CS 182 Lecture 19: Generative Adversarial Networks

7/18

Using LVM for generation:
→ sample $z \sim p(z)$
→ sample $x \sim p(x|z)$

"similar at the population level" → becomes hard to distinguish real/fake

GAN is a game. Train a network to "guess" real/fake. (classifier)
↳ 2-player.
↳ "discriminator": serves as loss function for "generator."

Objective of generator $G(z)$: make discriminator $D(x) = 0.5$ for all generated x (cannot tell).
↳ generate realistic images
↳ generate all possible realistic images (i.e. learn distribution).

zero-sum ↓ classic GAN 2-player game:

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[\log D(x)] + E_{z\sim p(z)}[\log(1 - D(G(z)))]$$

$$\approx \frac{1}{N}\sum_{i=1}^{N} \log D(x_i) \quad x_i \in \mathcal{D}_T \qquad \approx \frac{1}{N}\sum_{j=1}^{N} \log(1 - D(x_j)) \quad x_j = G(z_j)$$

more clearly: $\min_\theta \max_\phi V(\theta,\phi) = E_{x\sim p_{data}(x)}[\log D_\phi(x)] + E_{z\sim p(z)}[\log(1 - D_\phi(G_\theta(z)))]$

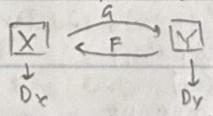
SGD → $\phi \leftarrow \phi + \alpha \nabla_\phi V(\theta,\phi) \approx \nabla_\phi \left(\frac{1}{N}\sum_{i=1}^{N} \log D_\theta(x_i) + \frac{1}{N}\sum_{j=1}^{N} \log(1 - D_\phi(x_j))\right)$ ← C-E loss
$\theta \leftarrow \theta - \alpha \nabla_\theta V(\theta,\phi)$ $x_i \in \mathcal{D}_T$ $x_j = G(z_j)$
$\approx \nabla_\theta \left(\frac{1}{N}\sum_{j=1}^{N} \log(1 - D_\phi(G_\theta(z_j)))\right)$

optimal discriminator (Bayes classifier) $= D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_G(x)}$ $x = G(z)$, $z \sim p(z)$.

Objective for G:
$V(D_G^*, G) =$
$E_{p_{data}(x)}[\log p_{data}(x) - \log(p_{data}(x) + p_G(x))] +$
$E_{p_G(x)}[\log p_G(x) - \log(p_{data}(x) + p_G(x))]$.
$= D_{JS}(p_{data} \| p_G)$
↑ Jensen-Shannon divergence.
- goes to 0 if dists match
+ symmetric

↳ practical generator loss: $E_{z\sim p(z)}[-\log D_\phi(G_\theta(z))]$
"maximize probability that image is real"
↳ better behaving gradients.

uses of GANs: conditional GAN (labelling)
CycleGAN ("translating")
↳ two (conditional) generators and two corresponding discriminators.



↳ incorporates cycle-consistency loss
Given $x \to \hat{Y} \to \hat{x}$, how close are $x/\hat{x}$?

i.e. easier to train!

improved GAN techniques:
→ least-squares GAN (LSGAN)
↳ discriminator outputs real-valued num
→ Wasserstein GAN (WGAN)
↳ discriminator is Lipschitz-continuous
→ gradient penalty
↳ discriminator is constrained to be continuous even harder
→ spectral norm
↳ discriminator is really constrained to be continuous
→ instance noise
↳ try to get $p_{data}(x)/p_G(x)$ overlap.

not just 0, 1.

good choices today!

"Earth mover's distance"

WGAN: accounts for "distance" b/w p_{data} and p_G.
↳ $W(p_{data}, p_G) = \inf_{\gamma \in \Gamma} E_{(x,y)\sim\gamma(x,y)}[\|x - y\|]$
$\gamma_X(x) = p_{data}(x)$. really complex...
$\gamma_Y(y) = p_G(x)$.

Kantorovich-Rubinstein duality → $= \sup_{\|f\|_L \le 1} E_{p_{data}}[f(x)] - E_{p_G(x)}[f(x)]$

NN: set of all 1-Lipschitz scalar functions
$|f(x) - f(y)| \le |x - y|$.
(bounded slope, not too steep).

gradient penalty: bounded slope.
update f_θ using gradient of
$E_{x\sim p_{data}}[f_\theta(x) - \lambda(\|\nabla_x f_\theta(x)\|_2 - 1)^2] - E_{z\sim p(z)}[f_\theta(G(z))]$
make norm of gradient close to 1.

$\sigma(W) = \max_{h:h\neq 0} \frac{\|Wh\|}{\|h\|} = \max_{\|h\|\le 1} \|Wh\|$ ← largest σ of W.

spectral norm: bound the Lipschitz constant in terms of singular values of each W_ℓ.
↳ max slope of $Wx + b$ is spectral norm: $W_\ell \leftarrow W_\ell / \sigma(W_\ell)$.