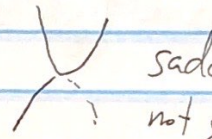
GANs: Generative Adversarial Networks

Goodfellow et al (2014)

Idea: adversarial training min-max objective



train 2 NNs trying to defeat one another

- G generator: noise z → data x
- D discriminator: classifies generated "fake" vs data "real"

Was SOTA on images until recently (now diffusion)

Still some advantages — computationally lighter & faster

GAN loss:

$$L(\theta,\phi) = -\left(\sum_{i \in \text{data}} \log D_\theta(x_i) + \sum_{i \in \text{gen}} \log\left(1 - D_\theta(G_\phi(z_i))\right) \right)$$

BCE for D but also loss for G!

but

$$\max_{\phi} \min_{\theta} L$$

→ balance two objectives

ϕ: good generator; θ: good discriminator

Why does this work??

- formally: $\min_\theta L \rightarrow D_{\hat{\theta}}(x) = \frac{p_{data}(x)}{p_{data}(x) + p_{G_\phi}(x)}$ (Neyman-Pearson lemma!)

plug back into L

$$L\big|_{D=D_{\hat{\theta}}} = -\left(\sum_{x \in \text{data}} \log \frac{p_{data}(x)}{p_{data}(x) + p_{G_\phi}(x)} + \sum_{x \in \text{gen}} \log \frac{p_{G_\phi}(x)}{p_{data}(x) + p_{G_\phi}(x)} \right)$$

$$\rightarrow -\int dx \left(p_{data} \log \frac{p_{data}}{p_{data}+p_{G_\phi}} + p_G \log \frac{p_{G_\phi}}{p_{data}+p_{G_\phi}} \right)$$

$$= -2JSD\left[p_{data}, p_{G_\phi}\right] + const$$

↓

Jensen Shannon Divergence — distance btw prob dist'ns!

$0 \leq JSD^{(pq)} \leq 1$

zero iff $P=Q$

$$\max_\phi L \Big|_{P=Q_\phi} = \min_\phi \left(2JSD\left[p_{data}, p_{G_\phi}\right] + const \right)$$

$$\Rightarrow \boxed{p_{G_\phi} = p_{data}}$$

(can also prove more straightforwardly from integral above)

So formally, GAN objective is achieved when G_ϕ matches data!

↓ key pt: p_{data} only learned implicitly!

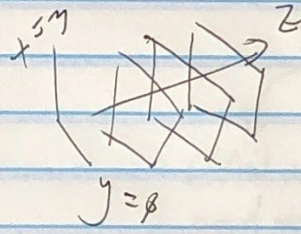
Unfortunately, can't train a GAN so easily, so formal derivation not 100% applicable...

→ go to GAN MNIST Demo

→ Training: Freeze G
Train D (batch real + batch gen)
Freeze D
Train G (batch gen)
repeat

CALOGAN example (1705.02355, 1712.10321)

- Early example of deep gen model for fast sim – surrogate modeling (first in HEP?)
- Used vanilla GAN for GEANT4 e^+, γ, π^+ calo showers by ATLAS ~~&~~ ECAL



3 layers of LAr + lead absorber

simplification: turn sandy fraction off (100% efficient)

"voxelization"

$3 \times 96 - 12 \times 12 - 12 \times 6$

$= 504$ total voxels

energy in each voxel

$\vec{E} \in R^{504}$

3 GANs: e^+, γ, π^+

Data: { 1–100 GeV unif E_{inc}
10k showers each generated w/ GEANT4

goal: learn $p(\vec{E} | E_{inc})$ to generate from

→ to overview of paper

main results: quality 50–50

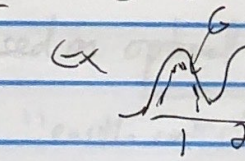
speed – very fast (but CPU/GPU comparisons subtleties)

metrics?

Problems w/ (Vanilla) GANs

- Convergence: min-max unstable
  - vanishing gradients
    - if D too powerful, can overwhelm G (100% separable, no useful gradients for G) → can't train to completion
    - if D too weak, random guessing, also no gradients for G → GAN never convergence, good performance quickly destabilized
- Model selection
  - D & G losses not correlated w/ quality
  - in industry / natural images often rely on "by eye" test!
  - hard to know when to stop training
- Mode collapse

  G & D stuck in vicious cycle



G learns to produce 1's
D fooled for a while
then realizes 2 are all real, guesses 1's
→ G will switch to making 2's

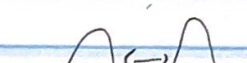
gan lab demo

Beyond vanilla GANs

- Many issues w/ GANs stem from loss fn

$D \to \min_G JSD(p_{data}, p_G)$

D trained to optimality.



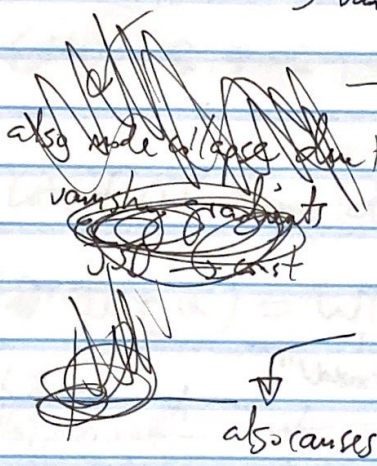
JSD = 1 whenever p_{data} p_G disjoint

bounded from above

saturates

→ vanishing gradients!

→ when G is very bad → no ability to improve!



So can't train D to optimality for vanilla GAN

also causes mode collapse, stems from suboptimal D.

- Wat: better GAN loss, unbounded from above

informative even for widely separated distns

→ one SOTA approach: Wasserstein GANs (1701.07875 Arjovsky, Chintala, Bottou)

Based on optimal transport theory

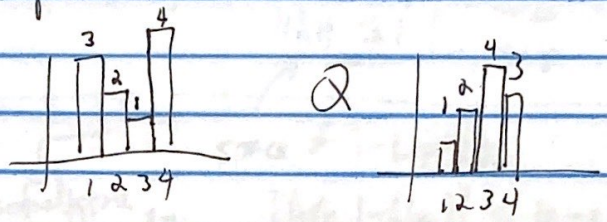
"earth mover's distance" aka "Wasserstein distance"

distance between distns P, Q

"work it takes to transform P into Q"

↳ right property: more separated P & Q, more work

Example:



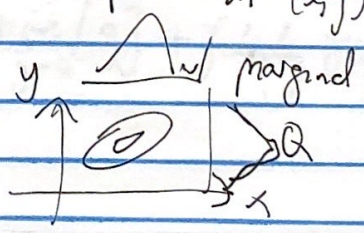
- transport 2 units $1 \to 2$ so 1 matches
- transport 2 units $2 \to 3$ so 2 matches
- transport 1 unit $4 \to 3$ so 3 & 4 match ✓

EMD: $2 \cdot 1 + 2 \cdot 1 + 1 \cdot 1 = 5$.

↳ Lots of OT theory later...

$$\text{EMD}(P,Q) = W_1(P,Q) = \min_{\gamma \in \Pi(P,Q)} \mathbb{E}_{(x,y)\sim\gamma}\left[\|x-y\|\right]$$

- like MSE in prob dist'n space!
- P, Q are dist'ns on $P(x), Q(x), x \in \mathbb{R}^d$
- "Wasserstein-1"
- expectation value over all dist'ns γ
- Euclidean dist'ce
- Wasserstein-k is $\sqrt[k]{\|x-y\|^k}$
- space of all prob dist'ns $\gamma(x,y)$ on $(x,y) \in \mathbb{R}^d \times \mathbb{R}^d$



$W_1 = 0 \iff P = Q$
unbounded for $P \neq Q$
$\uparrow \iff \wedge$ JSD $\to 1$
$W_1 \to \infty$

W_1 totally intractable...

→ Brilliant idea: use Kantorovich-Rubinstein duality

Amazing formula

K-R duality

$$W_1(P,Q) = \max_{\|h\|_L \le 1} \left[\mathop{\mathbb{E}}_{x\sim P}[h(x)] - \mathop{\mathbb{E}}_{y\sim Q}[h(y)] \right]$$

$\|h\|_L \le 1$: space of 1-Lipschitz

analytic for K-Lipschitz $\|h\|_L \le K$.

$|h(x_1) - h(x_2)| \le |x_1 - x_2| \quad \forall x_1, x_2$

$\Downarrow$

$|h'(x)| \le 1$ infinitesimal $\int$ by mean value thm

rough idea of pf: comes from Lagrange multipliers

$$\mathop{\mathbb{E}}_{x,y\sim\gamma}(\|x-y\|) = \int dx\,dy\, \gamma(x,y)\|x-y\|$$

$$+ \int \left(P(x) - \int \gamma(x,y)\,dy\right) f(x)\,dx$$

$$+ \int \left(Q(y) - \int \gamma(x,y)\,dx\right) g(y)\,dy$$

$$= \mathop{\mathbb{E}}_{x\sim P}[f] + \mathop{\mathbb{E}}_{y\sim Q}[g] + \int dx\,dy\, \gamma(x,y)\left(\|x-y\| - f(x) - g(y)\right)$$

$\vdots$

---

w/ K-R duality, W_1 becomes tractable!

Approx h w/ NN $\longrightarrow$ "critic" analog of discriminator

$\max_{\|h\|_L \le 1}(\cdots)$ $\longrightarrow$ train the critic!

WGAN Loss

$$L = \sum_{x \in \text{real}} h(x) - \sum_{z \in \text{latent}} h(G(z))$$

$$\min_G \max_{h \in \text{Lipschitz}} L$$

Enforcing Lipschitz: many approaches

- original paper: "weight clipping"
  - after each weight update
  - $w \to \text{clip}(w, -c, c)$ ($w = c$ if $w > c$; $w = -c$ if $w < -c$)
  - This enforces K-lipschitz crudely
  - roughly: NN output $h = g_1 \circ W_1 \circ g_2 \circ W_2 \circ \cdots \circ g_L \circ W_L(x)$
  - so $|\nabla_x h| \sim |\partial g_1' W_1 g_2' W_2 \cdots g_L' W_L|$
  - so if weights bounded, $|\nabla_x h|$ bounded & activation gradients further bounded ✓
- Better way: gradient penalty
  - add regularizer to loss $\delta L = \rho \left(|\nabla_x h| - 1 \right)^2$
  - might not be state of the art...
  - ρ: dist'n of pts $\hat{x} = tx + (1-t)y$
    - $x, y \sim$ data, gen
    - $t \sim U(0,1)$
  - Some theory behind this...
    - not clear why seems to require $|\nabla_x h| = 1$ instead of ≤ 1 ... but empirically works