UNIFYING GANS AND SCORE-BASED DIFFUSION AS GENERATIVE PARTICLE MODELS

NeurIPS 2023, December 10th – 16th

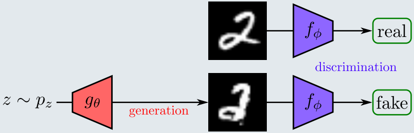
J.-Y. Franceschi,[1] M. Gartrell,[1]
L. Dos Santos,[1,*] T. Issenhuth,[1,2,*]
E. de Bézenac,[3,*] M. Chen,[4,*]
A. Rakotomamonjy[1,*]

[1]Criteo AI Lab, Paris, France
[2]LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, Marne-la-Vallée, France
[3]Seminar for Applied Mathematics, D-MATH, ETH Zürich, Zürich-8092, Switzerland
[4]Valeo.ai, Paris, France
[*]Randomly chosen order

CRITEO AI Lab

GANs

→ Generator trained by discriminating true vs fake data.



- Generator (manifold learning).
- Close to SOTA performance.
- Harder to optimize.
- Fast inference.

Diffusion

→ Learns to progressively reverse a data degradation process.

- No generator (operates on the data space).
- SOTA performance.
- Easier to optimize.
- Slow inference.

CRITEO AI Lab

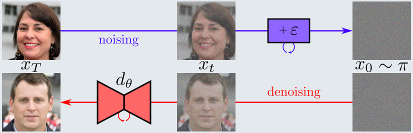
GANs

→ Generator trained by discriminating true vs fake data.

- Generator (manifold learning).
- Close to SOTA performance.
- Harder to optimize.
- Fast inference.

Diffusion

→ Learns to progressively reverse a data degradation process.

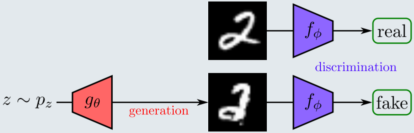


- No generator (operates on the data space).
- SOTA performance.
- Easier to optimize.
- Slow inference.

CRITEO AI Lab

GANs

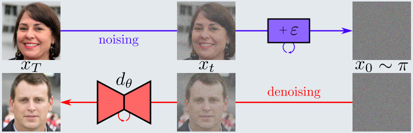
→ Generator trained by discriminating true vs fake data.



- Generator (manifold learning).
- Close to SOTA performance.
- Harder to optimize.
- Fast inference.

Diffusion

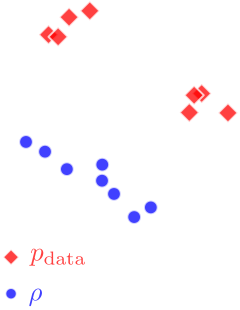
→ Learns to progressively reverse a data degradation process.



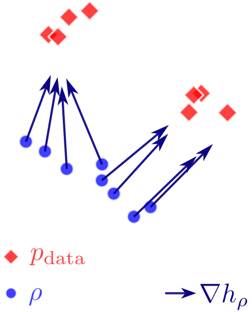
- No generator (operates on the data space).
- SOTA performance.
- Easier to optimize.
- Slow inference.

CRITEO AI Lab

- Models make a particle distribution ρ_t evolve with time t.



- Models make a particle distribution ρ_t evolve with time t.
- Particles follow a gradient vector field ∇h_{ρ_t}:
  - during inference;



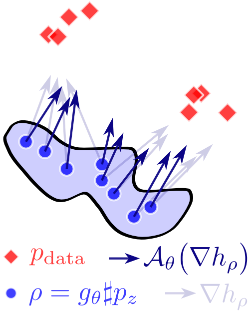
Definition (Particle Models, PMs)

$$x_0 \sim \pi = \rho_0, \qquad \mathrm{d}x_t = \nabla h_{\rho_t}(x_t)\,\mathrm{d}t,$$

where t is the inference time.

- Models make a particle distribution ρ_t evolve with time t.
- Particles follow a gradient vector field ∇h_{ρ_t}:
  - during inference;
  - or smoothed during generator training with loss:

$$\mathcal{L}_{\theta_t} = -\mathbb{E}_{z \sim p_z} h_{\rho_t}\big(g_{\theta_t}(z)\big).$$



Definition (Interacting Particle Models, Int-PMs)

$$\mathrm{d}g_{\theta_t}(z) = \eta\big[\mathcal{A}_{\theta_t}(z)\big]\big(\nabla h_{\rho_t}\big)\,\mathrm{d}t,$$

where t is the training time of the generator $g_{\theta_t}(z)$.

CRITEO AI Lab

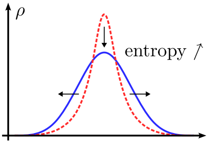
Model	Generator	Flow type ∇h_{ρ_t}

Model	Generator	Flow type ∇h_{ρ_t}
Score-based diffusion models	✗	$\alpha_t \nabla \log \left[p_{\mathrm{data}} \star k_{\mathrm{RBF}}^{\sigma(t)}\right] - \beta_t \nabla \log \rho_t$

Example for NCSN (Song et al., 2019) using Langevin sampling:

$$\mathrm{d}x_t = \nabla \log[p_{\mathrm{data}} \star k_{\mathrm{RBF}}^{\sigma}](x_t)\,\mathrm{d}t \boxed{+\sqrt{2}\,\mathrm{d}W_t},$$

$$\mathrm{d}x_t = \nabla \log[p_{\mathrm{data}} \star k_{\mathrm{RBF}}^{\sigma}](x_t)\,\mathrm{d}t \boxed{-\underbrace{\nabla \log \rho_t}_{\text{gen. score}}(x_t)\,\mathrm{d}t}.$$



- In diffusion, particles follow a **log ratio gradient**.

Model	Generator	Flow type ∇h_{ρ_t}
Score-based diffusion models	✗	$\alpha_t \nabla \log\left[p_{\text{data}} \star k_{\text{RBF}}^{\sigma(t)}\right] - \beta_t \nabla \log \rho_t$
GANs	✓	$-\nabla\left(c \circ f_{\rho_t}\right)$, where f_{ρ_t} is a discriminator between ρ_t and p_{data}

With gradient descent-ascent and generator loss:

$$\mathcal{L}_\theta = \mathbb{E}_{z \sim p_z}\left[\left(c \circ f_\rho\right)\left(g_\theta(z)\right)\right].$$

$\cancel{\min_g \max_f \mathcal{L}(f, g)}$

(Goodfellow et al., 2014)

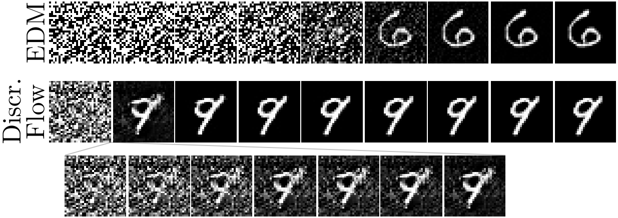
- In GANs, particles follow the **discriminator gradient**.

Model	Generator	Flow type ∇h_{ρ_t}
Score-based diffusion models Score GANs	✗ ✓	$\alpha_t \nabla \log \left[p_{\mathrm{data}} \star k_{\mathrm{RBF}}^{\sigma(t)} \right] - \beta_t \nabla \log \rho_t$
Discriminator Flows GANs	✗ ✓	$-\nabla \left(c \circ f_{\rho_t} \right)$, where f_{ρ_t} is a discriminator between ρ_t and p_{data}

Claim

It is possible to train:

- a generator with diffusion (Score GAN);
- a GAN without a generator (Discriminator Flow).



Animated samples:
https://jyfranceschi.fr/publications/gpm/.