

# NORMALIZING FLOWS INVERTIBLE NEURAL NETWORKS (INNs)

Jérémie Cabessa

Laboratoire DAVID, UVSQ

# INTRODUCTION

- ▶ On s'intéresse maintenant aux **modèles génératifs**.
- ▶ On possède des data d'apprentissages (e.g. des images) et on aimerait générer de nouvelles data "ressemblantes" à partir de ces dernières.
- ▶ **Idée générale:**
  1. On définit un réseau de neurones inversible qui transporte la distribution des data sur une loi normale centrée réduite.
  2. Pour générer de nouvelles data, on sample la loi normale et on applique la transformation inverse.

# INTRODUCTION

- ▶ On s'intéresse maintenant aux **modèles génératifs**.
- ▶ On possède des data d'apprentissages (e.g. des images) et on aimerait générer de nouvelles data “ressemblantes” à partir de ces dernières.
- ▶ **Idée générale:**
  1. On définit un réseau de neurones inversible qui transporte la distribution des data sur une loi normale centrée réduite.
  2. Pour générer de nouvelles data, on sample la loi normale et on applique la transformation inverse.

# INTRODUCTION

- ▶ On s'intéresse maintenant aux **modèles génératifs**.
- ▶ On possède des data d'apprentissages (e.g. des images) et on aimerait générer de nouvelles data “ressemblantes” à partir de ces dernières.
- ▶ **Idée générale:**
  1. On définit un réseau de neurones inversible qui transporte la distribution des data sur une loi normale centrée réduite.
  2. Pour générer de nouvelles data, on sample la loi normale et on applique la transformation inverse.

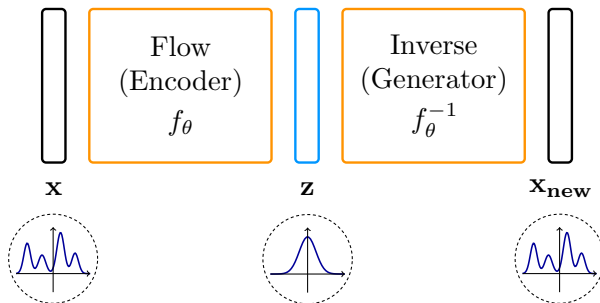
# INTRODUCTION

- ▶ On s'intéresse maintenant aux **modèles génératifs**.
- ▶ On possède des data d'apprentissages (e.g. des images) et on aimerait générer de nouvelles data “ressemblantes” à partir de ces dernières.
- ▶ **Idée générale:**
  1. On définit un réseau de neurones inversible qui transporte la distribution des data sur une loi normale centrée réduite.
  2. Pour générer de nouvelles data, on sample la loi normale et on applique la transformation inverse.

# INTRODUCTION

- ▶ On s'intéresse maintenant aux **modèles génératifs**.
- ▶ On possède des data d'apprentissages (e.g. des images) et on aimerait générer de nouvelles data “ressemblantes” à partir de ces dernières.
- ▶ **Idée générale:**
  1. On définit un réseau de neurones inversible qui transporte la distribution des data sur une loi normale centrée réduite.
  2. Pour générer de nouvelles data, on sample la loi normale et on applique la transformation inverse.

# INTRODUCTION



# INTRODUCTION

- ▶ Les concepts clés qui sous-tendent ces approches génératives sont:
  1. L'apprentissage d'une distribution de data.  
→ Density estimation
  2. Le transport de mesures.  
→ Transportation of measures



# INTRODUCTION

- ▶ Les concepts clés qui sous-tendent ces approches génératives sont:
  1. L'apprentissage d'une distribution de data.  
→ **Density estimation**
  2. Le transport de mesures.  
→ **Transportation of measures**

# INTRODUCTION

- ▶ Les concepts clés qui sous-tendent ces approches génératives sont:
  1. L'apprentissage d'une distribution de data.
    - **Density estimation**
  2. Le transport de mesures.
    - **Transportation of measures**

# DENSITY ESTIMATION

- Soit un dataset  $\mathcal{D} = \{\mathbf{x}_i \in \mathbb{R}^d : i = 1, \dots, n\}$  dont les éléments proviennent d'une distribution empirique inconnue  $p$ , i.e.,

$$\mathbf{x}_1, \dots, \mathbf{x}_n \sim p \quad (\text{i.i.d})$$

- *Apprendre la distribution  $p$  (density estimation)* signifie chercher un réseau de neurones  $p_\theta : \mathbb{R}^d \rightarrow \mathbb{R}$  de paramètres  $\theta$  tel que

$$p_\theta(\mathbf{x}) \simeq p(\mathbf{x}), \quad \forall x \in \mathbb{R}^d$$

- On cherche un réseau de neurones qui, pour tout élément  $\mathbf{x}$ , prédit la probabilité empirique  $p(\mathbf{x})$ .

# DENSITY ESTIMATION

- Soit un dataset  $\mathcal{D} = \{\mathbf{x}_i \in \mathbb{R}^d : i = 1, \dots, n\}$  dont les éléments proviennent d'une distribution empirique inconnue  $p$ , i.e.,

$$\mathbf{x}_1, \dots, \mathbf{x}_n \sim p \quad (\text{i.i.d})$$

- *Apprendre la distribution  $p$*  (density estimation) signifie chercher un réseau de neurones  $p_\theta : \mathbb{R}^d \rightarrow \mathbb{R}$  de paramètres  $\theta$  tel que

$$p_\theta(\mathbf{x}) \simeq p(\mathbf{x}), \quad \forall x \in \mathbb{R}^d$$

- On cherche un réseau de neurones qui, pour tout élément  $\mathbf{x}$ , prédit la probabilité empirique  $p(\mathbf{x})$ .

# DENSITY ESTIMATION

- Soit un dataset  $\mathcal{D} = \{\mathbf{x}_i \in \mathbb{R}^d : i = 1, \dots, n\}$  dont les éléments proviennent d'une distribution empirique inconnue  $p$ , i.e.,

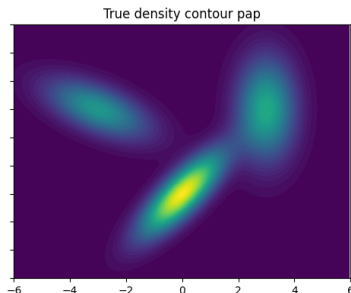
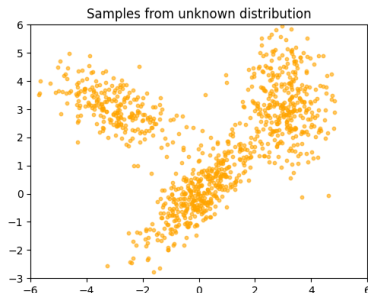
$$\mathbf{x}_1, \dots, \mathbf{x}_n \sim p \quad (\text{i.i.d})$$

- *Apprendre la distribution  $p$*  (density estimation) signifie chercher un réseau de neurones  $p_\theta : \mathbb{R}^d \rightarrow \mathbb{R}$  de paramètres  $\theta$  tel que

$$p_\theta(\mathbf{x}) \simeq p(\mathbf{x}), \quad \forall x \in \mathbb{R}^d$$

- On cherche un réseau de neurones qui, pour tout élément  $\mathbf{x}$ , prédit la probabilité empirique  $p(\mathbf{x})$ .

# DENSITY ESTIMATION



# DENSITY ESTIMATION

- ▶ On cherche un réseau de neurones  $p_\theta$  de paramètres  $\theta$  tel que

$$p_\theta(\mathbf{x}) \simeq p(\mathbf{x}), \forall x \in \mathbb{R}^d$$

- ▶ Pour cela, on cherche à minimiser la distance de Kullback-Leibler entre les lois  $p$  et  $p_\theta$

$$D_{\text{KL}}(p \parallel p_\theta) = \int_{-\infty}^{\infty} p(\mathbf{x}) \log \left( \frac{p(\mathbf{x})}{p_\theta(\mathbf{x})} \right) d\mathbf{x}$$

$$\begin{aligned} &= \int_{-\infty}^{\infty} p(\mathbf{x}) [\log p(\mathbf{x}) - \log p_\theta(\mathbf{x})] d\mathbf{x} \\ &= \mathbb{E}_p[\log p(\mathbf{x})] - \mathbb{E}_p[\log p_\theta(\mathbf{x})] \end{aligned}$$

# DENSITY ESTIMATION

- On cherche un réseau de neurones  $p_\theta$  de paramètres  $\theta$  tel que

$$p_\theta(\mathbf{x}) \simeq p(\mathbf{x}), \forall x \in \mathbb{R}^d$$

- Pour cela, on cherche à minimiser la distance de Kullback-Leibler entre les lois  $p$  et  $p_\theta$

$$\begin{aligned} D_{\text{KL}}(p \parallel p_\theta) &= \int_{-\infty}^{\infty} p(\mathbf{x}) \log \left( \frac{p(\mathbf{x})}{p_\theta(\mathbf{x})} \right) d\mathbf{x} \\ &= \int_{-\infty}^{\infty} p(\mathbf{x}) [\log(p(\mathbf{x})) - \log(p_\theta(\mathbf{x}))] d\mathbf{x} \\ &= \mathbb{E}_p[\log(p(\mathbf{x}))] - \mathbb{E}_p[\log(p_\theta(\mathbf{x}))] \end{aligned} \quad (1)$$



# DENSITY ESTIMATION

- On cherche un réseau de neurones  $p_\theta$  de paramètres  $\theta$  tel que

$$p_\theta(\mathbf{x}) \simeq p(\mathbf{x}), \forall \mathbf{x} \in \mathbb{R}^d$$

- Pour cela, on cherche à minimiser la distance de Kullback-Leibler entre les lois  $p$  et  $p_\theta$

$$\begin{aligned} D_{\text{KL}}(p \parallel p_\theta) &= \int_{-\infty}^{\infty} p(\mathbf{x}) \log \left( \frac{p(\mathbf{x})}{p_\theta(\mathbf{x})} \right) d\mathbf{x} \\ &= \int_{-\infty}^{\infty} p(\mathbf{x}) [\log(p(\mathbf{x})) - \log(p_\theta(\mathbf{x}))] d\mathbf{x} \\ &= \mathbb{E}_p[\log(p(\mathbf{x}))] - \mathbb{E}_p[\log(p_\theta(\mathbf{x}))] \end{aligned} \quad (1)$$

# DENSITY ESTIMATION

- On cherche un réseau de neurones  $p_\theta$  de paramètres  $\theta$  tel que

$$p_\theta(\mathbf{x}) \simeq p(\mathbf{x}), \forall x \in \mathbb{R}^d$$

- Pour cela, on cherche à minimiser la distance de Kullback-Leibler entre les lois  $p$  et  $p_\theta$

$$\begin{aligned} D_{\text{KL}}(p \parallel p_\theta) &= \int_{-\infty}^{\infty} p(\mathbf{x}) \log \left( \frac{p(\mathbf{x})}{p_\theta(\mathbf{x})} \right) d\mathbf{x} \\ &= \int_{-\infty}^{\infty} p(\mathbf{x}) [\log(p(\mathbf{x})) - \log(p_\theta(\mathbf{x}))] d\mathbf{x} \\ &= \mathbb{E}_p[\log(p(\mathbf{x}))] - \mathbb{E}_p[\log(p_\theta(\mathbf{x}))] \end{aligned} \quad (1)$$

# DENSITY ESTIMATION

- On cherche un réseau de neurones  $p_\theta$  de paramètres  $\theta$  tel que

$$p_\theta(\mathbf{x}) \simeq p(\mathbf{x}), \forall x \in \mathbb{R}^d$$

- Pour cela, on cherche à minimiser la distance de Kullback-Leibler entre les lois  $p$  et  $p_\theta$

$$\begin{aligned} D_{\text{KL}}(p \parallel p_\theta) &= \int_{-\infty}^{\infty} p(\mathbf{x}) \log \left( \frac{p(\mathbf{x})}{p_\theta(\mathbf{x})} \right) d\mathbf{x} \\ &= \int_{-\infty}^{\infty} p(\mathbf{x}) [\log(p(\mathbf{x})) - \log(p_\theta(\mathbf{x}))] d\mathbf{x} \\ &= \mathbb{E}_p[\log(p(\mathbf{x}))] - \mathbb{E}_p[\log(p_\theta(\mathbf{x}))] \end{aligned} \quad (1)$$

# DENSITY ESTIMATION

- Les paramètres  $\hat{\theta}$  qui minimisent  $D_{\text{KL}}(p \parallel p_{\theta})$  sont donc donnés par

$$\begin{aligned}\hat{\theta} &= \arg \min_{\theta} \left\{ D_{\text{KL}}(p \parallel p_{\theta}) \right\} \\ \text{by (1)} &= \arg \min_{\theta} \left\{ \mathbb{E}_p[\log(p(\mathbf{x}))] - \mathbb{E}_p[\log(p_{\theta}(\mathbf{x}))] \right\} \\ \text{term indep of } \theta &= \arg \min_{\theta} \left\{ -\mathbb{E}_p[\log(p_{\theta}(\mathbf{x}))] \right\} \quad (2)\end{aligned}$$

# DENSITY ESTIMATION

- Les paramètres  $\hat{\theta}$  qui minimisent  $D_{\text{KL}}(p \parallel p_{\theta})$  sont donc donnés par

$$\begin{aligned}\hat{\theta} &= \arg \min_{\theta} \left\{ D_{\text{KL}}(p \parallel p_{\theta}) \right\} \\ \text{by (1)} &= \arg \min_{\theta} \left\{ \mathbb{E}_p[\log(p(\mathbf{x}))] - \mathbb{E}_p[\log(p_{\theta}(\mathbf{x}))] \right\} \\ \text{term indep of } \theta &= \arg \min_{\theta} \left\{ -\mathbb{E}_p[\log(p_{\theta}(\mathbf{x}))] \right\}\end{aligned}\tag{2}$$

# DENSITY ESTIMATION

- Les paramètres  $\hat{\theta}$  qui minimisent  $D_{\text{KL}}(p \parallel p_{\theta})$  sont donc donnés par

$$\begin{aligned}\hat{\theta} &= \arg \min_{\theta} \left\{ D_{\text{KL}}(p \parallel p_{\theta}) \right\} \\ \text{by (1)} &= \arg \min_{\theta} \left\{ \mathbb{E}_p[\log(p(\mathbf{x}))] - \mathbb{E}_p[\log(p_{\theta}(\mathbf{x}))] \right\} \\ \text{term indep of } \theta &= \arg \min_{\theta} \left\{ -\mathbb{E}_p[\log(p_{\theta}(\mathbf{x}))] \right\}\end{aligned}\tag{2}$$

# DENSITY ESTIMATION

- Les paramètres  $\hat{\theta}$  qui minimisent  $D_{\text{KL}}(p \parallel p_{\theta})$  sont donc donnés par

$$\begin{aligned}\hat{\theta} &= \arg \min_{\theta} \left\{ D_{\text{KL}}(p \parallel p_{\theta}) \right\} \\ \text{by (1)} &= \arg \min_{\theta} \left\{ \mathbb{E}_p[\log(p(\mathbf{x}))] - \mathbb{E}_p[\log(p_{\theta}(\mathbf{x}))] \right\} \\ \text{term indep of } \theta &= \arg \min_{\theta} \left\{ -\mathbb{E}_p[\log(p_{\theta}(\mathbf{x}))] \right\}\end{aligned}\tag{2}$$

# DENSITY ESTIMATION

- ▶ On ne connaît pas la loi  $p$  mais on peut l'estimer de manière discrète à partir du dataset  $\mathcal{D}$

$$p(\mathbf{x}) = \begin{cases} 1/n & \text{si } \mathbf{x} = \mathbf{x}_i \text{ pour } i = 1, \dots, n \text{ (i.e. si } \mathbf{x} \in \mathcal{D}) \\ 0 & \text{sinon} \end{cases}$$

- ▶ Ainsi, l'espérance cherchée est estimée par

$$\mathbb{E}_p[\log(p_\theta(x))] \simeq \sum_{i=1}^n p(\mathbf{x}_i) \log(p_\theta(\mathbf{x}_i)) = \sum_{i=1}^n \frac{1}{n} \log(p_\theta(\mathbf{x}_i))$$

et les paramètres optimaux  $\hat{\theta}$  sont donc donnés par (by (2))

$$\hat{\theta} = \arg \min_{\theta} \left\{ - \sum_{i=1}^n \frac{1}{n} \log(p_\theta(\mathbf{x}_i)) \right\} \quad (3)$$



# DENSITY ESTIMATION

- ▶ On ne connaît pas la loi  $p$  mais on peut l'estimer de manière discrète à partir du dataset  $\mathcal{D}$

$$p(\mathbf{x}) = \begin{cases} 1/n & \text{si } \mathbf{x} = \mathbf{x}_i \text{ pour } i = 1, \dots, n \text{ (i.e. si } \mathbf{x} \in \mathcal{D}) \\ 0 & \text{sinon} \end{cases}$$

- ▶ Ainsi, l'espérance cherchée est estimée par

$$\mathbb{E}_p[\log(p_\theta(x))] \simeq \sum_{i=1}^n p(\mathbf{x}_i) \log(p_\theta(\mathbf{x}_i)) = \sum_{i=1}^n \frac{1}{n} \log(p_\theta(\mathbf{x}_i))$$

et les paramètres optimaux  $\hat{\theta}$  sont donc donnés par (by (2))

$$\hat{\theta} = \arg \min_{\theta} \left\{ - \sum_{i=1}^n \frac{1}{n} \log(p_\theta(\mathbf{x}_i)) \right\} \quad (3)$$

# DENSITY ESTIMATION

- On obtient alors la fonction de loss, appelée **negative log likelihood (NLL)** ou **negative log density**, qui permet d'apprendre la distribution empirique  $p$  (by (3))

$$\mathcal{L}(\theta) = -\frac{1}{n} \sum_{i=1}^n \log(p_{\theta}(\mathbf{x}_i)) = -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \log(p_{\theta}(\mathbf{x})) \quad (4)$$

- On peut donc entraîner un réseau de neurones multicouches  $p_{\theta}$  avec la loss  $\mathcal{L}(\theta)$  pour tout problème de density estimation.

# DENSITY ESTIMATION

- ▶ On obtient alors la fonction de loss, appelée **negative log likelihood (NLL)** ou **negative log density**, qui permet d'apprendre la distribution empirique  $p$  (by (3))

$$\mathcal{L}(\theta) = -\frac{1}{n} \sum_{i=1}^n \log(p_{\theta}(\mathbf{x}_i)) = -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \log(p_{\theta}(\mathbf{x})) \quad (4)$$

- ▶ On peut donc entraîner un réseau de neurones multicouches  $p_{\theta}$  avec la loss  $\mathcal{L}(\theta)$  pour tout problème de density estimation.

# DENSITY ESTIMATION

- ▶ En pratique, la density estimation via réseaux de neurones classiques et minimisation de la NLL fonctionne mal...
- ▶ Distributions multimodales difficiles à apprendre
- ▶ Distributions de haute dimension très difficiles à apprendre  
→ curse of dimensionality
- ▶ Il existe beaucoup d'autres méthodes très performantes.

# DENSITY ESTIMATION

- ▶ En pratique, la density estimation via réseaux de neurones classiques et minimisation de la NLL fonctionne mal...
- ▶ Distributions multimodales difficiles à apprendre
- ▶ Distributions de haute dimension très difficiles à apprendre  
→ curse of dimensionality
- ▶ Il existe beaucoup d'autres méthodes très performantes.

# DENSITY ESTIMATION

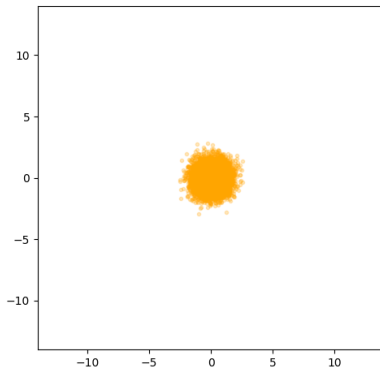
- ▶ En pratique, la density estimation via réseaux de neurones classiques et minimisation de la NLL fonctionne mal...
- ▶ Distributions multimodales difficiles à apprendre
- ▶ Distributions de haute dimension très difficiles à apprendre  
→ curse of dimensionality
- ▶ Il existe beaucoup d'autres méthodes très performantes.

# DENSITY ESTIMATION

- ▶ En pratique, la density estimation via réseaux de neurones classiques et minimisation de la NLL fonctionne mal...
- ▶ Distributions multimodales difficiles à apprendre
- ▶ Distributions de haute dimension très difficiles à apprendre  
→ curse of dimensionality
- ▶ Il existe beaucoup d'autres méthodes très performantes.

# DENSITY ESTIMATION

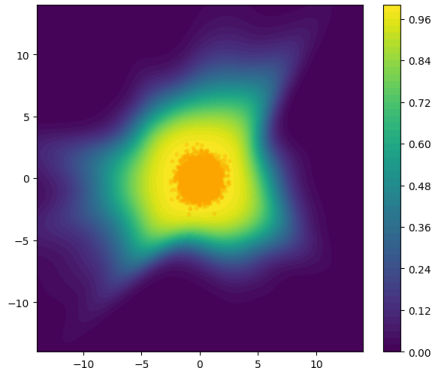
- ▶ Apprentissage de distributions multimodales (1, 2, 3 et 4 modes) par un MLP.
- ▶ Le MLP n'arrive pas à capturer les différents modes de manière discontinue.





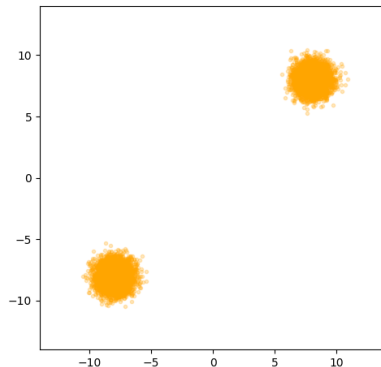
# DENSITY ESTIMATION

- ▶ Apprentissage de distributions multimodales (1, 2, 3 et 4 modes) par un MLP.
- ▶ Le MLP n'arrive pas à capturer les différents modes de manière discontinue.



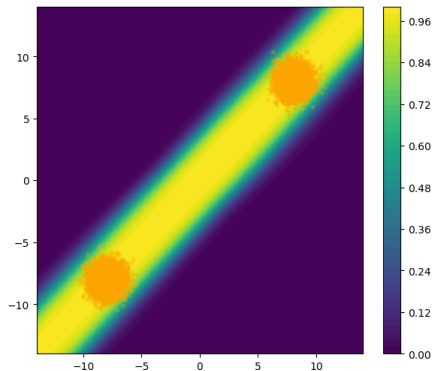
# DENSITY ESTIMATION

- ▶ Apprentissage de distributions multimodales (1, 2, 3 et 4 modes) par un MLP.
- ▶ Le MLP n'arrive pas à capturer les différents modes de manière discontinue.



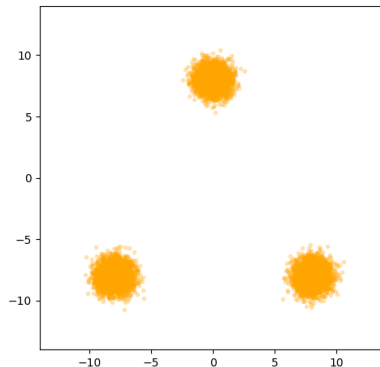
# DENSITY ESTIMATION

- ▶ Apprentissage de distributions multimodales (1, 2, 3 et 4 modes) par un MLP.
- ▶ Le MLP n'arrive pas à capturer les différents modes de manière discontinue.



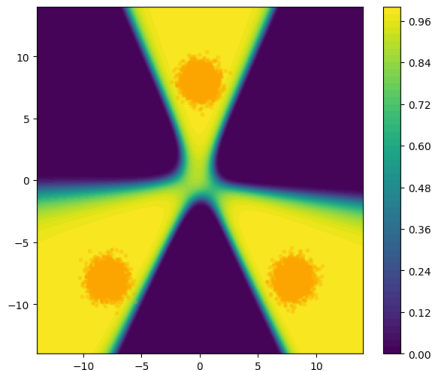
# DENSITY ESTIMATION

- ▶ Apprentissage de distributions multimodales (1, 2, 3 et 4 modes) par un MLP.
- ▶ Le MLP n'arrive pas à capturer les différents modes de manière discontinue.



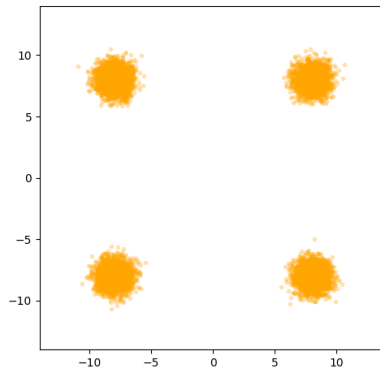
# DENSITY ESTIMATION

- ▶ Apprentissage de distributions multimodales (1, 2, 3 et 4 modes) par un MLP.
- ▶ Le MLP n'arrive pas à capturer les différents modes de manière discontinue.



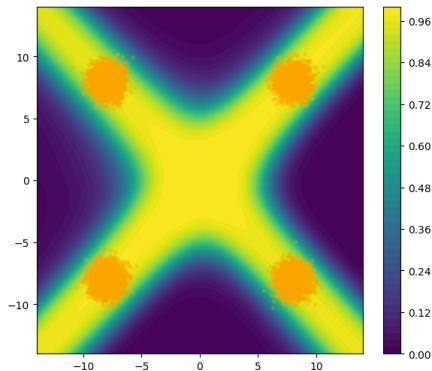
# DENSITY ESTIMATION

- ▶ Apprentissage de distributions multimodales (1, 2, 3 et 4 modes) par un MLP.
- ▶ Le MLP n'arrive pas à capturer les différents modes de manière discontinue.



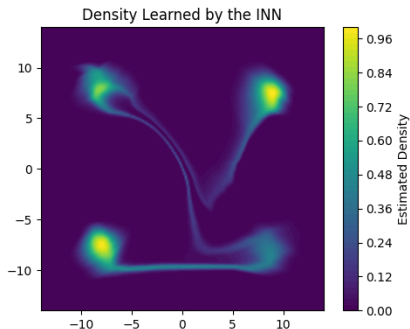
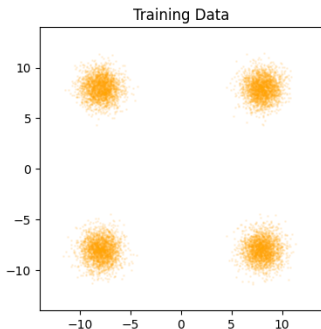
# DENSITY ESTIMATION

- ▶ Apprentissage de distributions multimodales (1, 2, 3 et 4 modes) par un MLP.
- ▶ Le MLP n'arrive pas à capturer les différents modes de manière discontinue.



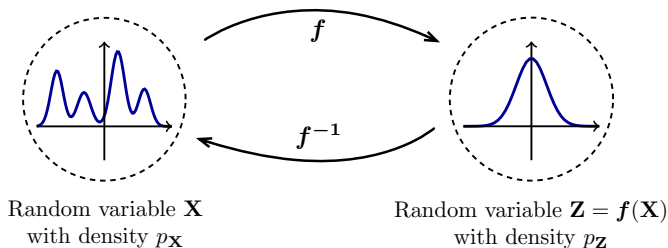
# DENSITY ESTIMATION

- ▶ Apprentissage d'une distributions multimodales (4 modes) par un réseau de neurones inversible (INN) (cf. slides suivants).
- ▶ C'est beaucoup mieux ! (malgré quelques filaments de continuité entre les modes)





## TRANSPORT DE MESURES



# TRANSPORT DE MESURES (DIM. 1)

- Soit  $X$  une variable aléatoire sur  $\mathbb{R}$  de fonction de densité  $p_X$  (PDF) et de répartition  $P_X$  (CDF).

Rappel:  $P_X(a) = \Pr(X \leq a) = \int_{-\infty}^a p_X(x)dx$

- Soit  $f : \mathbb{R} \rightarrow \mathbb{R}$  une fonction différentiable bijective.

Remarque:  $f$  continue et bijective  $\Rightarrow f$  monotone croissante ou monotone décroissante (sinon, on a des oscillations qui cassent l'injectivité et donc la bijectivité).

- Soit  $Z = f(X)$  la variable aléatoire obtenue par transformation de  $X$  via  $f$ . On cherche à exprimer la fonction de densité  $p_Z$  de  $Z$  en fonction de  $p_X$ .

# TRANSPORT DE MESURES (DIM. 1)

- Soit  $X$  une variable aléatoire sur  $\mathbb{R}$  de fonction de densité  $p_X$  (PDF) et de répartition  $P_X$  (CDF).

**Rappel:**  $P_X(a) = \Pr(X \leq a) = \int_{-\infty}^a p_X(x) dx$

- Soit  $f : \mathbb{R} \rightarrow \mathbb{R}$  une fonction différentiable bijective.

**Remarque:**  $f$  continue et bijective  $\Rightarrow f$  monotone croissante ou monotone décroissante (sinon, on a des oscillations qui cassent l'injectivité et donc la bijectivité).

- Soit  $Z = f(X)$  la variable aléatoire obtenue par transformation de  $X$  via  $f$ . On cherche à caractériser la fonction de densité  $p_Z$  de  $Z$  en fonction de  $p_X$ .

# TRANSPORT DE MESURES (DIM. 1)

- Soit  $X$  une variable aléatoire sur  $\mathbb{R}$  de fonction de densité  $p_X$  (PDF) et de répartition  $P_X$  (CDF).

**Rappel:**  $P_X(a) = Pr(X \leq a) = \int_{-\infty}^a p_X(x)dx$

- Soit  $f : \mathbb{R} \rightarrow \mathbb{R}$  une fonction différentiable bijective.

**Remarque:**  $f$  continue et bijective  $\Rightarrow f$  monotone croissante ou monotone décroissante (sinon, on a des oscillations qui cassent l'injectivité et donc la bijectivité).

- Soit  $Z = f(X)$  la variable aléatoire obtenue par transformation de  $X$  via  $f$ .

# TRANSPORT DE MESURES (DIM. 1)

- Soit  $X$  une variable aléatoire sur  $\mathbb{R}$  de fonction de densité  $p_X$  (PDF) et de répartition  $P_X$  (CDF).

**Rappel:**  $P_X(a) = \Pr(X \leq a) = \int_{-\infty}^a p_X(x)dx$

- Soit  $f : \mathbb{R} \rightarrow \mathbb{R}$  une fonction différentiable bijective.

**Remarque:**  $f$  continue et bijective  $\Rightarrow f$  monotone croissante ou monotone décroissante (sinon, on a des oscillations qui cassent l'injectivité et donc la bijectivité).

- Soit  $Z = f(X)$  la variable aléatoire obtenue par transformation de  $X$  via  $f$ .

# TRANSPORT DE MESURES (DIM. 1)

- ▶ Soit  $X$  une variable aléatoire sur  $\mathbb{R}$  de fonction de densité  $p_X$  (PDF) et de répartition  $P_X$  (CDF).

**Rappel:**  $P_X(a) = Pr(X \leq a) = \int_{-\infty}^a p_X(x)dx$

- ▶ Soit  $f : \mathbb{R} \rightarrow \mathbb{R}$  une fonction différentiable bijective.

**Remarque:**  $f$  continue et bijective  $\Rightarrow f$  monotone croissante ou monotone décroissante (sinon, on a des oscillations qui cassent l'injectivité et donc la bijectivité).

- ▶ Soit  $Z = f(X)$  la variable aléatoire obtenue par transformation de  $X$  via  $f$ . On cherche à exprimer la fonction de densité  $p_Z$  de  $Z$  en fonction de  $p_X$ .

# TRANSPORT DE MESURES (DIM. 1)

- ▶ Soit  $X$  une variable aléatoire sur  $\mathbb{R}$  de fonction de densité  $p_X$  (PDF) et de répartition  $P_X$  (CDF).

**Rappel:**  $P_X(a) = Pr(X \leq a) = \int_{-\infty}^a p_X(x)dx$

- ▶ Soit  $f : \mathbb{R} \rightarrow \mathbb{R}$  une fonction différentiable bijective.

**Remarque:**  $f$  continue et bijective  $\Rightarrow f$  monotone croissante ou monotone décroissante (sinon, on a des oscillations qui cassent l'injectivité et donc la bijectivité).

- ▶ Soit  $Z = f(X)$  la variable aléatoire obtenue par transformation de  $X$  via  $f$ . On cherche à exprimer la fonction de densité  $p_Z$  de  $Z$  en fonction de  $p_X$ .

# TRANSPORT DE MESURES (DIM. 1)

- ▶ Soit  $X$  une variable aléatoire sur  $\mathbb{R}$  de fonction de densité  $p_X$  (PDF) et de répartition  $P_X$  (CDF).

**Rappel:**  $P_X(a) = Pr(X \leq a) = \int_{-\infty}^a p_X(x)dx$

- ▶ Soit  $f : \mathbb{R} \rightarrow \mathbb{R}$  une fonction différentiable bijective.

**Remarque:**  $f$  continue et bijective  $\Rightarrow f$  monotone croissante ou monotone décroissante (sinon, on a des oscillations qui cassent l'injectivité et donc la bijectivité).

- ▶ Soit  $Z = f(X)$  la variable aléatoire obtenue par transformation de  $X$  via  $f$ . On cherche à exprimer la fonction de densité  $p_Z$  de  $Z$  en fonction de  $p_X$ .



# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone croissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \leq f^{-1}(z) \quad (5)$$

$$(f^{-1}(z))' \geq 0 \Rightarrow \left| (f^{-1}(z))' \right| = (f^{-1}(z))' \quad (6)$$

Ainsi

$$\begin{aligned} P_Z(z) &= Pr(Z \leq z) = Pr(f(X) \leq z) \\ &\stackrel{(5)}{=} Pr(X \leq f^{-1}(z)) = P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{dP_X(f^{-1}(z))}{dz} = \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{dx}{dz} \\ &= \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{d(f^{-1}(z))}{dz} \stackrel{(6)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{d(f^{-1}(z))}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone croissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \leq f^{-1}(z) \quad (5)$$

$$(f^{-1}(z))' \geq 0 \Rightarrow \left| (f^{-1}(z))' \right| = (f^{-1}(z))' \quad (6)$$

Ainsi

$$\begin{aligned} P_Z(z) &= Pr(Z \leq z) = Pr(f(X) \leq z) \\ &\stackrel{(5)}{=} Pr(X \leq f^{-1}(z)) = P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{dP_X(f^{-1}(z))}{dz} = \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{dx}{dz} \\ &= \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{d(f^{-1}(z))}{dz} \stackrel{(6)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{d(f^{-1}(z))}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone croissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \leq f^{-1}(z) \quad (5)$$

$$(f^{-1}(z))' \geq 0 \Rightarrow \left| (f^{-1}(z))' \right| = (f^{-1}(z))' \quad (6)$$

Ainsi

$$\begin{aligned} P_Z(z) &= Pr(Z \leq z) = Pr(f(X) \leq z) \\ &\stackrel{(5)}{=} Pr(X \leq f^{-1}(z)) = P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{dP_X(f^{-1}(z))}{dz} = \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{dx}{dz} \\ &= \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{d(f^{-1}(z))}{dz} \stackrel{(6)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{d(f^{-1}(z))}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone croissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \leq f^{-1}(z) \quad (5)$$

$$(f^{-1}(z))' \geq 0 \Rightarrow \left| (f^{-1}(z))' \right| = (f^{-1}(z))' \quad (6)$$

Ainsi

$$P_Z(z) = \Pr(Z \leq z) = \Pr(f(X) \leq z)$$

$$\stackrel{(5)}{=} \Pr(X \leq f^{-1}(z)) = P_X(f^{-1}(z))$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{dP_X(f^{-1}(z))}{dz} = \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{dx}{dz} \\ &= \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{d(f^{-1}(z))}{dz} \stackrel{(6)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{d(f^{-1}(z))}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone croissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \leq f^{-1}(z) \quad (5)$$

$$(f^{-1}(z))' \geq 0 \Rightarrow \left| (f^{-1}(z))' \right| = (f^{-1}(z))' \quad (6)$$

Ainsi

$$\begin{aligned} P_Z(z) &= Pr(Z \leq z) = Pr(f(X) \leq z) \\ &\stackrel{(5)}{=} Pr(X \leq f^{-1}(z)) = P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{dP_X(f^{-1}(z))}{dz} = \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{dx}{dz} \\ &= \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{d(f^{-1}(z))}{dz} \stackrel{(6)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{d(f^{-1}(z))}{dz} \right| \end{aligned}$$

## TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone croissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \leq f^{-1}(z) \quad (5)$$

$$(f^{-1}(z))' \geq 0 \Rightarrow \left| (f^{-1}(z))' \right| = (f^{-1}(z))' \quad (6)$$

Ainsi

$$\begin{aligned} P_Z(z) &= Pr(Z \leq z) = Pr(f(X) \leq z) \\ &\stackrel{(5)}{=} Pr(X \leq f^{-1}(z)) = P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{dP_X(f^{-1}(z))}{dz} = \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{dx}{dz} \\ &= \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{d(f^{-1}(z))}{dz} \stackrel{(6)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{d(f^{-1}(z))}{dz} \right| \end{aligned}$$

## TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone croissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \leq f^{-1}(z) \quad (5)$$

$$(f^{-1}(z))' \geq 0 \Rightarrow \left| (f^{-1}(z))' \right| = (f^{-1}(z))' \quad (6)$$

Ainsi

$$\begin{aligned} P_Z(z) &= Pr(Z \leq z) = Pr(f(X) \leq z) \\ &\stackrel{(5)}{=} Pr(X \leq f^{-1}(z)) = P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{dP_X(f^{-1}(z))}{dz} = \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{dx}{dz} \\ &= \frac{dP_X(f^{-1}(z))}{dx} \cdot \frac{d(f^{-1}(z))}{dz} \stackrel{(6)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{d(f^{-1}(z))}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone décroissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \geq f^{-1}(z) \quad (7)$$

$$(f^{-1}(z))' \leq 0 \Rightarrow \left| (f^{-1}(z))' \right| = -(f^{-1}(z))' \quad (8)$$

Ainsi

$$P_Z(z) = Pr(Z \leq z) = Pr(f(X) \leq y)$$

$$\stackrel{(7)}{=} Pr(X \geq f^{-1}(z)) = 1 - P_X(f^{-1}(z))$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dz} \\ &= \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{dx}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{df^{-1}(z)}{dz} \\ &= -p_X(f^{-1}(z)) \cdot \frac{df^{-1}(z)}{dz} \stackrel{(8)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \end{aligned}$$



# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone décroissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \geq f^{-1}(z) \quad (7)$$

$$(f^{-1}(z))' \leq 0 \Rightarrow \left| (f^{-1}(z))' \right| = -(f^{-1}(z))' \quad (8)$$

Ainsi

$$P_Z(z) = Pr(Z \leq z) = Pr(f(X) \leq y)$$

$$\stackrel{(7)}{=} Pr(X \geq f^{-1}(z)) = 1 - P_X(f^{-1}(z))$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dz} \\ &= \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{dx}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{df^{-1}(z)}{dz} \\ &= -p_X(f^{-1}(z)) \cdot \frac{df^{-1}(z)}{dz} \stackrel{(8)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone décroissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \geq f^{-1}(z) \quad (7)$$

$$(f^{-1}(z))' \leq 0 \Rightarrow \left| (f^{-1}(z))' \right| = -(f^{-1}(z))' \quad (8)$$

Ainsi

$$\begin{aligned} P_Z(z) &= Pr(Z \leq z) = Pr(f(X) \leq y) \\ &\stackrel{(7)}{=} Pr(X \geq f^{-1}(z)) = 1 - P_X(f^{-1}(z)) \\ p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dz} \\ &= \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{dx}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{df^{-1}(z)}{dz} \\ &= -p_X(f^{-1}(z)) \cdot \frac{df^{-1}(z)}{dz} \stackrel{(8)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone décroissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \geq f^{-1}(z) \quad (7)$$

$$(f^{-1}(z))' \leq 0 \Rightarrow \left| (f^{-1}(z))' \right| = -(f^{-1}(z))' \quad (8)$$

Ainsi

$$P_Z(z) = \Pr(Z \leq z) = \Pr(f(X) \leq y)$$

$$\stackrel{(7)}{=} \Pr(X \geq f^{-1}(z)) = 1 - P_X(f^{-1}(z))$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dz} \\ &= \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{dx}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{df^{-1}(z)}{dz} \\ &= -p_X(f^{-1}(z)) \cdot \frac{df^{-1}(z)}{dz} \stackrel{(8)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \end{aligned}$$

## TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone décroissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \geq f^{-1}(z) \quad (7)$$

$$(f^{-1}(z))' \leq 0 \Rightarrow \left| (f^{-1}(z))' \right| = -(f^{-1}(z))' \quad (8)$$

Ainsi

$$\begin{aligned} P_Z(z) &= \Pr(Z \leq z) = \Pr(f(X) \leq y) \\ &\stackrel{(7)}{=} \Pr(X \geq f^{-1}(z)) = 1 - P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dz} \\ &= \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{dx}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{df^{-1}(z)}{dz} \\ &= -p_X(f^{-1}(z)) \cdot \frac{df^{-1}(z)}{dz} \stackrel{(8)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone décroissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \geq f^{-1}(z) \quad (7)$$

$$(f^{-1}(z))' \leq 0 \Rightarrow \left| (f^{-1}(z))' \right| = -(f^{-1}(z))' \quad (8)$$

Ainsi

$$\begin{aligned} P_Z(z) &= \Pr(Z \leq z) = \Pr(f(X) \leq y) \\ &\stackrel{(7)}{=} \Pr(X \geq f^{-1}(z)) = 1 - P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dz} \\ &= \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{dx}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{df^{-1}(z)}{dz} \\ &= -p_X(f^{-1}(z)) \cdot \frac{df^{-1}(z)}{dz} \stackrel{(8)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone décroissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \geq f^{-1}(z) \quad (7)$$

$$(f^{-1}(z))' \leq 0 \Rightarrow \left| (f^{-1}(z))' \right| = -(f^{-1}(z))' \quad (8)$$

Ainsi

$$\begin{aligned} P_Z(z) &= \Pr(Z \leq z) = \Pr(f(X) \leq y) \\ &\stackrel{(7)}{=} \Pr(X \geq f^{-1}(z)) = 1 - P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dz} \\ &= \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{dx}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{df^{-1}(z)}{dz} \\ &= -p_X(f^{-1}(z)) \cdot \frac{df^{-1}(z)}{dz} \stackrel{(8)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

Si  $f$  est monotone décroissante, alors  $f^{-1}$  également, et on a

$$f(X) \leq z \Leftrightarrow X \geq f^{-1}(z) \quad (7)$$

$$(f^{-1}(z))' \leq 0 \Rightarrow \left| (f^{-1}(z))' \right| = -(f^{-1}(z))' \quad (8)$$

Ainsi

$$\begin{aligned} P_Z(z) &= \Pr(Z \leq z) = \Pr(f(X) \leq y) \\ &\stackrel{(7)}{=} \Pr(X \geq f^{-1}(z)) = 1 - P_X(f^{-1}(z)) \end{aligned}$$

$$\begin{aligned} p_Z(z) &= \frac{dP_Z(z)}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dz} \\ &= \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{dx}{dz} = \frac{d(1 - P_X(f^{-1}(z)))}{dx} \cdot \frac{df^{-1}(z)}{dz} \\ &= -p_X(f^{-1}(z)) \cdot \frac{df^{-1}(z)}{dz} \stackrel{(8)}{=} p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \end{aligned}$$

# TRANSPORT DE MESURES (DIM. 1)

En résumé, on a le **théorème de changement de variable** suivant:

## THEOREM

*Soient  $X$  une variable aléatoire sur  $\mathbb{R}$ ,  $f : \mathbb{R} \rightarrow \mathbb{R}$  une fonction différentiable bijective et  $Z = f(X)$ . La fonction de densité de  $Z$  est donnée par:*

$$p_Z(z) = p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \quad (9)$$



# TRANSPORT DE MESURES (DIM. 1)

En résumé, on a le **théorème de changement de variable** suivant:

## THEOREM

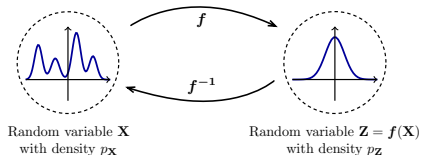
*Soient  $X$  une variable aléatoire sur  $\mathbb{R}$ ,  $f : \mathbb{R} \rightarrow \mathbb{R}$  une fonction différentiable bijective et  $Z = f(X)$ . La fonction de densité de  $Z$  est donnée par:*

$$p_Z(z) = p_X(f^{-1}(z)) \cdot \left| \frac{df^{-1}(z)}{dz} \right| \quad (9)$$

# TRANSPORT DE MESURES (DIM. 1)

- ▶ Dans notre cas, on cherchera plutôt à exprimer  $p_X$  en fonction de  $p_Z$ , qui sera une loi normale centrée réduite (simple).
- ▶ En remplaçant  $X$ ,  $Z$  et  $f$  par  $Z$ ,  $X$  et  $f^{-1}$ , respectivement, dans le théorème, on a:

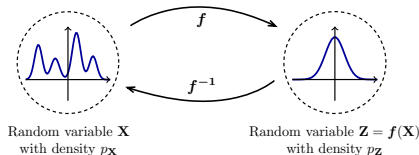
$$p_X(x) = p_Z(f(x)) \cdot \left| \frac{df(x)}{dx} \right| = p_Z(f(x)) \cdot |J_f(x)| \quad (10)$$



# TRANSPORT DE MESURES (DIM. 1)

- ▶ Dans notre cas, on cherchera plutôt à exprimer  $p_X$  en fonction de  $p_Z$ , qui sera une loi normale centrée réduite (simple).
- ▶ En remplaçant  $X$ ,  $Z$  et  $f$  par  $Z$ ,  $X$  et  $f^{-1}$ , respectivement, dans le théorème, on a:

$$p_X(x) = p_Z(f(x)) \cdot \left| \frac{df(x)}{dx} \right| = p_Z(f(x)) \cdot |J_f(x)| \quad (10)$$



# TRANSPORT DE MESURES (DIM. $d > 1$ )

Dans le cas multidimensionnel, le **théorème de changement de variable** se généralise ainsi:

## THEOREM

Soient  $\mathbf{X}$  une variable aléatoire sur  $\mathbb{R}^d$ ,  $f : \mathbb{R}^d \rightarrow \mathbb{R}^d$  une fonction différentiable bijective et  $\mathbf{Z} = f(\mathbf{X})$ . La fonction de densité de  $\mathbf{Z}$  est donnée par:

$$p_{\mathbf{Z}}(\mathbf{z}) = p_{\mathbf{X}}(f^{-1}(\mathbf{z})) \cdot \left| \det \left( \frac{\partial f^{-1}(\mathbf{z})}{\partial \mathbf{z}} \right) \right|$$

$$\text{où } \frac{\partial f^{-1}(\mathbf{z})}{\partial \mathbf{z}} = \begin{bmatrix} \frac{\partial f_1^{-1}(\mathbf{z})}{\partial z_1} & \dots & \frac{\partial f_1^{-1}(\mathbf{z})}{\partial z_d} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_d^{-1}(\mathbf{z})}{\partial z_1} & \dots & \frac{\partial f_d^{-1}(\mathbf{z})}{\partial z_d} \end{bmatrix} \text{ est le Jacobien de } f^{-1}(\mathbf{z}).$$

# TRANSPORT DE MESURES (DIM. $d > 1$ )

Dans le cas multidimensionnel, le **théorème de changement de variable** se généralise ainsi:

## THEOREM

*Soient  $\mathbf{X}$  une variable aléatoire sur  $\mathbb{R}^d$ ,  $f : \mathbb{R}^d \rightarrow \mathbb{R}^d$  une fonction différentiable bijective et  $\mathbf{Z} = f(\mathbf{X})$ . La fonction de densité de  $\mathbf{Z}$  est donnée par:*

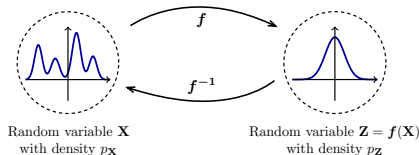
$$p_{\mathbf{Z}}(\mathbf{z}) = p_{\mathbf{X}}(f^{-1}(\mathbf{z})) \cdot \left| \det \left( \frac{\partial f^{-1}(\mathbf{z})}{\partial \mathbf{z}} \right) \right|$$

$$\text{où } \frac{\partial f^{-1}(\mathbf{z})}{\partial \mathbf{z}} = \begin{bmatrix} \frac{\partial f_1^{-1}(\mathbf{z})}{\partial z_1} & \dots & \frac{\partial f_1^{-1}(\mathbf{z})}{\partial z_d} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_d^{-1}(\mathbf{z})}{\partial z_1} & \dots & \frac{\partial f_d^{-1}(\mathbf{z})}{\partial z_d} \end{bmatrix} \text{ est le Jacobien de } f^{-1}(\mathbf{z}).$$

# TRANSPORT DE MESURES (DIM. $d > 1$ )

- Dans notre cas, on cherchera plutôt à exprimer  $p_{\mathbf{X}}$  en fonction de  $p_{\mathbf{Z}}$ , qui sera une loi normale centrée réduite (simple).
- En remplaçant  $\mathbf{X}$ ,  $\mathbf{Z}$  et  $f$  par  $\mathbf{Z}$ ,  $\mathbf{X}$  et  $f^{-1}$ , respectivement, dans le théorème, on a:

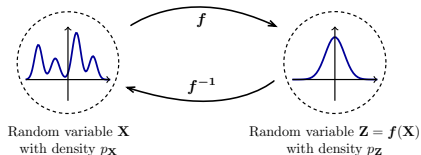
$$p_{\mathbf{X}}(\mathbf{x}) = p_{\mathbf{Z}}(f(\mathbf{x})) \cdot \left| \det \left( \frac{d f(\mathbf{x})}{d \mathbf{x}} \right) \right| = p_{\mathbf{Z}}(f(\mathbf{x})) \cdot |\det(J_f(\mathbf{x}))| \quad (11)$$



## TRANSPORT DE MESURES (DIM. $d > 1$ )

- ▶ Dans notre cas, on cherchera plutôt à exprimer  $p_{\mathbf{X}}$  en fonction de  $p_{\mathbf{Z}}$ , qui sera une loi normale centrée réduite (simple).
- ▶ En remplaçant  $\mathbf{X}$ ,  $\mathbf{Z}$  et  $f$  par  $\mathbf{Z}$ ,  $\mathbf{X}$  et  $f^{-1}$ , respectivement, dans le théorème, on a:

$$p_{\mathbf{X}}(\mathbf{x}) = p_{\mathbf{Z}}(f(\mathbf{x})) \cdot \left| \det \left( \frac{d f(\mathbf{x})}{d \mathbf{x}} \right) \right| = p_{\mathbf{Z}}(f(\mathbf{x})) \cdot |\det(J_f(\mathbf{x}))| \quad (11)$$



# NORMALIZING FLOWS

- ▶ Les modèles de **flots génératifs (normalizing flows, flow-based generative models)**, en particulier les **réseaux de neurones inversibles**, utilisent le *théorème du changement de variable* pour:
  1. apprendre la distribution des data plus efficacement ;
  2. générer des data.



# RÉSEAUX DE NEURONES INVERSIBLES (INNs)

- Un **réseau de neurones inversible** (invertible neural networks, INN) est un réseau de neurones dont la fonction associée

$$f_{\theta} : \mathbb{R}^d \rightarrow \mathbb{R}^d$$

satisfait les propriétés suivantes:

1.  $f_{\theta}$  est bijective ;
2. l'inverse  $f_{\theta}^{-1}$  est facile à calculer ;
3. le Jacobien  $\frac{\partial f_{\theta}}{\partial \mathbf{x}}$  et son déterminant sont faciles à calculer.

# RÉSEAUX DE NEURONES INVERSIBLES (INNs)

- Un **réseau de neurones inversible** (invertible neural networks, INN) est un réseau de neurones dont la fonction associée

$$f_{\theta} : \mathbb{R}^d \rightarrow \mathbb{R}^d$$

satisfait les propriétés suivantes:

1.  $f_{\theta}$  est bijective ;
2. l'inverse  $f_{\theta}^{-1}$  est facile à calculer ;
3. le Jacobien  $\frac{\partial f_{\theta}}{\partial \mathbf{x}}$  et son déterminant sont faciles à calculer.

# RÉSEAUX DE NEURONES INVERSIBLES (INNs)

- Un **réseau de neurones inversible** (**invertible neural networks, INN**) est un réseau de neurones dont la fonction associée

$$f_{\theta} : \mathbb{R}^d \rightarrow \mathbb{R}^d$$

satisfait les propriétés suivantes:

1.  $f_{\theta}$  est bijective ;
2. l'inverse  $f_{\theta}^{-1}$  est facile à calculer ;
3. le Jacobien  $\frac{\partial f_{\theta}}{\partial \mathbf{x}}$  et son déterminant sont faciles à calculer.

# RÉSEAUX DE NEURONES INVERSIBLES (INNs)

- Un **réseau de neurones inversible** (invertible neural networks, INN) est un réseau de neurones dont la fonction associée

$$f_{\theta} : \mathbb{R}^d \rightarrow \mathbb{R}^d$$

satisfait les propriétés suivantes:

1.  $f_{\theta}$  est bijective ;
2. l'inverse  $f_{\theta}^{-1}$  est facile à calculer ;
3. le Jacobien  $\frac{\partial f_{\theta}}{\partial \mathbf{x}}$  et son déterminant sont faciles à calculer.

# INN – REALNVP

- ▶ Un **real-valued non-volume preserving network (RealNVP)** est un INNs composé de couches bijectives (invertibles)  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  appelées “coupling layers” (dim.  $d$  conservée).

[Dinh et al., 2017]

- ▶ Le réseau est entraîné de telle sorte que la composition de toutes ses couches  $f_\theta = l_n \circ \dots \circ l_1$  réalise pas à pas (interpolation) un *transport de mesure inversible* :

*les data originales sont transportées sur une distribution normale centrée réduite.*

- ▶ Ensuite, pour *générer* de nouvelles data, on sample  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  et on applique la transformation inverse

$$\mathbf{x}_{\text{new}} = f_\theta^{-1}(\mathbf{z}) = l_1^{-1} \circ \dots \circ l_n^{-1}(\mathbf{z})$$

# INN – REALNVP

- ▶ Un **real-valued non-volume preserving network (RealNVP)** est un INN composé de couches bijectives (invertibles)  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  appelées “coupling layers” (dim.  $d$  conservée).  
[Dinh et al., 2017]
- ▶ Le réseau est entraîné de telle sorte que la composition de toutes ses couches  $f_\theta = l_n \circ \dots \circ l_1$  réalise pas à pas (interpolation) un *transport de mesure inversible* :

*les data originales sont transportées sur une distribution normale centrée réduite.*

- ▶ Ensuite, pour *générer* de nouvelles data, on sample  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  et on applique la transformation inverse

$$\mathbf{x}_{\text{new}} = f_\theta^{-1}(\mathbf{z}) = l_1^{-1} \circ \dots \circ l_n^{-1}(\mathbf{z})$$

# INN – REALNVP

- Un **real-valued non-volume preserving network (RealNVP)** est un INN composé de couches bijectives (inversibles)  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  appelées “coupling layers” (dim.  $d$  conservée).

[Dinh et al., 2017]

- Le réseau est entraîné de telle sorte que la composition de toutes ses couches  $f_\theta = l_n \circ \dots \circ l_1$  réalise pas à pas (interpolation) un *transport de mesure inversible* :

*les data originales sont transportées sur une distribution normale centrée réduite.*

- Ensuite, pour *générer* de nouvelles data, on sample  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  et on applique la transformation inverse

$$\mathbf{x}_{\text{new}} = f_\theta^{-1}(\mathbf{z}) = l_1^{-1} \circ \dots \circ l_n^{-1}(\mathbf{z})$$

# INN – REALNVP

- Un **real-valued non-volume preserving network (RealNVP)** est un INN composé de couches bijectives (inversibles)  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  appelées “coupling layers” (dim.  $d$  conservée).

[Dinh et al., 2017]

- Le réseau est entraîné de telle sorte que la composition de toutes ses couches  $f_\theta = l_n \circ \dots \circ l_1$  réalise pas à pas (interpolation) un *transport de mesure inversible* :

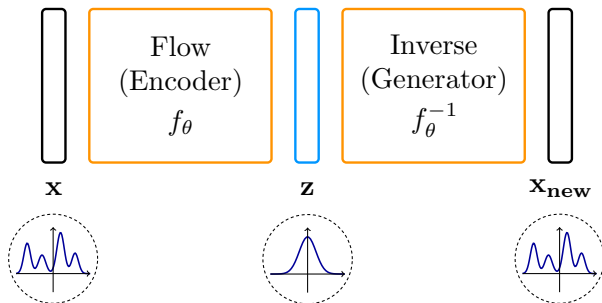
*les data originales sont transportées sur une distribution normale centrée réduite.*

- Ensuite, pour *générer* de nouvelles data, on sample  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  et on applique la transformation inverse

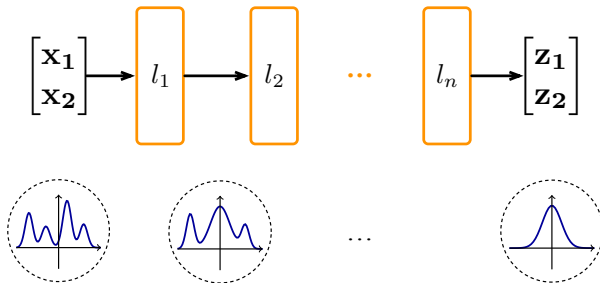
$$\mathbf{x}_{\text{new}} = f_\theta^{-1}(\mathbf{z}) = l_1^{-1} \circ \dots \circ l_n^{-1}(\mathbf{z})$$



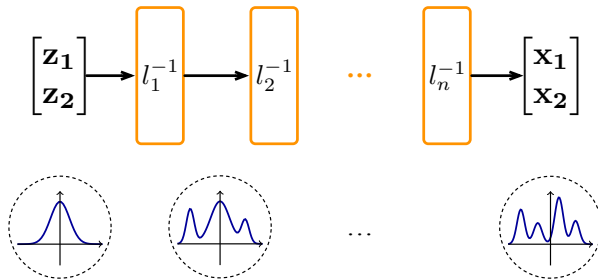
## INN – REALNVP



## INN – REALNVP



## INN – REALNVP



# INN – REALNVP: LAYER

- ▶ Chaque couche  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  réalise une transformation bijective facile à inverser.
- ▶ L'idée est de séparer l'input en deux parties  $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2] \in \mathbb{R}^d$ , où  $\mathbf{x}_1 \in \mathbb{R}^{k_i}$  et  $\mathbf{x}_2 \in \mathbb{R}^{d-k_i}$ .
- ▶ La couche  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  implémente la transformation affine suivante:

$$\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \xrightarrow{l_i} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \odot \exp(s_i(\mathbf{x}_1)) + t_i(\mathbf{x}_1) \end{bmatrix}$$

où  $s_i$  (scale function) et  $t_i$  (translation function) sont des réseaux de neurones simples.

# INN – REALNVP: LAYER

- ▶ Chaque couche  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  réalise une transformation bijective facile à inverser.
- ▶ L'idée est de séparer l'input en deux parties  $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2] \in \mathbb{R}^d$ , où  $\mathbf{x}_1 \in \mathbb{R}^{k_i}$  et  $\mathbf{x}_2 \in \mathbb{R}^{d-k_i}$ .
- ▶ La couche  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  implémente la transformation affine suivante:

$$\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \xrightarrow{l_i} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \odot \exp(s_i(\mathbf{x}_1)) + t_i(\mathbf{x}_1) \end{bmatrix}$$

où  $s_i$  (scale function) et  $t_i$  (translation function) sont des réseaux de neurones simples.

# INN – REALNVP: LAYER

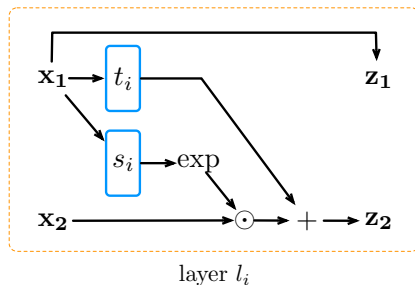
- ▶ Chaque couche  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  réalise une transformation bijective facile à inverser.
- ▶ L'idée est de séparer l'input en deux parties  $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2] \in \mathbb{R}^d$ , où  $\mathbf{x}_1 \in \mathbb{R}^{k_i}$  et  $\mathbf{x}_2 \in \mathbb{R}^{d-k_i}$ .
- ▶ La couche  $l_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$  implémente la transformation affine suivante:

$$\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \xrightarrow{l_i} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \odot \exp(s_i(\mathbf{x}_1)) + t_i(\mathbf{x}_1) \end{bmatrix}$$

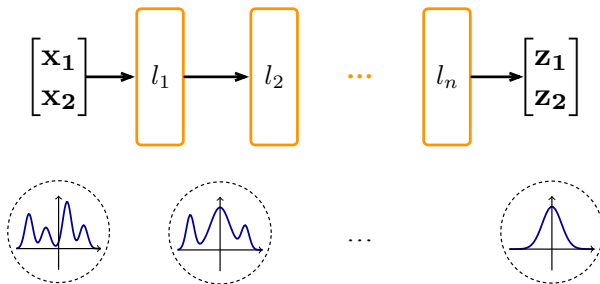
où  $s_i$  (scale function) et  $t_i$  (translation function) sont des réseaux de neurones simples.

## INN – REALNVP: LAYER

$$\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \xrightarrow{l_i} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \odot \exp(s_i(\mathbf{x}_1)) + t_i(\mathbf{x}_1) \end{bmatrix}$$



## INN – REALNVP: FULL NETWORK





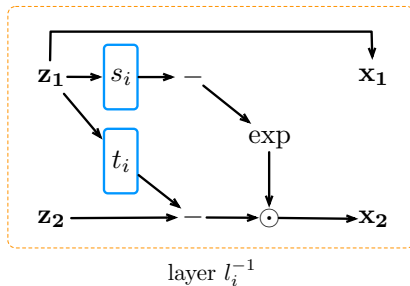
## INN – REALNVP: INVERSE LAYER

- Grâce à cette architecture, la transformation inverse de chaque couche  $l_i$ , notée  $l_i^{-1} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ , peut se calculer simplement:

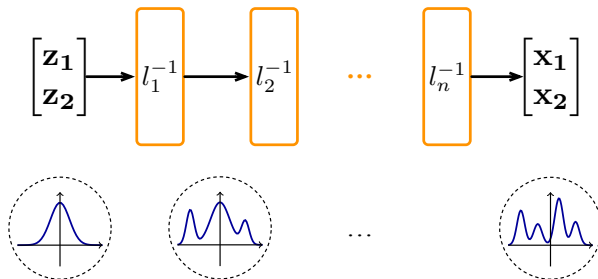
$$\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} \xrightarrow{l_i^{-1}} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{y}_1 \\ (\mathbf{z}_2 - t_i(\mathbf{z}_1)) \odot \exp(-s_i(\mathbf{z}_1)) \end{bmatrix}$$

### INN – REALNVP: INVERSE LAYER

$$\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} \xrightarrow{l_i^{-1}} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{z}_1 \\ (\mathbf{z}_2 - t_i(\mathbf{z}_1)) \odot \exp(-s_i(\mathbf{z}_1)) \end{bmatrix}$$



## INN – REALNVP: FULL INVERSE NETWORK



# CALCUL DU JACOBIEN (1 COUCHE)

- Le Jacobien associé à chaque couche  $l_i$  est une matrice triangulaire inférieure :

$$\mathbf{z} = l_i(\mathbf{x}) = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \odot \exp(s_i(\mathbf{x}_1)) + t_i(\mathbf{x}_1) \end{bmatrix}$$

$$J_{l_i} := \frac{\partial l_i(\mathbf{x})}{\partial \mathbf{x}} = \frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \begin{bmatrix} \mathbf{I}_{k_i} & \mathbf{0} \\ \frac{\partial \mathbf{z}_2}{\partial \mathbf{x}_1} & \text{diag}[\exp(s_i(\mathbf{x}_1))] \end{bmatrix}$$

où le terme  $\frac{\partial \mathbf{z}_2}{\partial \mathbf{x}_1}$  est non trivial (ce qui ne pose aucun problème).

- Ainsi, le déterminant du Jacobien se calcule simplement :

$$|\det(J_{l_i})| = \left| \det \left( \frac{\partial \mathbf{z}}{\partial \mathbf{x}} \right) \right| = \exp \left( \sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1) \right) \quad (12)$$

# CALCUL DU JACOBIEN (1 COUCHE)

- Le Jacobien associé à chaque couche  $l_i$  est une matrice triangulaire inférieure :

$$\mathbf{z} = l_i(\mathbf{x}) = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \odot \exp(s_i(\mathbf{x}_1)) + t_i(\mathbf{x}_1) \end{bmatrix}$$

$$J_{l_i} := \frac{\partial l_i(\mathbf{x})}{\partial \mathbf{x}} = \frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \begin{bmatrix} \mathbf{I}_{k_i} & \mathbf{0} \\ \frac{\partial \mathbf{z}_2}{\partial \mathbf{x}_1} & \text{diag}[\exp(s_i(\mathbf{x}_1))] \end{bmatrix}$$

où le terme  $\frac{\partial \mathbf{z}_2}{\partial \mathbf{x}_1}$  est non trivial (ce qui ne pose aucun problème).

- Ainsi, le déterminant du Jacobien se calcule simplement :

$$|\det(J_{l_i})| = \left| \det\left(\frac{\partial \mathbf{z}}{\partial \mathbf{x}}\right) \right| = \exp\left(\sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1)\right) \quad (12)$$

# CALCUL DU JACOBIEN (RÉSEAU TOTAL)

- La transformation totale du RealNVP est donnée par

$$f_{\theta}(\mathbf{x}) = l_n \circ l_{n-1} \circ \cdots \circ l_1(\mathbf{x}) = \mathbf{z}$$

$$\mathbf{x}_0 = \mathbf{x}, \quad \mathbf{x}_1 = l_1(\mathbf{x}_0), \quad \dots, \quad \mathbf{z} = \mathbf{x}_n = l_n(\mathbf{x}_{n-1})$$

- Pour cette transformation totale, en appliquant la 'chain rule', le Jacobien et son déterminant sont donnés par :

$$J_{f_{\theta}}(\mathbf{x}) = J_{l_n}(\mathbf{x}_{n-1}) \cdots J_{l_2}(\mathbf{x}_1) J_{l_1}(\mathbf{x}_0)$$

$$|\det J_{f_{\theta}}(\mathbf{x})| = \prod_{i=1}^n |\det J_{l_i}(\mathbf{x}_{i-1})| \stackrel{(12)}{=} \prod_{i=1}^n \exp \left( \sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1) \right) \quad (13)$$

# CALCUL DU JACOBIEN (RÉSEAU TOTAL)

- La transformation totale du RealNVP est donnée par

$$f_{\theta}(\mathbf{x}) = l_n \circ l_{n-1} \circ \dots \circ l_1(\mathbf{x}) = \mathbf{z}$$

$$\mathbf{x}_0 = \mathbf{x}, \quad \mathbf{x}_1 = l_1(\mathbf{x}_0), \dots, \mathbf{z} = \mathbf{x}_n = l_n(\mathbf{x}_{n-1})$$

- Pour cette transformation totale, en appliquant la 'chain rule', le Jacobien et son déterminant sont donnés par :

$$J_{f_{\theta}}(\mathbf{x}) = J_{l_n}(\mathbf{x}_{n-1}) \dots J_{l_2}(\mathbf{x}_1) J_{l_1}(\mathbf{x}_0)$$

$$|\det J_{f_{\theta}}(\mathbf{x})| = \prod_{i=1}^n |\det J_{l_i}(\mathbf{x}_{i-1})| \stackrel{(12)}{=} \prod_{i=1}^n \exp \left( \sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1) \right) \quad (13)$$

## INN – ENTRAÎNEMENT

- ▶ Soit un dataset  $\mathcal{D} = \{\mathbf{x}_i \in \mathbb{R}^d : i = 1, \dots, n\}$  dont les éléments proviennent d'une distribution empirique  $p$ .
- ▶ Soit la distribution normale centrée réduite  $p_{\mathbf{Z}}$ .
- ▶ Soit un réseau de neurones inversible  $f_{\theta} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ .
- ▶ Par le *théorème de changement de variable*, la distribution  $p_{\mathbf{X}}$  qui est *transportée* par  $f_{\theta}$  sur  $p_{\mathbf{Z}}$  est donnée par :

$$\begin{aligned}
 p_{\mathbf{X}}(\mathbf{x}) &\stackrel{(11)}{=} p_{\mathbf{Z}}(f_{\theta}(\mathbf{x})) \cdot |\det(J_{f_{\theta}}(\mathbf{x}))| \\
 &\stackrel{(13)}{=} p_{\mathbf{Z}}(f_{\theta}(\mathbf{x})) \cdot \prod_{i=1}^n \exp\left(\sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1)\right)
 \end{aligned} \tag{14}$$



## INN – ENTRAÎNEMENT

- Soit un dataset  $\mathcal{D} = \{\mathbf{x}_i \in \mathbb{R}^d : i = 1, \dots, n\}$  dont les éléments proviennent d'une distribution empirique  $p$ .
- Soit la distribution normale centrée réduite  $p_{\mathbf{Z}}$ .
- Soit un réseau de neurones inversible  $f_{\theta} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ .
- Par le *théorème de changement de variable*, la distribution  $p_{\mathbf{X}}$  qui est *transportée* par  $f_{\theta}$  sur  $p_{\mathbf{Z}}$  est donnée par :

$$\begin{aligned}
 p_{\mathbf{X}}(\mathbf{x}) &\stackrel{(11)}{=} p_{\mathbf{Z}}(f_{\theta}(\mathbf{x})) \cdot |\det(J_{f_{\theta}}(\mathbf{x}))| \\
 &\stackrel{(13)}{=} p_{\mathbf{Z}}(f_{\theta}(\mathbf{x})) \cdot \prod_{i=1}^n \exp\left(\sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1)\right)
 \end{aligned} \tag{14}$$

## INN – ENTRAÎNEMENT

- Soit un dataset  $\mathcal{D} = \{\mathbf{x}_i \in \mathbb{R}^d : i = 1, \dots, n\}$  dont les éléments proviennent d'une distribution empirique  $p$ .
- Soit la distribution normale centrée réduite  $p_{\mathbf{Z}}$ .
- Soit un réseau de neurones inversible  $f_{\theta} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ .
- Par le *théorème de changement de variable*, la distribution  $p_{\mathbf{X}}$  qui est *transportée* par  $f_{\theta}$  sur  $p_{\mathbf{Z}}$  est donnée par :

$$\begin{aligned}
 p_{\mathbf{X}}(\mathbf{x}) &\stackrel{(11)}{=} p_{\mathbf{Z}}(f_{\theta}(\mathbf{x})) \cdot |\det(J_{f_{\theta}}(\mathbf{x}))| \\
 &\stackrel{(13)}{=} p_{\mathbf{Z}}(f_{\theta}(\mathbf{x})) \cdot \prod_{i=1}^n \exp\left(\sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1)\right)
 \end{aligned} \tag{14}$$

## INN – ENTRAÎNEMENT

- Soit un dataset  $\mathcal{D} = \{\mathbf{x}_i \in \mathbb{R}^d : i = 1, \dots, n\}$  dont les éléments proviennent d'une distribution empirique  $p$ .
- Soit la distribution normale centrée réduite  $p_{\mathbf{Z}}$ .
- Soit un réseau de neurones inversible  $f_{\theta} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ .
- Par le *théorème de changement de variable*, la distribution  $p_{\mathbf{X}}$  qui est *transportée* par  $f_{\theta}$  sur  $p_{\mathbf{Z}}$  est donnée par :

$$\begin{aligned}
 p_{\mathbf{X}}(\mathbf{x}) &\stackrel{(11)}{=} p_{\mathbf{Z}}(f_{\theta}(\mathbf{x})) \cdot |\det(J_{f_{\theta}}(\mathbf{x}))| \\
 &\stackrel{(13)}{=} p_{\mathbf{Z}}(f_{\theta}(\mathbf{x})) \cdot \prod_{i=1}^n \exp\left(\sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1)\right)
 \end{aligned} \tag{14}$$

## INN – ENTRAÎNEMENT

- ▶ On entraîne  $f_\theta$  pour que la distribution  $p_{\mathbf{X}}$  qui est transportée par  $f_\theta$  sur  $p_{\mathbf{Z}}$  soit le plus proche possible de la distribution empirique  $p$ , i.e.,  $p_{\mathbf{X}}(\mathbf{x}) \simeq p(\mathbf{x})$ ,  $\forall \mathbf{x} \in \mathbb{R}^d$ .
- ▶ On se ramène alors à un problème de *d'apprentissage de la distribution  $p$  via  $p_{\mathbf{X}}$* .
- ▶ Pour cela, on minimise la negative log likelihood (NLL) :

$$\begin{aligned}
 \mathcal{L}(\theta) &\stackrel{(4)}{=} -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \log(p_{\mathbf{X}}(\mathbf{x})) \\
 &\stackrel{(14)}{=} -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \log \left( p_{\mathbf{Z}}(f_\theta(\mathbf{x})) \cdot \prod_{i=1}^n \exp \left( \sum_{j=1}^{d-k_i} s_i(\mathbf{x}_j) \right) \right) \\
 &= -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \left[ \log(p_{\mathbf{Z}}(f_\theta(\mathbf{x}))) + \sum_{i=1}^n \log \left( \exp \left( \sum_{j=1}^{d-k_i} s_i(\mathbf{x}_j) \right) \right) \right]
 \end{aligned}$$

## INN – ENTRAÎNEMENT

- ▶ On entraîne  $f_\theta$  pour que la distribution  $p_{\mathbf{X}}$  qui est transportée par  $f_\theta$  sur  $p_{\mathbf{Z}}$  soit le plus proche possible de la distribution empirique  $p$ , i.e.,  $p_{\mathbf{X}}(\mathbf{x}) \simeq p(\mathbf{x})$ ,  $\forall \mathbf{x} \in \mathbb{R}^d$ .
- ▶ On se ramène alors à un problème de *d'apprentissage de la distribution  $p$  via  $p_{\mathbf{X}}$* .
- ▶ Pour cela, on minimise la negative log likelihood (NLL) :

$$\begin{aligned}
 \mathcal{L}(\theta) &\stackrel{(4)}{=} -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \log(p_{\mathbf{X}}(\mathbf{x})) \\
 &\stackrel{(14)}{=} -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \log \left( p_{\mathbf{Z}}(f_\theta(\mathbf{x})) \cdot \prod_{i=1}^n \exp \left( \sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1) \right) \right) \\
 &= -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \left[ \log(p_{\mathbf{Z}}(f_\theta(\mathbf{x}))) + \sum_{i=1}^n \log \left( \exp \left( \sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1) \right) \right) \right]
 \end{aligned}$$

## INN – ENTRAÎNEMENT

- ▶ On entraîne  $f_\theta$  pour que la distribution  $p_{\mathbf{X}}$  qui est transportée par  $f_\theta$  sur  $p_{\mathbf{Z}}$  soit le plus proche possible de la distribution empirique  $p$ , i.e.,  $p_{\mathbf{X}}(\mathbf{x}) \simeq p(\mathbf{x})$ ,  $\forall \mathbf{x} \in \mathbb{R}^d$ .
- ▶ On se ramène alors à un problème de *d'apprentissage de la distribution  $p$  via  $p_{\mathbf{X}}$* .
- ▶ Pour cela, on minimise la negative log likelihood (NLL) :

$$\begin{aligned}
 \mathcal{L}(\theta) &\stackrel{(4)}{=} -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \log(p_{\mathbf{X}}(\mathbf{x})) \\
 &\stackrel{(14)}{=} -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \log \left( p_{\mathbf{Z}}(f_\theta(\mathbf{x})) \cdot \prod_{i=1}^n \exp \left( \sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1) \right) \right) \\
 &= -\frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \left[ \log(p_{\mathbf{Z}}(f_\theta(\mathbf{x}))) + \sum_{i=1}^n \log \left( \exp \left( \sum_{i=1}^{d-k_i} s_i(\mathbf{x}_1) \right) \right) \right]
 \end{aligned}$$

# INN – GÉNÉRATION

- ▶ Le réseau de neurones inversible  $f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^d$  peut ensuite être utilisé pour la *génération* de data.
- ▶ Pour générer une data  $\mathbf{x}_{\text{new}}$  :
  1. On sample  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  ;
  2. On applique la transformation inverse

$$\mathbf{x}_{\text{new}} = f_\theta^{-1}(\mathbf{z}) = l_1^{-1} \circ \dots \circ l_n^{-1}(\mathbf{z})$$

# INN – GÉNÉRATION

- ▶ Le réseau de neurones inversible  $f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^d$  peut ensuite être utilisé pour la *génération* de data.
- ▶ Pour générer une data  $\mathbf{x}_{\text{new}}$  :
  1. On sample  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  ;
  2. On applique la transformation inverse

$$\mathbf{x}_{\text{new}} = f_\theta^{-1}(\mathbf{z}) = l_1^{-1} \circ \dots \circ l_n^{-1}(\mathbf{z})$$



# INN – GÉNÉRATION

- ▶ Le réseau de neurones inversible  $f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^d$  peut ensuite être utilisé pour la *génération* de data.
- ▶ Pour générer une data  $\mathbf{x}_{\text{new}}$  :
  1. On sample  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  ;
  2. On applique la transformation inverse

$$\mathbf{x}_{\text{new}} = f_\theta^{-1}(\mathbf{z}) = l_1^{-1} \circ \dots \circ l_n^{-1}(\mathbf{z})$$

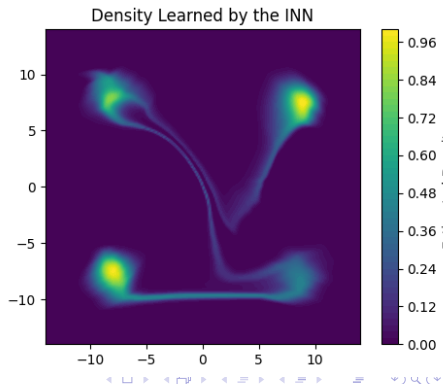
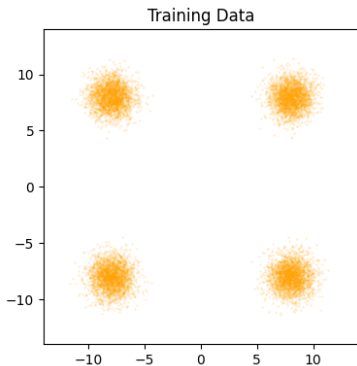
# INN – GÉNÉRATION

- ▶ Le réseau de neurones inversible  $f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^d$  peut ensuite être utilisé pour la *génération* de data.
- ▶ Pour générer une data  $\mathbf{x}_{\text{new}}$  :
  1. On sample  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  ;
  2. On applique la transformation inverse

$$\mathbf{x}_{\text{new}} = f_\theta^{-1}(\mathbf{z}) = l_1^{-1} \circ \dots \circ l_n^{-1}(\mathbf{z})$$

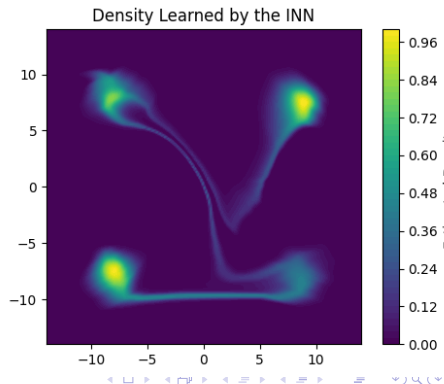
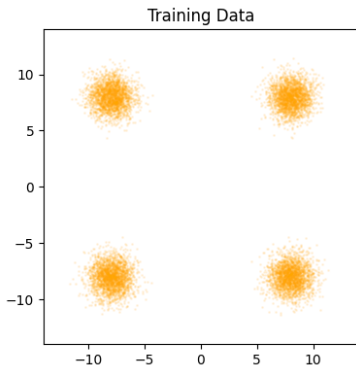
## EXEMPLE

- ▶ Apprentissage d'une distributions multimodales (4 modes) par un RealNVP.
- ▶ On entraîne  $f_\theta$  comme décrit précédemment et on utilise ensuite la formule (14) pour évaluer la densité des points du plan.



## EXEMPLE

- ▶ Apprentissage d'une distributions multimodales (4 modes) par un RealNVP.
- ▶ On entraîne  $f_\theta$  comme décrit précédemment et on utilise ensuite la formule (14) pour évaluer la densité des points du plan.



# EXEMPLE

- ▶ Entraînement d'un RealNVP sur le dataset MNIST et génération de data...
- ▶ On pourrait utiliser d'autres architectures inversibles plus appropriées aux images.



## EXEMPLE

- ▶ Entraînement d'un RealNVP sur le dataset MNIST et génération de data...
- ▶ On pourrait utiliser d'autres architectures inversibles plus appropriées aux images.



# BIBLIOGRAPHIE



Dinh, L., Sohl-Dickstein, J., and Bengio, S. (2017).

Density estimation using real NVP.

*In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings.* OpenReview.net.



Ermon, S. and Grover, A. (2023).

Normalizing flows.

<https://deepgenerativemodels.github.io/notes/flow/>.

Accessed: 2025-04-28.



Weng, L. (2018).

Flow-based deep generative models.

[lilianweng.github.io](https://lilianweng.github.io).



Wikipedia contributors (2024).

Flow-based generative model.

[https://en.wikipedia.org/wiki/Flow-based\\_generative\\_model#cite\\_note-27](https://en.wikipedia.org/wiki/Flow-based_generative_model#cite_note-27).

Accessed: 2025-04-28.