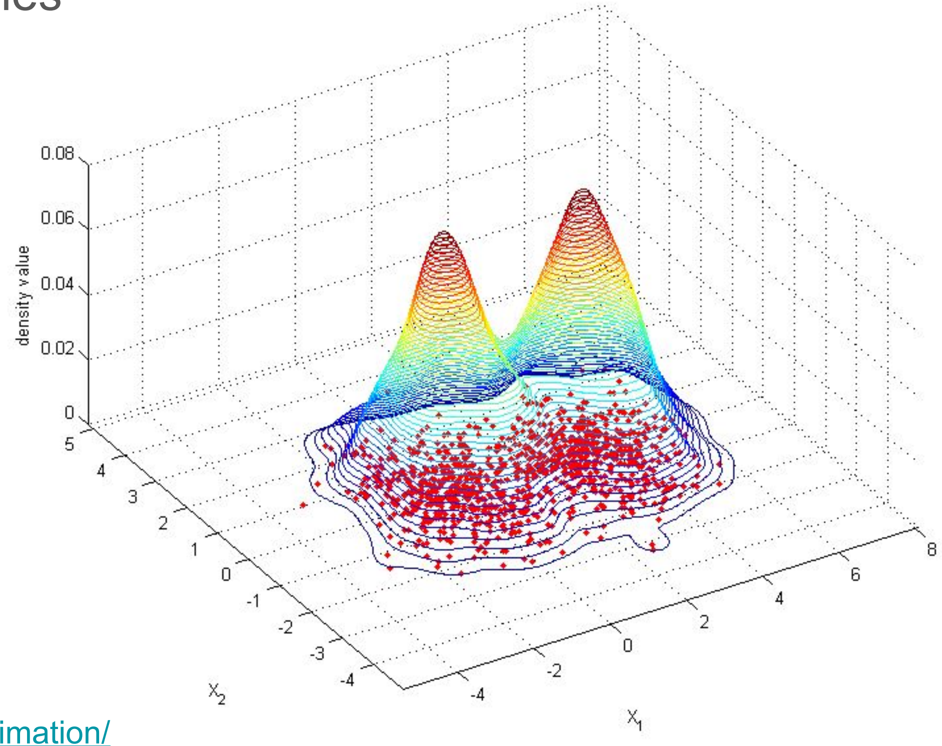
Challenge: understand the structure of the data

Probability density estimation

UNIVERSITÉ DU LUXEMBOURG

Parametric vs non-parametric approaches

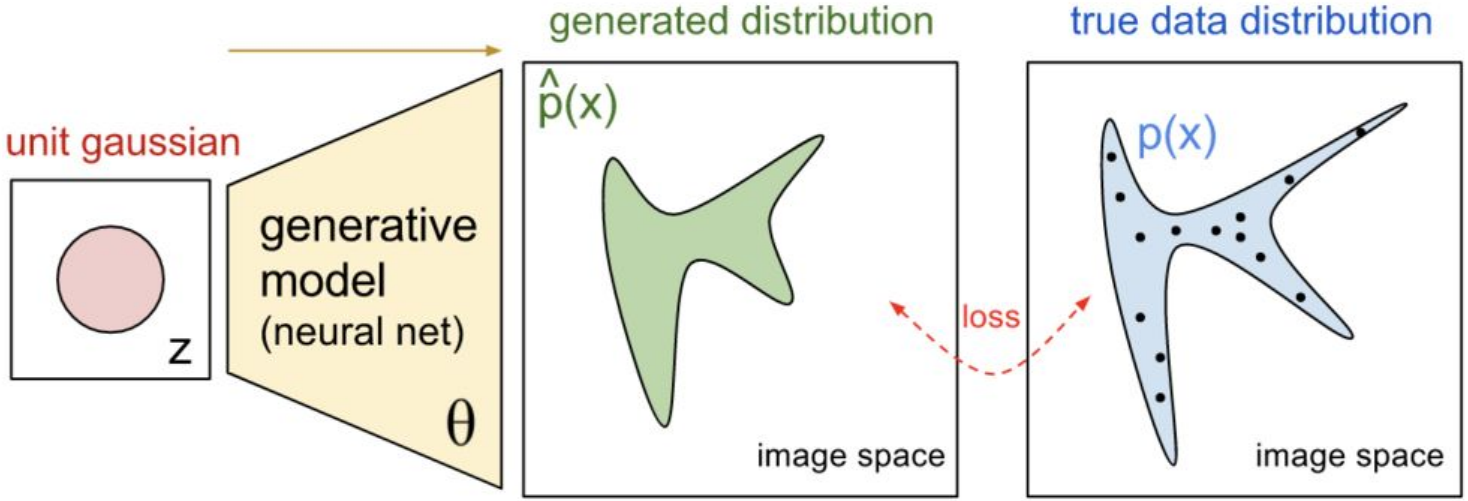




https://machinelearningmastery.com/probability-density-estimation/

Framework

Learn $p_{model}(\boldsymbol{x}) \approx p_{data}(\boldsymbol{x})$, however $p_{data}(\boldsymbol{x})$ is unknown!



https://openai.com/blog/generative-models/

Examples

https://openai.com/blog/generative-models/

How to estimate density?

Explicitly

- Tractable density: Autoregressive, flow-based, and diffusion models
- Intractable density: VAEs

Implicitly

- GANs

More approaches also available but they are not that much popular; see e.g. Fig.9 in https://arxiv.org/pdf/1701.00160.pdf

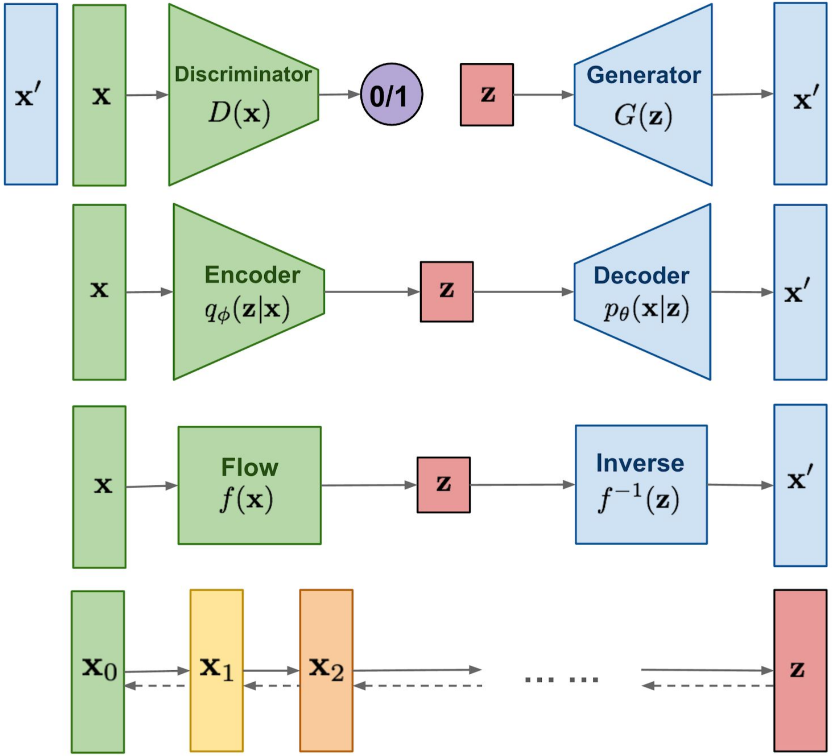
Summary of approaches

GAN: Adversarial training

VAE: maximize variational lower bound

Flow-based models: Invertible transform of distributions

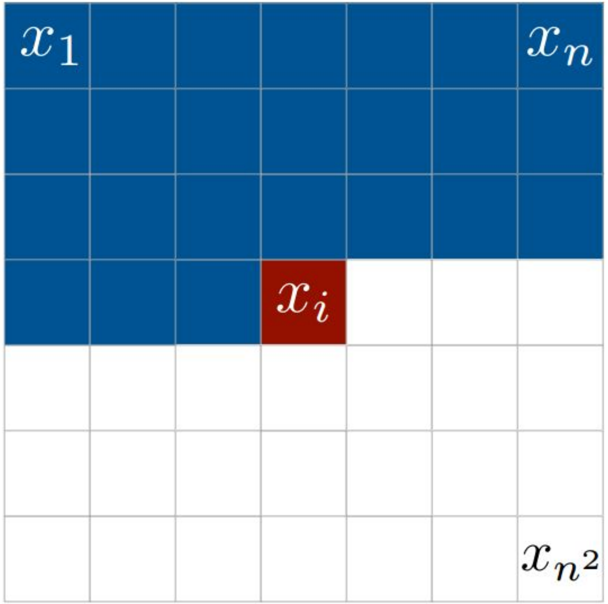
Diffusion models: Gradually add Gaussian noise and then reverse



https://lilianweng.github.io/lil-log/2021/07/11/diffusion-models.html

Autoregressive models

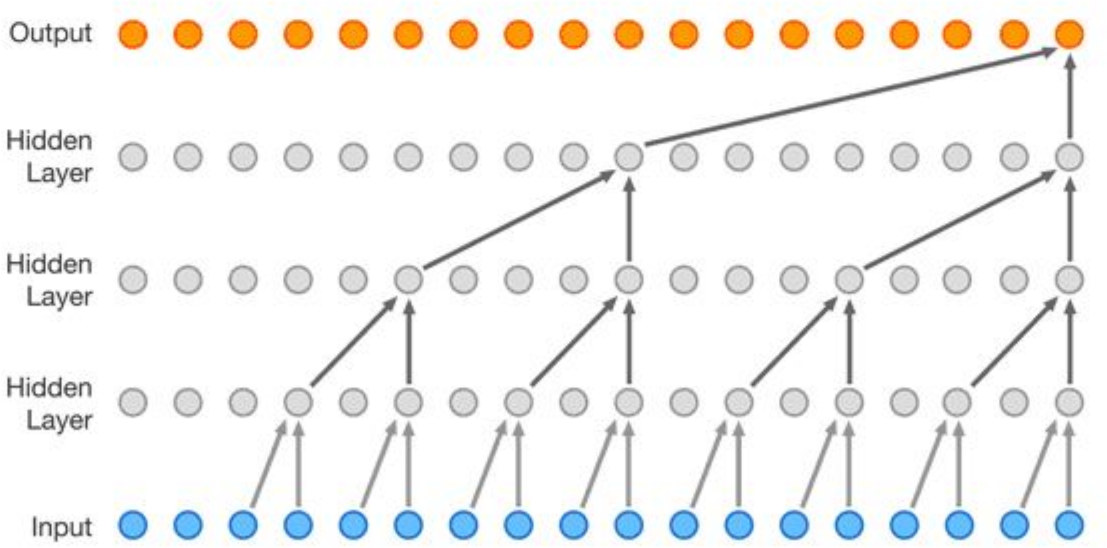
$$p_\theta(x) = \prod_{i=1}^{n} p_\theta(x_i | x_1, \ldots, x_{i-1})$$



https://towardsdatascience.com/32d192911173

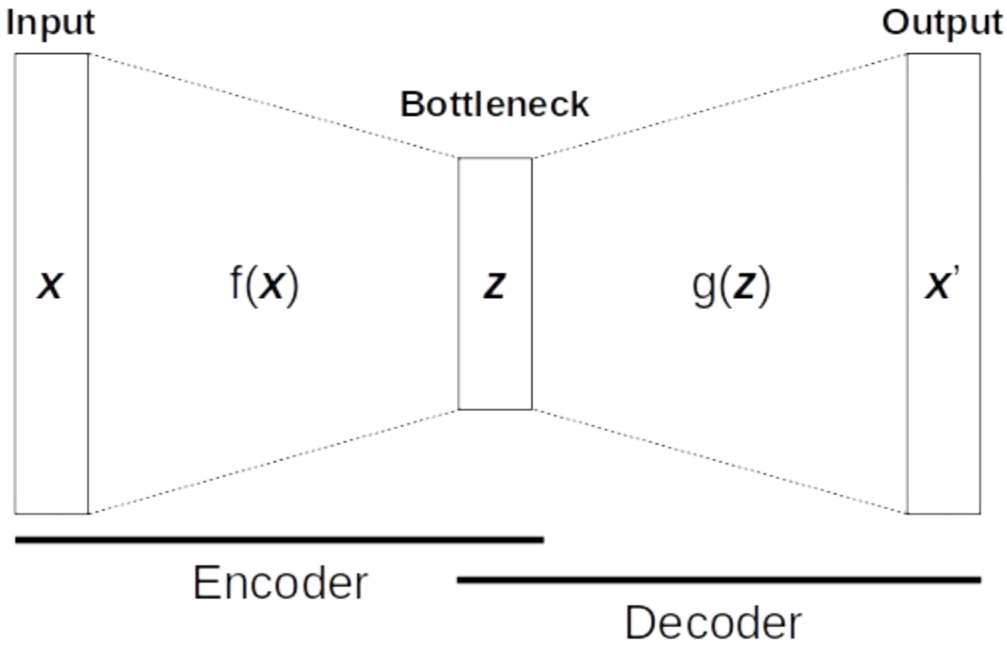
Examples: PixelRNN and PixelCNN

Autoregressive models



https://deepmind.com/blog/article/wavenet-generative-model-raw-audio

Recap: Autoencoders



Variational inference

Intractable likelihood: $p_\theta(x) = \int p_\theta(z) p_\theta(x|z) dz$

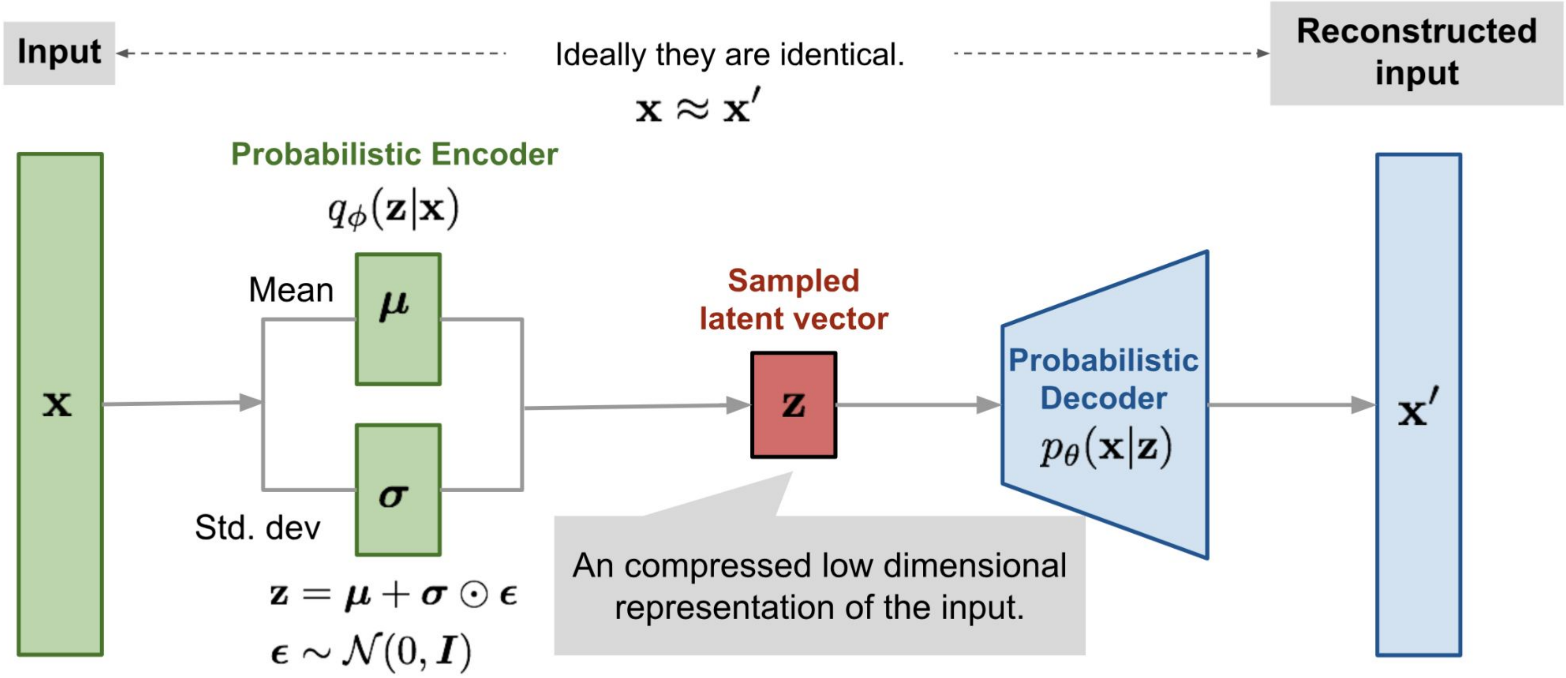
Posterior density also intractable: $p_\theta(z|x) = \frac{p_\theta(x|z) p_\theta(z)}{p_\theta(x)}$

Solutions: (1) *probabilistic* encoder and decoder: $q_\phi(z|x) \approx p_\theta(x|z)$

(2) ELBO criterion: $\log p(x) \geq \mathcal{L}(x; \theta, \phi)$

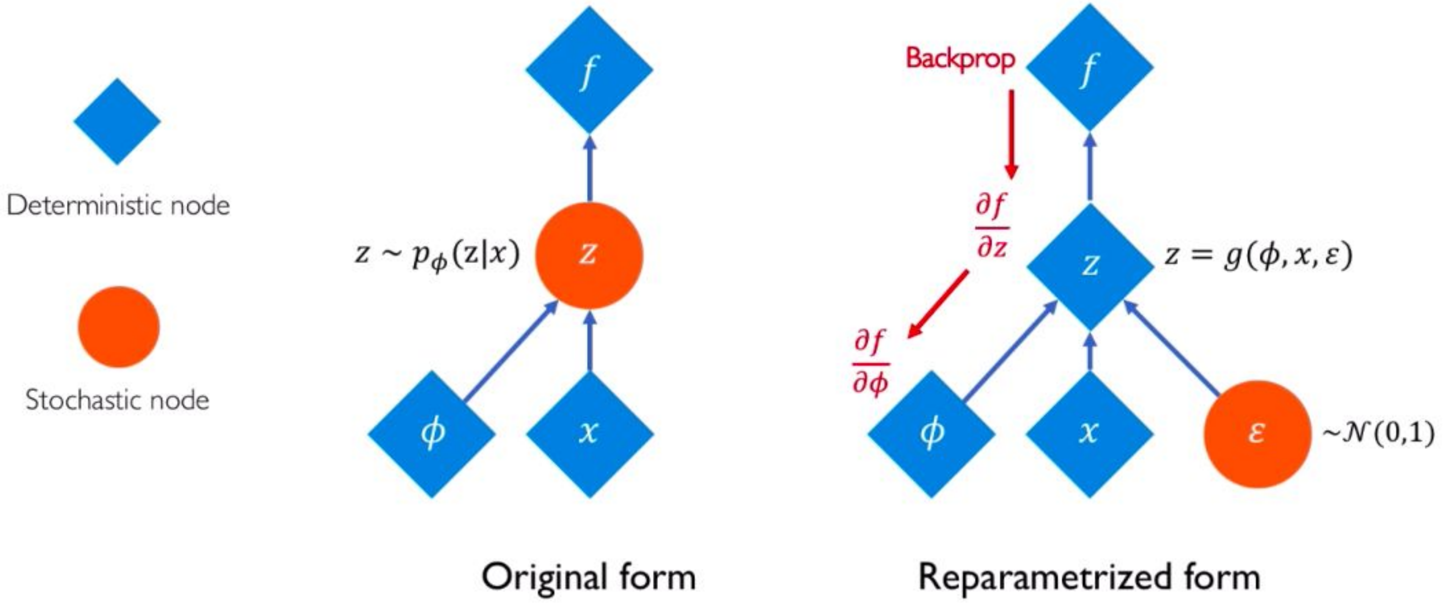
See also
https://fdocuments.in/document/notes-on-variational-autoencoders-dmmmlvaepdf-notes-on-variational-autoencoders.html?page=1

Variational Autoencoder (VAE)



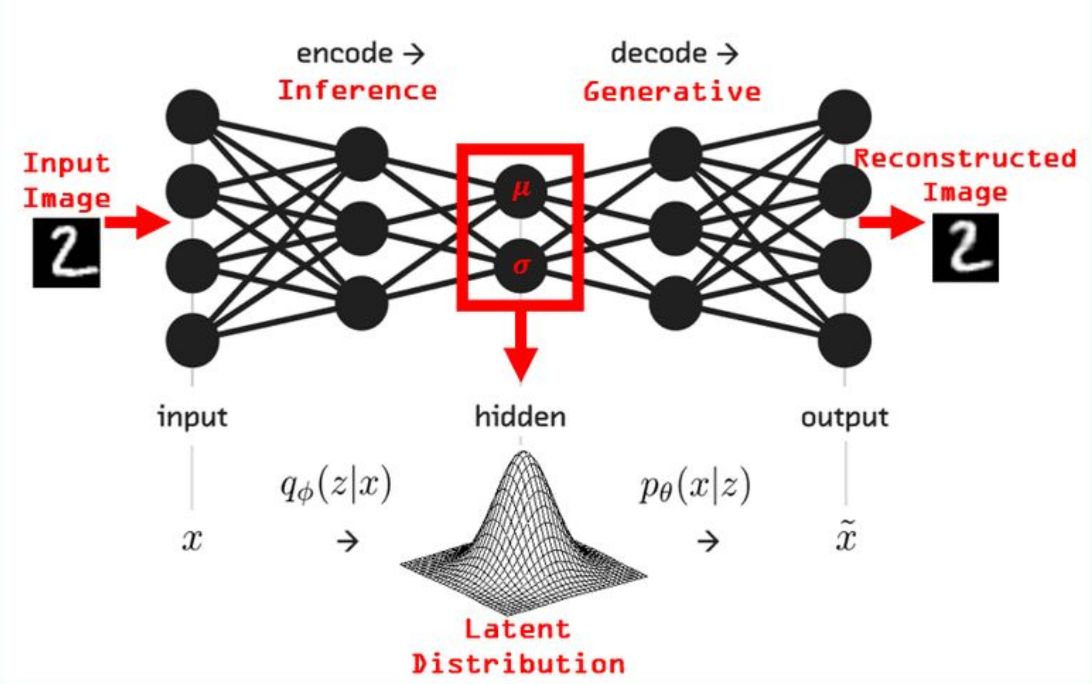
https://lilianweng.github.io/lil-log/2018/08/12/from-autoencoder-to-beta-vae.html

Variational Autoencoder (VAE)



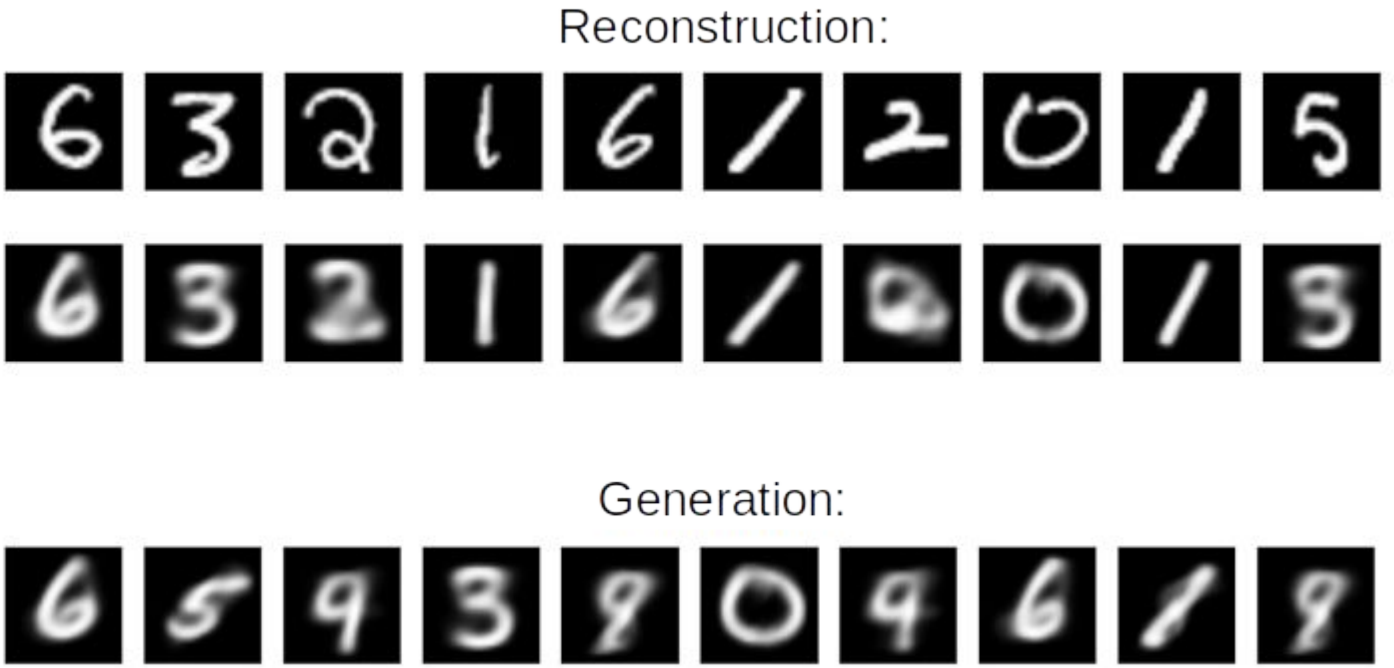
Source: MIT 6.S191 (2020): Introduction to Deep Learning - Deep Generative Modeling

Variational Autoencoder (VAE)

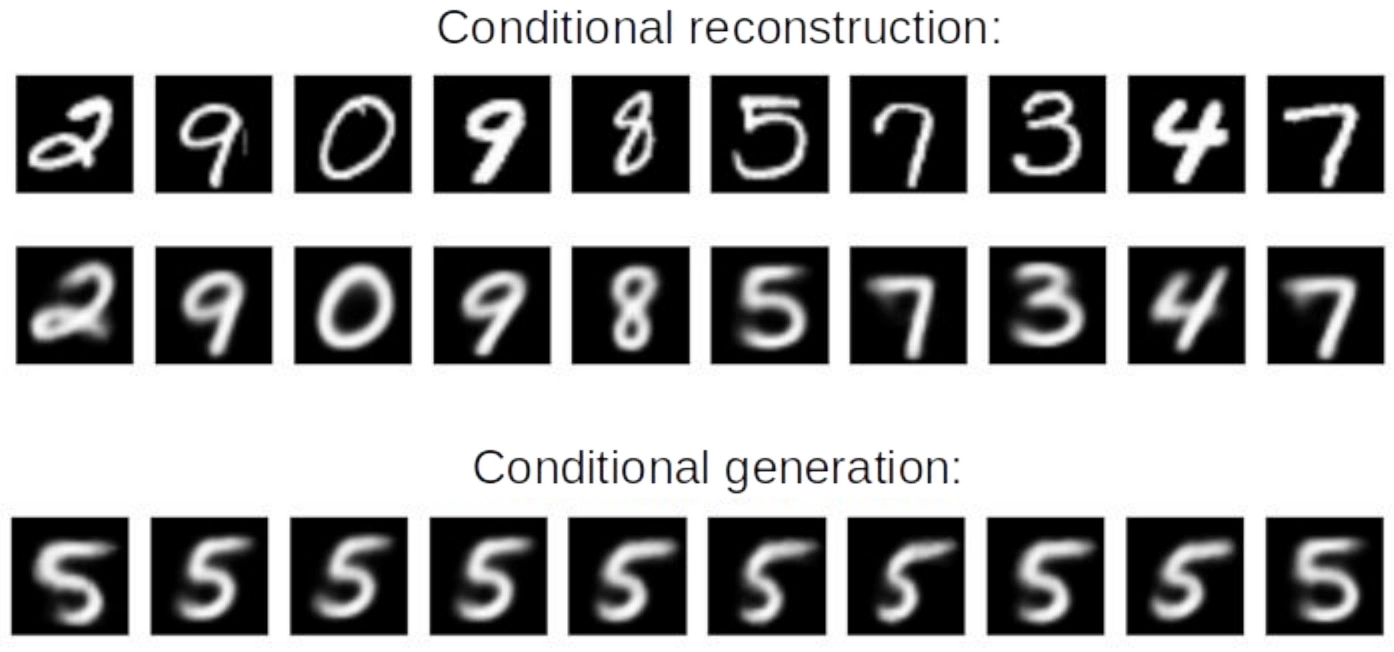


$$\begin{aligned}\mathcal{L}(\mathbf{x};\theta) &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\left[\log p(\mathbf{x},\mathbf{z}) - \log q(\mathbf{z}|\mathbf{x})\right] \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\left[\log p(\mathbf{x}|\mathbf{z})\right] - D_{KL}(q(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))\end{aligned}$$

Variational Autoencoder (VAE)



Variational Autoencoder (VAE)



Generative Adversarial Network (GAN)

UNIVERSITÉ DU LUXEMBOURG

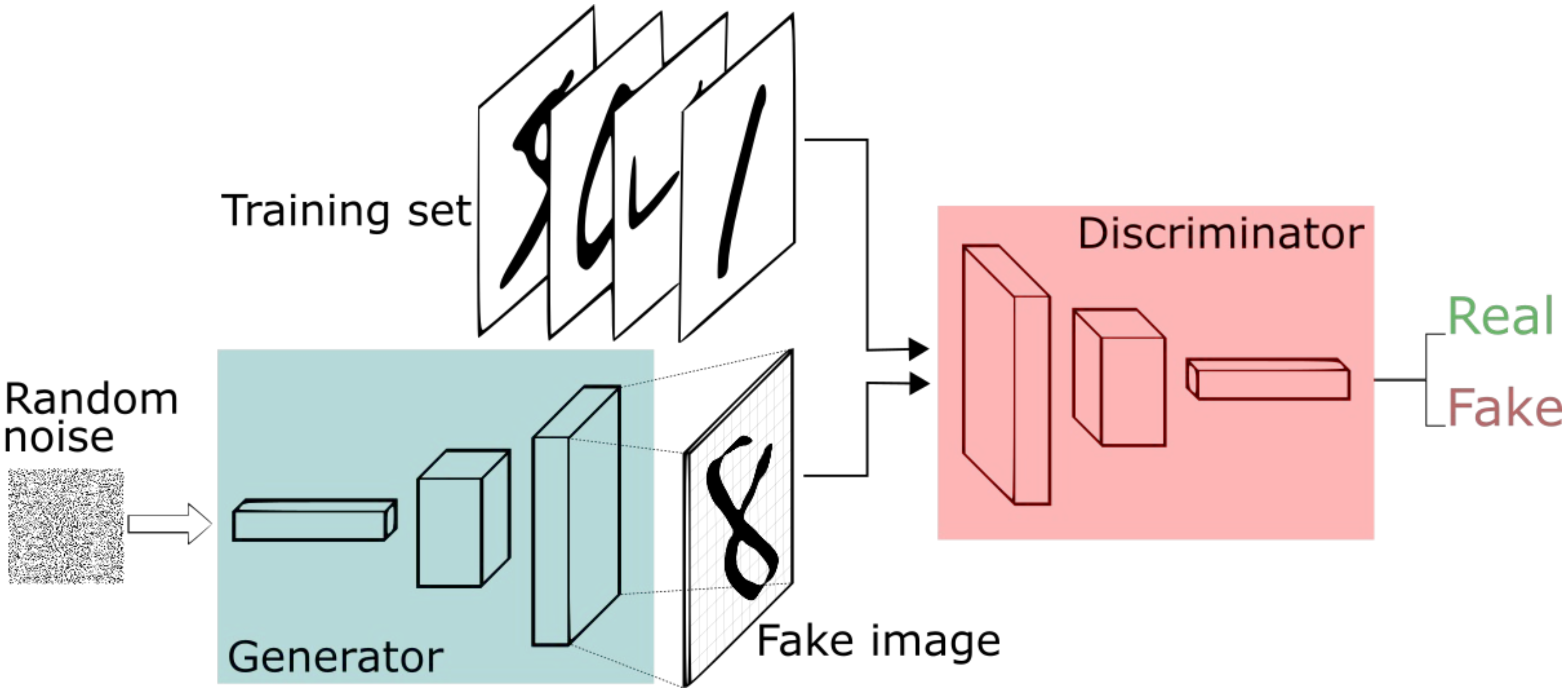
Forget about *explicit* density estimation

We care most about sampling!

Game-theoretic approach (Nash equilibrium)

- **Generator:** generate real-looking data
- **Discriminator:** distinguish between real and fake data

Generative Adversarial Network (GAN)



Generative Adversarial Network (GAN)

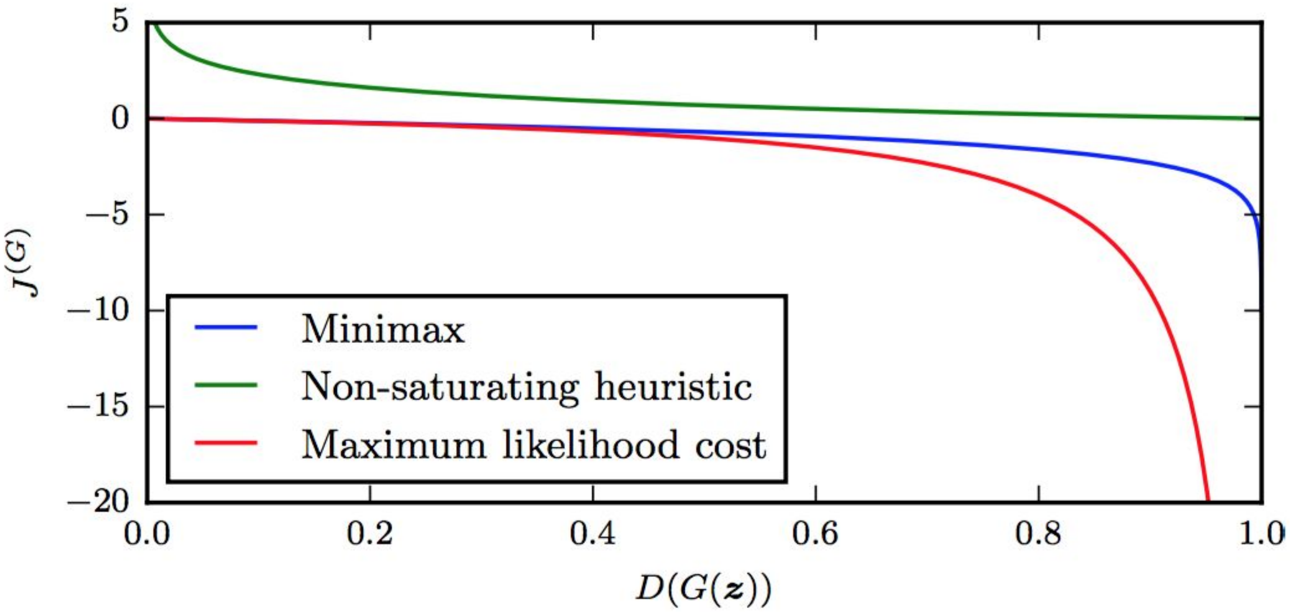
Minimax objective function:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p(x)} \log D(x) + \mathbb{E}_{z\sim p(z)} \log(1 - D(G(z)))$$

Non-saturating version:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p(x)} \log D(x) - \mathbb{E}_{z\sim p(z)} \log D(G(z))$$

Generative Adversarial Network (GAN)



https://danieltakeshi.github.io/2017/03/05/understanding-generative-adversarial-networks/

Generative Adversarial Network (GAN)

Adversarial training:

1. Gradient ascent on discriminator: $D(\boldsymbol{x}) \rightarrow 1$ and $D(G(\boldsymbol{z})) \rightarrow 0$
2. Gradient descent on generator: $D(G(\boldsymbol{z})) \rightarrow 1$

→ Better: gradient ascend on non-saturating generator loss

Training GANs is (really) hard

https://www.slideshare.net/EmanueleGhelfi/gan-theory-and-applications-143737572

Training GANs is (really) hard

Highly sensitive to hyperparameter selection

Non-convergence: model parameters oscillate (a lot)

Mode collapse: generator produces few varieties samples

Diminished gradient: discriminator quickly succeeds so generator cannot learn

On Nash equilibrium

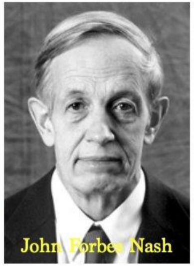
convex concave

Min-max game: $\min_{x} \max_{y} f(x, y) \geqslant \max_{y} \min_{x} f(x, y)$

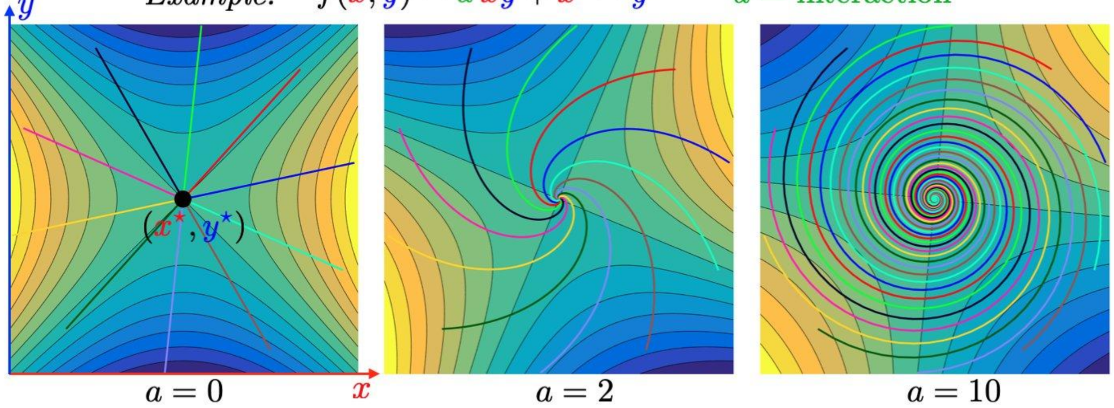
Saddle point $(x^\star, y^\star)$: $f(x^\star, y) \leqslant f(x^\star, y^\star) \leqslant f(x, y^\star)$

$\longrightarrow$ Strong duality: $\min_{x} \max_{y} f(x, y) = \max_{y} \min_{x} f(x, y)$

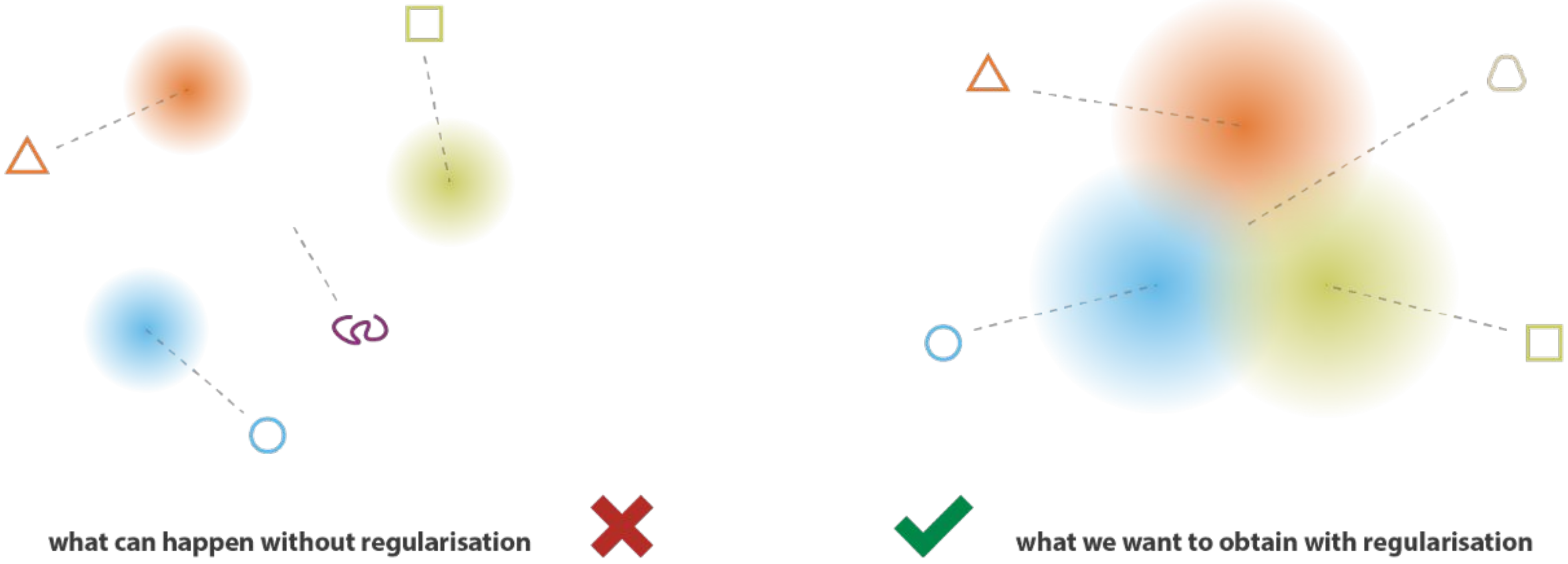
Gradient descent: $\begin{cases} x_{k+1} = x_k - \tau \nabla_x f(x_k, y_k) \\ y_{k+1} = y_k + \tau \nabla_y f(x_k, y_k) \end{cases}$



Example: $f(x, y) = a\, xy + x^2 - y^2$ $\quad a =$ interaction

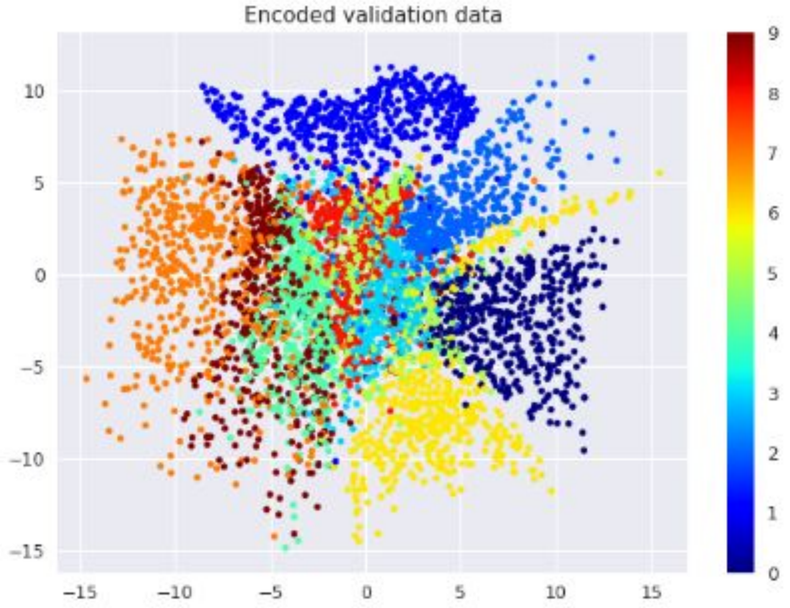


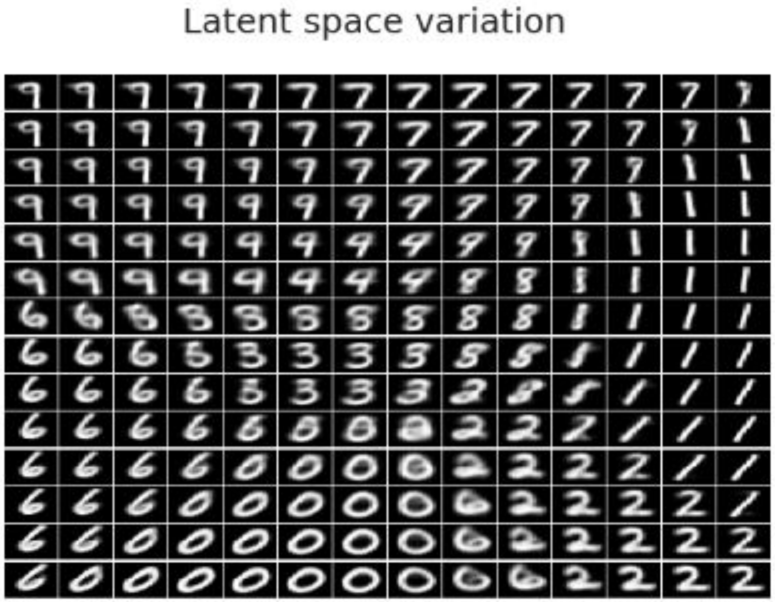
Latent space interpolation in VAEs



https://towardsdatascience.com/f70510919f73

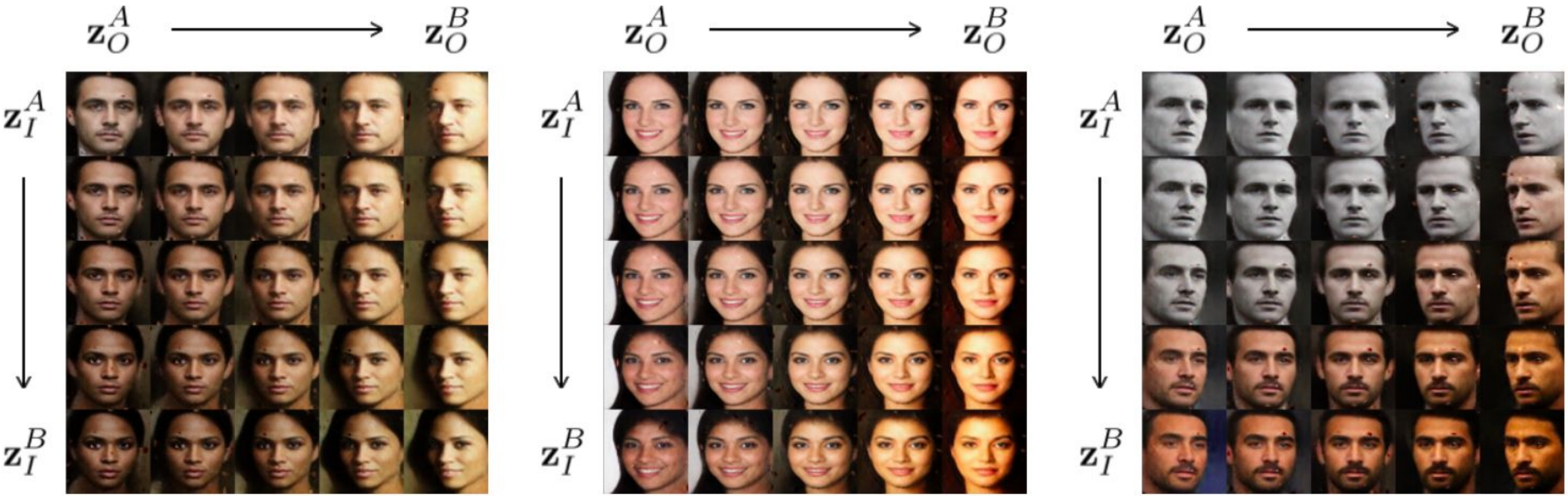
Latent space interpolation in GANs





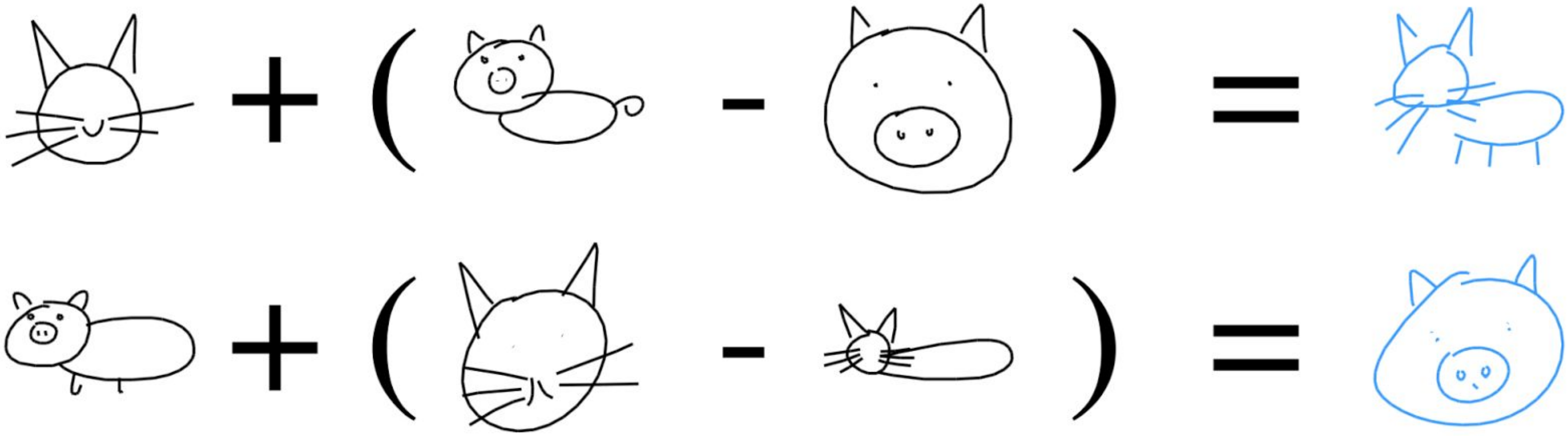
https://medium.com/jungle-book/9dd64e9628e6

Semantic decomposition of latent spaces



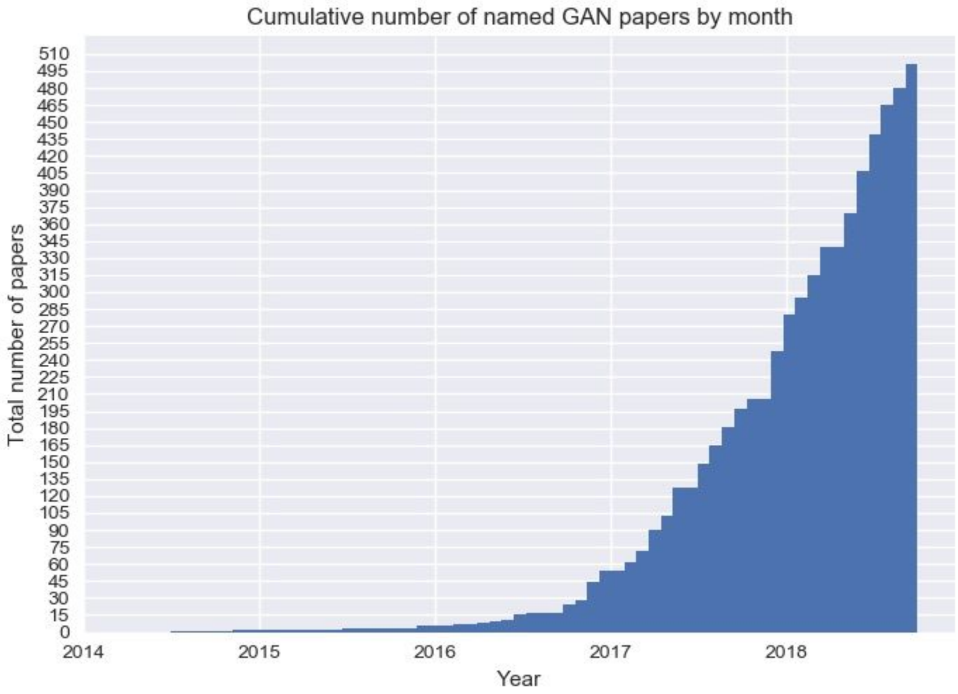
https://arxiv.org/pdf/1705.07904v2.pdf

Latent space arithmetic



https://arxiv.org/pdf/1704.03477.pdf

The GAN Zoo

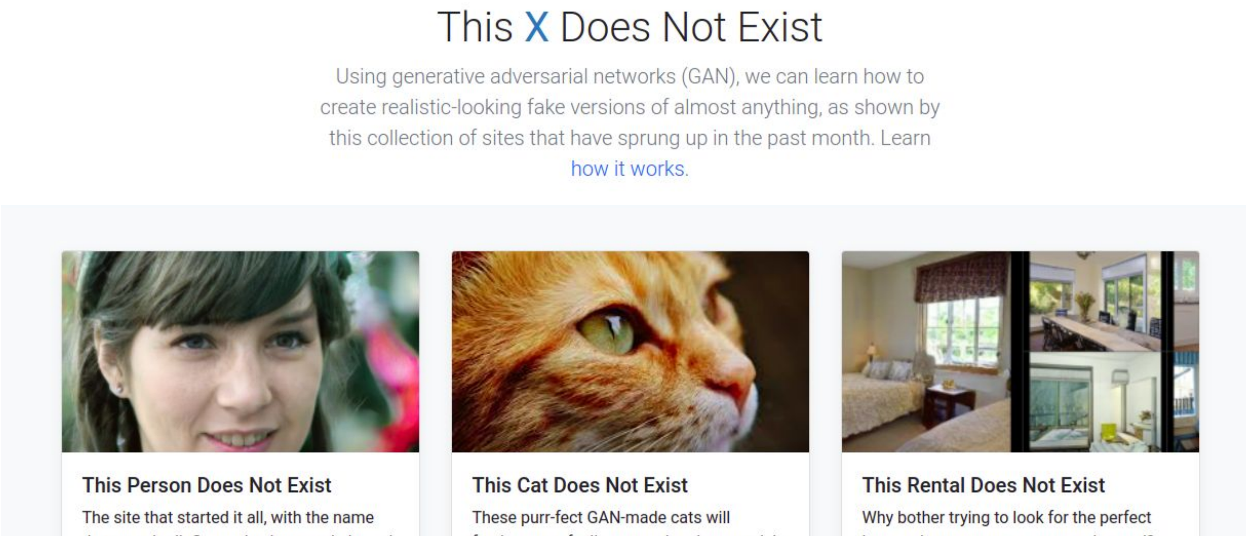


https://github.com/hindupuravinash/the-gan-zoo

Live GAN example applications

https://thisxdoesnotexist.com/

x = { person, cat, rental, waifu, url, startup, question, resume, emotion, … }



Classic GANs

pix2pix (2017)

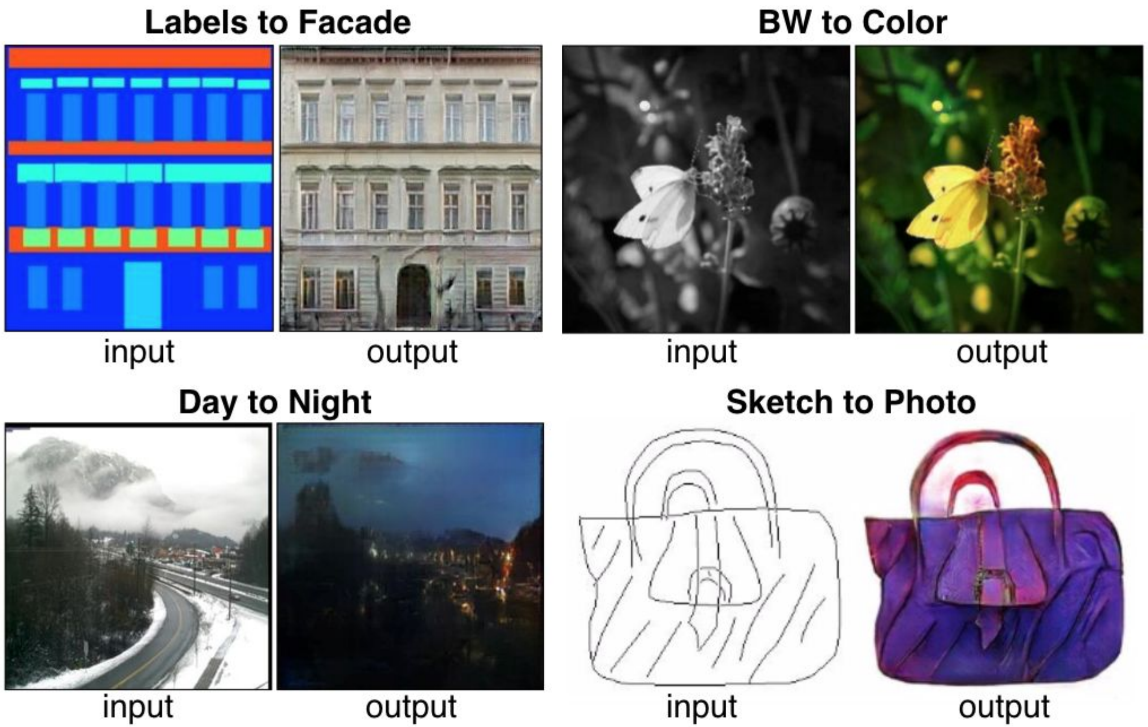
CycleGAN (2017)

PGGAN (2018)

GauGAN (2019)

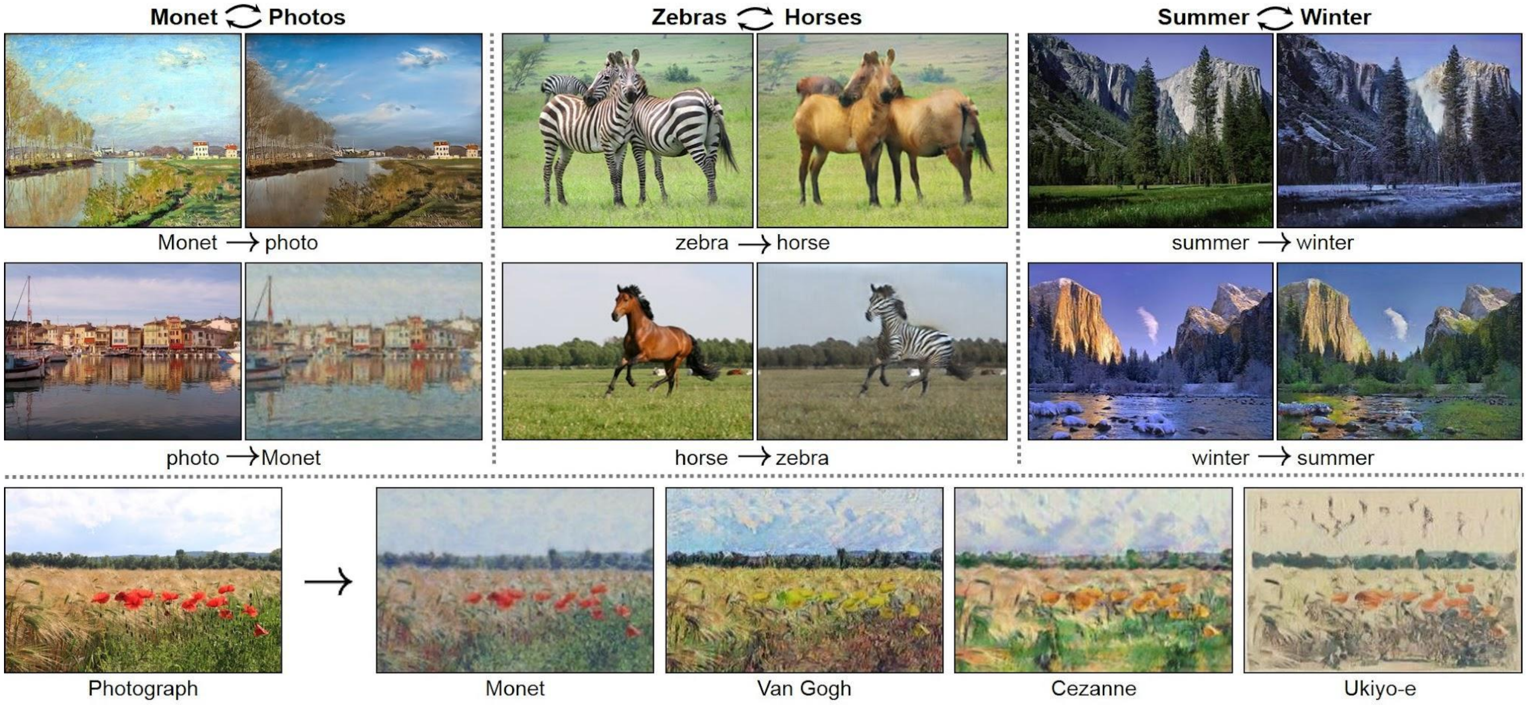
StyleGAN (2019, 2020, 2021)

Classic GAN: **pix2pix**



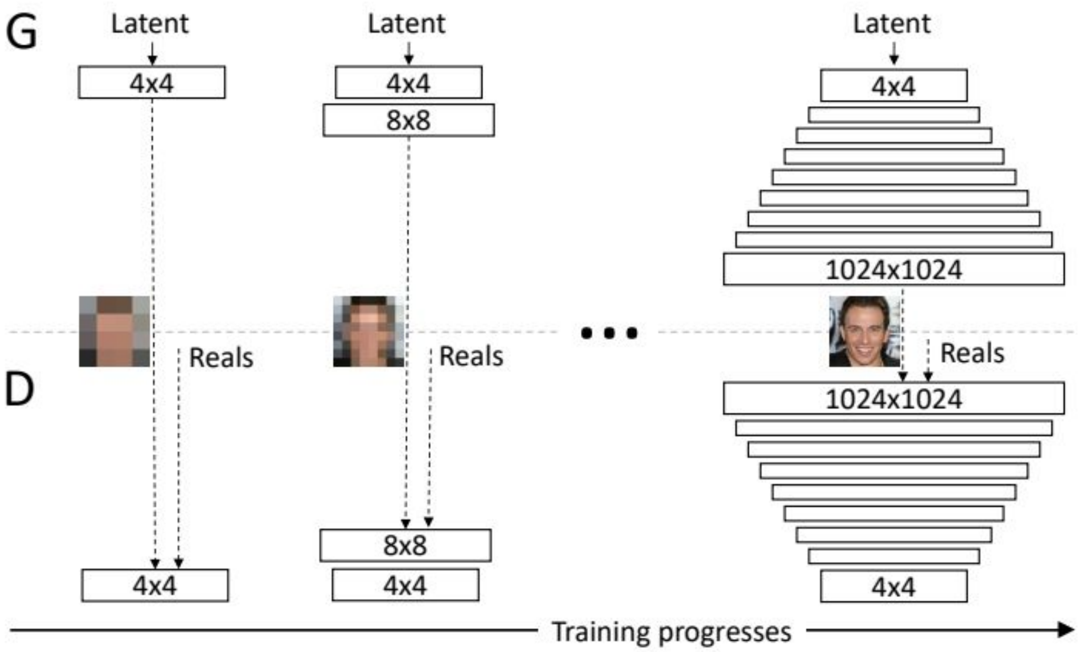
https://phillipi.github.io/pix2pix/

Classic GAN: **CycleGAN**



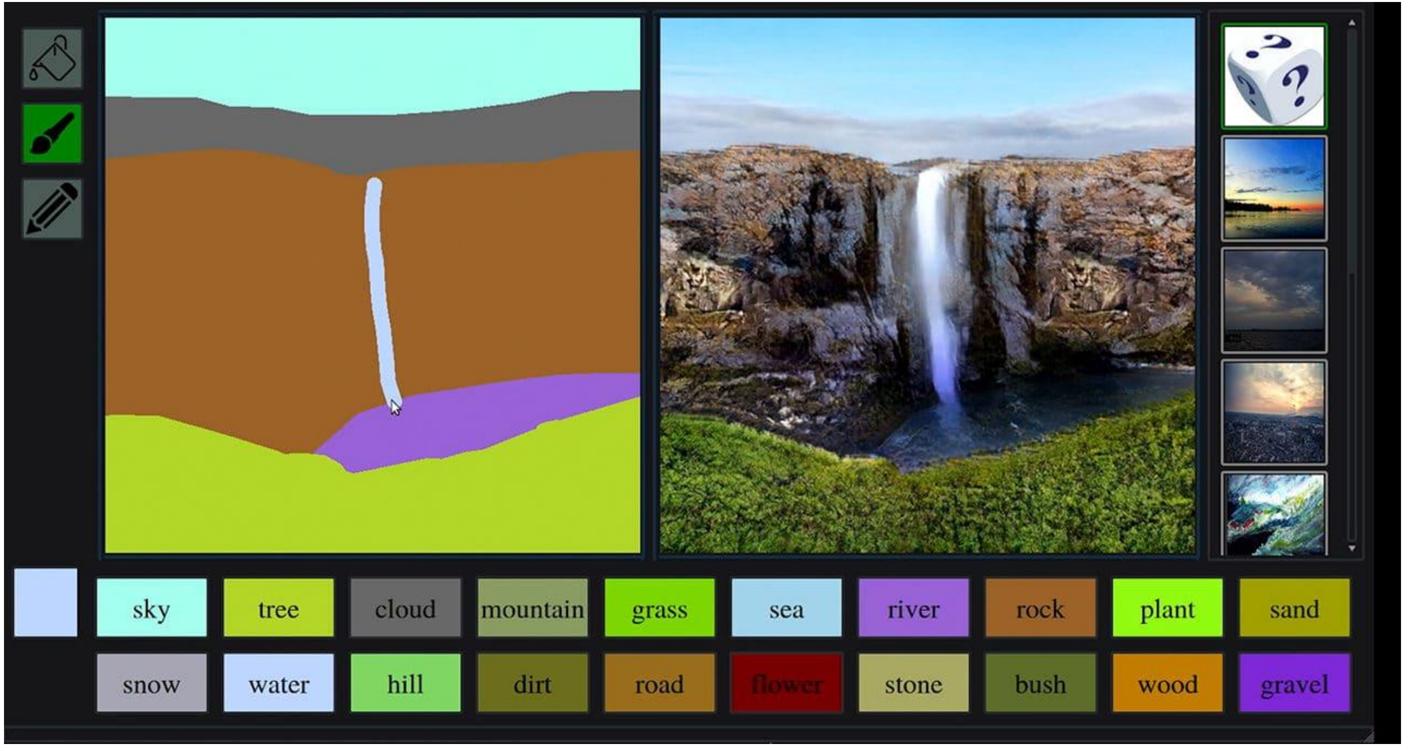
https://junyanz.github.io/CycleGAN/

Classic GAN: **PGGAN**



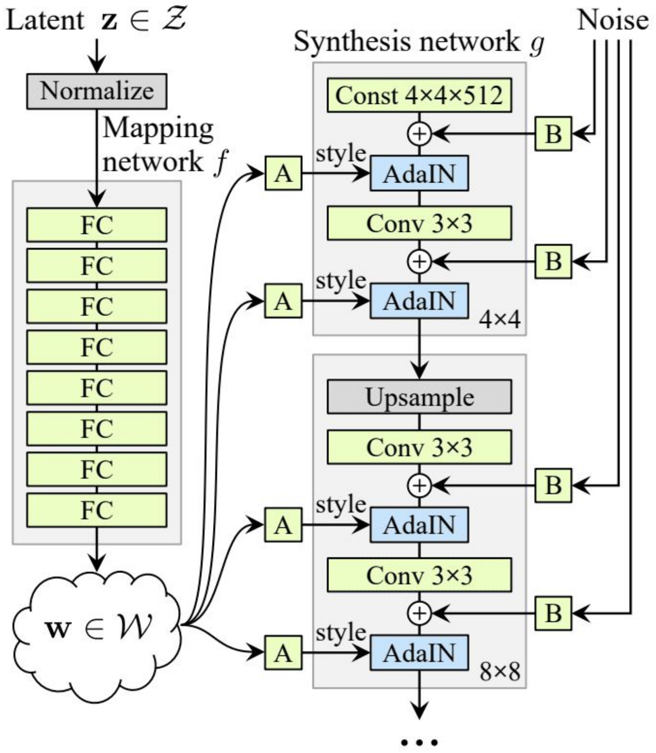
https://machinelearningmastery.com/introduction-to-progressive-growing-generative-adversarial-networks/

Classic GAN: **GauGAN**



https://blog.paperspace.com/nvidia-gaugan-introduction/

Classic GAN: **StyleGAN**



https://machinelearningmastery.com/introduction-to-style-generative-adversarial-network-stylegan/

Current challenges in generative modeling

Weak evaluation measures: FID, IS, KID, AIS, MS-SSIM, etc.

Low coverage of latent spaces

Entangled latent features

Uncontrolled sampling

Sequential, discrete data is understudied

Some tricks and tips

https://towardsdatascience.com/c9071159628

https://medium.com/@jonathan_hui/819a86b3750b

https://medium.com/@utk.is.here/edd529764aa9

https://github.com/soumith/ganhacks