Deep Generative Models:
Generative Adversarial Network

David Métivier

INRAE science for people, life & earth

Mistea Mathématiques, Informatique et STatistique pour l'Environnement et l'Agronomie

Département MathNum

Generative models

Generative models Build random generators

Goal : Learn the distribution data of a dataset $\mathbf{X}$.

- It can learn the multivariate distribution of pixels (forming an image)
- Time series
- Next word in a sentence

Mathematically :

$$\mathbf{X} = G_\theta(\mathbf{Z})$$

- G_θ : Generator (Neural Networks, Markov Chain, Random forest, . . .)
- $\mathbf{Z}$: Latent variable with **known** distributions (generally Gaussian)

Generative models

Generative models Build random generators

Goal : Learn the distribution data of a dataset $\mathbf{X}$.

- It can learn the multivariate distribution of pixels (forming an image)
- Time series
- Next word in a sentence

Mathematically :

$$\mathbf{X} = G_\theta(\mathbf{Z})$$

- G_θ : Generator (Neural Networks, Markov Chain, Random forest, ...)
- $\mathbf{Z}$: Latent variable with **known** distributions (generally Gaussian)

Example Observations $X \sim \mathcal{N}(0, 1)$. We fix $Z \sim \mathbb{U}([0, 1])$

G_θ such that $G_\theta \approx \phi_{\mathcal{N}}^{-1}$ (quantile function)

Why should it work ?

- Which type of noise ?
- How to build a generator ?
- Which loss, distance, or divergence function ?

Why should it work ?

- Which type of noise ?
- How to build a generator ?
- Which loss, distance, or divergence function ?

Theorem 1

For any random variable $X \in \mathbb{R}^d$, there is a measurable $\phi : [0,1] \to \mathbb{R}^d$ such that

$$\phi(U) \overset{dist.}{=} X$$

$\Rightarrow$ In theory $Z \sim \mathbb{U}(0,1)$ is enough !
In practice, ϕ is very ugly.
$\Rightarrow \Rightarrow$ Increase latent dimension

Model choice

Plenty of ways of building G_θ

- Classic stats/ML models (Random Forest, Gaussian Mixtures, ...)

Model choice

Plenty of ways of building G_θ

- Classic stats/ML models (Random Forest, Gaussian Mixtures, ...)
- Deep Generative Models
  - GAN : historically the first to display amazing results.

Model choice

Plenty of ways of building G_θ

- Classic stats/ML models (Random Forest, Gaussian Mixtures, ...)
- Deep Generative Models
  - GAN : historically the first to display amazing results.
  - Plenty of variants today (conditional GAN, Wasserstein GAN, CycleGAN, StyleGAN)

Model choice

Plenty of ways of building G_θ

- Classic stats/ML models (Random Forest, Gaussian Mixtures, ...)
- Deep Generative Models
  - GAN : historically the first to display amazing results.
  - Plenty of variants today (conditional GAN, Wasserstein GAN, CycleGAN, StyleGAN)
  - Other Deep Generative Models (Diffusion Model, VAE, ...)

What are GANs (Goodfellow et al. 2014)

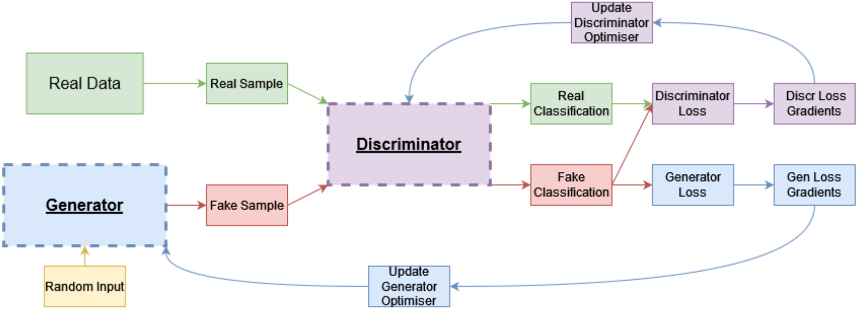
In the relatively short history of deep learning, GANs have significantly revolutionized the world of AI and made their mark with popular demonstrators such as https://thisxdoesnotexist.com/

GANs can learn very complex G_θ like distribution of images of realistic people, beach, mountain etc.

GANs : Architecture and Loss function

$$\min_G \max_D \text{Loss}(D,G) = \mathbb{E}_{\mathbf{x}\sim\text{data}}\left[\log D(\mathbf{x})\right] + \mathbb{E}_{\mathbf{z}\sim f_{\mathbf{Z}}}\left[\log\left(1 - D(G(\mathbf{z}))\right)\right]$$

- G Generator (a classifier Neural Network)
- D Discriminator (a Neural Network)
- $\mathbf{X} \sim \text{data}$
- $\mathbf{Z} \sim f_z$ noise (latent vector) of dimension l



GAN limitations

- Very fragile equilibrium between Discriminator and Generator e.g. mode collapse (generator learns too fast compared to the discriminator)
- Vanishing gradients
- Needs a lot of resources with huge architectures