Apprentissage profond non supervisé

Bruno Sauvalle

Centre de Robotique

7 mars 2023

Sommaire

Introduction

Apprentissage de la distribution des données

Apprentissage de représentations

Modèles génératifs

Sommaire

Introduction

Apprentissage de la distribution des données

Apprentissage de représentations

Modèles génératifs

L'apprentissage non supervisé, c'est quoi ?

Apprentissage supervisé

On considère un dataset $(x_i, y_i)_{1 \leq i \leq N}$, où les y_i sont les labels associés aux données x_i.
On veut apprendre la fonction $p(y \mid x)$.

Apprentissage non supervisé

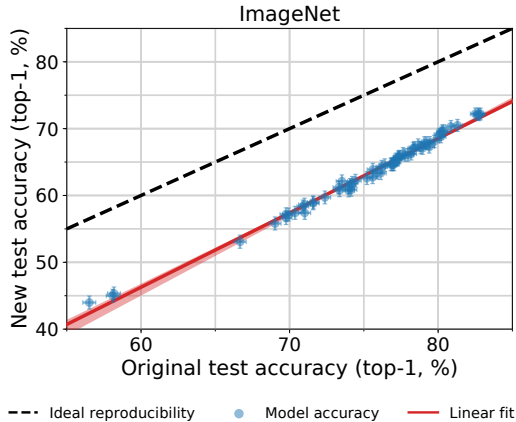
On considère un dataset $(x_i)_{1 \leq i \leq N}$.
On souhaiterait :

- apprendre la distribution des données $p(x)$
- construire une "bonne" représentation des données x_i
- générer des données suivant la même distribution que les x_i

L'apprentissage non supervisé, pourquoi ?

- très large quantité de données non labellisées disponibles
- coût de la production de données labellisées
- erreurs sur les données labellisées
- problème de robustesse aux changements de distribution ou aux outliers

généralisation sur Imagenet : Imagenet-v2



B. Recht et al : Do ImageNet Classifiers Generalize to ImageNet? - ICML 2019

L'apprentissage non supervisé vu par Yann Le Cun (NIPS 2016)

- **"Pure" Reinforcement Learning** (cherry)
  - The machine predicts a scalar reward given once in a while.
  - **A few bits for some samples**
- **Supervised Learning** (icing)
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - **10→10,000 bits per sample**
- **Unsupervised/Predictive Learning** (cake)
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - **Millions of bits per sample**

- **(Yes, I know, this picture is slightly offensive to RL folks. But I'll make it up)**

Sommaire

Introduction

Apprentissage de la distribution des données

Apprentissage de représentations

Modèles génératifs

Apprendre la distribution des données

Hypothèse de base : Les données d'entrées $(x_i)_{1 \leq i \leq N}$ sont considérées comme des échantillons indépendants et identiquement distribués issus d'une unique distribution $p(x)$.

Notion de dimension : nombre de variables scalaires nécessaires pour représenter la donnée

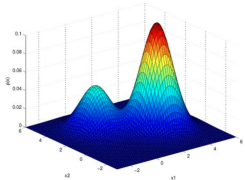
exemples d'images du dataset FFHQ : $d = 3 \times 1024 \times 1024 = 3\,145\,728$

Apprendre la distribution des données en petite dimension : modélisation statistique classique

méthodes paramétriques

exemple : modélisation d'une distribution comme un mélange de gaussiennes

$$p(x) \simeq \sum_{k=1}^{g} \pi_k \frac{1}{\sqrt{(2\pi)^k \det(\Sigma_k)}} e^{-\frac{1}{2}(x-\mu_k)^t \Sigma_k^{-1}(x-\mu_k)}$$

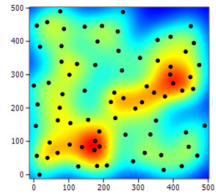


exemple de mélange de deux gaussiennes

méthodes non paramétriques

exemple : estimation par noyau (KDE)

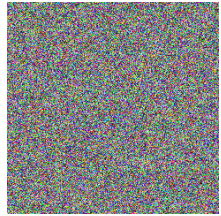
$$p(x) \simeq \frac{1}{N} \sum_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{\|x-x_i\|^2}{2\sigma^2}}$$



exemple d'estimation de densité par noyau

Apprentissage de la distribution des données : grande dimension

Problème : En grande dimension, p(x) est généralement nul ou quasiment nul sur la quasi-totalité de l'espace envisagé.



example d'image RGB échantillonnée suivant une distribution uniforme

On recherche à modéliser la distribution des données dans un espace quasiment vide !

Clustering

Identifier au préalable des clusters dans cet espace ?

Clustering en basse dimension - Méthode des k-moyennes

Faible complexité en pratique: $Nk \times$ *nombre d'iterations*
Méthode se prête bien au parallélisme distribué, peut donc être utilisé sur des datasets extrêmement grands

Clustering en haute dimension - la malédiction de la dimensionnalité

- Méthodes classiques de clustering inopérantes en grande dimension
- Malédiction de la grande dimension : en grande dimension, et après renormalisation des variables, des échantillons aléatoires sont approximativement tous situés à la même distance les uns des autres.

  exemple : images noir/blanc d = 1000x1000.
  $D(x,y) = \frac{1}{d}\sum_{i=1}^{d}(x_i - y_i)^2$
  Peut s'interpréter comme la moyenne de 1000 000 variables aléatoires indépendantes et identiquement distribuées égales à zero ou 1.
  théorème central limite : $D \simeq \mathbb{E}[D] + \frac{\sigma}{\sqrt{d}}\epsilon = \frac{1}{2} \pm \frac{1}{2000}$

Clustering en haute dimension - quelle métrique ?

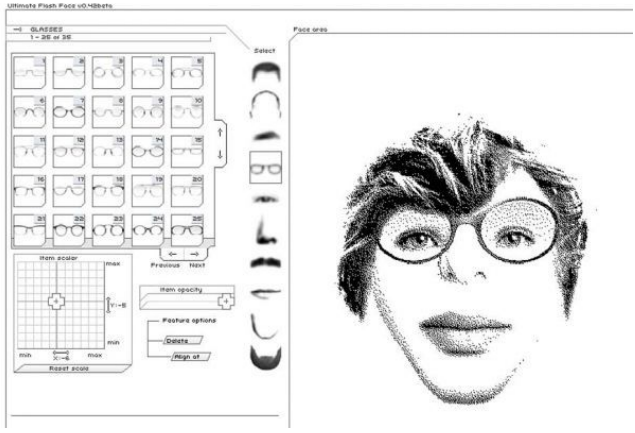
La distance euclidienne n'est pas forcément une métrique pertinente !

Pas d'algorithme universel. Nécessaire d'utiliser des techniques d'apprentissage profond pour :

- réduire la dimensionnalité des données
- obtenir des métriques induites pertinentes

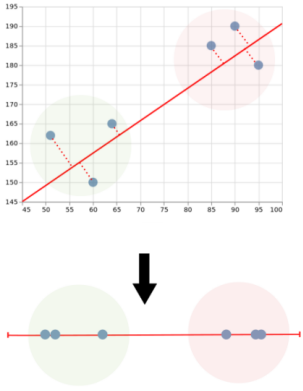
Réduction de dimensionnalité

Hypothèses : Les données x_i d'entrée ne sont pas denses dans l'espace de dimension d, mais sont en réalité concentrées sur une variété de beaucoup plus petite dimension qu'il faut découvrir.

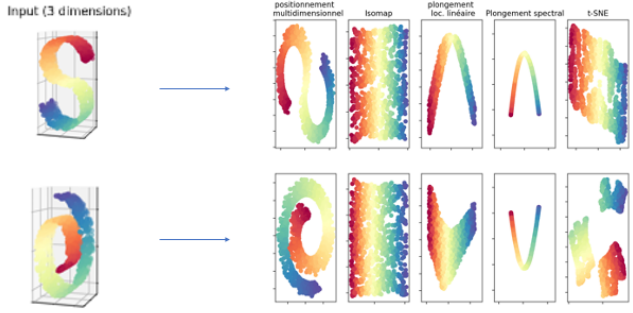


exemple de portrait-robot

Réduction de dimensionnalité en petite dimension



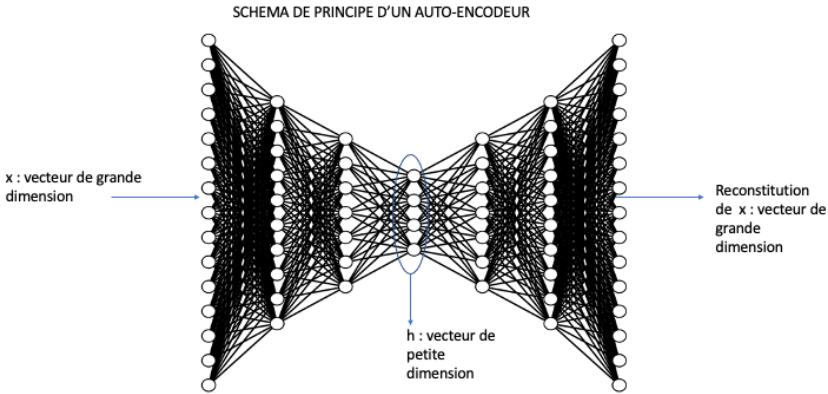
analyse en composante principale



exemples de réduction de dimensionnalité non linéaires

Réduction de dimensionnalité en grande dimension : les auto-encodeurs

- encodeur : convertit la donnée d'entrée x en un vecteur de petite dimension h
- décodeur : essaye de régénérer x à partir de h
- entraînement conjoint de l'encodeur et du décodeur, avec pour objectif de réduire l'erreur de reconstruction



Sommaire

Introduction

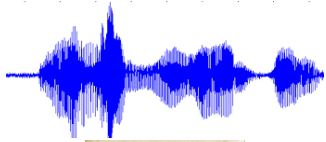
Apprentissage de la distribution des données

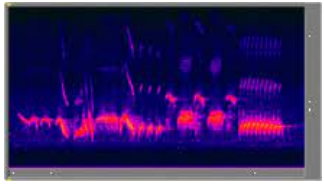
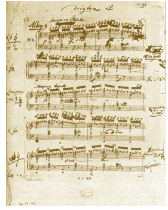
Apprentissage de représentations

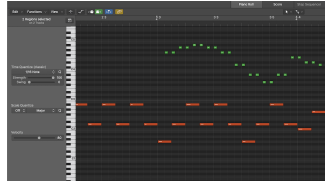
Modèles génératifs

La notion de représentation

Une même donnée peut être représentée de nombreuses façons.







Quelle représentation construire et utiliser ?

Qu'est-ce qu'une bonne représentation ?

- compression, réduction de dimensionnalité
- représente non la simple apparence, mais le contenu sémantique des données
- invariance à certaines transformations (bruit,mouvements de caméra..)
- identifier les variables latentes à l'origine de la génération des données
- améliore la performance des modèles utilisant cette representation

La notion de transfer learning

- Un modèle de classification d'image entraîné sur Imagenet peut être réutilisé et faire l'objet d'un "fine-tuning" sur d'autres dataset plus réduits.
  Ceci permet d'obtenir de bien meilleurs résultats sur le nouveau dataset qu'avec une initialisation aléatoire.
- Les feature maps produites par un modèle de classification peuvent être réutilisées pour faire de la détection d'objet ou de la segmentation

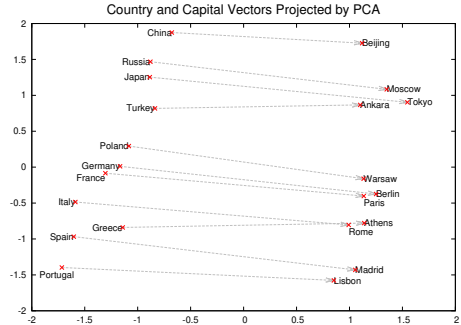
Apprentissage auto-supervisé de représentations non structurées

Peut-on obtenir des résultats équivalents ou meilleurs que le transfer learning sans données labellisées ?
Oui !
domaine de recherche très actif actuellement.
Entraînement sur des "pretext tasks" qui nécessitent une compréhension sémantique de la donnée d'entrée.

Les représentations de mot (2013)

Méthodes permettant de représenter chaque mot par un vecteur.

- La proximité entre les représentations est souvent associée à une proximité sémantique.
- Des opérations vectorielles peuvent avoir un sens sur le plan sémantique.



exemples de représentations de mot

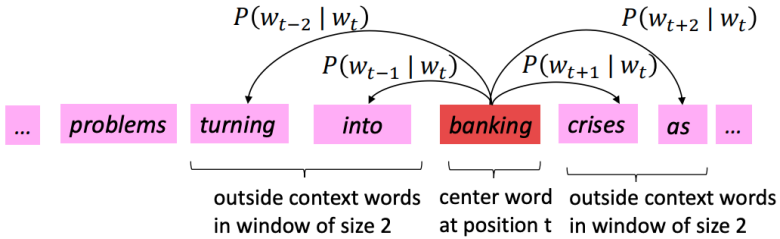
Exemple de représentation de mot : WordtoVec

Pretext task : prévoir les mots situés à proximité d'un mot donné

Maximiser $\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c}\log(p(w_{t+j} \mid w_t))$

avec

$$p(w_O \mid w_I) = \frac{\exp({v'}^t_{w_O} v_{w_I})}{\sum_{w=1}^{W}\exp({v'}^t_{w} v_{w_I})}$$



Les représentations de mot contextuelles : BERT (2019)

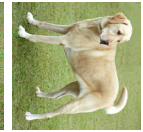
Objectif : associer à chaque mot ou partie d'un mot un vecteur qui reflète la signification du mot compte tenu du contexte où il apparaît.
Deux pretext tasks :

- prédiction des tokens manquant dans une phrase
- détecte si deux phrases données sont consécutives ou non

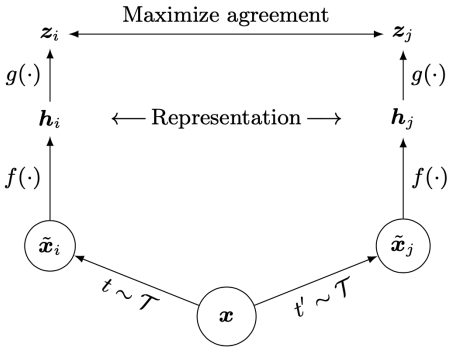
A révolutionné le traitement du language naturel en obtenant après fine-tuning de meilleurs résultats que les modèles spécialisés sur plus de onze taches différentes (compréhension de phrases, questions/réponses, inférences élémentaires).

Exemple d'apprentissage auto-supervisé de représentations non structurées dans le domaine de la vision : SimCLR

One demande au modèle de produire des représentations le plus stables possibles par rapport à l'action de certaines transformations qui préservent le contenu sémantique de l'image.



SimCLR : entraînement

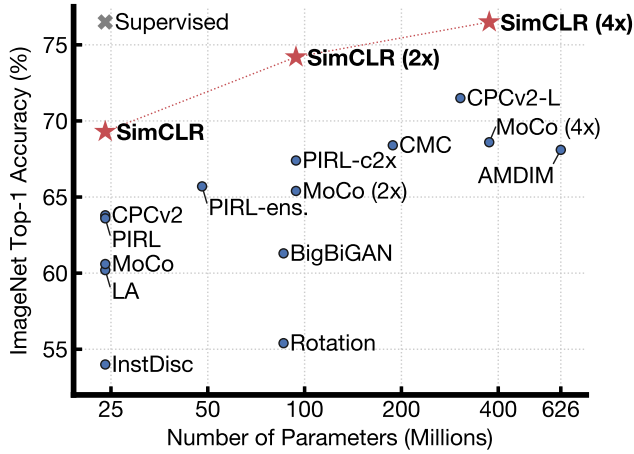


modèle SimCLR

$$\mathrm{sim}(u, v) = u^\top v / \|u\| \|v\|$$

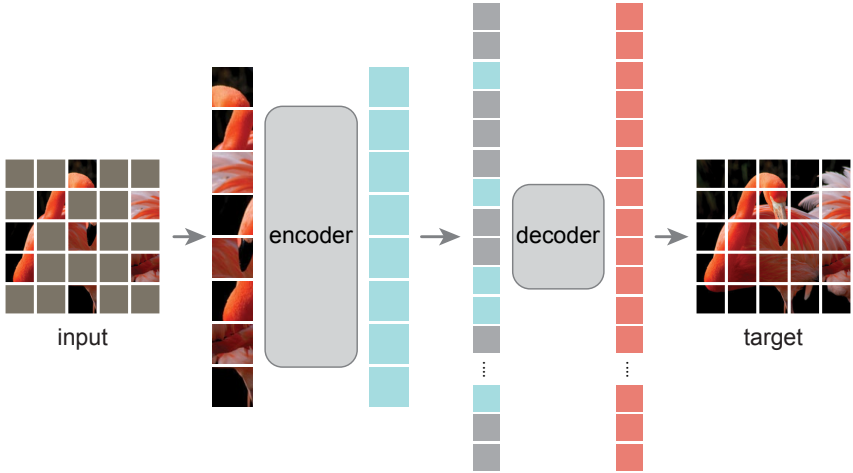
$$l(i, j) = -\log \frac{\exp(\mathrm{sim}(z_i, z_j)/\tau)}{\sum_{k=1, k \neq i}^{2N} \exp(\mathrm{sim}(z_i, z_k)/\tau)}$$

SimCLR : performances



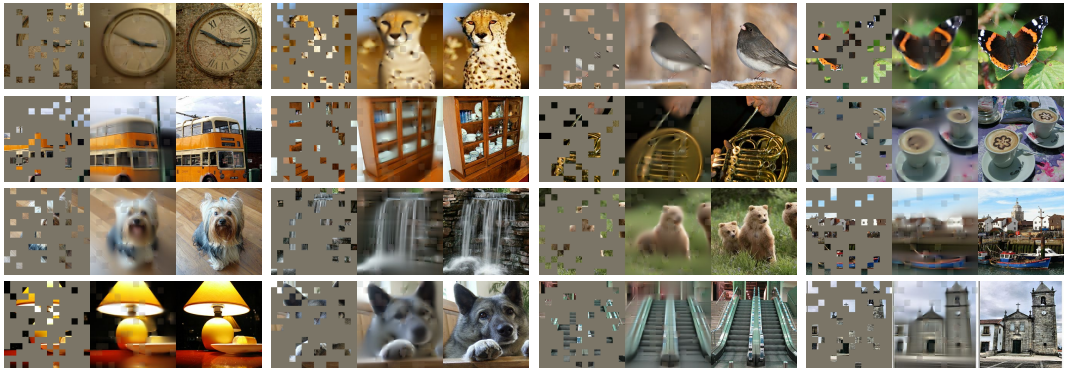
Résultats obtenus par un classificateur linéaire utilisants les features générées par SimCLR, sans fine-tuning

Le modèle MAE (masked autoencoder)



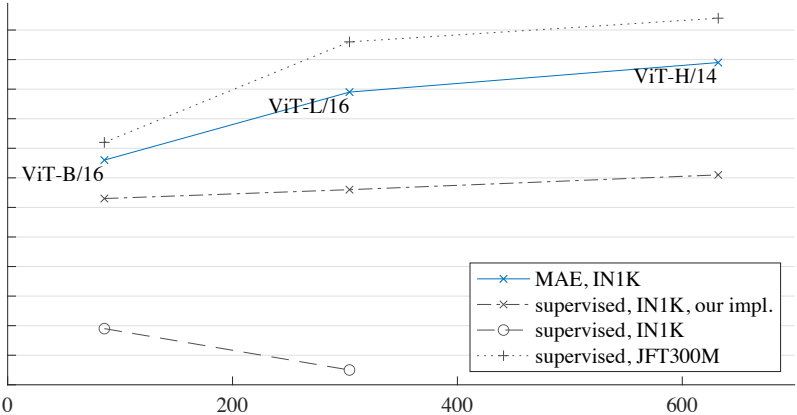
modèle mae

MAE : exemples de reconstruction



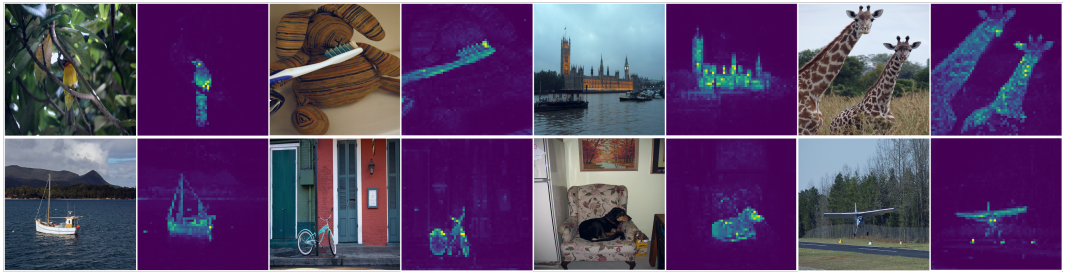
dataset Imagenet-val, taux d'occultation 80%

MAE : résultats



résultats sur imagenet après fine-tuning en fonction du nombre de paramètres du modèle. Imagenet1K : 1,3 M images; JFT300M : 300 M images

Propriétés émergentes des modèles entraînés par auto-supervision



Cartes d'attention extraites d'un modèle entraîné par auto-supervision suivant la méthode DINO

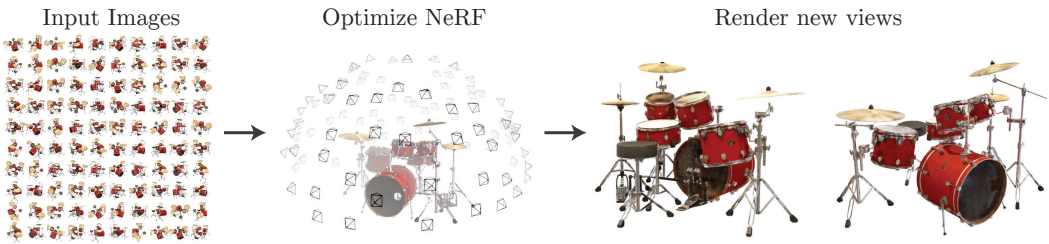
Avantages et inconvénients des représentations non structurées

- ✓ meilleures performances finales, même lorsque l'on fait le pre-training sur le dataset labellisé.
- ✓ permet de réduire la quantité de données labellisées nécessaire
- ✓ réduit le risque d'overfitting
- ✓ propriétés émergentes
- ✗ représentations non interprétables directement

Apprentissage auto-supervisé de représentations structurées

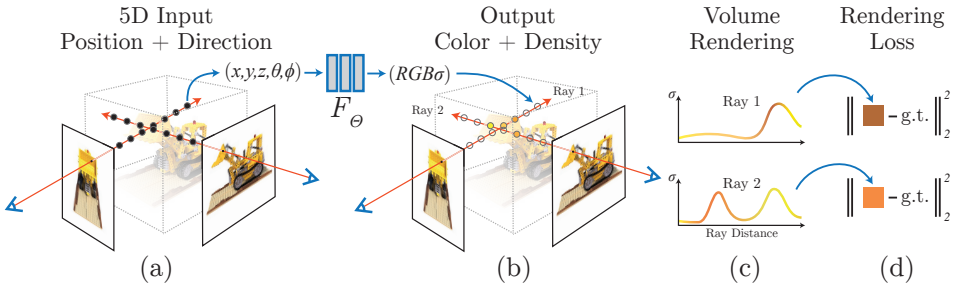
On suppose que les données ont été générés suivant un mécanisme connu.
On optimise un modèle pour qu'il reconstruise les données d'entrée suivant ce mécanisme.

Exemple de représentation structurée : les NeRF (Neural radiance fields)



NeRF

- Hypothèse : l'image est générée par ray-tracing en partant d'un champ 5D de densité et de couleur.
- Ce champ de densité et de couleur est modélisé par un réseau de neurones.



Représentations structurées : représentations centrées sur les objets

- Hypothèse : l'image est composée d'un arrière plan et de différents objets qui peuvent s'occulter.
- Représentation de l'arrière plan par un vecteur et de chaque objet par sa position, sa taille, sa profondeur et un vecteur.
- Permet de segmenter des objets de façon non supervisée sous réserve de disposer d'un dataset où ces objets apparaissent à des positions variables.

Avantages et inconvénients des représentations structurées

✓ La représentation est directement interprétable et utilisable

✗ Beaucoup moins général et flexible que les représentations non structurées.

Sommaire

Introduction

Apprentissage de la distribution des données

Apprentissage de représentations

Modèles génératifs

La notion d'échantillonnage

objectif

générer de façon aléatoire des échantillons x_i' indépendants qui suivent la même distribution que les données d'entrée x_i

pour quoi faire ?

- générer des images, du son, du texte
- Les méthodes d'échantillonnages aléatoires développées de façon non supervisée peuvent généralement facilement être adaptées à une génération conditionnelle (prédiction structurée).
- En grande dimension, $p(x)$ est inutilisable pour de nombreuses applications.
  exemple : estimation d'une espérance en dimension 1000 :
  $\mathbb{E}(f(x_1, ...x_{1000})) = \int \int ... \int \int f(x_1, .., x_{1000}) p(x_1, .., x_{1000}) dx_1 .. dx_{1000}$
  Calcul d'espérance immédiat par la méthode de Monte-Carlo avec des échantillons $\mathbb{E}(f(x)) \simeq \frac{1}{N} \sum_i^N f(x_i')$

Le problème de l'évaluation des modèles génératifs

Comment évaluer la qualité d'un modèle génératif ?

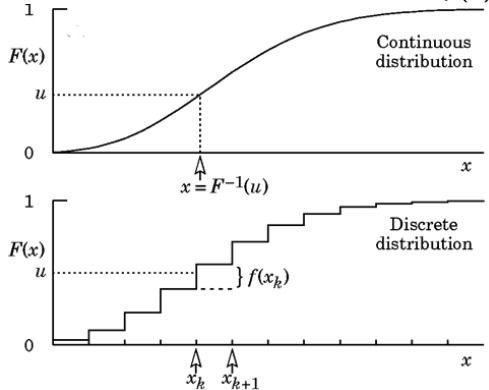
- Evaluer la qualité des images générées
- Vérifier qu'il ne se contente pas de recopier des données d'entraînement
- Vérifier que tous les modes de la distribution sont bien générés

Pas de méthode universelle
Principal critère utilisé dans le domaine de la vision : Frechet Inception Distance (FID)
On compare les moyennes μ et les matrices de covariance Σ du vecteur produit par un modèle "Inception" entraîné sur Imagenet lorsque ce modèle est appliqué au dataset d'entrée et les échantillons produits par le GAN.

$$FID = \|\mu_{data} - \mu_{fake}\|^2 + tr(\Sigma_{data} + \Sigma_{fake} + (\Sigma_{fake}^{\frac{1}{2}} \Sigma_{data} \Sigma_{fake}^{\frac{1}{2}})^{\frac{1}{2}})$$

L'échantillonnage en utilisant $p(x)$ est simple en dimension 1, mais ce n'est pas le cas pour les dimensions supérieures.

- facile en dimension 1 si l'on connait p(x)



- Pas de méthode simple en dimension > 1.

Modèles autorégressifs

Toute distribution conjointe peut s'écrire comme le produit de distributions de dimension 1.
$p(x_0, x_1, ..., x_n) = p(x_0) \prod_{i=1}^{n} p(x_i \mid x_{i-1}, x_{i-2}, .., x_0)$
Pour obtenir un échantillon $(x_0, x_1, .., x_n)$ il suffit donc de :

- échantillonner x_0 suivant $p(x_0)$
- évaluer la distribution $p(x_1 \mid x_0)$
- échantillonner x_1 suivant la distribution $p(x_1 \mid x_0)$
- évaluer la distribution $p(x_2 \mid x_1, x_0)$
- échantillonner x_1 suivant la distribution $p(x_2 \mid x_1, x_0)$
- etc.

Applications des modèles autorégressifs

Modèles de language :

- x_i représente un mot ou une partie d'un mot.
- Modélisation des $p(x_i \mid x_{i-1}, x_{i-2}, .., x_0)$ par un transformer decoder unique entrainé sur un large corpus de textes.

Avantages et inconvénients des modèles autorégressifs

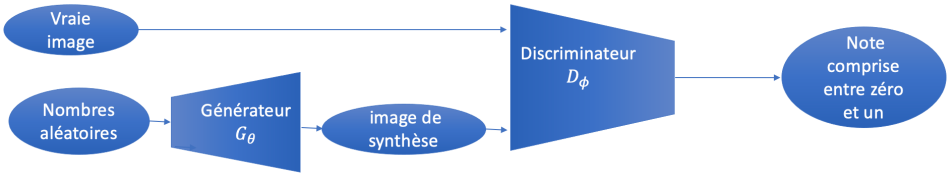
- ✓ résultats spectaculaires obtenus par les modèles de language autorégressifs
- ✓ qualité des représentations obtenues
- ✓ estimation de densité

- ✗ Processus d'échantillonnage séquentiel inadapté pour la génération d'images : une inférence nécessaire pour chaque x_i, impossible à paralléliser.

comprendre ChatGPT

- Reinforcement learning with human feedback
- supervised fine-tuning
- self-supervised pretraining

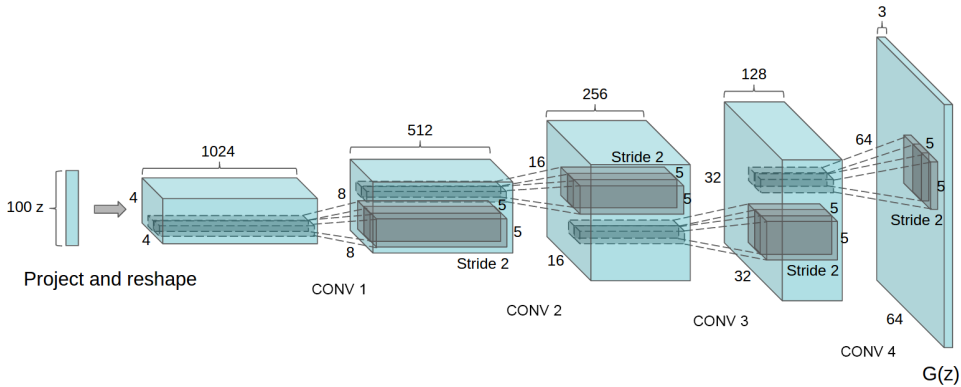
Réseaux antagonistes génératifs (GAN,2014)



optimisation : $\min_\theta \max_\phi \mathbb{E}_{x\sim p_{data}}[\log(D_\phi(x))] + \mathbb{E}_{h\sim\mathcal{N}}[\log(1 - D_\phi(G_\theta(h))]$

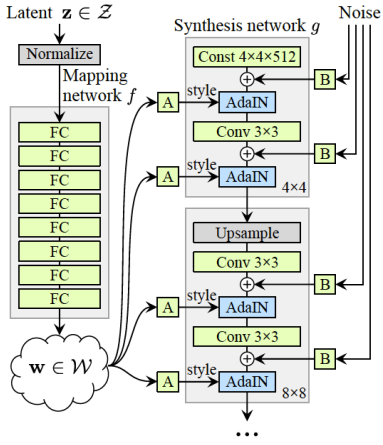
exemple d'architecture : DCGAN (2016)

modèle comportant essentiellement des couches de convolution transposée pour le générateur et des couches de convolution pour le discriminateur.

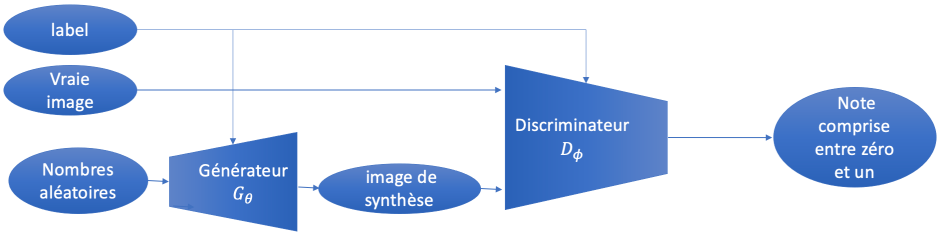


exemple d'architecture : StyleGAN (2018)

Les moyennes et variances des valeurs des features maps correspondent au style global de l'image.

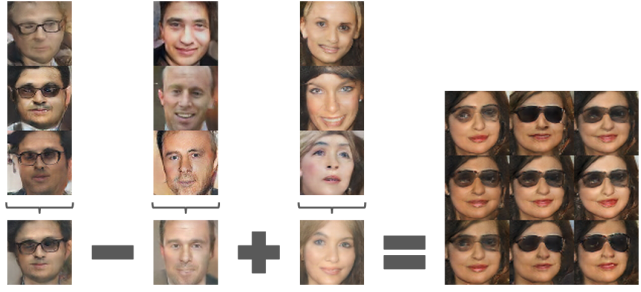


GAN conditionnel



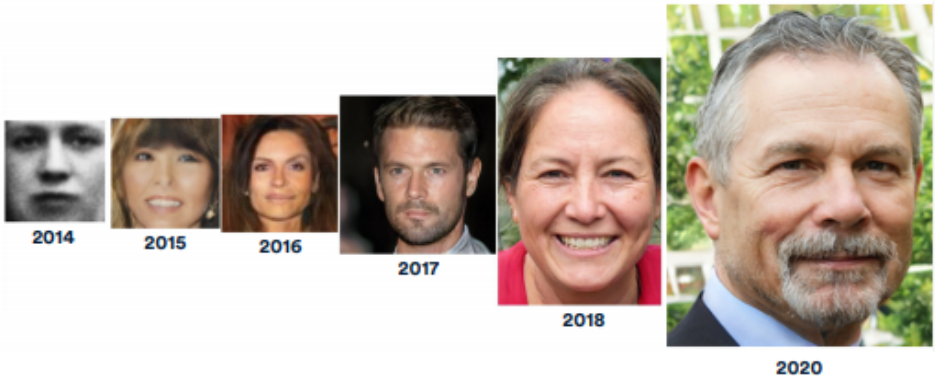
Espace latent et manipulations d'image

Certaines directions dans l'espace latent peuvent être associées à des transformations interprétables.



Avantages et inconvénients des GAN

- ✓ Excellente qualité pour les images
- ✓ Inférence rapide



- ✗ Entraînement peut être instable.
- ✗ Certains modes peuvent ne pas être générés.

Auto-encodeurs variationnels (VAE, 2013)

- générateur stochastique $p_\theta(x \mid z)$ conditionné à des variables latentes qui suivent une distribution fixée $p(z)$. On suppose que $p_\theta(x \mid z)$ est facile à échantillonner. $p_\theta(x) = \int p_\theta(x, z)dz = \int p_\theta(x \mid z)p(z)dz$
- On introduit aussi un encodeur stochastique $q_\phi(z \mid x)$ qui va être amené à modéliser la probabilité $p_\theta(z \mid x)$.

Distance de Kullback-Leibler

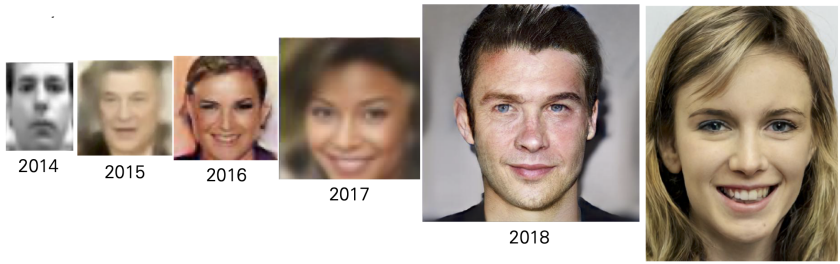
- Soit deux distributions de probabilité $q(x)$ et $p(x)$, alors
  $D_{KL}(q,p) = \int q(x) \log \frac{q(x)}{p(x)} dx \geq 0$
- Preuve :
  Le logarithme est concave,
  donc $\mathbb{E}[\log(x)] \leq \log(\mathbb{E}[x])$,
  donc $\int q(x) \log \frac{p(x)}{q(x)} dx = \mathbb{E}_q[\log \frac{p(x)}{q(x)}] \leq \log(\mathbb{E}_q[\frac{p(x)}{q(x)}]) = \log(\int q(x) \frac{p(x)}{q(x)} dx) = 0$
- $D_{KL}(p,q)$ peut être calculé explicitement si q et p sont des gaussiennes.
  $D_{KL}(p,q) = \log(\frac{\sigma_2}{\sigma_1}) + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\sigma_2^2} - \frac{1}{2}$

Auto-encodeurs variationnels - entraînement

- $p_\theta(x,z) = p_\theta(x \mid z)p(z) = p_\theta(z \mid x)p_\theta(x)$, donc $p_\theta(x) = \frac{p_\theta(x|z)p(z)}{p_\theta(z|x)} \frac{q_\phi(z|x)}{q_\phi(z|x)}$
- $\log(p_\theta(x)) = \log(p_\theta(x|z)) - \log(\frac{q_\phi(z|x)}{p(z)}) + \log(\frac{q_\phi(z|x)}{p_\theta(z|x))})$
  et en prenant l'espérance pour $z \sim q_\phi(z|x)$
- $\log(p_\theta(x)) = \mathbb{E}[\log(p_\theta(x|z))] - \mathbb{E}[\log(\frac{q_\phi(z|x)}{p(z)})] + \mathbb{E}[\log(\frac{q_\phi(z|x)}{p_\theta(z|x))})]$
- On note
  $\mathcal{L}(x) = \mathbb{E}[\log(p_\theta(x|z))] - \mathbb{E}[\log(\frac{q_\phi(z|x)}{p(z)})] = \mathbb{E}[\log(p_\theta(x|z))] - D_{KL}(q_\phi(z \mid x), p(z))$
- Ajuster ϕ pour maximizer $\mathcal{L}(x)$ équivaut à minimiser
  $\mathbb{E}[\log(\frac{q_\phi(z|x)}{p_\theta(z|x))})] = D_{KL}(q_\phi(z \mid x), p_\theta(z \mid x))$.
- Donc ajuster θ et ϕ pour maximiser $\sum_{i=1}^{N} \mathcal{L}(x)$ permet à la fois de maximiser $\mathbb{E}(\log(p_\theta(x))$ et d'avoir $q_\phi(z \mid x) \simeq p_\theta(z|x)$.

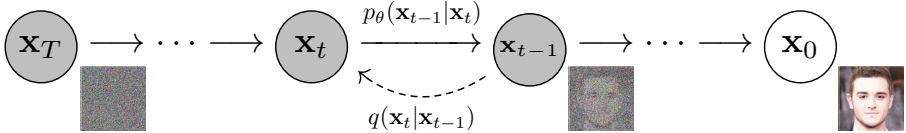
Auto-encodeurs variationnels - avantages et inconvénient

- ✓ Comporte un encodeur.
- ✓ Bonne qualité des variables latentes
- ✓ Cadre théorique extrêmement général, nombreuses déclinaisons possibles
- ✓ Convergence ne pose pas de difficulté
- ✗ qualité des images obtenues inférieure aux GANs et modèles de diffusion



Modèles de diffusion (2015)

- Dégradation progressive d'une image en y ajoutant du bruit.
  $q(x_t \mid x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I_n)$
  $x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}\epsilon,\ \epsilon \sim \mathcal{N}(0, I_n)$



- apprentissage d'un processus inverse de la forme:
  $p_\theta(x_{t-1} \mid x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$
- même principe d'optimisation que pour un auto-encodeur variationel, mais encodeur q fixe et pas de réduction de dimensionnalité

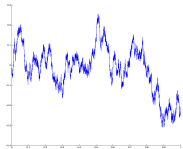
Modèles de diffusion et chaînes de Markov

- La séquence d'images $(x_0, x_1, ..x_T)$ forme une chaîne de Markov :
  $p(x_t \mid x_{t-1}, .., x_0) = p(x_t \mid x_{t-1})$
- Si une sequence de variable aléatoire $(x_0, x_1, .., x_T)$ forme une chaîne de Markov, alors la sequence inverse $(x_T, x_{T-1}, .., x_0)$ est aussi une chaîne de Markov.
  Preuve :
  $p(x_{t-1} \mid x_t, x_{t+1}, .., x_T)$
  $= \frac{p(x_{t-1}, x_t, x_{t+1}, .., x_T)}{p(x_t, x_{t+1}, ..x_T)}$
  $= \frac{p(x_{t-1}, x_t) p(x_{t+1}, ..x_T \mid x_{t-1}, x_t)}{p(x_t) p(x_{t+1}, ..x_T \mid x_t)}$
  $= p(x_{t-1} \mid x_t)$

On peut donc factoriser et échantillonner de deux façons la distribution conjointe
$p(x_0, x_1, .., x_T) = p(x_0)p(x_1 \mid x_0)..p(x_T \mid x_{T-1}) = p(x_T)p(x_{T-1} \mid x_T)..p(x_0 \mid x_1)$

Modèles de diffusion et équations différentielles stochastiques

- Si l'on suppose T très grand (donc β_t très petit), alors les équations $x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}\epsilon$, $\epsilon \sim \mathcal{N}(0, I_n)$ peuvent être considérées comme la discrétisation d'une équation différentielle stochastique $dx = -\frac{1}{2}\beta_t x dt + \sqrt{\beta_t}dw$



exemple de mouvement brownien

- Si un processus $(X_t)_{0\leq t\leq T}$ est solution d'une équation différentielle stochastique du type $dx = f(x,t)dt + \sigma(t)dw$, alors le processus inverse est solution de l'équation différentielle stochastique $dx = (f(x,t) - \sigma(t)^2\nabla_x \log p_t(x))dt + \sigma(t)d\bar{w}$.
- Il est donc légitime pour T très grand de modéliser le processus inverse comme une séquence de distributions gaussiennes.

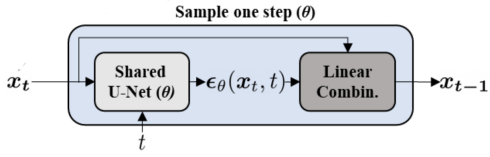
Entraînement d'un modèle de diffusion (DDPM, 2020)

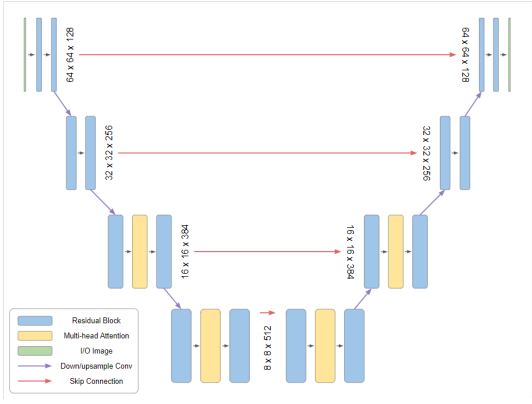
- matrices de covariance $\Sigma_\theta(x_t, t) = \sigma_t^2 I_n$ fixées à l'avance.
- espérance $\mu_\theta(x_t, t)$
  On peut décrire x_t and fonction de x_0 comme
  $x_t = \sqrt{1-\bar{\beta}_t}x_0 + \sqrt{\bar{\beta}_t}\epsilon_t$, $\epsilon_t \sim \mathcal{N}(0, I_n)$ avec $\bar{\beta}_t = 1 - \prod_{s=1}^{t}(1-\beta_t)$

  L'entraînement se fait en demandant au modèle de prédire ϵ_t en fonction de x_t
  $\min \mathbb{E}_{x_0,t,\epsilon}[\gamma_t \|\epsilon - \epsilon_\theta(\sqrt{1-\bar{\beta}_t}x_0 + \sqrt{\bar{\beta}_t}\epsilon, t)\|^2]$

  L'espérance de x_{t-1} sachant x_t et ϵ_t (donc x_0) est :
  $\mu_\theta(x_t, t) = \frac{1}{\sqrt{1-\beta_t}}(x_t - \frac{\beta_t}{\sqrt{\bar{\beta}_t}}\epsilon_\theta(x_t, t))$

Architecture d'un modèle de diffusion





architecture d'un U-net

Modèles de diffusion - principales variantes

- denoising diffusion implicit models (DDIM,2020) : En utilisant un processus de diffusion non markovien, il est possible de diminuer le nombre d'étapes nécessaires pour l'échantilllonnage et de remplacer le générateur stochastique par un générateur déterministe une fois que x_T a été échantillonné.
- diffusion dans l'espace latent (2022) : réduction de dimensionnalité grâce à un autoencodeur, puis modèle de diffusion dans l'espace latent.

Avantages et inconvénients de modèles de diffusion

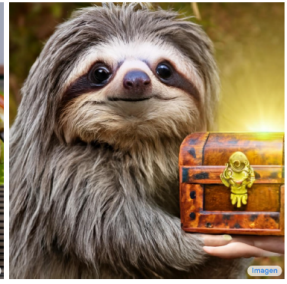
- ✓ entraînement stable
- ✓ résultats de grande qualité
- ✗ temps de calcul pour l'entraînement et l'inférence
- ✗ pas de variables latentes

Modèles mélant génération autogressive et modèle de diffusion: Imagen (2022)

- Objectif : générer des images à partir de texte
- Utilisation d'un modèle de language (T5) gelé
- génération des images par un modèle de diffusion conditionné aux représentations produites par le modèle de language

Teddy bears swimming at the Olympics 400m Butterfly event.

A cute corgi lives in a house made out of sushi.



A cute sloth holding a small treasure chest. A bright golden glow is coming from the chest.

images générées par Imagen à partir de textes

Questions ?