Information Theoretic Foundations of Generative Models

Part I: Generative Adversarial Networks

Lalitha Sankar and Monica Welfert

ISIT 2025 Tutorial, Ann Arbor, MI

June 19, 2025

Supported in part by NSF grants CIF-1815361, CIF-2007688, DMS-2134256, and SCH-2205080

Overview of Part I

The Need for Synthetic Data

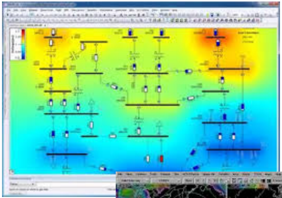
Generative Adversarial Networks: Theory

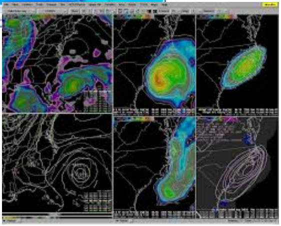
Generative Adversarial Networks: Practice

Motivation

The Need for Synthetic Data

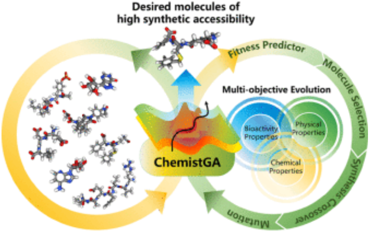
- Complex networks (e.g., electric power, flight, robotics) need to be designed and tested with synthetic/simulated data
- Complex systems involve randomness and require probabilistic modeling
- Design and testing of algorithms require simulated data





What Can Synthetic Data Do for Us?

- **Increasing need for generative methodologies** for autonomous systems, drug discovery, interactive agents, ...



Generative methods grounded in theoretical guarantees are key

The Need for Synthetic Data

Data-driven distribution modeling: holy grail of statistics

- Backbone of simulation models for complex systems
- Increasingly complex distributions are "designed" to match observed data (e.g., arc-tan exponential, Bernoulli-gamma-Gaussian, ...)
- Computational complexity is significant and continual need for updating models

The Need for Synthetic Data

Data-driven distribution modeling: holy grail of statistics

- Backbone of simulation models for complex systems
- Increasingly complex distributions are "designed" to match observed data (e.g., arc-tan exponential, Bernoulli-gamma-Gaussian, ...)
- Computational complexity is significant and continual need for updating models

What does modern ML have to offer?

- Shallow neural networks: universal approximators of smooth functions
- Deeper models used in practice to capture complex functionals
- Made possible by massive compute (GPUs) but theory matters

The Need for Synthetic Data

Data-driven distribution modeling: holy grail of statistics

- Backbone of simulation models for complex systems
- Increasingly complex distributions are "designed" to match observed data (e.g., arc-tan exponential, Bernoulli-gamma-Gaussian, ...)
- Computational complexity is significant and continual need for updating models

What does modern ML have to offer?

- Shallow neural networks: universal approximators of smooth functions
- Deeper models used in practice to capture complex functionals
- Made possible by massive compute (GPUs) but theory matters

This tutorial: Generative Adversarial Networks (GANs) and Diffusion Models

Generative Adversarial Networks: Theoretical Formulation

From Latent Models to GANs

Latent Models for Data Distributions

- Models high-dimensional data $\mathbf{x}$ as generated by a lower dimensional latent vector $\mathbf{z}$ (simplest model: $\mathbf{x} = \mathbf{W}\mathbf{z} + \mu + \mathbf{e}$)
- Many approaches considered:
  - graphical models[1]
  - probabilistic PCA[2]
  - Undirected (Boltzmann machines)[3], deep belief nets[4]
- Typically learn distributions up to a normalization constant
- Sampling requires MCMC methods

Latent Models served as a starting point for GANs (Goodfellow et al., 2014)

---

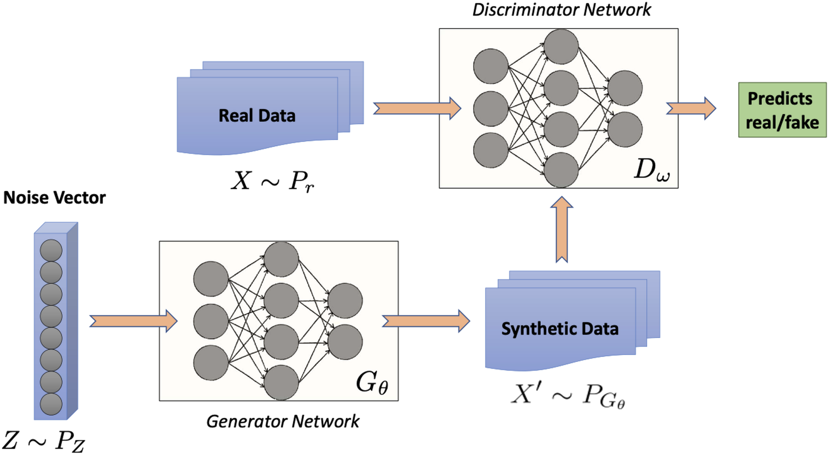
[1] Bartholomew (1987)

[2] Tipping and Bishop (1999)

[3] Hinton et al. (2006)

[4] Salakhutdinov and Hinton (2009)

Generative Adversarial Networks (GANs)



A zero-sum game between two agents via a coupled objective[1]

[1] Goodfellow et al. (2014)

Optimization Problem in GANs

- GANs are an adversarial min-max game between two agents:
  - generator $G_\theta : \mathcal{Z} \to \mathcal{X}$, $\theta \in \Theta$ and
  - discriminator $D_\omega : \mathcal{X} \to [0,1]$, $\omega \in \Omega$
- The discriminator $D_\omega \in [0,1]$ is a soft binary classifier
  - $D_\omega(x)$: probability that $x \in \mathcal{X}$ labeled as real (label $Y = 1$)
  - $1 - D_\omega(x)$: probability that $x \in \mathcal{X}$ labeled as fake (label $Y = 0$)
- The two-agent game involves a common coupled value function $V(\theta, \omega)$ leading to the min-max optimization:

$$\inf_{\theta \in \Theta} \sup_{\omega \in \Omega} V(\theta, \omega)$$

- $V(\theta, \omega)$ chosen for D to maximize and G to minimize
- Classifier D: maximize (average) negative loss $\ell(y, D_\omega(x))$

$$V_{\mathsf{VG}}(\theta, \omega) = \mathbb{E}_{X \sim P_r}[-\ell(1, D_\omega(X))] + \mathbb{E}_{X \sim P_{G_\theta}}[-\ell(0, D_\omega(X))]$$

Vanilla GAN: Value Function and Optimal Strategies

Goodfellow *et al.* (2014) introduced the (now called) *vanilla GAN*

- Log-loss:
  $\ell(y, D_\omega(x)) = -\log(yD_\omega(x) + (1-y)(1-D_\omega(x)), y \in \{0,1\}$

$$V_{\mathsf{VG}}(\theta, \omega) = \mathbb{E}_{X\sim P_r}[\log D_\omega(X)] + \mathbb{E}_{X\sim P_{G_\theta}}[\log(1 - D_\omega(X))]$$

- Assuming sufficiently large Ω, for fixed G_θ the optimal discriminator is

$$D_{\omega^*}(x) = \frac{p_r(x)}{p_r(x) + p_{G_\theta}(x)}$$

where p_r and p_{G_θ} are the densities of P_r and P_{G_θ}, respectively

Vanilla GAN: Optimal Strategies

- Fixed G_θ: the optimal discriminator is given by

$$D_{\omega^*}(x) = \frac{p_r(x)}{p_r(x) + p_{G_\theta}(x)}$$

- For this D_{ω^*}, the generator's optimization simplifies to

$$\boxed{\inf_{\theta \in \Theta} 2D_{\mathsf{JS}}(P_r \| P_{G_\theta}) - \log(4)}$$

where $D_{\mathsf{JS}}(P_r \| P_{G_\theta})$ is the **Jensen-Shannon divergence**:

$$D_{\mathsf{JS}}(P \| Q) = \frac{1}{2} D_{\mathsf{KL}}(P \| M) + \frac{1}{2} D_{\mathsf{KL}}(Q \| M), \quad M = \frac{P+Q}{2}$$

- $P_r = P_{G_\theta}$ iff $D_{\omega^*}(x) = \frac{1}{2}$ for all $x \in \mathcal{X}$ and $D_{\mathsf{JS}}(P_r \| P_{G_\theta}) = 0$

Beyond vanilla GAN

- JSD objective resulted from choosing the logistic/CE loss
- Can we obtain other probability similarity/distance metrics?
  - by changing the value function?
  - considering different classification losses?
- Nowozin et al. (2016): start with variational form of f-divergence[1] involving the convex conjugate[2]:

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

[2] Every convex, lower-semicontinuous function f has a convex conjugate function f^*, also known as Fenchel conjugate

Beyond vanilla GAN: f-GANs

- Nowozin *et al.* (2016) proposed f-GANs minimizing any f-divergence

$$V_f(\theta,\omega) = \mathbb{E}_{X\sim P_r}[D_\omega(X)] + \mathbb{E}_{X\sim P_{G_\theta}}[-f^*(D_\omega(X))] \quad ,$$

$D_\omega : \mathcal{X} \to \mathbb{R}$ and $f^*(t) = \sup_{u\in \mathrm{dom}(f)}[tu - f(u)]$

- Lower bound on f-divergence is tight if:

$$\exists\, \omega^* \in \Omega : D_{\omega^*}(x) = f'\left(\frac{p_r(x)}{p_{G_\theta}(x)}\right) \Rightarrow \sup_{\omega\in\Omega} V_f(\theta,\omega) = D_f(P_r\|P_{G_\theta})$$

Name	$D_f(P\|Q)$	Generator $f(u)$	$T^*(x)$
Kullback-Leibler	$\int p(x)\log\frac{p(x)}{q(x)}\,\mathrm{d}x$	$u\log u$	$1+\log\frac{p(x)}{q(x)}$
Reverse KL	$\int q(x)\log\frac{q(x)}{p(x)}\,\mathrm{d}x$	$-\log u$	$-\frac{q(x)}{p(x)}$
Pearson χ^2	$\int \frac{(q(x)-p(x))^2}{p(x)}\,\mathrm{d}x$	$(u-1)^2$	$2(\frac{p(x)}{q(x)}-1)$
Squared Hellinger	$\int\left(\sqrt{p(x)}-\sqrt{q(x)}\right)^2\,\mathrm{d}x$	$(\sqrt{u}-1)^2$	$(\sqrt{\frac{p(x)}{q(x)}}-1)\cdot\sqrt{\frac{q(x)}{p(x)}}$
Jensen-Shannon	$\frac{1}{2}\int p(x)\log\frac{2p(x)}{p(x)+q(x)}+q(x)\log\frac{2q(x)}{p(x)+q(x)}\,\mathrm{d}x$	$-(u+1)\log\frac{1+u}{2}+u\log u$	$\log\frac{2p(x)}{p(x)+q(x)}$
GAN	$\int p(x)\log\frac{2p(x)}{p(x)+q(x)}+q(x)\log\frac{2q(x)}{p(x)+q(x)}\,\mathrm{d}x-\log(4)$	$u\log u-(u+1)\log(u+1)$	$\log\frac{p(x)}{p(x)+q(x)}$

Table of f-divergences and corresponding $D_{\omega^*}(=T^*(x))$ (Nowozin et al., 2016)

Beyond vanilla GAN (VG): CPE Loss-Based GANs

- Value function in terms of a general classification loss

$$V_{\text{gen}}(\theta, \omega) = \mathbb{E}_{X \sim P_r}[-\ell(1, D_\omega(X))] + \mathbb{E}_{X \sim P_{G_\theta}}[-\ell(0, D_\omega(X))]$$

- Kurri et al. (2021) consider a broad class of class probability estimation (CPE) losses[1]

[1] Kurri et al. (2021)

Beyond vanilla GAN (VG): CPE Loss-Based GANs

- $\ell(y, \hat{y})$ - any *class probability estimation* (CPE) loss
  - $\hat{y} \in [0, 1]$ is a soft prediction of $y \in \{0, 1\}$
- Kurri et al. (2021) showed that $V(\theta, \omega)$ expressed using ℓ yields a meaningful objective provided ℓ satisfies:

$$\ell(1, \hat{y}) + \ell(0, \hat{y}) \geq \ell\,(1, 0.5) + \ell(0, 0.5)$$

- α-GANs (Kurri et al., 2021) use the CPE loss function α-loss, $\alpha \in (0, 1) \cup (1, \infty]$ (Sypherd et al., 2019) given by:

$$\ell_\alpha(y, \hat{y}) = \frac{\alpha}{\alpha - 1}\left(1 - y\hat{y}^{\frac{\alpha-1}{\alpha}} - (1 - y)(1 - \hat{y})^{\frac{\alpha-1}{\alpha}}\right)$$

  - Recovers binary cross-entropy loss as $\alpha \to 1$

α-GANs

- The value function for α-GANs is given by[1]

$$V_\alpha(\theta,\omega) = \mathbb{E}_{X\sim P_r}[-\ell_\alpha(1, D_\omega(X))] + \mathbb{E}_{X\sim P_{G_\theta}}[-\ell_\alpha(0, D_\omega(X))]$$

- Optimal discriminator

$$D_{\omega^*}(x) = \frac{p_r(x)^\alpha}{p_r(x)^\alpha + p_{G_\theta}(x)^\alpha}, \quad x \in \mathcal{X}$$

- $\alpha \to 0$: discriminator becomes less confident $\to$ outputs uniform predictions
- $\alpha = 1$: discriminator recovers true posterior of mixture distribution $\frac{1}{2}(P_r + P_{G_\theta})$
- $\alpha \to \infty$: discriminator becomes very confident and in the limit

$$D_{\omega^*}(x) = \begin{cases} 1 & \text{if } p_r(x) > p_{G_\theta}(x), \\ 1/2 & \text{if } p_r(x) = p_{G_\theta}(x), \\ 0 & \text{o.w.} \end{cases} \quad x \in \mathcal{X}$$

[1] Kurri et al. (2022)

α-GANs

$$V_\alpha(\theta,\omega) = \mathbb{E}_{X\sim P_r}[-\ell_\alpha(1, D_\omega(X))] + \mathbb{E}_{X\sim P_{G_\theta}}[-\ell_\alpha(0, D_\omega(X))]$$

- α-GAN formulation recovers a class of f-GANs that minimize the symmetric Arimoto divergence $D_{f_\alpha}(P_r||P_{G_\theta})$ using

$$f_\alpha(u) = \frac{\alpha}{\alpha-1}\left((1+u^\alpha)^{\frac{1}{\alpha}} - (1+u) - 2^{\frac{1}{\alpha}} + 2\right), \quad u \geq 0$$

$\alpha \to 1$	$\alpha = 1/2$	$\alpha \to \infty$
Vanilla GAN	Hellinger GAN	TV GAN

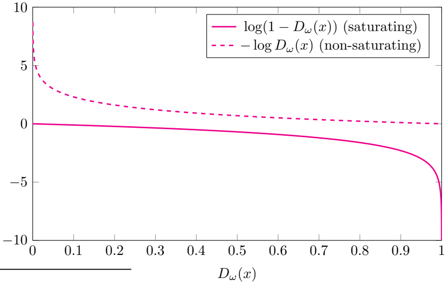
Single-objective GANs, even with tunable losses or different divergences, suffer from training instabilities!

Non-Saturating Vanilla GAN: Addressing Training Instabilities

$$V_{\mathsf{VG}}(\theta,\omega) = \mathbb{E}_{X\sim P_r}[\log D_\omega(X)] + \mathbb{E}_{X\sim P_{G_\theta}}[\log(1 - D_\omega(X))]$$

- Saturing VGANs (above) suffer from exploding & vanishing gradients
- Addressed via a dual-objective *non-saturating* (NS) vanilla GAN[1]

$$\text{D: } \sup_{\omega\in\Omega} V_{\mathsf{VG}}(\theta,\omega) \quad , \qquad \text{G: } \inf_{\theta\in\Theta} V_{\mathsf{VG}}^{\mathsf{NS}}(\theta,\omega) := \mathbb{E}_{X\sim P_{G_\theta}}[-\log D_\omega(X)]$$



[1] Goodfellow et al. (2014)

Wasserstein GANs

- Arjovsky and Bottou (2017) argue for a distributional closeness measure that accounts for misaligned supports
  - JSD fails to provide gradients to bring close distributions with misaligned supports
- Propose the Wasserstein-1 distance to obtain a Wasserstein GAN:

$$\inf_{\theta \in \Theta} \sup_{\phi \in \Phi : \|f_\phi\|_L \leq 1} \mathbb{E}_{X \sim P_r}[f_\phi(X)] - \mathbb{E}_{X \sim P_{G_\theta}}[f_\phi(X)]$$

- Minimizing Wasserstein-1: problem of finding the best transportation plan over the space of couplings (P_r, P_{G_θ})
- Maps to finding a good <u>critic</u> f_ϕ that maximally highlights the differences between P_r and P_{G_θ}
- Function class Φ captures bounded 1-Lipschitz functions f
  - Enforced in practice via weight-clipping or gradient penalty

Generative Adversarial Networks: Practical Challenges and Successes

GAN Training Instabilities and Challenges

GANs face several key challenges impacting their training stability[1]:

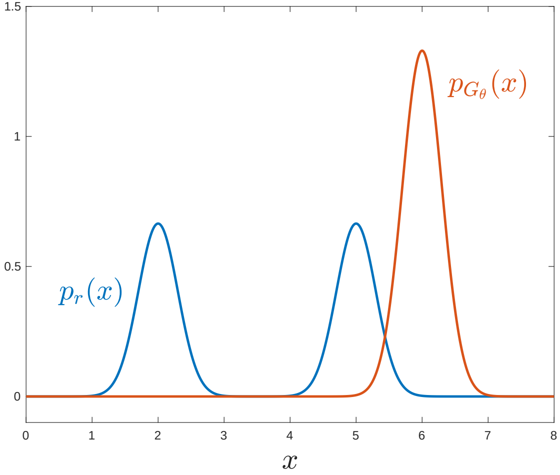
1. **Vanishing gradients**: Minimal gradients limit the generator's ability to learn from discriminator feedback
2. **Exploding gradients**: Excessively large gradients cause unstable generator updates
3. **Mode collapse**: Generator maps many inputs to similar outputs, failing to capture the full data distribution
4. **Sensitivity to initialization and hyperparameters**: Small changes in architecture or training settings can drastically affect outcomes
5. **Memorization**: With limited data, GANs can overfit by memorizing training samples instead of learning generalizable patterns[2]

[1] Goodfellow (2016); Wiatrak et al. (2019); Welfert et al. (2024)

[2] Feng et al. (2021); Nagarajan et al. (2018)

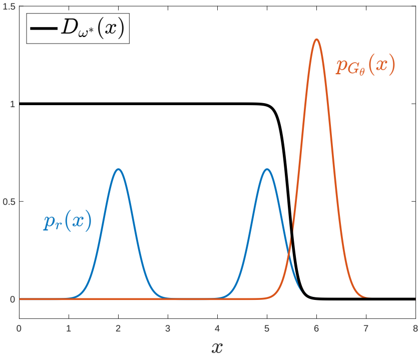
GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(6, 0.3^2)$



GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(6, 0.3^2)$

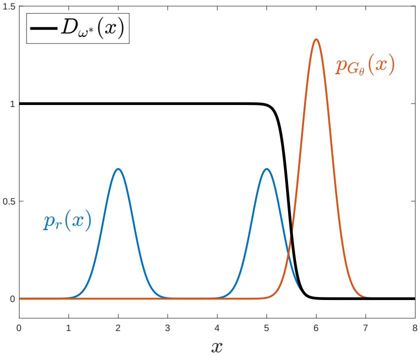


$$D_{\omega^*}(x) = \frac{p_r(x)}{p_r(x) + p_{G_\theta}(x)}$$

(Optimal discriminator)

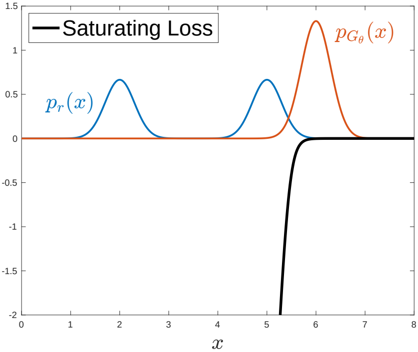
GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(6, 0.3^2)$



$$D_{\omega^*}(x) = \frac{p_r(x)}{p_r(x) + p_{G_\theta}(x)}$$

(Optimal discriminator)

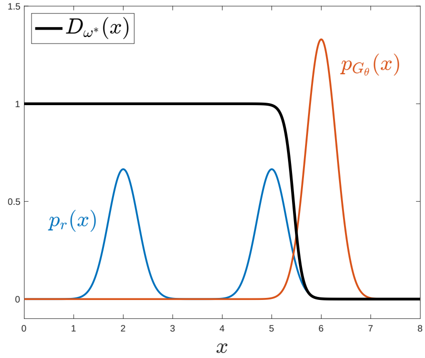


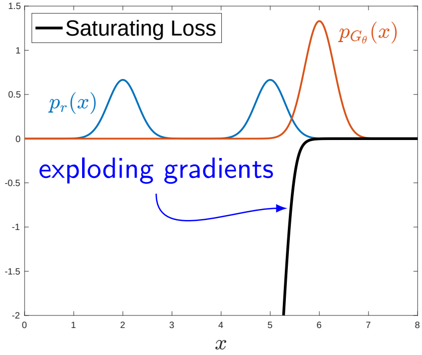
$$\ell_{G_\theta} = \log(1 - D_{\omega^*}(x))$$

(Generator's loss)

GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(6, 0.3^2)$

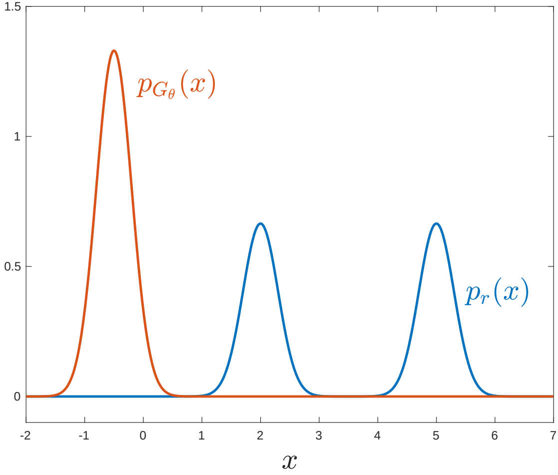




Overconfident $D_\omega \rightarrow$ exploding gradients for G_θ near real data
$\rightarrow$ unstable updates, samples pushed far from real data

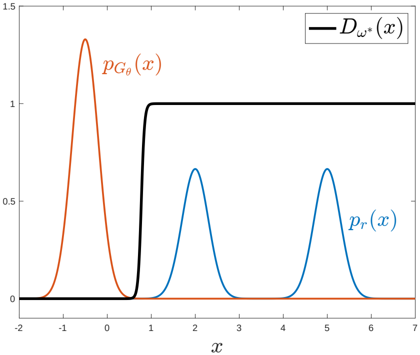
GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(-0.5, 0.3^2)$



GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(-0.5, 0.3^2)$

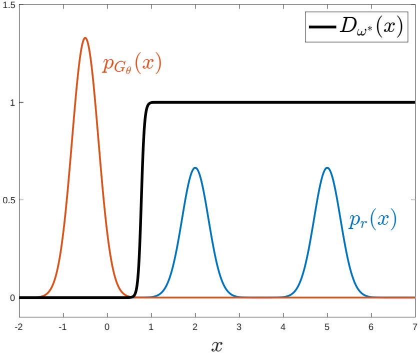


$$D_{\omega^*}(x) = \frac{p_r(x)}{p_r(x) + p_{G_\theta}(x)}$$

(Optimal discriminator)

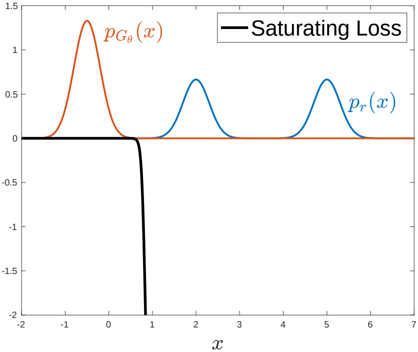
GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(-0.5, 0.3^2)$



$$D_{\omega^*}(x) = \frac{p_r(x)}{p_r(x) + p_{G_\theta}(x)}$$

(Optimal discriminator)

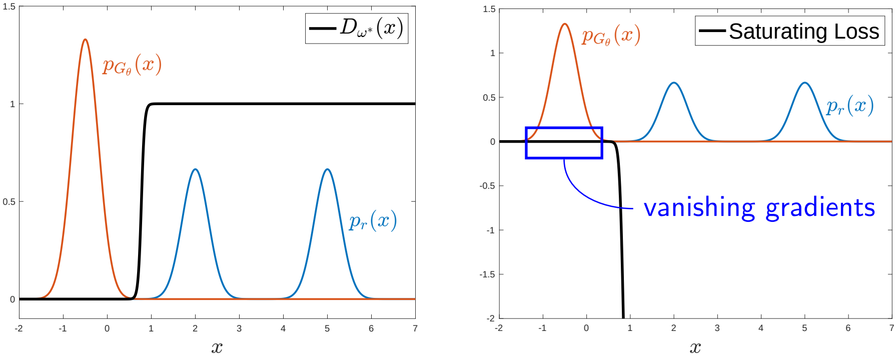


$$\ell_{G_\theta} = \log(1 - D_{\omega^*}(x))$$

(Generator's loss)

GAN Training Instabilities: Toy Example

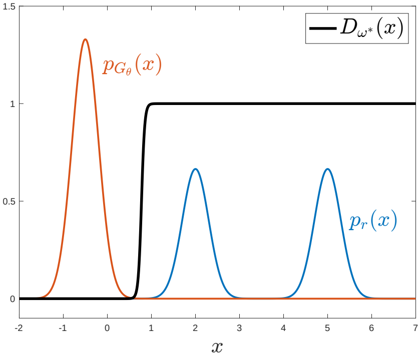
Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(-0.5, 0.3^2)$

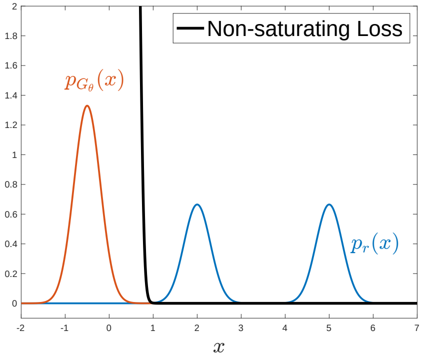


Overconfident $D_\omega \rightarrow$ vanishing gradients for G_θ on samples far from real data $\rightarrow$ mode collapse

GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(-0.5, 0.3^2)$





$$D_{\omega^*}(x) = \frac{p_r(x)}{p_r(x) + p_{G_\theta}(x)}$$

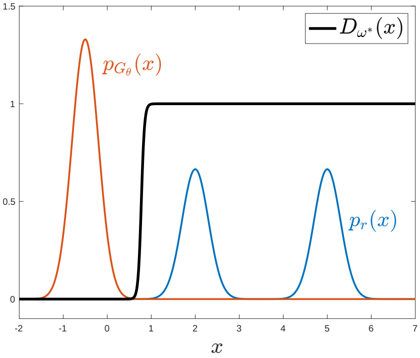
(Optimal discriminator)

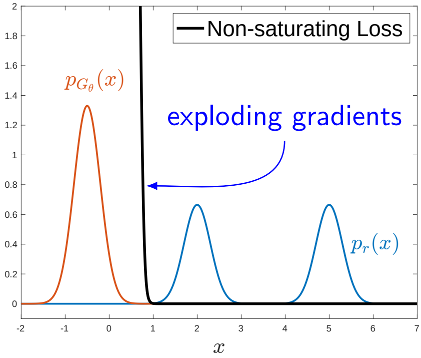
$$\ell_{G_\theta} = -\log(D_{\omega^*}(x))$$

(Generator's loss)

GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2)$, $P_{G_\theta} = \mathcal{N}(-0.5, 0.3^2)$

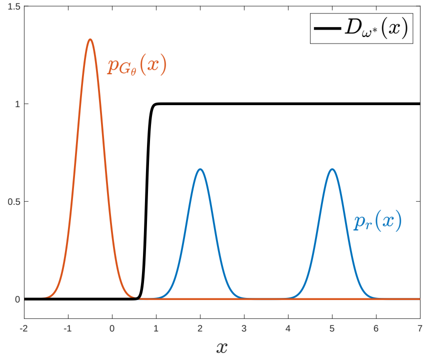


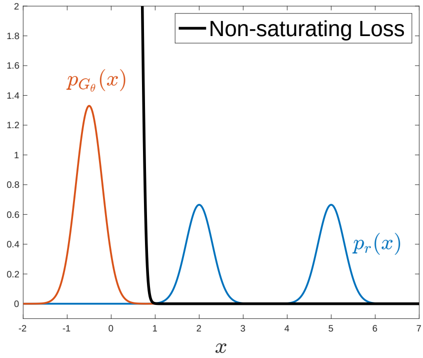


Non-saturating loss mitigates vanishing gradients but can still suffer from exploding gradients, leading to oscillatory dynamics and mode collapse

GAN Training Instabilities: Toy Example

Toy example: $P_r = 0.5\mathcal{N}(2, 0.3^2) + 0.5\mathcal{N}(5, 0.3^2),\ P_{G_\theta} = \mathcal{N}(-0.5, 0.3^2)$





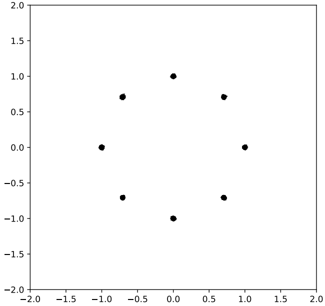
Are instabilities due to assuming optimal D_{ω^*}?
Or do they also arise in practical, non-ideal training settings?

Training Instabilities in Practice

- In practice, the discriminator is usually **not trained to convergence** at each step
- Instead, generator and discriminator are updated in an **alternating** manner
- Despite this heuristic, **training instabilities persist**:
  - Vanishing/exploding gradients
  - Oscillatory behavior
  - Mode collapse
- **Stability is highly sensitive** to architecture, loss function, and hyperparameter choice

Training Instabilities in Practice: Synthetic 2D Gaussian Ring

- **Dataset:** Synthetic 2D Gaussian Ring
  - 8-component Gaussian mixture with equal priors
  - Means placed uniformly on unit circle
  - Isotropic variance: 10^{-4}
- **Architecture:**
  - Generator (G_θ): 4 fully connected layers, 400 units each
  - Discriminator (D_ω): 4 fully connected layers, 200 units each
- **Training:** 400 epochs using Adam optimizer

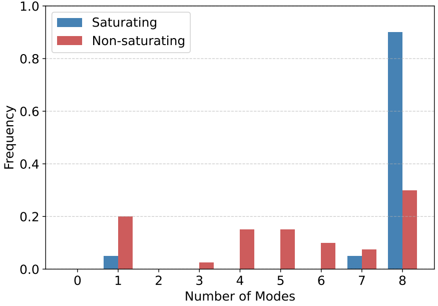


Real Data

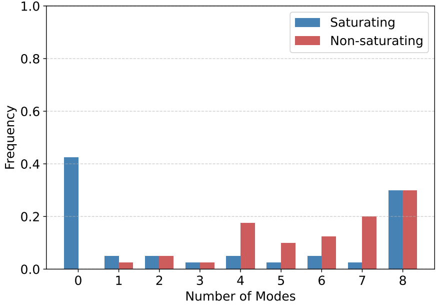
2D Gaussian Ring: Sensitivity to Hyperparameter Initialization

Distribution of Captured Modes Across 40 Random Seeds

Adam Optimizer $\beta_2 = 0.99$



Adam Optimizer $\beta_2 = 0.999$

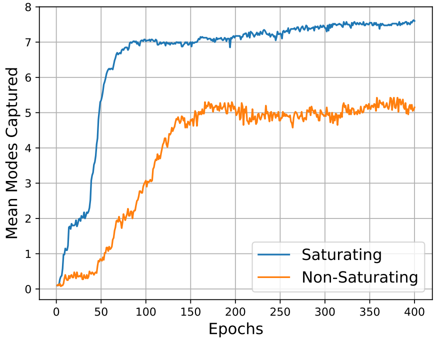


Number of modes captured varies significantly across seeds and hyperparameter settings, highlighting challenges in robustness

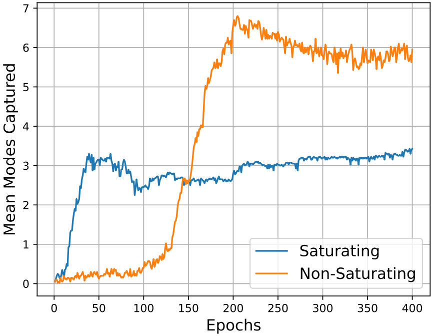
2D Gaussian Ring: Sensitivity to Hyperparameter Initialization

Average Number of Modes Captured (across 40 seeds) vs. Epochs

Adam Optimizer $\beta_2 = 0.99$



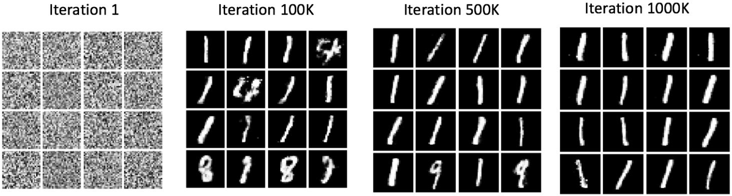
Adam Optimizer $\beta_2 = 0.999$



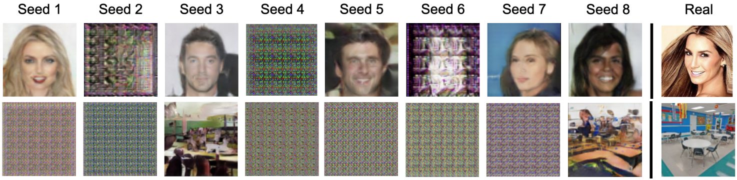
Small changes to hyperparameters can significantly impact mode coverage, highlighting sensitivity in training dynamics

Training Instabilities in Practice: Image Datasets

Mode collapse

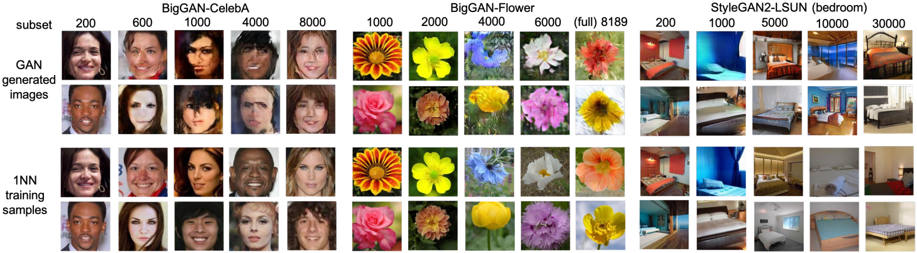


Non-convergence



Images from Mi et al. (2018) (MNIST) and Welfert et al. (2024) (CelebA/LSUN)

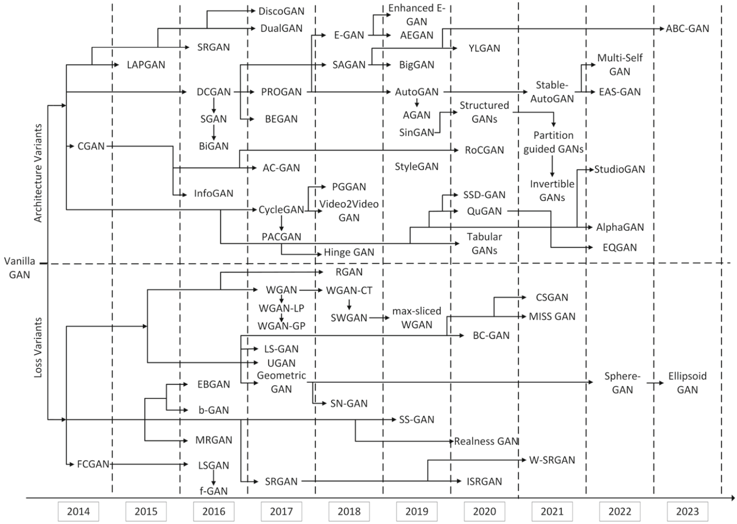
Training Instabilities in Practice: Image Datasets



GANs can replicate training data but memorization decays exponentially with increasing dataset size and complexity[1]

[1] Feng et al. (2021); Nagarajan et al. (2018)
Image from Feng et al. (2021)

Alleviating Training Instabilities and Improving Performance



Timeline of some GAN variants (Ahmad et al., 2024)

Loss Function Variants in GANs

Key formulations based on different objectives for improved training dynamics:

- **WGAN-GP** (Gulrajani et al., 2017): Adds gradient penalty to WGAN to enforce Lipschitz continuity robustly
- **Least Squares GAN (LSGAN)** (Mao et al., 2017): Uses squared loss to smooth discriminator outputs and reduce vanishing gradients
- **Geometric GAN** (Lim and Ye, 2017): Employs a margin-based hinge loss that improves stability
- **(α_D, α_G)-GAN** (Welfert et al., 2023): Dual-objective framework with tunable control over each player's objective using α-loss

SOTA Architectural Variants of GANs

- **StyleGAN (v1-v3, XL, T)** (Karras et al., 2019, 2020, 2021; Sauer et al., 2022, 2023): Style-based generators with
  - adaptive instance normalization
  - progressive growing
  - alias-free convolution (v3)
  - high-resolution scaling (XL)
  - transformer-based architectures (T)
- **GigaGAN** (Kang et al., 2023): Scalable transformer-convolutional architecture for high-resolution conditional image generation
- **BigGAN** (Brock et al., 2019): Scaled-up class-conditional GAN for high-fidelity generation on ImageNet
- **Encoder-Decoder GANs** (e.g., Pix2PixHD, CycleGAN) (Wang et al., 2018; Zhu et al., 2017): Used in image-to-image translation and domain adaptation

SOTA models combine architecture and loss design:
StyleGAN + NS loss/WGAN-GP, BigGAN + hinge loss

Training Techniques for More Stable GAN Training

- **Spectral Normalization** (Miyato et al., 2018): Limits discriminator gradients via Lipschitz constraints
- **Gradient Penalty / Regularization** (Gulrajani et al., 2017)
- **Two Time-Scale Updates** (Heusel et al., 2017): Different learning rates for discriminator and generator
- **Normalization Layers** (Batch, Instance) (Ioffe and Szegedy, 2015; Ulyanov et al., 2017): Stabilize training by normalizing activations
- **Label Smoothing** (Salimans et al., 2016): Reduces discriminator overconfidence
- **Feature Matching / Minibatch Discrimination** (Salimans et al., 2016): Promote diversity by comparing intermediate activations or minibatches
- **Noise Injection** (Karras et al., 2019)

Combining multiple techniques is key to stable GAN training
Key advantage: Fast Sampling (unlike diffusion models)

Realistic Faces Generated by GANs

Which Face is Real?

https://www.whichfaceisreal.com/

This Person Does Not Exist!

https://thispersondoesnotexist.com/

References i

D. J. Bartholomew. Latent variable models and factor analysis, Nov. 1987.

Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 61(3):611–622, 1999. ISSN 13697412, 14679868. URL http://www.jstor.org/stable/2680726.

G. E. Hinton, S. Osindero, and Y. W. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.

Ruslan Salakhutdinov and Geoffrey Hinton. Deep boltzmann machines. In David van Dyk and Max Welling, editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 448–455, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL https://proceedings.mlr.press/v5/salakhutdinov09a.html.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Gowtham R Kurri, Tyler Sypherd, and Lalitha Sankar. Realizing gans via a tunable loss function. *arXiv preprint arXiv:2106.05232*, 2021.

References ii

Tyler Sypherd, Mario Diaz, Lalitha Sankar, and Peter Kairouz. A tunable loss function for binary classification. In *IEEE International Symposium on Information Theory (ISIT)*, pages 2479–2483, 2019.

Gowtham R. Kurri, Monica Welfert, Tyler Sypherd, and Lalitha Sankar. α-GAN: Convergence and estimation guarantees. In *IEEE ISIT*, pages 276–281, 2022.

Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. *arXiv:1701.04862*, 2017.

Ian Goodfellow. NIPS 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

Maciej Wiatrak, Stefano V Albrecht, and Andrew Nystrom. Stabilizing generative adversarial networks: A survey. *arXiv:1910.00927*, 2019.

Monica Welfert, Gowtham R Kurri, Kyle Otstot, and Lalitha Sankar. Addressing gan training instabilities via tunable classification losses. *IEEE Journal on Selected Areas in Information Theory*, 2024.

Qianli Feng, Chenqi Guo, Fabian Benitez-Quiroz, and Aleix M. Martinez. When do GANs replicate? on the choice of dataset size. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6701–6710, October 2021.

Vaishnavh Nagarajan, Colin Raffel, and Ian J Goodfellow. Theoretical insights into memorization in gans. In *NeurIPS Integration of Deep Learning Theories Workshop*, 2018.

References iii

Lu Mi, Macheng Shen, and Jingzhao Zhang. A probe towards understanding GAN and VAE models. *arXiv preprint arXiv:1812.05676*, 2018.

Zeeshan Ahmad, Zain ul Abidin Jaffri, Meng Chen, and Shudi Bao. Understanding GANs: Fundamentals, variants, training challenges, applications, and open problems. *Multimedia Tools and Applications*, pages 1–77, 2024.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017.

Xudong Mao, Qing Li, Haoran Xie, Raymond Y.K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

Jae Hyun Lim and Jong Chul Ye. Geometric GAN. *arXiv preprint arXiv:1705.02894*, 2017.

Monica Welfert, Kyle Otstot, Gowtham R. Kurri, and Lalitha Sankar. (α_D, α_G)-GANs: Addressing GAN training instabilities via dual objectives. In *IEEE ISIT*, 2023.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of styleGAN. In *CVPR*, 2020.

References iv

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *NeurIPS*, 2021.

Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH 2022 Conference Proceedings*, 2022. URL `https://doi.org/10.1145/3528233.3530738`.

Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. StyleGAN-t: Unlocking the power of GANs for fast large-scale text-to-image synthesis. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 30105–30118, 2023.

Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*, 2019.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017.

References v

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6924–6932, 2017.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Thank You!