

GENERATIVE ADVERSARIAL NETWORKS

INTRODUCTION AU DEEP LEARNING

Théo Lopès-Quintas

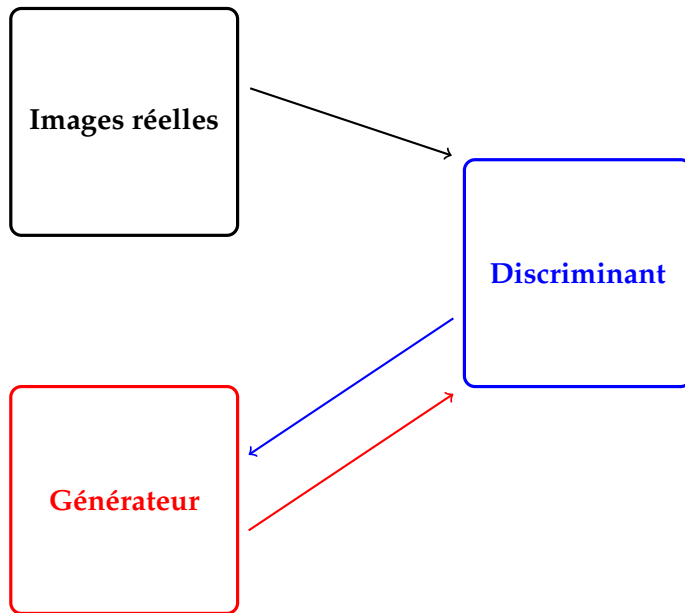
BPCE Payment Services,
Université Paris Dauphine

2024 - 2025

| | | |
|----------|-------------------------------------|----------|
| 1 | Introduction | 1 |
| 2 | Comment entraîner? | 3 |
| 3 | Les problèmes | 6 |

INTRODUCTION

L'IDÉE



INTRODUCTION

LES AUTEURS

Ian Goodfellow a travaillé chez Google puis OpenAI. Il travaille actuellement chez Deepmind après avoir été directeur de la recherche pour Apple.



Chez Deepmind travaille également Mehdi Mirza et David Warde-Farley. Jean Pouget-Abadie travaille actuellement chez Google Research et Bing Xu chez Deepmind. Le point commun entre tous ces chercheurs, et ceux que nous n'avons pas cité, est l'université de Montréal. Tous ces chercheurs se sont trouvés au même endroit sous la direction de Yoshua Bengio pour développer, entre autre, cette idée.

Un chercheur isolé n'existe pas

— Cédric Villani, *Pour faire naître une idée* (2015)

COMMENT ENTRAÎNER?

RAPPEL DE L'OBJECTIF

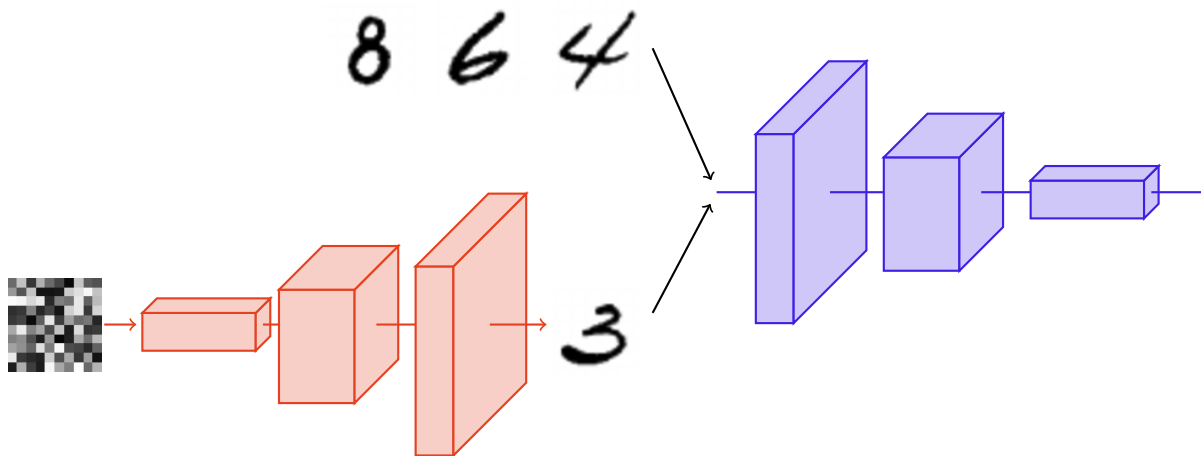


Figure – Architecture d'un *Generative Adversarial Networks* avec un **générateur** et un **discriminant**

COMMENT ENTRAÎNER ?

FONCTION DE PERTE

Soit $d, d' \in \mathbb{N}^*$ les dimensions respectivement du modèle générateur et discriminant. La fonction de perte du système est :

$$(\Theta^*, \Psi^*) = \min_{\Theta \in \mathbb{R}^d} \max_{\Psi \in \mathbb{R}^{d'}} \mathbb{E}_{x \sim \text{réel}} \left[\log \left(\overbrace{D_{\Psi}}^{\text{Discriminant}}(x) \right) \right] + \mathbb{E}_{z \sim \text{aléatoire}} \left[\log \left(1 - D_{\Psi}(\underbrace{G_{\Theta}}_{\text{Générateur}}(z)) \right) \right]$$

COMMENT ENTRAÎNER ?

FONCTION DE PERTE

Soit $d, d' \in \mathbb{N}^*$ les dimensions respectivement du modèle générateur et discriminant. La fonction de perte du système est :

$$(\Theta^*, \Psi^*) = \min_{\Theta \in \mathbb{R}^d} \max_{\Psi \in \mathbb{R}^{d'}} \mathbb{E}_{x \sim \text{réel}} \left[\log \left(\overset{\text{Discriminant}}{\underset{\downarrow}{D_{\Psi}}}(x) \right) \right] + \mathbb{E}_{z \sim \text{aléatoire}} \left[\log \left(1 - D_{\Psi}(\overset{\text{Générateur}}{\underset{\uparrow}{G_{\Theta}}}(z)) \right) \right]$$

Exercice 1

On s'intéresse à la dynamique d'apprentissage.

1. De quoi dépend le gradient pour Θ ? Et pour Ψ ?
2. Quand est-ce que l'équilibre sera atteint ?

LES PROBLÈMES ...

GRADIENT VANISHING STRIKES BACK

Rappelons le problème d'optimisation :

$$(\Theta^*, \Psi^*) = \min_{\Theta \in \mathbb{R}^d} \max_{\Psi \in \mathbb{R}^{d'}} \mathbb{E}_{x \sim \text{réel}} \left[\log \left(\overset{\text{Discriminant}}{\underset{\downarrow}{D_\Psi}}(x) \right) \right] + \mathbb{E}_{z \sim \text{aléatoire}} \left[\log \left(1 - D_\Psi \left(\overset{\text{Générateur}}{\uparrow}{G_\Theta}(z) \right) \right) \right]$$

Exercice 2

Nous sommes dans le cas où le discriminateur est très bon.

1. Que se passe-t-il pour le gradient du générateur ?
2. Pourquoi remplacer $\log(1 - D_\Psi(G_\Theta(z)))$ par $\log(D_\Psi(G_\theta))$ et réaliser une montée de gradient au lieu d'une descente de gradient règle le problème ?

LES PROBLÈMES ...

MODE COLLAPSE

L'équilibre est obtenu quand le générateur réussit à produire des images telle que le modèle discriminant ne peut pas les dissocier des vraies images. Une manière de le réaliser est de ne générer qu'un seul *type* d'image plutôt que la totalité de la diversité d'image présente dans le dataset. Ce phénomène s'appelle le **mode collapse**.



Figure – Exemple d'un mode collapse avec MNIST

LES PROBLÈMES ...

MESURER LA PERFORMANCE

Mesurer la performance d'un tel modèle n'est pas simple : comment s'assurer que le modèle génère **bien** toute la **diversité** de la distribution réelle ? Il n'y a pas de réel consensus parmi les nombreuses métriques proposées, nous en présentons deux.

Inception Score (IS)

Pour chaque image générée, on calcule la distribution de la classe prédite par le modèle Inception v3. En répétant l'opération pour chaque image générée, on peut calculer la divergence de KL entre la distribution d'une image et la distribution moyenne des images.

Fréchet Inception Distance (FID)

En utilisant le modèle Inception v3, on mesure la distance entre les features de la couche précédant la couche de classification entre les images réelles et générées. La distance sélectionnée ici est la distance de Fréchet.