

UNIVERSITÉ PARIS  
DAUPHINE

# SAN : METRIZABILITY OF GAN

EVELINA SPAC  
CAPUCINE BRISSON

# PLAN

- Motivation & Contexte
- Notion de métrisabilité
- Slicing Adversarial Network (SAN)
- Résultats
- Perspectives et pistes d'amélioration

# **SLICING ADVERSARIAL NETWORK (SAN)**

**Comment entraîner les GAN de manière à ce qu'ils minimisent réellement une distance entre la distribution des données réelles et celle des données générées ?**

# CONCEPT DE MÉTRISABILITÉ - 3 CONDITIONS :

## OPTIMALITÉ DIRECTIONNELLE

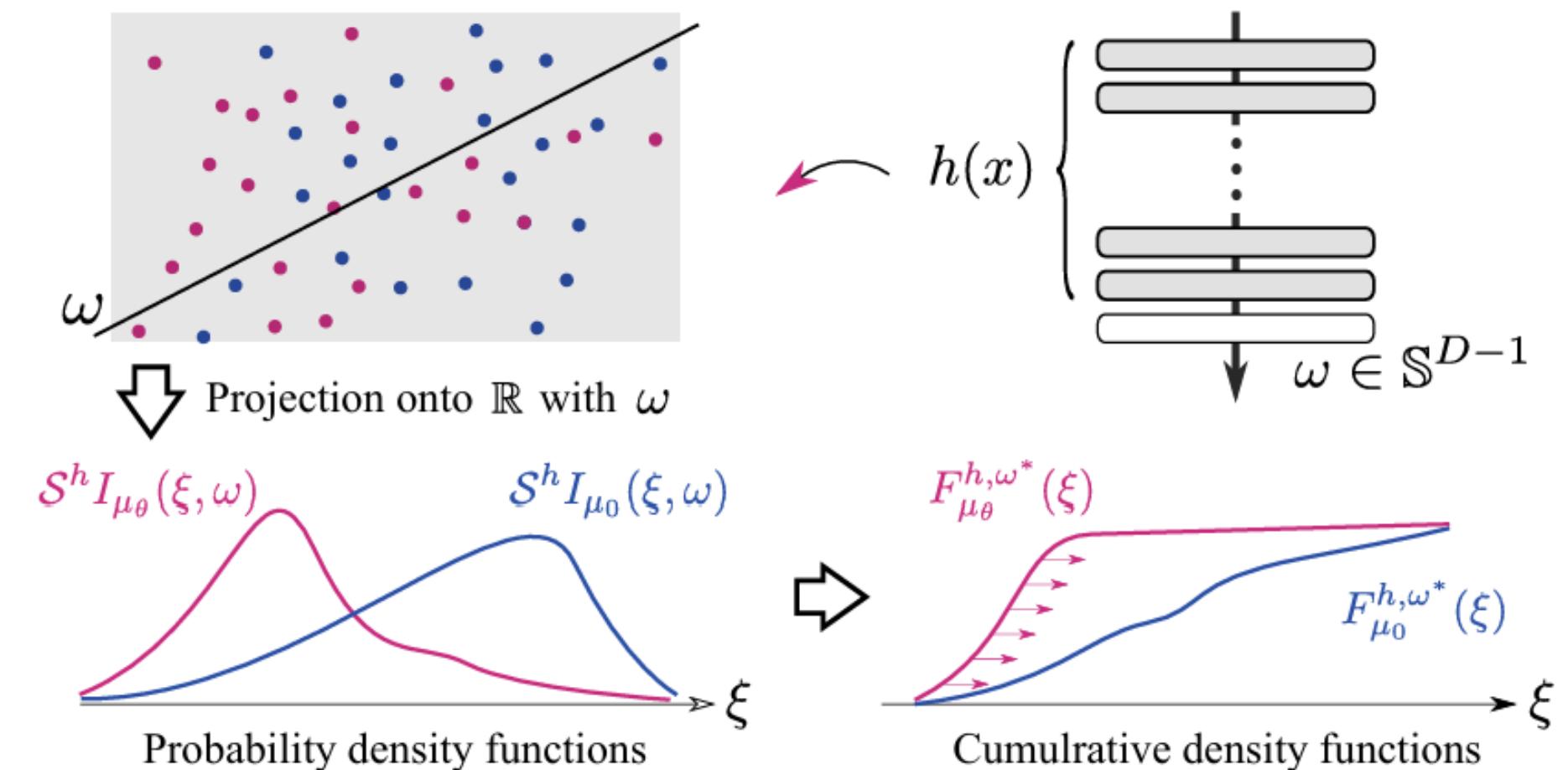
$$\omega^* = \arg \max_{\omega \in S^{D-1}} d_{\langle \omega, h \rangle}(\mu_0, \mu_\theta)$$

## SÉPARABILITÉ

$$F_{\mu_0}^{h,\omega^*}(\xi) \leq F_{\mu_\theta}^{h,\omega^*}(\xi), \quad \forall \xi \in R$$

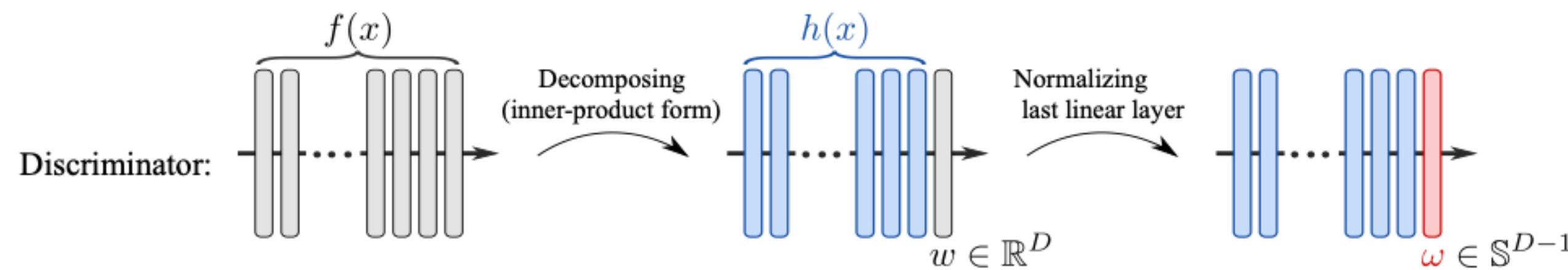
## INJECTIVITÉ

Préservation de l'information dans l'espace de caractéristiques



# IMPLEMENTATION DE SAN

SAN repose sur une stratégie d'optimisation découplée.



On sépare le problème d'optimisation du discriminateur en deux parties pour apprendre :

- la direction de projection optimale : composante directionnelle
- la représentation des données : composante fonctionnelle

$$\max_{\omega \in S^{d-1}, h \in \mathcal{F}(\mathcal{X})} \mathcal{V}^{\text{SAN}}(\omega, h; \mu_\theta) := \mathcal{V}(\langle \omega^-, h \rangle; \mu_\theta) + \lambda d_{\langle \omega, h^- \rangle}(\mu_0, \mu_\theta),$$

# FONCTIONS DE PERTES DU DISCRIMINATEUR

**Hinge Loss**, permet au discriminateur d'être entraîné sur des distributions tronquées

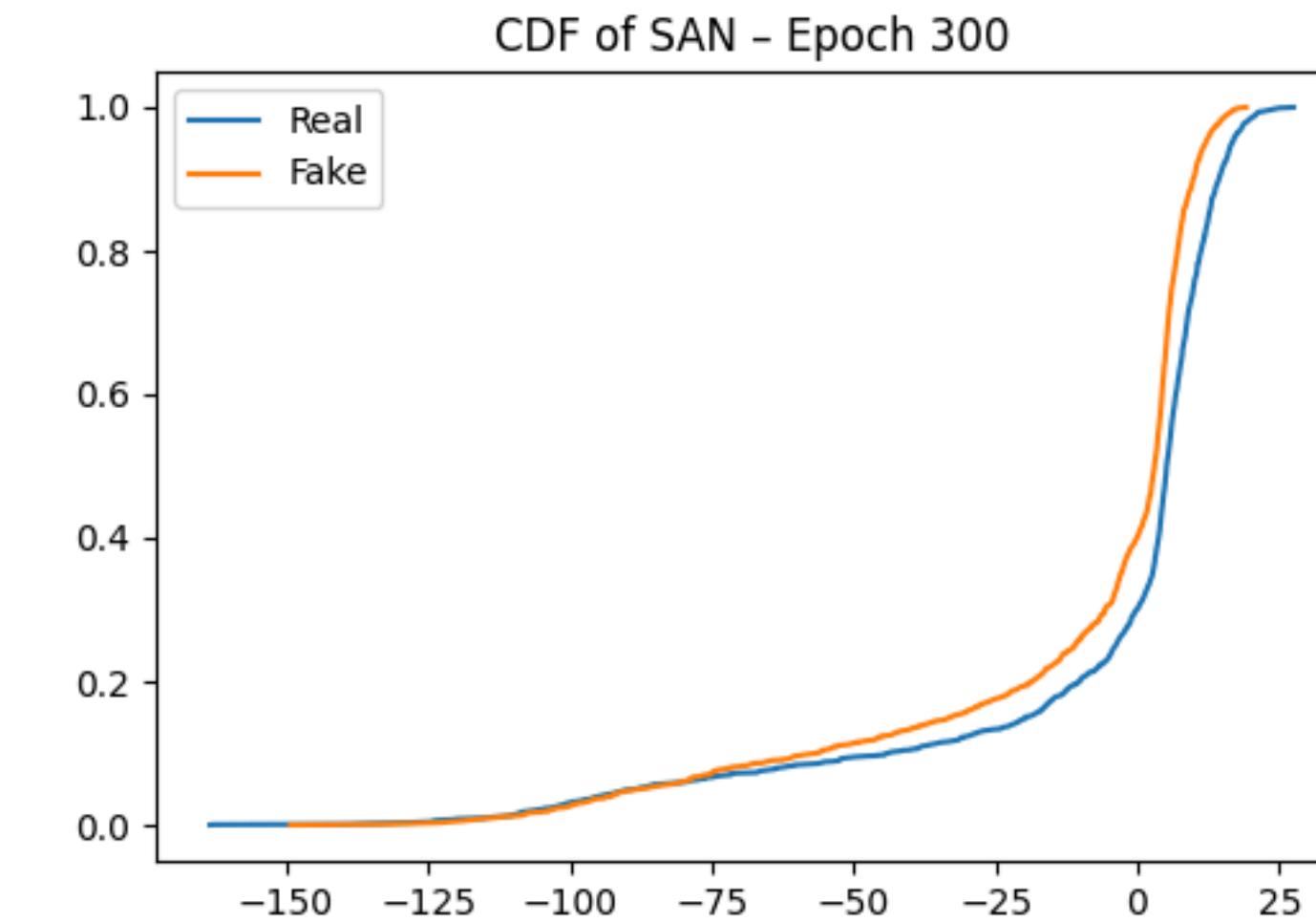
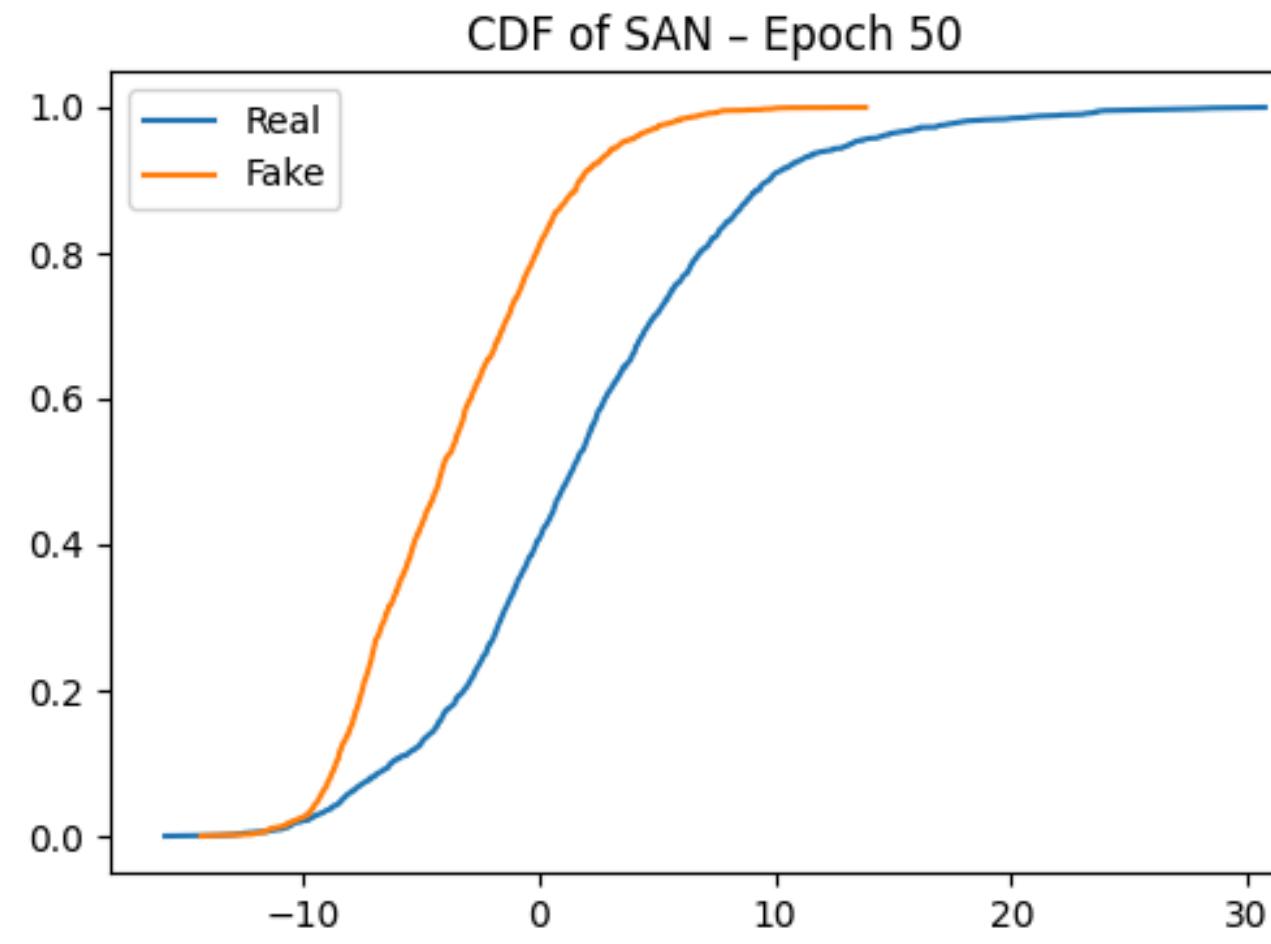
$$\mathcal{V}_{\text{hinge}}(\langle \omega, h \rangle; \mu_\theta) := E_{x \sim \mu_0} [\min(0, -1 + \langle \omega, h(x) \rangle)] + E_{x \sim \mu_\theta} [\min(0, -1 - \langle \omega, h(x) \rangle)].$$

**La loss directionnelle**

$$\sup_{\|\omega\|=1} (E_{x \sim \mu_0} [\langle \omega, h(x) \rangle] - E_{x \sim \mu_\theta} [\langle \omega, h(x) \rangle]).$$

Ainsi, le discriminateur apprend à bien représenter les données pour fournir au générateur un gradient informatif.

# SEPARABILITÉ DES FONCTIONS DE RÉPARTITIONS



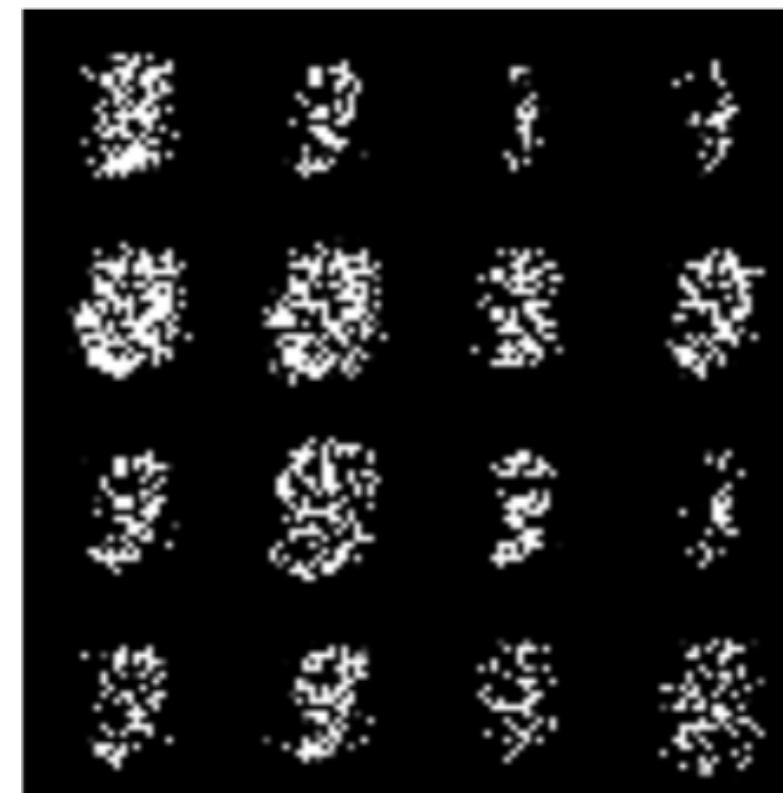
Evolution des CDF pendant l'entraînement d'un SAN GAN

# RÉSULTATS DES MODÈLES SUR MNIST

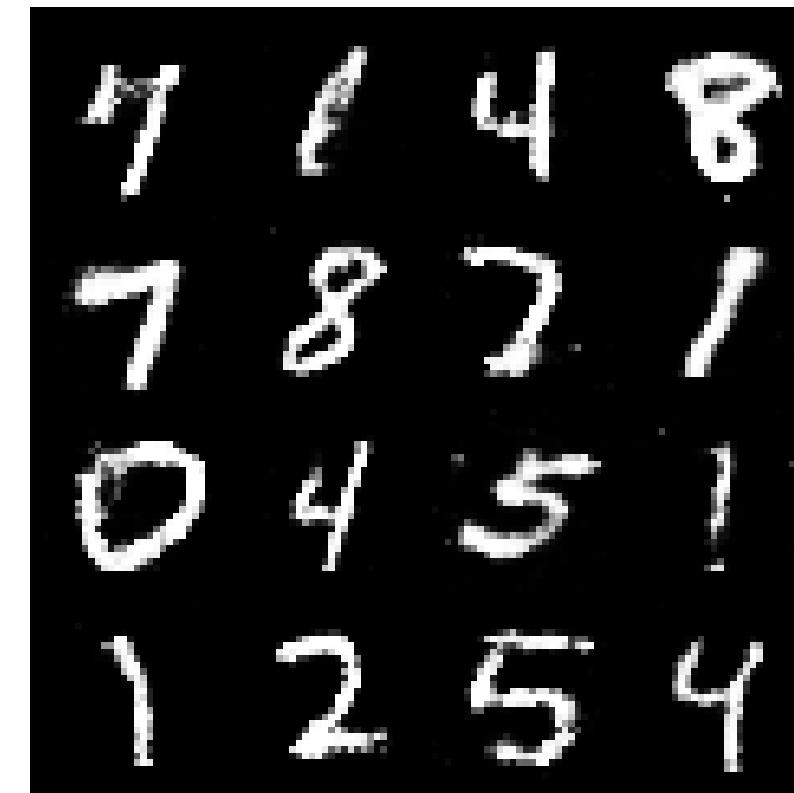
GAN Classique : Précision : 0.52 -- Recall : 0.23

SAN avec 100 epochs : Précision : 0.67 -- Recall : 0.53

SAN avec 300 epochs : Précision : 0.76 -- Recall : 0.59



SW-GAN avec 100 epochs et batch\_size  
64



SAN avec 100 epochs



SAN avec 300 epochs et batch\_size 128

# EXPERIENCES ET PISTES D'AMÉLIORATION

## EXPERIENCES

- Entraînement de SAN avec  $batch\_size = 64$  et  $128$
- Passage de *Leaky ReLU* de  $0.2$  à  $0.1$
- Test avec un *learning rate* de  $0.001$  et  $0.002$
- Test avec un *learning rate scheduler* : *ExponentialLR* ( $gamma = 0.995$ )

## PISTES D'AMÉLIORATION

- Implémenter un **SAN conditionnel** sur les classes en incorporant les classes des paramètres dans le générateur et le discriminant.
- Implémenter SAN sur des GANs déjà performants : **BigGAN**, **StyleGAN**