- anomaly detection
  - (low $p(x)$ — rare?
  - learn $p(x|y)$ for $y \in CR$
    - interpolate y into SR
    - set background model, fully data driven for x

useful beyond anomaly detection of course!

- .... probably more applications out there!

---

4 classes of NN-based GMs:

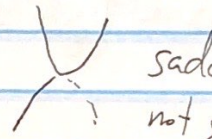
- GANs
- VAEs
- Flows (also DE)
- Diffusion (also DE)

Plan is to introduce class to each one, with examples

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!

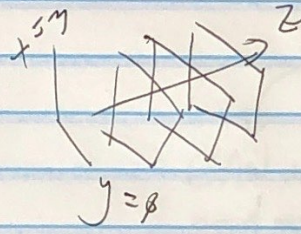
Unfortunately, can't train a GAN so easily, so formal derivation not 100% applicable...

→ 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?