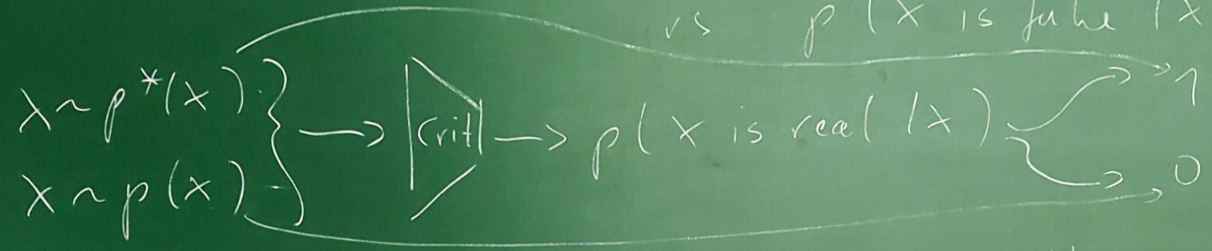
Generative Adversarial Networks (GANs) [Goodfellow et al. 2014]

- dominant generative model 2014 – 2019/20
  [now: diffusion models, transformers are better]
- idea: learn quality criterion (instead of hand-crafted formula)
  - new neural network: "discriminator" or "critic"
    classifier $p(x \text{ is real} \mid x)$
    vs $p(x \text{ is fake} \mid x) = 1 - p(x \text{ is real} \mid x)$



- train decoder ("generator") jointly with critic

- training becomes an "arms race" or "game"
  - critic becomes better at distinguishing reals and fakes
  - decoder becomes better at fooling the critic

$\Rightarrow$ training objective

$$\hat{C}, \hat{D} = \underset{D}{\operatorname{argmin}} \underset{C}{\operatorname{argmax}} \; \mathbb{E}_{x \sim p^*(x)} \left[\log p_C(x \text{ is real} \mid x) \right] + \mathbb{E}_{z \sim q(z)} \left[\log \underbrace{\left(1 - p_C(\text{real} \mid D(z)) \right)}_{p(x \text{ is fake} \mid x = D(z))} \right]$$

(generated data)

- at optimal convergence, we have $p(x) = p^*(x)$

proof: consider expression $a \log y + b \log(1-y)$ with $a, b \neq 0$, $y \in [0,1]$

set derivative $\frac{\partial}{\partial y} \ldots \doteq 0$ $\quad y = \frac{a}{a+b}$

let $y = p^*(x \text{ is real} \mid x)$ by the optimal critic, $a = p^*(x)$, $b = p(x)$

$$\text{loss}(D) = \mathbb{E}_{x \sim p^*(x)} \left[\log \frac{p^*(x)}{p^*(x) + p(x)} \right] + \mathbb{E}_{x \sim p(x)} \left[\log \left(\frac{p(x)}{p^*(x) + p(x)} \right) \right]$$

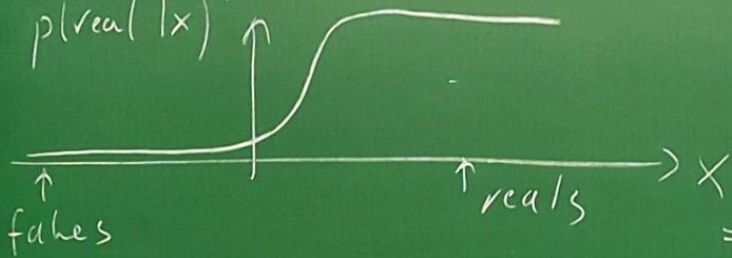
$$= KL\left[p^*(x) \,\middle\|\, \frac{p^*(x)+p(x)}{2}\right] + KL\left[p(x) \,\middle\|\, \frac{p^*(x)+p(x)}{2}\right] - \log 4$$

$$\geq -\log 4$$

minimum is achieved if both $KL = 0$

$$p^*(x) = \frac{p^*(x)+p(x)}{2} \quad \text{and} \quad p(x) = \frac{p^*(x)+p(x)}{2} \iff \boxed{p(x) = p^*(x)}$$

- If critic is perfect, loss(D) is convex, so gradient descent finds global optimum
- two crucial differences in practice
  - training with perfect critic does not work. for a bad decoder, recognizing fakes is very easy $\Rightarrow$ loss gradient ≈ 0 $\Rightarrow$ no training signal



$\Rightarrow$ train D and C jointly, i.e. start with D and C both randomly initialized

$\Rightarrow$ alternating optimization: 1 step ∇D
1...4 steps ∇C

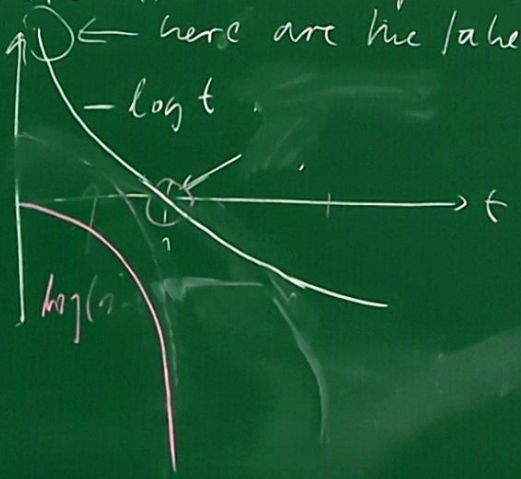
- use non-saturating loss : replace $\log(1 - p(\text{real}|x))$ with $-\log p(\text{real}|x)$ in decoder training step

$$\hat{C} = \arg\max_C \mathbb{E}_{x \sim p^*(x)}\left[\log p_C(\text{real}|x)\right] + \mathbb{E}_{z \sim q(z)}\left[\log(1 - p_C(\text{real}|D(z)))\right]$$

$$\hat{D} = \arg\min_D \mathbb{E}_{z \sim q(z)}\left[-\log p_C(\text{real}|D(z))\right]$$

alternate between the two losses

more informative gradients for D, but the global optimum $p(x) = p^*(x)$ is preserved



why? . unclear, find out!

- GANs defined state-of-the-art up to 2019/20
- variants: - Wasserstein GAN: uses alternative loss for critic
  but is not really better
  - Cycle GAN. replace $q(z)$ with true distribution over another variety of real data

$p^*(x)$: distr. of daylight photos, $\tilde{p}^t(\tilde{x})$: distr night photos
distr of satellite images maps

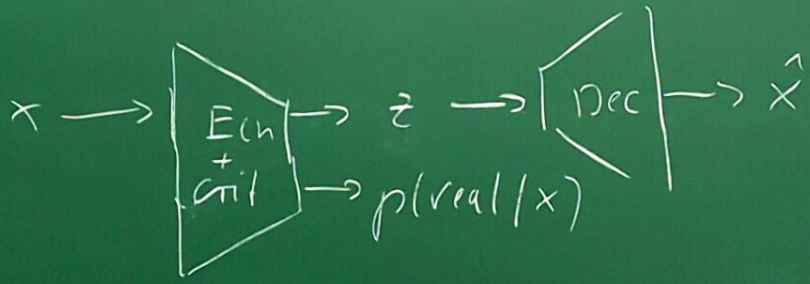
- two cases: ① paired training set: same x from both variants
  ② unpaired training: no overlap between instances

supervised learning — unsupervised learning

minimize $\|\tilde{x}_i - E(x_i)\|^2$

benefit: new "cycle losses": $\|x - D(E(x))\|^2$ and $\|\tilde{x} - E(D(\tilde{x}))\|^2$
should be small both in ① and ② optimize quality of instances

- invertible GAN: add an encoder to original GAN



recreating codes: $z(x)$ and distinguishing reals from fakes can use same image features
⇒ single network for encoder & critic

⇒ many additional losses as in cycle GAN (esp. cycle losses)

cycles $z \to \hat{x} \to \hat{z}$, $x \to z \to \hat{x}$ [$x \to z \to \hat{x} \to \hat{z} \to \hat{\hat{x}}$ etc]
(self-consistency)

push forward $E_{\#} p^*(x) \approx q(z)$

critic $p_C(\text{real} \mid x \sim p(x)) \approx p_C(\text{fake} \mid x \sim p(x))$ (fakes look like reals)

bad: hyperparameter optimization is hard

without critic, checking with $D(\tilde{x}) = \tilde{x}$ and $E(x) = x$ would optimize the cycle losses

$\Rightarrow$ critic ensures that $p_E(\tilde{x}) = E_\# p^*(x) \approx \hat{p}^*(\tilde{x})$

$p_D(x) = D_\# \tilde{p}^*(\tilde{x}) \approx p^*(x)$

} optimize quality of distributions

[if paired TS: two additional losses $\|\tilde{x}_i - E(x_i)\|^2$

$\|x_i - D(\tilde{x}_i)\|^2$]

difference to autoencoder: no bottleneck

$x \rightarrow \boxed{E} \rightarrow \tilde{x} \rightarrow \boxed{D} \rightarrow x$

& additional losses for training