RCP211 – Modèles génératifs

Motivations – lien avec l'estimation de densité – autoencodeurs

---

Arnaud Breloy arnaud.breloy@lecnam.net

28 octobre 2025

Conservatoire national des arts & métiers

Table of contents

1. Motivations

2. Lien avec l’estimation de densité

3. Chaînes de Markov

4. Auto-encodeurs

Motivations

Modèle prédictif/modèle génératif

Modélisation décisionnelle : prédire une variable y à partir d'une observation $\mathbf{x}$

$\Longrightarrow$ modélisation de la probabilité conditionnelle $p(y|\mathbf{x})$

- *Exemple :* connaissant une image, quelle est la race du chien représenté ?

Modèle génératif : produire une observation $\tilde{\mathbf{x}}$ qui ressemble aux observations $\mathbf{x}$

$\Longrightarrow$ modélisation de la probabilité $p(\mathbf{x})$

- *Exemple :* connaissant un ensemble d'images de chien, produire une nouvelle image de chien.

Génération de texte

Produire du texte qui *ressemble à*, produire du texte à partir d'une *accroche* ou de voix.

Applications

- Agents conversationnels
- Traduction automatique
- *Reporting*
- *Marketing*

☹ Propagande

☹ Désinformation

Exemples : BERT, GPT, LLaMA, Whisper…

$\Longrightarrow$ RCP217

Synthèse vocale

Produire un son, généralement conditionné à du texte ou à un autre son.

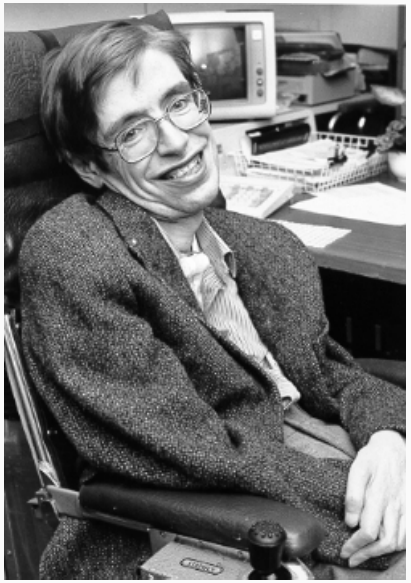
Applications

- *Text-to-speech*
- Robots « parlants »
- Doublage
- Jeux vidéo
- Musique

☹ Usurpation d'identité

☹ Désinformation

Exemples : MusicLM, WaveNet, PaddleSpeech.



Synthèse d’image

Produire une image qui *ressemble à*, qui *contient X*

Applications

- Effets spéciaux
- Simulations
- Retouche d’image
- ☹ Usurpation d’identité
- ☹ Désinformation

https://thispersondoesnotexist.com/

Exemples : Dall-E, Midjourney, StableDiffusion.

https://openai.com/blog/dall-e/

Cadre formel de la modélisation générative

Considérons deux variables aléatoires X et Y :

- X est la variable explicative (l'observation),
- Y est la variable à expliquer (la classe).

Modèle discriminatif

On cherche à déterminer les valeurs que peut prendre Y en fonction de X, c'est-à-dire la probabilité conditionnelle :

$$\mathbb{P}(Y|X)$$

Connaissant X, quelle sont les valeurs probables pour Y?

En pratique

X et Y sont généralements à valeurs dans un ensemble discret (mais de grande dimension).

Cadre formel de la modélisation générative

Considérons deux variables aléatoires X et Y :

- X est la variable explicative (l'observation),
- Y est la variable à expliquer (la classe).

Modèle discriminatif

On cherche à déterminer les valeurs que peut prendre Y en fonction de X, c'est-à-dire la probabilité conditionnelle :

$$\mathbb{P}(Y|X)$$

Connaissant X, quelle sont les valeurs probables pour Y?

En pratique

X et Y sont généralements à valeurs dans un ensemble discret (mais de grande dimension).

Cadre formel de la modélisation générative

Considérons deux variables aléatoires X et Y :

- X est la variable explicative (l'observation),
- Y est la variable à expliquer (la classe).

Modèle discriminatif

On cherche à déterminer les valeurs que peut prendre Y en fonction de X, c'est-à-dire la probabilité conditionnelle :

$$\mathbb{P}(Y|X)$$

Connaissant X, quelle sont les valeurs probables pour Y?

En pratique

X et Y sont généralements à valeurs dans un ensemble discret (mais de grande dimension).

Un exemple

Considérons le jeu de données suivant :

x_1	x_2	y
0.5	-1.	0
0.0	-1.2	0
-0.5	0.0	1

Probabilités conditionelles

x_1	x_2	$\mathbb{P}(y=0\|x_1,x_2)$	$\mathbb{P}(y=1\|x_1,x_2)$
0.5	-1.	1.0	0.0
0.0	-1.2	1.0	0.0
-0.5	0.0	0.0	1.0

Modèle discriminatif paramétrisé par θ

- minimisation de la KL entre les prédictions $\hat{Y} = \mathbb{P}_\theta(Y|X)$ et $\mathbb{P}(Y|X)$

$$\mathcal{L}_{\mathrm{KL}} = \mathbb{P}(y=0|X)\log\frac{\mathbb{P}(y=0|X)}{\mathbb{P}(\hat{y}=0|X)} + \mathbb{P}(y=1|X)\log\frac{\mathbb{P}(y=1|X)}{\mathbb{P}(\hat{y}=1|X)}$$

Limites des modèles discriminatifs

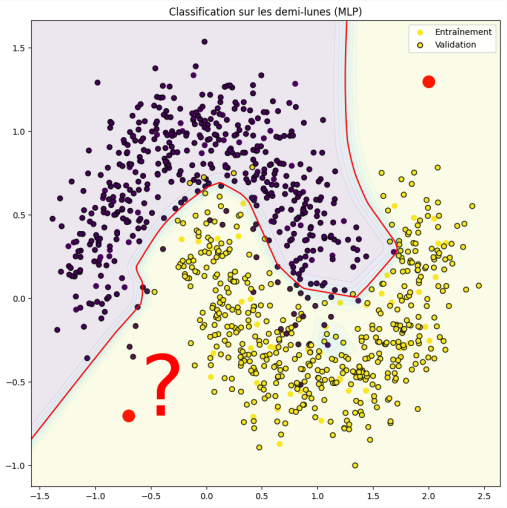
- Le modèle discriminatif apprend directement $\mathbb{P}(Y|X)$
- On classe les observations sans savoir « pourquoi »

Limite

Avec un modèle discriminatif, il est impossible de répondre à la question :

À quoi ressemble une observation de la classe y?

Le modèle discriminatif apprend les frontières entre classes, mais **pas** la forme des classes.



Modèle génératif

Considérons deux variables aléatoires X et Y :

- X est la variable explicative (l'observation),
- Y est la variable à expliquer (la classe).

Modèle génératif

On cherche à déterminer quelles valeurs X sont susceptibles d'avoir provoqué Y, c'est-à-dire :

$$\mathbb{P}(X|Y)$$

ou, si $\mathbb{P}(Y)$ est connue, la probabilité conjointe d'avoir X et Y :

$$\mathbb{P}(X, Y) = \mathbb{P}(X|Y) \cdot \mathbb{P}(Y) \ \ (= \mathbb{P}(Y|X) \cdot \mathbb{P}(X))$$

Modèle génératif

Considérons deux variables aléatoires X et Y :

- X est la variable explicative (l'observation),
- Y est la variable à expliquer (la classe).

Modèle génératif

On cherche à déterminer quelles valeurs X sont susceptibles d'avoir provoqué Y, c'est-à-dire :

$$\mathbb{P}(X|\, Y)$$

ou, si $\mathbb{P}(\, Y)$ est connue, la probabilité conjointe d'avoir X et Y :

$$\mathbb{P}(X,\, Y) = \mathbb{P}(X|\, Y) \cdot \mathbb{P}(\, Y) \;\; (= \mathbb{P}(\, Y|X) \cdot \mathbb{P}(X))$$

Lien entre modèle discriminatif et modèle génératif

On peut transformer un modèle génératif en modèle discriminatif en utilisant le **théorème de Bayes** :

Formule de Bayes

$$\mathbb{P}(Y|X) = \frac{\mathbb{P}(X|Y) \cdot \mathbb{P}(Y)}{\mathbb{P}(X)}$$

Dans cette formulation, $\mathbb{P}(Y)$ est l'a priori bayésien.

Transformation

Si je connais une approximation de $\mathbb{P}(X|Y)$ alors je peux construire le classifieur :

$$\arg\max_i \mathbb{P}(Y = y_i|X) = \arg\max_i \mathbb{P}(X|Y = y_i) \cdot \mathbb{P}(Y = y_i)$$

$\implies$ $\mathbb{P}(X)$ ne dépend pas de y_i et n'intervient pas dans l'$\arg\max$!

Lien entre modèle discriminatif et modèle génératif

On peut transformer un modèle génératif en modèle discriminatif en utilisant le **théorème de Bayes** :

Formule de Bayes

$$\mathbb{P}(Y|X) = \frac{\mathbb{P}(X|Y) \cdot \mathbb{P}(Y)}{\mathbb{P}(X)}$$

Dans cette formulation, $\mathbb{P}(Y)$ est l'a priori bayésien.

Transformation

Si je connais une approximation de $\mathbb{P}(X|Y)$ alors je peux construire le classifieur :

$$\arg\max_i \mathbb{P}(Y = y_i|X) = \arg\max_i \mathbb{P}(X|Y = y_i) \cdot \mathbb{P}(Y = y_i)$$

$\Longrightarrow$ $\mathbb{P}(X)$ ne dépend pas de y_i et n'intervient pas dans l'$\arg\max$!

Un exemple

4 points :

x	y
$1.$	0
$1.$	0
$2.$	0
$2.$	1

Probabilité d'appartenir à la classe y sachant x : $\mathbb{P}(Y|X)$

	$y=0$	$y=1$
$x=1.0$	$1.$	$0.$
$x=2.0$	0.5	0.5

Probabilité d'avoir x **et** y : $\mathbb{P}(X, Y) = \mathbb{P}(Y|X)\mathbb{P}(X)$

	$y=0$	$y=1$
$x=1.0$	0.5	0.
$x=2.0$	0.25	0.25

Avantages des modèles génératifs

Génération

La connaissance de $\mathbb{P}(X, Y)$ permet de produire de nouvelles données en échantillonnant dans la distribution jointe.

Compréhension

La connaissance de $\mathbb{P}(X|\, Y)$ permet de comprendre quelles données x sont les plus plausibles pour une catégorie y donnée.

Combinaison

Si j'ajoute une classe y_{n+1}, il me suffit d'estimer $\mathbb{P}(X|\, Y = y_{n+1})$ pour avoir la connaissance de $\mathbb{P}(X|\, Y \cup \{y_{n+1}\})$. Autrement dit, il est facile d'enrichir le modèle a posteriori.

Avantages des modèles génératifs

Génération

La connaissance de $\mathbb{P}(X, Y)$ permet de produire de nouvelles données en échantillonnant dans la distribution jointe.

Compréhension

La connaissance de $\mathbb{P}(X|\, Y)$ permet de comprendre quelles données x sont les plus plausibles pour une catégorie y donnée.

Combinaison

Si j'ajoute une classe y_{n+1}, il me suffit d'estimer $\mathbb{P}(X|\, Y = y_{n+1})$ pour avoir la connaissance de $\mathbb{P}(X|\, Y \cup \{y_{n+1}\})$. Autrement dit, il est facile d'enrichir le modèle a posteriori.

Avantages des modèles génératifs

Génération

La connaissance de $\mathbb{P}(X, Y)$ permet de produire de nouvelles données en échantillonnant dans la distribution jointe.

Compréhension

La connaissance de $\mathbb{P}(X|\, Y)$ permet de comprendre quelles données x sont les plus plausibles pour une catégorie y donnée.

Combinaison

Si j'ajoute une classe y_{n+1}, il me suffit d'estimer $\mathbb{P}(X|\, Y = y_{n+1})$ pour avoir la connaissance de $\mathbb{P}(X|\, Y \cup \{y_{n+1}\})$. Autrement dit, il est facile d'enrichir le modèle a posteriori.

Points de difficultés

Performance des modèles discriminatifs

Un modèle discriminatif estime directement $\mathbb{P}(Y|X)$. Cette approche directe tend en pratique à être plus simple et plus performante en classification.

Dimensionalité

En général, X est de grande dimension. Par conséquent, $\mathbb{P}(X|Y)$ peut être difficile à estimer dans un modèle génératif.

Quiz

Un modèle $\mathcal{M}$ renvoie pour une donnée une probabilité p. Ce modèle est-il génératif ou discriminatif ?

1. 1. Discriminatif
2. 2. Génératif
3. 3. On ne peut pas savoir.

Lien avec l’estimation de densité

Estimation de densité

Densité de probabilité

On appelle **densité de probabilité** de la variable aléatoire X à valeurs dans $\mathbb{R}^d$ une fonction f telle que, pour tout pavé $A \subset \mathbb{R}^d$:

$$\mathbb{P}(X \in A) = \int_A f(x)\mathrm{d}x$$

Estimation de densité

À partir de $(x_1, \ldots, x_n)$ observations de X, on cherche $\hat{f}$:

$$\|\hat{f} - f\| \leq \epsilon \ .$$

$\rightarrow$ on modélise $\mathbb{P}(X)$

$\rightarrow$ idem à un modèle génératif (sans l'aspect conditionnel)

Méthode des noyaux

Méthode des noyaux

Soient $x_1, \ldots, x_n$ des observations de X et Φ une fonction noyau.

On définit la densité approchée par :

$$x \to \hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} \Phi(x, x_i)$$

c'est-à-dire la somme des noyaux centrés sur chaque observation.

La méthode des noyaux donne une estimation de densité non-paramétrique :
$\to$ nombre de "paramètres" augmente avec la quantité de données

Le modèle de mélange gaussien : un modèle génératif

Hypothèse

La densité f recherchée est une somme de gaussiennes.

Modèle de mélange gaussien

On cherche une densité de la forme :

$$\hat{f}_{\alpha,\theta}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i \Phi_i(\mathbf{x}|\theta_i)$$

- Φ_i une loi normale paramétrisée par θ_i,
- α_i le poids de la composante i, avec $\sum_i \alpha_i = 1$,
- m le nombre de **composantes** du mélange.

Hypothèse raisonnable car pour toute variable aléatoire X, il existe une séquence $X_1, X_2, \ldots, X_n$ de mélange gaussien tel que $X_n \to f$.

Le modèle de mélange gaussien : un modèle génératif

Hypothèse

La densité f recherchée est une somme de gaussiennes.

Modèle de mélange gaussien

On cherche une densité de la forme :

$$\hat{f}_{\alpha,\theta}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i \Phi_i(\mathbf{x}|\theta_i)$$

- Φ_i une loi normale paramétrisée par θ_i,
- α_i le poids de la composante i, avec $\sum_i \alpha_i = 1$,
- m le nombre de **composantes** du mélange.

Modèle paramétrique : on cherche les paramètres θ_i des gaussiennes qui minimisent l'erreur. Leur nombre ne dépend pas du nombre d'observations.

Mélange gaussien : génération

Une fois les paramètres du mélange fixés, on peut échantillonner la nouvelle variable aléatoire $\hat{X}$ de probabilité :

$$\mathbb{P}(\hat{X}) = \sum_{i=1}^{m} \alpha_i \mathcal{N}(\mu_i, \sigma_i)$$

Il est également possible d'échantillonner sur une composante (= une classe) spécifique suivant la loi :

$$\mathbb{P}(X|\, Y = i) = \mathcal{N}(\mu_i, \sigma_i)$$

Mélange gaussien : classification

La classe de x est la valeur de i pour laquelle $\mathbb{P}(Y = i|X = x)$ est la plus élevée. Par le théorème de Bayes :

$$\begin{aligned}\arg\max_i \left[\log \mathbb{P}(Y = i|X = x)\right] &= \arg\max_i \left[\log \frac{\mathbb{P}(X = x|Y = i) \cdot \mathbb{P}(Y = i)}{\mathbb{P}(X = x)}\right] \\ &= \arg\max_i \left[\log \mathbb{P}(X = x|Y = i) + \log \mathbb{P}(Y = i)\right] \\ &= \arg\max_i \left[\log \Phi_i(x|\theta_i) + \log \alpha_i\right]\end{aligned} \tag{1}$$

où α_i et θ_i sont respectivement le poids et les paramètres de la i^{e} composante du mélange.

Interprétation

La classe de x est la composante pour laquelle x a la plus haute (log) vraisemblance.

Chaînes de Markov

Chaîne de Markov

Définition

Un **processus de Markov** (à temps discret) est une séquence $(X_i)_{1 \leq i \leq \dots}$ où X est une variable aléatoire qui prend ses valeurs dans un espace d'états E. On dit que X_n est l'état du processus à l'instant n.

Si $|E|$ est fini, on parle de **chaîne de Markov**.

Propriété de Markov

La prédiction du futur ne nécessite pas de connaître le passé, seulement le présent :

$$\mathbb{P}(X_{n+1} = j | X_0 = i_0, X_1 = i_1, \dots, X_n = i) = \mathbb{P}(X_{n+1} = j | X_n = i)$$

Chaîne de Markov

Définition

Un **processus de Markov** (à temps discret) est une séquence $(X_i)_{1 \leq i \leq \dots}$ où X est une variable aléatoire qui prend ses valeurs dans un espace d'états E. On dit que X_n est l'état du processus à l'instant n.

Si $|E|$ est fini, on parle de **chaîne de Markov**.

Propriété de Markov

La prédiction du futur ne nécessite pas de connaître le passé, seulement le présent :

$$\mathbb{P}(X_{n+1} = j | X_0 = i_0, X_1 = i_1, \dots, X_n = i) = \mathbb{P}(X_{n+1} = j | X_n = i)$$

Chaîne de Markov

Définition

Un **processus de Markov** (à temps discret) est une séquence $(X_i)_{1 \leq i \leq \dots}$ où X est une variable aléatoire qui prend ses valeurs dans un espace d'états E. On dit que X_n est l'état du processus à l'instant n.

Si $|E|$ est fini, on parle de **chaîne de Markov**.

Propriété de Markov

La prédiction du futur ne nécessite pas de connaître le passé, seulement le présent :

$$\mathbb{P}(X_{n+1} = j | X_0 = i_0, X_1 = i_1, \dots, X_n = i) = \mathbb{P}(X_{n+1} = j | X_n = i)$$

Modèle autorégressif analogue à AR(1)

Probabilités de transition

Homogénéité

Généralement, on suppose que la probabilité de passer d'un état i à un état j ne dépend pas du temps :

$$\mathbb{P}(X_{n+1} = j | X_n = i) = \mathbb{P}(X_n = j | X_{n-1} = i)$$

On note

$$p_{i,j} := \mathbb{P}(X_1 = j | X_0 = i)$$

la **probabilité de transition** de l'état i à l'état j.

Si n est entier, on peut construire $M = (p_{i,j})_{1 \leq i \leq n, 1 \leq j \leq n}$ la **matrice de transition**.

La chaîne de Markov comme modèle génératif

Estimation de la matrice de transition

À partir de séquences observées, on peut estimer la matrice de transition $\mathbb{P}(X_t = j | X_{t-1} = i)$.

- Approche fréquentiste
  - on compte les nombres d'occurrences des paires d'états (i, j)

Échantillonnage

Connaissant X_0 et M, on peut générer la séquence $X_1, X_2, \ldots, X_n$ la plus probable :

- $X_t = i$
- à chaque t, $X_{t+1} = \arg\max_j p_{i,j}$ (déterministe)
  - on peut aussi échantillonner de façon stochastique en pondérant selon $p_{i,j}$

Chaîne de Markov d'ordre k

On parle de chaîne de Markov d'ordre k lorsque la prédiction du futur ne nécessite pas de connaître plus de k pas de temps dans le passé :

$$\mathbb{P}(X_{n+1} = j | X_0 = i_0, \ldots, X_n = i) = \mathbb{P}(X_{n+1} = j | X_n = i, \ldots, X_{n-k} = i_{n-k}) \, .$$

Dans ce cas, la matrice de transition est multidimensionnelle et on cherche à estimer

$$p_{i_1, i_2, \ldots, i_k} = \mathbb{P}(X_t = i_k | X_{t-1} = i_{k-1}, X_{t-2} = i_{k-2}, \ldots, X_{t-k} = i_1) \, .$$

Attention

La complexité du problème est exponentielle selon k. Si le cardinal de l'espace d'états E est grand, la matrice de transition complète est énorme (on peut parfois s'en sortir si beaucoup de transitions sont impossibles, i.e. $p_{i,j} = 0$).

Génération de texte

Une phrase est une séquence de mots.

- X_i : le i^{e} mot de la phrase,
- E : n mots du dictionnaire,
- M : matrice de transition $n \times n$.

Modélisation par une chaîne de Markov

On suppose que l'occurrence d'un mot ne dépend que des k mots qui le précèdent (on s'intéresse aux transitions dans les « k-grammes »).

$\Longrightarrow$ chaîne de Markov d'ordre k

Exemple

Jeu de données (“corpus”)

- « Ali est ici. »
- « Ali aime le bleu. »
- « Le vélo d’Ali est bleu. »
- « Qui est Ali ? »

Vocabulaire : E = {Ali, est, ici, “.”, aime, le, bleu, vélo, d’, qui, “ ?”}

Bi-grammes commençant par *Ali* :

- “Ali, est” (2 fois),
- “Ali, aime” (1 fois),
- “Ali, ?” (1 fois).

$p_{i,j}$	Ali	est	ici	“.”	aime	le	bleu	vélo	d’	qui	“ ?”
Ali	0	0.5	0	0	0.25	0	0	0	0	0	0.25
…						…					

Quiz

Considérons le corpus où les suites de mots :

- (1) « j'aime l'intelligence artificielle »
- (2) « machine learning »

n'apparaîssent jamais.

En utilisant une chaîne de Markov (d'ordre 1), laquelle de ces propositions est vrai :

1. Je peux générer (1) et (2)
2. Je peux générer (1) mais pas (2)
3. Je peux générer (2) mais pas (1)
4. Je ne peux générer ni (1), ni (2)

Auto-encodeurs

Définition

Soit une variable aléatoire X à valeurs dans $\mathbb{R}^n$. Un réseau de neurones artificiels **auto-encodeur** modélise une fonction $\mathcal{H}$ telle que :

$$\|\mathcal{H}(x) - x\| \leq \epsilon$$

L'auto-encodeur se décompose en deux parties :

- un encodeur, i.e. une fonction $\mathcal{E} : \mathbb{R}^n \to \mathbb{R}^d$,
- un décodeur, i.e. une fonction $\mathcal{D} : \mathbb{R}^d \to \mathbb{R}^n$.

avec $\mathcal{H} = \mathcal{D} \circ \mathcal{E}$.

L'objectif de déterminer $\mathcal{E}$ et $\mathcal{D}$ tels que $\mathcal{D} \circ \mathcal{E} \approx \mathbf{Id}$.

Réduction de dimension

En principe, $d \leq n$: l'encodeur **réduit la dimension** de x.

On note $z = \mathcal{E}(x)$ le **code** associé à x.

Apprentissage

L'auto-encodeur $\mathcal{H}$ apprend à **reconstruire** l'entrée x. On cherche donc les poids θ tels que :

$$\theta^* = \arg\min_{\theta} \mathcal{L}(x, \hat{x}) = \|\mathcal{D}(\mathcal{E}(x)) - x\|$$

où $\mathcal{L}$ est une fonction de coût de régression (typiquement, erreur quadratique moyenne ou erreur absolue).

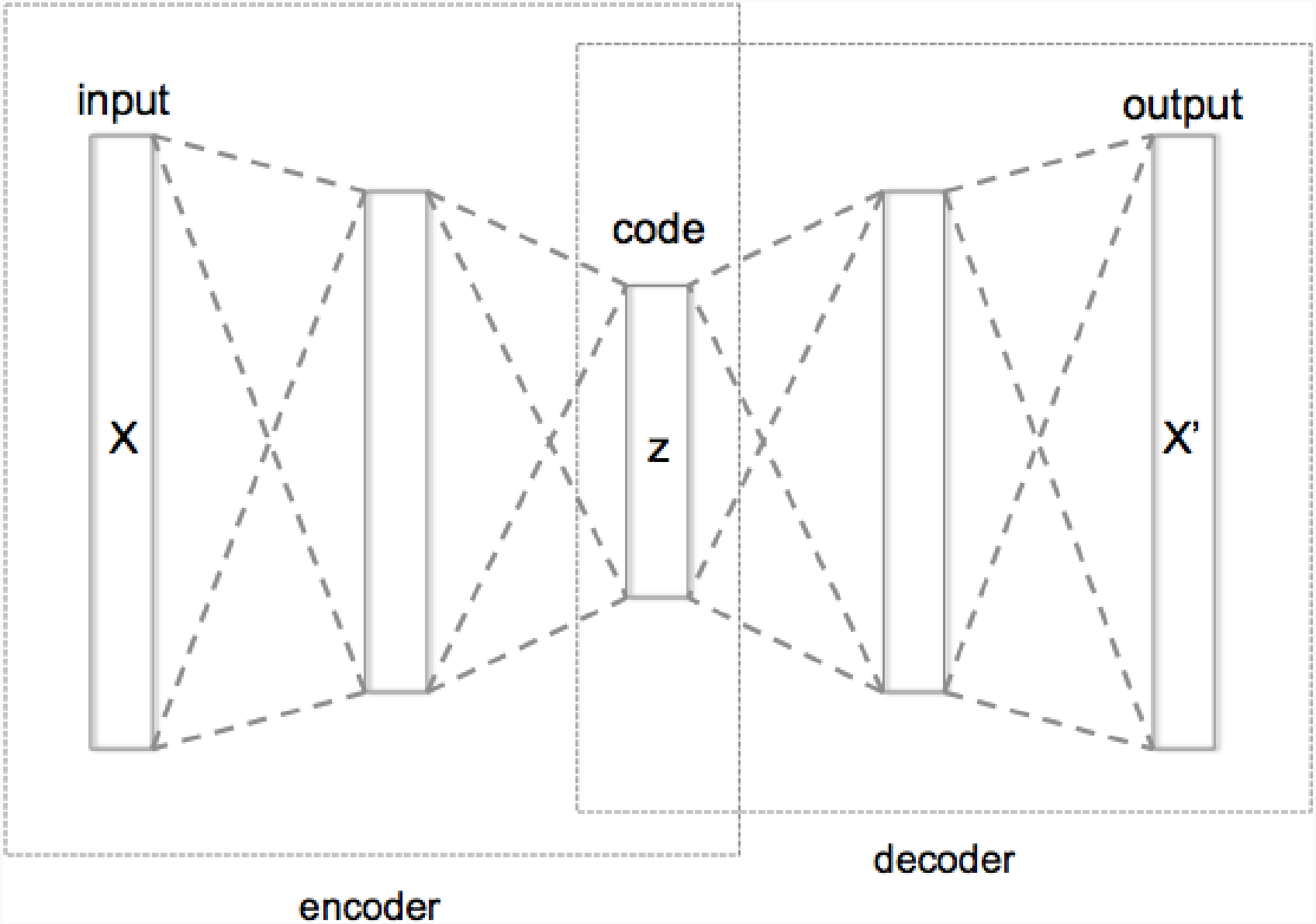
Encodeur

L'encodeur réduit la dimension de x.

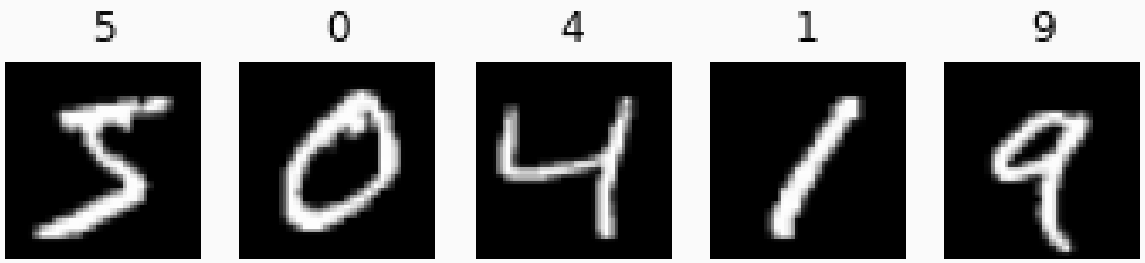
Décodeur

Le décodeur reconstruit x à partir du code réduit z.

Structure visuelle



Exemple sur MNIST

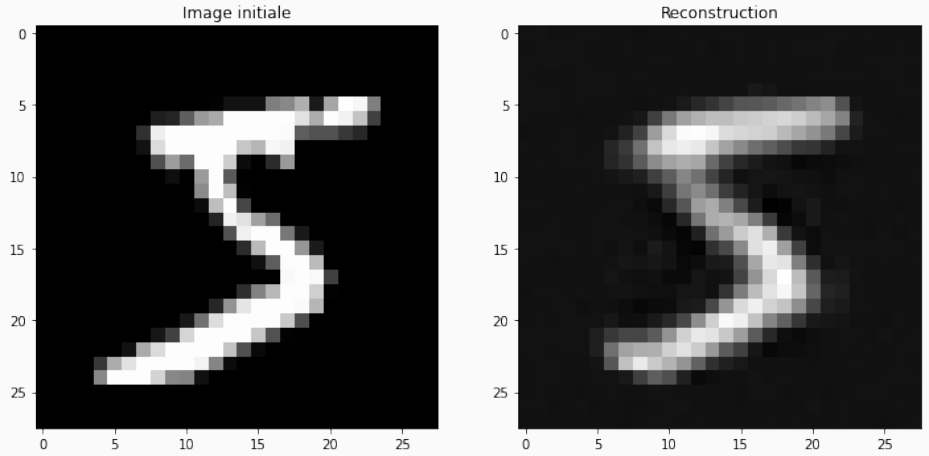


Encodeur :

- FC $(28 \times 28, 1024)$ + ReLU
- FC $(1024, 256)$ + ReLU
- FC $(256, 128)$

Décodeur :

- FC $(128, 256)$ + ReLU
- FC $(256, 1024)$ + ReLU
- FC $(1024, 28 \times 28)$



Auto-encodeur entièrement connecté à une couche

Le perceptron à une seule couche cachée forme un auto-encodeur simple :

$$\mathbf{z} = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b})$$

$$\widehat{\mathbf{x}} = \sigma'(\mathbf{W}'\mathbf{z} + \mathbf{b}')$$

avec W, W' les matrices de poids, b, b' les vecteurs de biais et σ la non-linéarité.

Lien avec l'analyse en composantes principales

Dans le cas où $\sigma = \mathrm{Id}$, alors l'auto-encodeur réalise une opération analogue à l'analyse en composantes principales (sans l'orthogonalité).

Intérêt des auto-encodeurs

Les auto-encodeurs réalisent une **réduction de dimension avec perte minimale d'information**.

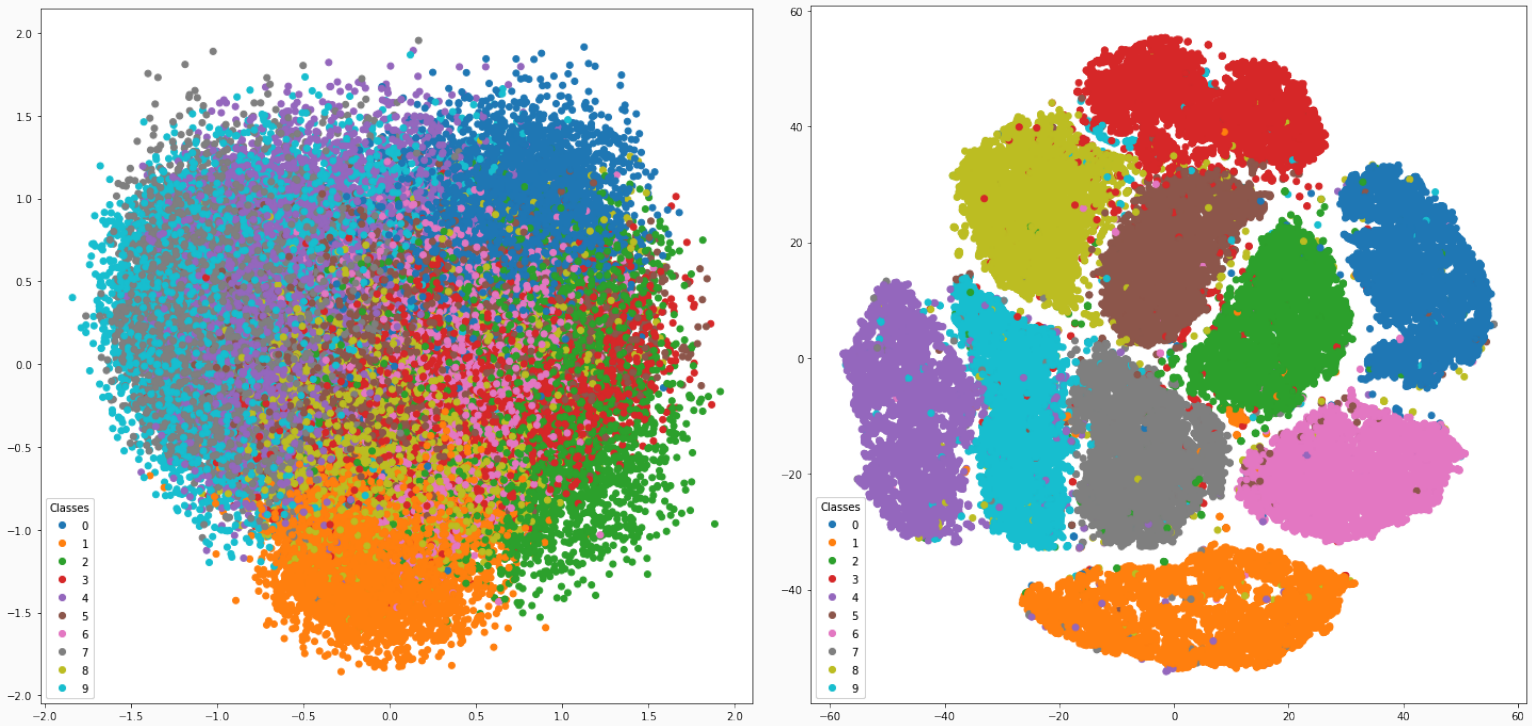
Comparaison avec l'analyse en composantes principales

- Si $\sigma = \mathrm{Id}$, AE $\approx$ ACP,
- Sinon, AE $\approx$ kernel-ACP où le noyau est appris automatiquement.

L'AE non-linéaire est plus riche que l'ACP.

Intérêt des auto-encodeurs

Les auto-encodeurs réalisent une **réduction de dimension avec perte minimale d'information**.



Comparaison d'une projection par ACP (à gauche) et par auto-encodeur + t-SNE (à droite) sur MNIST.

L'espace des codes $\mathcal{Z} = \mathbb{R}^d$ est appelé **espace latent**.

Génération

Le décodeur $\mathcal{D}$ est un modèle génératif $\mathbb{P}(X|z)$.

$\Longrightarrow$ Échantillonner dans $\mathcal{Z}$ permet de produire des observations $x \in \mathbb{R}^n$.

Espace latent

Définition

Un **espace latent** est un espace caché qui explique bien les données.

Des différences importantes dans l'espace des observations peuvent être expliquée par de faibles variations dans l'espace latent du fait de régularités (par exemple, même sémantique).

Exemples

Plongements lexicaux (*Word2Vec*...), *feature maps* d'un CNN, espace intermédiaire d'un auto-encodeur, projection par t-SNE.

Mélanges

Interpolation dans l'espace latent

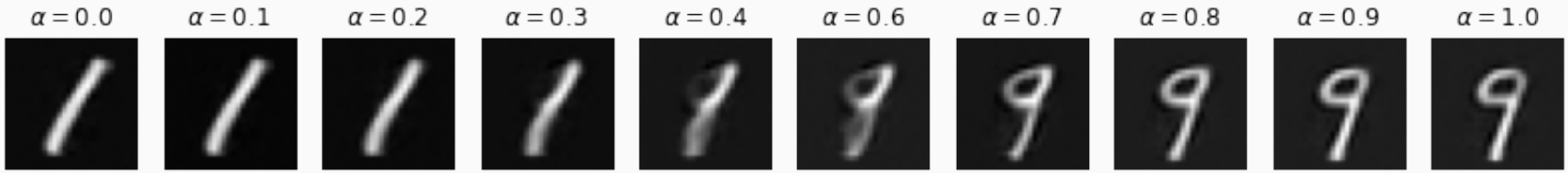
Considérons deux observations x_1 et x_2 et leurs codes z_1 et z_2. Elles peuvent être par exemple de classes différentes y_1 et y_2.

Il est possible de "reconstruire" une observation x à mi-chemin en reconstruisant le code moyen :

$$x_{\text{moy}} = \mathcal{D}\left(\frac{z_1 + z_2}{2}\right)$$

Plus généralement, il est possible d'interpoler linéairement entre x_1 et x_2 :

$$x_\alpha = \mathcal{D}\left(\alpha \cdot z_1 + (1 - \alpha) \cdot z_2\right)$$



Limites des auto-encodeurs

- On ne connaît pas la distribution $\mathbb{P}(\mathbf{z})$ qui sous-tend l'espace latent $\mathcal{Z}$
  - on ne peut donc pas échantillonner directement $P(X) = P(X|z) \cdot P(z)$
  - (on pourrait l'estimer)
- Quelle dimension d donner à $\mathcal{Z}$?
- L'encodeur n'est généralement pas injectif : deux observations $x_1 \neq x_2$ peuvent avoir des projections $z_1 \approx z_2$
- Aucune garantie que les codes z se situant dans des « trous » dans $\mathcal{Z}$ (zones de faible densité) aient du sens une fois décodés.

Limites des auto-encodeurs

- On ne connaît pas la distribution $\mathbb{P}(\mathbf{z})$ qui sous-tend l'espace latent $\mathcal{Z}$
  - on ne peut donc pas échantillonner directement $P(X) = P(X|z) \cdot P(z)$
  - (on pourrait l'estimer)
- Quelle dimension d donner à $\mathcal{Z}$?
- L'encodeur n'est généralement pas injectif : deux observations $x_1 \neq x_2$ peuvent avoir des projections $z_1 \approx z_2$
- Aucune garantie que les codes z se situant dans des « trous » dans $\mathcal{Z}$ (zones de faible densité) aient du sens une fois décodés.

Limites des auto-encodeurs

- On ne connaît pas la distribution $\mathbb{P}(\mathbf{z})$ qui sous-tend l'espace latent $\mathcal{Z}$
  - on ne peut donc pas échantillonner directement $P(X) = P(X|z) \cdot P(z)$
  - (on pourrait l'estimer)
- Quelle dimension d donner à $\mathcal{Z}$?
- L'encodeur n'est généralement pas injectif : deux observations $x_1 \neq x_2$ peuvent avoir des projections $z_1 \approx z_2$
- Aucune garantie que les codes z se situant dans des « trous » dans $\mathcal{Z}$ (zones de faible densité) aient du sens une fois décodés.

Limites des auto-encodeurs

- On ne connaît pas la distribution $\mathbb{P}(\mathbf{z})$ qui sous-tend l'espace latent $\mathcal{Z}$
  - on ne peut donc pas échantillonner directement $P(X) = P(X|z) \cdot P(z)$
  - (on pourrait l'estimer)
- Quelle dimension d donner à $\mathcal{Z}$?
- L'encodeur n'est généralement pas injectif : deux observations $x_1 \neq x_2$ peuvent avoir des projections $z_1 \approx z_2$
- Aucune garantie que les codes z se situant dans des « trous » dans $\mathcal{Z}$ (zones de faible densité) aient du sens une fois décodés.

Quiz

Quel est l'avantage principal d'un auto-encodeur par rapport à une analyse en composantes principales ?

1. L'auto-encodeur est plus rapide
2. L'auto-encodeur a une plus grande capacité
3. L'auto-encodeur a moins de paramètres
4. L'auto-encodeur a une meilleure reconstruction