

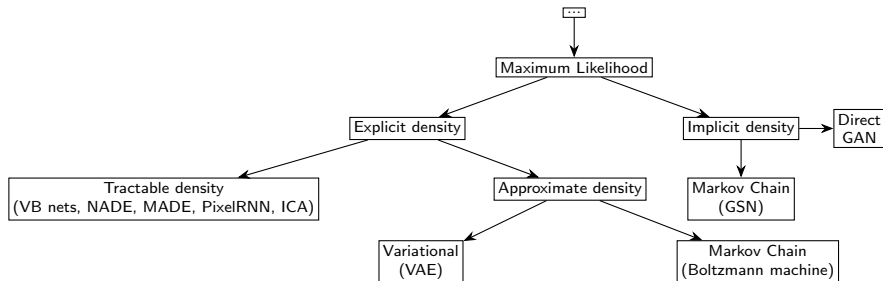
A Mathematical Perspective on GANs and f-GANs

May 2025

Key References on Generative Adversarial Networks

- The original GAN formulation introducing the adversarial training framework. [Goodfellow et al., 2014]
- Wasserstein GANs (WGAN): applies the Wasserstein distance to improve training stability. [Arjovsky et al., 2017]
- A concise mathematical survey of GAN developments and challenges. [Wang, 2020]

Taxonomy of Generative Models



Adapted from Goodfellow (2016)

Prescribed vs. Implicit Probabilistic Models

Definition from [Mohamed and Lakshminarayanan, 2016]

- **Prescribed probabilistic models** are those that provide an explicit parametric specification of the distribution of an observed random variable x , specifying a log-likelihood function

$$\log q_{\theta}(x)$$

with parameters θ .

- **Implicit probabilistic models** that define a stochastic procedure that directly generates data. Natural approach for problems
 - climate and weather,
 - population genetics,
 - ecology.

Comparative Overview of Generative Models

| Model | Likelihood | Sampling | Quality | Stability | Coverage |
|-------------------|-----------------------|------------------------------|---------------------------|----------------------|-----------|
| GANs | Implicit | <i>Very fast</i> | High realism | Unstable | Poor |
| VAEs | Explicit (ELBO) | Fast | Moderate (blurry) | <i>Stable</i> | Good |
| Normalizing Flows | Exact, tractable | Fast | Moderate | <i>Stable</i> | Good |
| Autoregressive | Exact, tractable | Slow (sequential) | Excellent (sequential) | <i>Stable</i> | Excellent |
| Diffusion Models | Score-based tractable | <i>Very slow (iterative)</i> | <i>Very high fidelity</i> | <i>Highly stable</i> | Good |

Table: Comparison of Generative Models

Generative Modeling: Normalizing Flows and VAEs

Design parameterized densities with huge capacity!

- **Normalizing flows:** a sequence of invertible non-linear transformations to a simple base distribution $p_Z(z)$:

$$p(\mathbf{x} \mid \theta_{0:k}) = p_Z(z), \quad z = f_{\theta_k}^{-1} \circ \dots \circ f_{\theta_0}^{-1}(\mathbf{x}).$$

Each $f_{\theta_j}^{-1}$ must be invertible with a tractable log-determinant of its Jacobian.

- **Variational Autoencoders (VAEs):** latent-variable models where inference networks specify the decoder parameters:

$$p(\mathbf{x}, y \mid \theta) = p(\mathbf{x} \mid f_{\theta}(y)) p_Y(y).$$

The marginal likelihood $p(\mathbf{x})$ is maximized via the ELBO.

GANs: Density-Free Models

”Many training methods rely on obtaining a probability density for $G_\theta(\mu)$; for example, this is used in normalising flows. However, this is not in general computable, perhaps due to the complicated internal structure of G_θ . Instead, GANs examine the statistics of samples from $G_\theta(\mu)$, and seek to match the statistics of the model to the statistics of the data. Most typically, this is a learnt scalar statistic, called the discriminator. An optimally-trained generator is one for which

$$\mathbb{E}_{X \sim \text{model}}[F(X)] = \mathbb{E}_{X \sim \text{data}}[F(X)]$$

for all statistics F , so that there is no possible statistic (or “witness function” in the language of integral probability metrics) that the discriminator may learn to represent, so as to distinguish real from fake.” [Kidger et al., 2021].

GANs: Density-Free Models

Generative Adversarial Networks (GANs) instead use an unrestricted generator $G_{\theta_g}(z)$ such that

$$p(x \mid \theta_g) = p_Z(\{z\}), \quad \text{where } \{z\} = G_{\theta_g}^{-1}(x).$$

- **Problem:** the inverse image of $G_{\theta_g}(z)$ may be huge!
- **Problem:** it's likely intractable to preserve volume through $G(z; \theta_g)$.

So, we can't evaluate $p(x \mid \theta_g)$ and we can't learn θ_g by maximum likelihood.

GANs: Discriminators

GANs learn by comparing model samples with examples from \mathcal{D} .

- Sampling from the generator is easy:

$$\hat{x} = G_{\theta_g}(\hat{z}), \quad \hat{z} \sim p_Z(z).$$

- Given a sample \hat{x} , a discriminator tries to distinguish it from true examples:

$$D(x) = \Pr(x \sim p_{\text{data}}).$$

- The discriminator “supervises” the generator network.

GANs: Generator + Discriminator

- Let $z \in \mathbb{R}^m$ and $p_Z(z)$ be a simple base distribution.
- The generator

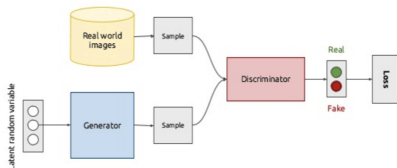
$$G_{\theta_g}(z): \mathbb{R}^m \longrightarrow \tilde{\mathcal{D}}$$

is a deep neural network.

- The discriminator

$$D_{\theta_d}(x): \mathcal{D} \cup \tilde{\mathcal{D}} \longrightarrow (0, 1)$$

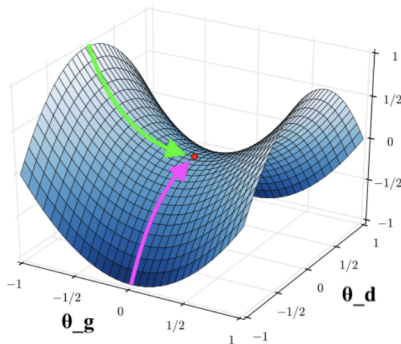
is also a deep neural network.



Saddle-Point Optimization

Learn $G_{\theta_g}(z)$ and $D_{\theta_d}(x)$ jointly via the objective $V(\theta_d, \theta_g)$:

$$\min_{\theta_g} \max_{\theta_d} \underbrace{\mathbb{E}_{x \sim p_{\text{data}}} [\log D_{\theta_d}(x)]}_{\text{likelihood of true data}} + \underbrace{\mathbb{E}_{z \sim p_Z(z)} [\log(1 - D_{\theta_d}(G_{\theta_g}(z)))]}_{\text{likelihood of generated data}}.$$



Problems with GANs

- **Vanishing gradients:** the discriminator becomes “too good” and the generator gradient vanishes.
- **Non-Convergence:** the generator and discriminator oscillate without reaching an equilibrium.
 - **Mode Collapse:** the generator distribution collapses to a small set of examples.
 - **Mode Dropping:** the generator distribution doesn't fully cover the data distribution.

Problems: Non-Convergence

Simultaneous gradient descent is not guaranteed to converge for minimax objectives.

- Goodfellow et al. only showed convergence when updates are made in the function space [Goodfellow et al., 2014].
- The parameterization of D_{θ_d} and G_{θ_g} results in a highly non-convex objective. (see Supplementary Material [Luo and Yang, 2024])
- In practice, training tends to oscillate—updates “undo” each other.

A Possible Solution: Alternative Divergences

There are a large variety of divergence measures for distributions:

- **Integral Probability Metrics:** (e.g. Earth Movers Distance, Maximum Mean Discrepancy)

$$\gamma_{\mathcal{F}}(P\|Q) = \sup_{f \in \mathcal{F}} \left| \int f \, dP - \int f \, dQ \right|$$

- Wasserstein GANs [Arjovsky et al., 2017], Cramer GANs [Bellemare et al., 2017], Sobolev GANs [Mroueh et al., 2017] and more.
- **f -Divergences:** (e.g. Jensen–Shannon, Kullback–Leibler)

$$D_f(P\|Q) = \int_{\mathcal{X}} q(x) f\left(\frac{p(x)}{q(x)}\right) dx$$

- GANs [Goodfellow et al., 2014], f -GANs [Nowozin et al., 2016], and more.

Definition of a Divergence

A *divergence* is any functional

$$D(P\|Q) = D(P(x) \| Q(x))$$

that measures the discrepancy between two probability distributions P and Q . It typically satisfies:

- $D(P\|Q) \geq 0$ (*non-negativity*)
- $D(P\|Q) = 0$ if and only if $P = Q$ almost everywhere

Example: the *Kullback–Leibler divergence*

$$D_{\text{KL}}(P\|Q) = \int_{\mathcal{X}} p(x) \log \frac{p(x)}{q(x)} dx.$$

Classical GAN as Jensen–Shannon Divergence Minimization

Minimax objective:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}} [\log D(x)] + \mathbb{E}_{z \sim p_Z} [\log(1 - D(G(z)))].$$

Optimal discriminator for fixed G (see [Wang, 2020]):

$$D_G^*(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_G(x)}.$$

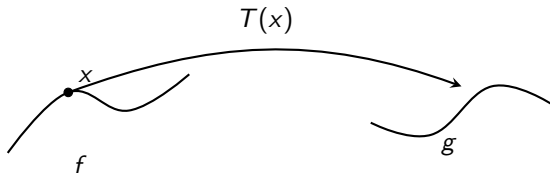
Plugging D_G^* back in gives

$$\begin{aligned} V(D_G^*, G) &= \int p_{\text{data}}(x) \log \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_G(x)} dx \\ &\quad + \int p_G(x) \log \frac{p_G(x)}{p_{\text{data}}(x) + p_G(x)} dx \\ &= -\log 4 + 2 \text{JS}(p_{\text{data}} \| p_G). \end{aligned}$$

Monge Problem

- Given a “source” density $f(x)$ and a “target” density $g(y)$.
- Seek a transport map T that pushes f onto g , i.e. $T_{\#}f = g$.
- Minimize the total transport cost:

$$\min_T \int_{\mathcal{X}} |x - T(x)| f(x) dx.$$



Wasserstein Distance (Intuitive View)

- Think of P and Q as two piles of “earth” (mass) on a space \mathcal{X} .
- A *transport plan* $\gamma(x, y)$ tells you how much mass to move from point x (in P) to point y (in Q).
- Each unit of mass moved from x to y incurs a cost $d(x, y)$, the distance between x and y .
- The Wasserstein distance $W_1(P, Q)$ is the *minimum total cost* (per unit mass) over all valid transport plans:

$$W_1(P, Q) = \min_{\gamma \in \Pi(P, Q)} \int_{\mathcal{X} \times \mathcal{X}} d(x, y) d\gamma(x, y).$$

- In other words, it's the least “work” needed to reshape P into Q .

Wasserstein-1 Distance: Primal vs. Dual

Primal (Monge–Kantorovich)

$$W_1(P, Q) = \inf_{\gamma \in \Pi(P, Q)} \int_{\mathcal{X} \times \mathcal{X}} d(x, y) \, d\gamma(x, y),$$

where $\Pi(P, Q)$ is the set of all couplings of P and Q .

Kantorovich–Rubinstein Dual

$$W_1(P, Q) = \sup_{\substack{f: \mathcal{X} \rightarrow \mathbb{R} \\ \|f\|_{\text{Lip}} \leq 1}} \left\{ \mathbb{E}_{x \sim P} [f(x)] - \mathbb{E}_{x \sim Q} [f(x)] \right\}.$$

Here $\|f\|_{\text{Lip}} \leq 1$ means $|f(x) - f(y)| \leq d(x, y)$ for all x, y .

In WGANs, the critic f_w is a neural network designed to approximate this supremum under a (soft) 1-Lipschitz constraint.

Wasserstein GAN: Applying the Dual

Replace the JS-based min-max with the Kantorovich–Rubinstein dual for W_1 :

$$\min_G \max_{f: \|f\|_{\text{Lip}} \leq 1} \left\{ \mathbb{E}_{x \sim p_{\text{data}}} [f(x)] - \mathbb{E}_{z \sim p_z} [f(G_\theta(z))] \right\}.$$

- f is now called the *critic* (not a discriminator) and must be 1-Lipschitz.
- For fixed G , the inner maximization approximates the Wasserstein-1 distance $W_1(p_{\text{data}}, p_G)$.
- Minimizing w.r.t. G then pushes p_G toward p_{data} in the Earth–Mover metric.

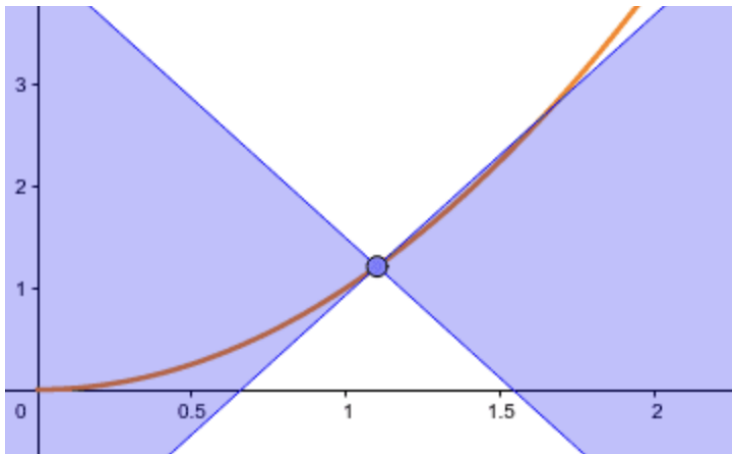


Figure: Example of a non-Lipschitz function

The f -divergence Family

- For distributions P, Q on \mathcal{X} with densities p, q :

$$D_f(P\|Q) = \int_{\mathcal{X}} q(x) f\left(\frac{p(x)}{q(x)}\right) dx.$$

- Here $f: \mathbb{R}_+ \rightarrow \mathbb{R}$ is convex, lower-semicontinuous, and $f(1) = 0$.
- Different choices of f yield KL, JS, Hellinger, ...

$$f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y), \quad \theta \in [0, 1].$$

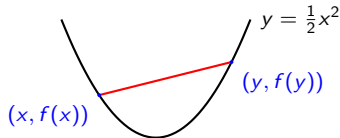


Figure: Convex function. The straight line (in red) joining two points on the curve stays above the curve itself, illustrating convexity.

f-GAN: Generalized GAN via f -Divergences

1. f -Divergence and Dual Representation

$$D_f(P\|Q) = \int q(x) f\left(\frac{p(x)}{q(x)}\right) dx = \sup_{T:\mathcal{X}\rightarrow\mathbb{R}} \left\{ \mathbb{E}_{x\sim P}[T(x)] - \mathbb{E}_{x\sim Q}[f^*(T(x))] \right\},$$

where $f^*(t) = \sup_{u>0} \{u t - f(u)\}$ (convex conjugate or Fenchel–Legendre transform).

2. f-GAN Minimax Objective

$$\min_{\theta} \max_{\phi} \mathbb{E}_{x\sim p_{\text{data}}} [T_{\phi}(x)] - \mathbb{E}_{z\sim p_z} [f^*(T_{\phi}(G_{\theta}(z)))].$$

f-GAN: Generalized GAN via f -Divergences

3. Special Cases

- *Jensen–Shannon GAN*: $f(u) = u \log u - (u + 1) \log \frac{u+1}{2}$,
 $f^*(t) = \log(1 + e^t)$.
- *KL GAN*: $f(u) = u \log u$.
- *Reverse KL GAN*: $f(u) = -\log u$.
- *Pearson χ^2 GAN, Hellinger GAN, ...*

Advantages of f-GAN I

- **Flexible Divergence Choice:** Supports any f -divergence (e.g., KL, reverse KL, Pearson χ^2 , total variation).
- **Unified Framework:** Training is a variational lower bound on an arbitrary f -divergence.
- **More Stable Training:** Choose divergences whose convex conjugates yield well-behaved, non-saturating gradients.
- **Links to Likelihood Methods:** Relates adversarial training to variational inference (e.g., VAEs), enabling hybrid objectives.
- **Control Over Diversity vs. Fidelity:** Select divergences to emphasize mode coverage (e.g., reverse KL) or high-density regions.
- **Empirical Gains:** Better divergence metrics, more stable loss curves, and improved Inception/FID scores in some benchmarks.

References I



Arjovsky, M., Chintala, S., and Bottou, L. (2017).

Wasserstein GAN.

In International Conference on Machine Learning (ICML).



Bellemare, M. G., Danihelka, I., Dabney, W., Mohamed, S., Lakshminarayanan, B., Hoyer, S., and Munos, R. (2017).

The cramer distance as a solution to biased wasserstein gradients.

arXiv preprint arXiv:1705.10743.







Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014).

Generative adversarial nets.

In Advances in Neural Information Processing Systems (NeurIPS).

References II

-  Kidger, P., Foster, J., Li, X., and Lyons, T. J. (2021).
Neural sdes as infinite-dimensional gans.
In International conference on machine learning, pages 5453–5463.
PMLR.
-  Luo, Y. and Yang, Z. (2024).
Dyngan: Solving mode collapse in gans with dynamic clustering.
IEEE Transactions on Pattern Analysis and Machine Intelligence.
-  Mohamed, S. and Lakshminarayanan, B. (2016).
Learning in implicit generative models.
arXiv preprint arXiv:1610.03483.
-  Mroueh, Y., Li, C.-L., Sercu, T., Raj, A., and Cheng, Y. (2017).
Sobolev gan.
arXiv preprint arXiv:1711.04894.

References III



Nowozin, S., Cseke, B., and Tomioka, R. (2016).

f-GAN: Training generative neural samplers using variational divergence minimization.

arXiv preprint arXiv:1606.00709.



Wang, Y. (2020).

A mathematical introduction to generative adversarial nets (gan).

arXiv preprint arXiv:2009.00169.