Deep Generative Models

Changyou Chen

Department of Computer Science and Engineering
Universitpy at Buffalo, SUNY
changyou@buffalo.edu

April 16, 2019

Training GANs: A Two-Player Game

A minimax objective function:

$$\min_{\boldsymbol{\theta}_g} \max_{\boldsymbol{\theta}_d} \left[\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log D_{\boldsymbol{\theta}_d}(\mathbf{x}) + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \log \left(1 - D_{\boldsymbol{\theta}_d}(G_{\boldsymbol{\theta}_g}(\mathbf{z}))\right)\right]$$

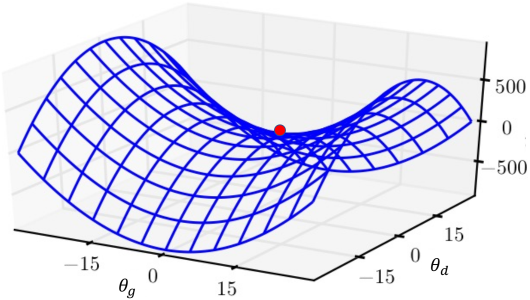
- What does an optimal solution of GAN look like?
  - A local minimum/maximum?
  - Or $\cdots$

Training GANs: A Two-Player Game

A minimax objective function:

$$\min_{\boldsymbol{\theta}_g} \max_{\boldsymbol{\theta}_d} \left[\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log D_{\boldsymbol{\theta}_d}(\mathbf{x}) + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \log\left(1 - D_{\boldsymbol{\theta}_d}(G_{\boldsymbol{\theta}_g}(\mathbf{z}))\right)\right]$$

- The optimal solution for the min-max procedure is a saddle point of the GAN objective.



Training GANs: A Two-Player Game

$$\min_{\boldsymbol{\theta}_g} \max_{\boldsymbol{\theta}_d} \left[\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log D_{\boldsymbol{\theta}_d}(\mathbf{x}) + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \log\left(1 - D_{\boldsymbol{\theta}_d}(G_{\boldsymbol{\theta}_g}(\mathbf{z}))\right)\right]$$

Training

for number of training iterations **do**

for k steps **do**

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D_{\theta_d}(x^{(i)}) + \log(1 - D_{\theta_d}(G_{\theta_g}(z^{(i)})))\right]$$

end for

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Update the generator by ascending its stochastic gradient (improved objective):

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log(D_{\theta_d}(G_{\theta_g}(z^{(i)})))$$

end for

- No best rule for choosing k.

Training GANs: A Two-Player Game

$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log D_{\theta_d}(\mathbf{x}) + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \log\left(1 - D_{\theta_d}(G_{\theta_g}(\mathbf{z}))\right)\right]$$

Training

for number of training iterations **do**

for k steps **do**

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D_{\theta_d}(x^{(i)}) + \log(1 - D_{\theta_d}(G_{\theta_g}(z^{(i)})))\right]$$

end for

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Update the generator by ascending its stochastic gradient (improved objective):

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log(D_{\theta_d}(G_{\theta_g}(z^{(i)})))$$

end for

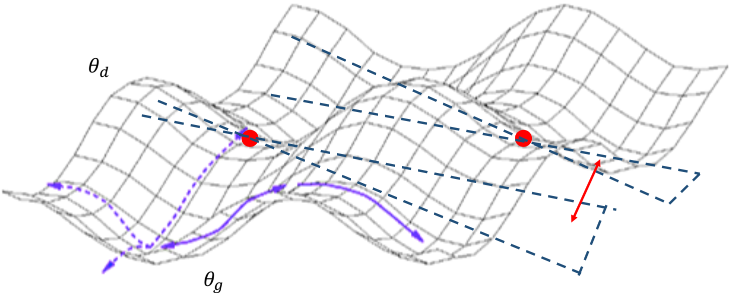
- No best rule for choosing k.

Training GANs: A Two-Player Game

A minimax objective function:

$$\min_{\boldsymbol{\theta}_g} \max_{\boldsymbol{\theta}_d} \left[\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log D_{\boldsymbol{\theta}_d}(\mathbf{x}) + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \log\left(1 - D_{\boldsymbol{\theta}_d}(G_{\boldsymbol{\theta}_g}(\mathbf{z}))\right)\right]$$

- Which one is the solution?

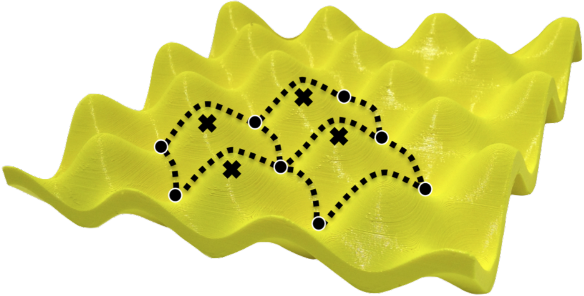


Training GANs: A Two-Player Game

A minimax objective function:

$$\min_{\boldsymbol{\theta}_g} \max_{\boldsymbol{\theta}_d} \left[\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log D_{\boldsymbol{\theta}_d}(\mathbf{x}) + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \log\left(1 - D_{\boldsymbol{\theta}_d}(G_{\boldsymbol{\theta}_g}(\mathbf{z}))\right)\right]$$

- Jointly training two networks is challenging, can be unstable:
  - Can have many saddle points $\Rightarrow$ many unstable sub-optima.

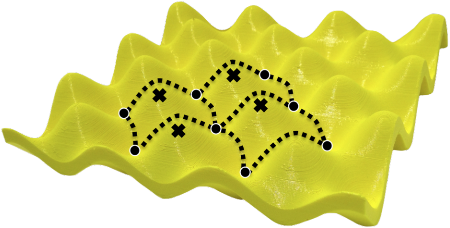


Training GANs: A Two-Player Game

A minimax objective function:

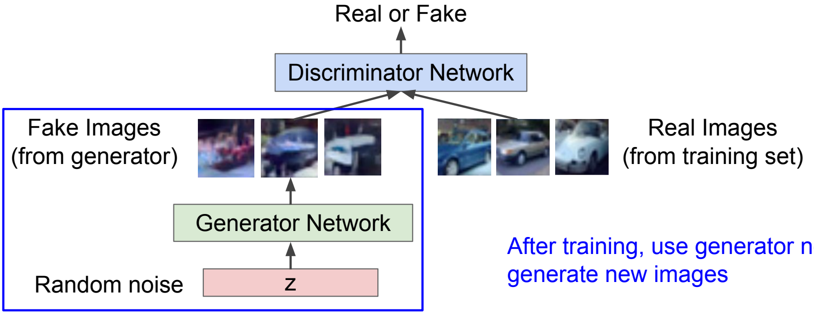
$$\min_{\boldsymbol{\theta}_g} \max_{\boldsymbol{\theta}_d} \left[\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log D_{\boldsymbol{\theta}_d}(\mathbf{x}) + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \log\left(1 - D_{\boldsymbol{\theta}_d}(G_{\boldsymbol{\theta}_g}(\mathbf{z}))\right)\right]$$

- Jointly training two networks is challenging, can be unstable:
  - Can have many saddle points $\Rightarrow$ many unstable sub-optima.
- Choosing objectives with better loss landscapes helps training, or designing better training algorithms, are active areas of research.



Training GANs: A Two-player Game

- $D_{\theta_d}(\mathbf{x})$: Discriminator (parameterized by θ_d) takes input as an image $\mathbf{x}$, and outputs likelihood in $[0, 1]$ to tell if it is a real image or not.
- $G_{\theta_g}(\mathbf{z})$: Generator (parameterized by θ_g) takes input as a random noise $\mathbf{z}$, and outputs an image.

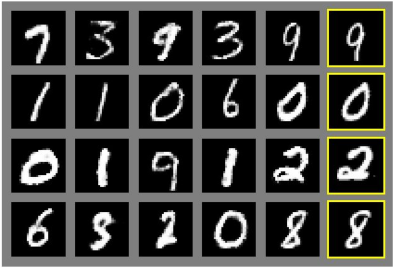


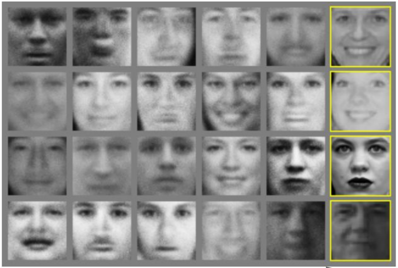
After training, use generator network to generate new images

Denton *et al*, 2015

GAN: Generated Samples

- Generator and discriminator as MLPs (left); Generator as deconvolutional NN, discriminator as a CNN (right).





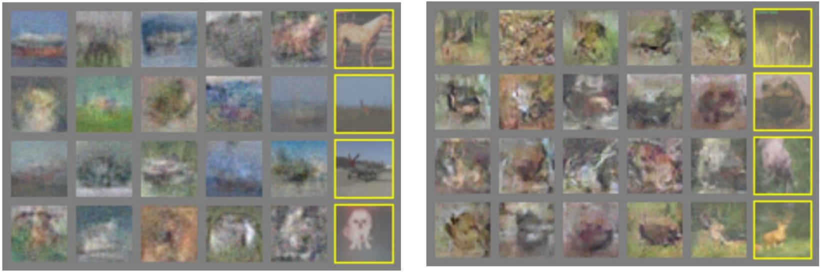
Nearest neighbor from training set

Goodfellow *et al*, NIPS 2014

GAN: Generated Samples

- Generator and discriminator as MLPs (left); Generator as deconvolutional NN, discriminator as a CNN (right).

Generated samples (CIFAR-10)



Nearest neighbor from training set

- Not great, as the generator and discriminator need to be carefully designed.

Goodfellow *et al*, NIPS 2014

Theoretical Properties of GANs

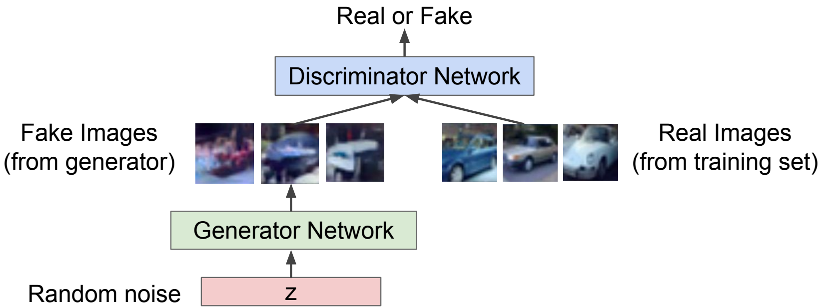
Recap: A Two-Player Game

A minimax objective function:

$$\min_{\boldsymbol{\theta}_g} \max_{\boldsymbol{\theta}_d} \left[\mathbb{E}_{\mathbf{x} \sim p_{\text{data}}} \log D_{\boldsymbol{\theta}_d}(\mathbf{x}) + \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z})} \log \left(1 - D_{\boldsymbol{\theta}_d}(G_{\boldsymbol{\theta}_g}(\mathbf{z})) \right) \right]$$

What is under going in GANs

Distribution matching between data distribution $p_{\text{data}}(\mathbf{x})$ and generator distribution $p_g(x)$.



GAN as Distribution Matching

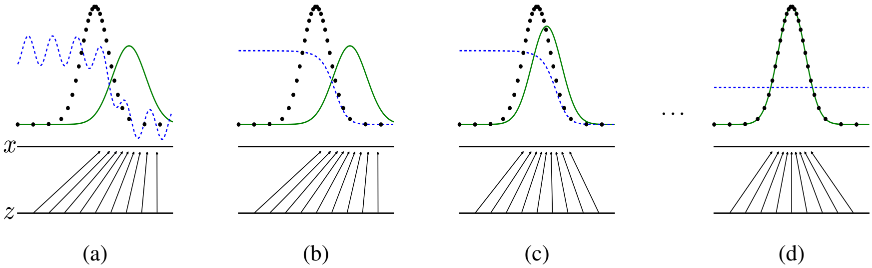


Figure 1: Generative adversarial nets are trained by simultaneously updating the **d**iscriminative distribution (D, blue, dashed line) so that it discriminates between samples from the data generating distribution (black, dotted line) $p_{\boldsymbol{x}}$ from those of the **g**enerative distribution p_g (G) (green, solid line). The lower horizontal line is the domain from which $\boldsymbol{z}$ is sampled, in this case uniformly. The horizontal line above is part of the domain of $\boldsymbol{x}$. The upward arrows show how the mapping $\boldsymbol{x} = G(\boldsymbol{z})$ imposes the non-uniform distribution p_g on transformed samples. G contracts in regions of high density and expands in regions of low density of p_g. (a) Consider an adversarial pair near convergence: p_g is similar to p_{data} and D is a partially accurate classifier. (b) In the inner loop of the algorithm D is trained to discriminate samples from data, converging to $D^*(\boldsymbol{x}) = \frac{p_{\text{data}}(\boldsymbol{x})}{p_{\text{data}}(\boldsymbol{x})+p_g(\boldsymbol{x})}$. (c) After an update to G, gradient of D has guided $G(\boldsymbol{z})$ to flow to regions that are more likely to be classified as data. (d) After several steps of training, if G and D have enough capacity, they will reach a point at which both cannot improve because $p_g = p_{\text{data}}$. The discriminator is unable to differentiate between the two distributions, i.e. $D(\boldsymbol{x}) = \frac{1}{2}$.

Objective Reformulation

$$\min_{\boldsymbol{\theta}_g} \max_{\boldsymbol{\theta}_d} \left[\underbrace{\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log D_{\boldsymbol{\theta}_d}(\mathbf{x}) + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \log \left(1 - D_{\boldsymbol{\theta}_d}(G_{\boldsymbol{\theta}_g}(\mathbf{z}))\right)}_{V(G,D)} \right]$$

$$\Rightarrow V(G,D) = \int_x p_{\text{data}(\mathbf{x})} \log(D(\mathbf{x}))\mathrm{d}\,\mathbf{x} + \int_z p(\mathbf{z}) \log(1 - D(G(\mathbf{z})))\mathrm{d}\,\mathbf{z}$$

$$\overset{\text{change of r.v.}}{\Rightarrow} V(G, D) = \int_x \left(p_{\text{data}(\mathbf{x})} \log(D(\mathbf{x})) + p_g(\mathbf{x}) \log(1 - D(\mathbf{x}))\right) \mathrm{d}\,\mathbf{x}$$

Underlying assumption

The mapping $G_{\boldsymbol{\theta}_g}$ is invertible:

- Not generally satisfied in DNN, but just use in practice.

Global Optimality of p_g

$$V(G, D) = \int_x \left(p_{\text{data}(\mathbf{x})} \log(D(\mathbf{x})) + p_g(\mathbf{x}) \log(1 - D(\mathbf{x})) \right) \mathrm{d}\,\mathbf{x}$$

Theorem

For G fixed, the optimal discriminator is

$$D_G^*(\mathbf{x}) = \frac{p_{data}(\mathbf{x})}{p_{data}(\mathbf{x}) + p_g(\mathbf{x})}$$

Objective Reformulation

$$V(G, D) = \int_x \left(p_{\text{data}(\mathbf{x})} \log(D(\mathbf{x})) + p_g(\mathbf{x}) \log(1 - D(\mathbf{x}))\right) \mathrm{d}\,\mathbf{x}$$

The GAN objective can be reformulated as

$$\min_G C(G) \triangleq \min_G \max_D V(G, D)$$

$$= \min_G \left\{ \underbrace{\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \left[\log \frac{p_{\text{data}}(\mathbf{x})}{p_{\text{data}}(\mathbf{x}) + p_g(\mathbf{x})}\right] + \mathbb{E}_{\mathbf{x}\sim p_g} \left[\log \frac{p_g(\mathbf{x})}{p_{\text{data}}(\mathbf{x}) + p_g(\mathbf{x})}\right]}_{C(G)} \right\}$$

Global Optimality of GAN

Theorem

The global minimum of the virtual training criterion $C(G)$ is achieved if and only if $p_g = p_{data}$. At that point, $C(G)$ achieves the value $-\log 4$, i.e.,

$$\min_G C(G) = -\log 4 \ .$$

Jensen-Shannon Divergence

Definition (Jensen-Shannon Divergence)

The Jensen-Shannon divergence between two distributions $p_1(x)$ and $p_2(x)$ is defined as:

$$JSD(p_1 \| p_2) \triangleq \frac{1}{2}\left(KL\left(p_1 \| \frac{p_1 + p_2}{2}\right) + KL\left(p_2 \| \frac{p_1 + p_2}{2}\right)\right)$$

Jensen-Shannon Divergence

Definition (Jensen-Shannon Divergence)

The Jensen-Shannon divergence between two distributions $p_1(x)$ and $p_2(x)$ is defined as:

$$JSD(p_1 \| p_2) \triangleq \frac{1}{2}\left(KL\left(p_1 \| \frac{p_1 + p_2}{2}\right) + KL\left(p_2 \| \frac{p_1 + p_2}{2}\right)\right)$$

The training criterion $C(G)$ of GAN can be written in terms of the Jensen-Shannon divergence:

$$C(G) = -\log(4) + 2JSD\left(p_{\text{data}} \| p_g\right)$$
$$\Rightarrow G^* = \arg\min_G C(G) = \arg\min_G JSD\left(p_{\text{data}} \| p_g\right)$$

Jensen-Shannon Divergence

Definition (Jensen-Shannon Divergence)

The Jensen-Shannon divergence between two distributions $p_1(x)$ and $p_2(x)$ is defined as:

$$JSD(p_1 \| p_2) \triangleq \frac{1}{2}\left(KL\left(p_1 \middle\| \frac{p_1+p_2}{2}\right) + KL\left(p_2 \middle\| \frac{p_1+p_2}{2}\right)\right)$$

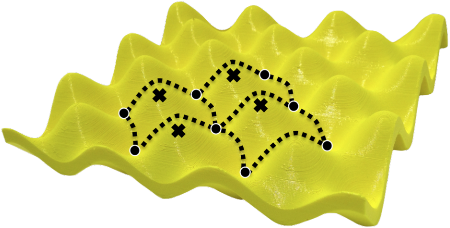
The training criterion $C(G)$ of GAN can be written in terms of the Jensen-Shannon divergence:

$$C(G) = -\log(4) + 2JSD\left(p_{\text{data}} \| p_g\right)$$
$$\Rightarrow G^* = \arg\min_G C(G) = \arg\min_G JSD\left(p_{\text{data}} \| p_g\right)$$

Distribution matching!

Convergence of the Algorithm

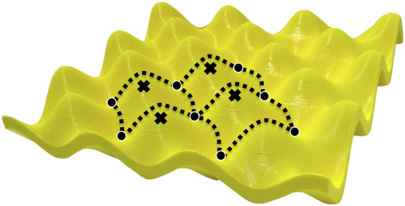
Intuitively not necessary converge



- Hard to say which one is the global optima.
- Not able to reach the globally optimal saddle point.

Convergence of the Algorithm

Intuitively not necessary converge



Theorem

If G and D have enough capacity, and at each step of Algorithm 1, the discriminator is allowed to reach its optimum given G, and p_g is updated so as to improve the criterion

$$\mathbb{E}_{\mathbf{x}\sim p_{data}} \log D_G^*(\mathbf{x}) + \mathbb{E}_{\mathbf{x}\sim p_g} \log (1 - D_G^*(\mathbf{x})) \ ,$$

then p_g converges to p_{data}.

Proof Idea

$$V(G, D) = \int_x \left(p_{\text{data}(\mathbf{x})} \log(D(\mathbf{x})) + p_g(\mathbf{x}) \log(1 - D(\mathbf{x}))\right) \mathrm{d}\,\mathbf{x}$$

GAN objective

$$\begin{aligned}
&\min_G C(G) \triangleq \min_G \max_D V(G, D) \\
&= -\log(4) + 2 \min_G JSD\left(p_{\text{data}} \| p_g\right) \\
&= -\log(4) + \min_G KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_g}{2}\right) + KL\left(p_g \| \frac{p_{\text{data}} + p_g}{2}\right)
\end{aligned}$$

Proof Idea

GAN objective

$$
\begin{aligned}
\min_G C(G) &\triangleq \min_G \max_D V(G, D) \\
&= -\log(4) + 2 \min_G JSD\left(p_{\text{data}} \| p_g\right) \\
&= -\log(4) + \min_G KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_g}{2}\right) + KL\left(p_g \| \frac{p_{\text{data}} + p_g}{2}\right)
\end{aligned}
$$

Alternative GAN objective in the space of probability distributions

$$
\min_{p_g} \mathcal{F}(p_g) \triangleq \min_{p_g} KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_g}{2}\right) + KL\left(p_g \| \frac{p_{\text{data}} + p_g}{2}\right) .
$$

Proof Idea

$$\mathcal{F}(p_g) = KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_g}{2}\right) + KL\left(p_g \| \frac{p_{\text{data}} + p_g}{2}\right) .$$

- To ensure global optima, we need to prove $\mathcal{F}(p_g)$ is convex w.r.t. p_g.
  - If convex, we can find the global optima (*i.e.*, $p_g = p_{\text{data}}$) by doing gradient descent on p_g.
  - Convexity in space of distributions!
- Specifically, we need to prove for two distributions p_1 and p_2 and $\lambda \in [0, 1]$:

$$\mathcal{F}(\lambda p_1 + (1 - \lambda)p_2) \leq \lambda\mathcal{F}(p_1) + (1 - \lambda)\mathcal{F}(p_2) .$$

Proof Idea

$$\mathcal{F}(p_g) = KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_g}{2}\right) + KL\left(p_g \| \frac{p_{\text{data}} + p_g}{2}\right) .$$

- To ensure global optima, we need to prove $\mathcal{F}(p_g)$ is convex w.r.t. p_g.
  - If convex, we can find the global optima (*i.e.*, $p_g = p_{\text{data}}$) by doing gradient descent on p_g.
  - Convexity in space of distributions!
- Specifically, we need to prove for two distributions p_1 and p_2 and $\lambda \in [0, 1]$:

$$\mathcal{F}(\lambda p_1 + (1 - \lambda)p_2) \leq \lambda \mathcal{F}(p_1) + (1 - \lambda)\mathcal{F}(p_2) .$$

Proof Idea

$$\mathcal{F}(p_g) = KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_g}{2}\right) + KL\left(p_g \| \frac{p_{\text{data}} + p_g}{2}\right) .$$
$$\Rightarrow \mathcal{F}(\lambda p_1 + (1-\lambda)p_2) \le \lambda \mathcal{F}(p_1) + (1-\lambda)\mathcal{F}(p_2)$$

- Let $p = \lambda p_1 + (1-\lambda)p_2$, we need to prove:

$$\begin{aligned}
&KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p}{2}\right) + KL\left(p \| \frac{p_{\text{data}} + p}{2}\right) \\
\le& \lambda KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_1}{2}\right) + (1-\lambda) KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_2}{2}\right) \\
&+ \lambda KL\left(p_1 \| \frac{p_{\text{data}} + p_1}{2}\right) + (1-\lambda) KL\left(p_2 \| \frac{p_{\text{data}} + p_2}{2}\right)
\end{aligned}$$

Proof Idea

$$
\begin{aligned}
&KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p}{2}\right) + KL\left(p \| \frac{p_{\text{data}} + p}{2}\right) \\
\le &\lambda KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_1}{2}\right) + (1 - \lambda) KL\left(p_{\text{data}} \| \frac{p_{\text{data}} + p_2}{2}\right) \\
&+ \lambda KL\left(p_1 \| \frac{p_{\text{data}} + p_1}{2}\right) + (1 - \lambda) KL\left(p_2 \| \frac{p_{\text{data}} + p_2}{2}\right)
\end{aligned}
$$

Lemma (Convexity of KL divergence)

Let a_1, b_1 and a_2, b_2 be probability distributions over x, and $\lambda \in (0, 1)$. Define $a = \lambda a_1 + (1 - \lambda) a_2$, $b = \lambda b_1 + (1 - \lambda) b_2$. Then

$$KL(a \| b) \le \lambda KL(a_1 \| b_1) + (1 - \lambda) KL(a_2 \| b_2)$$

Proof Idea

$$KL\left(a\|b\right) \leq \lambda KL\left(a_1\|b_1\right) + (1-\lambda)KL\left(a_2\|b_2\right)$$

$$p = \lambda p_1 + (1-\lambda)p_2$$

Proof of

$$KL\left(p_{\mathbf{data}}\|\frac{p_{\mathbf{data}}+p}{2}\right) \leq \lambda KL\left(p_{\mathbf{data}}\|\frac{p_{\mathbf{data}}+p_1}{2}\right) + (1-\lambda)KL\left(p_{\mathbf{data}}\|\frac{p_{\mathbf{data}}+p_2}{2}\right)$$

Proof Idea

$$KL(a\|b) \leq \lambda KL(a_1\|b_1) + (1-\lambda)KL(a_2\|b_2)$$

$$p = \lambda p_1 + (1-\lambda)p_2$$

Proof of $KL\left(p_{\mathbf{data}}\|\frac{p_{\mathbf{data}}+p}{2}\right) \leq \lambda KL\left(p_{\mathbf{data}}\|\frac{p_{\mathbf{data}}+p_1}{2}\right) + (1-\lambda)KL\left(p_{\mathbf{data}}\|\frac{p_{\mathbf{data}}+p_2}{2}\right)$

Proof.

Let $a_1 = a_2 = p_{\text{data}}$, $b_1 = \frac{p_{\text{data}}+p_1}{2}$, $b_2 = \frac{p_{\text{data}}+p_2}{2}$. Substituting these into the KL inequality, we get the conclusion. □

Proof Idea

$$KL\left(a \| b\right) \leq \lambda KL\left(a_1 \| b_1\right) + (1-\lambda) KL\left(a_2 \| b_2\right)$$

$$p = \lambda p_1 + (1-\lambda) p_2$$

Proof of $KL\left(p \| \frac{p_{\textbf{data}}+p}{2}\right) \leq \lambda KL\left(p_1 \| \frac{p_{\textbf{data}}+p_1}{2}\right) + (1-\lambda) KL\left(p_2 \| \frac{p_{\textbf{data}}+p_2}{2}\right)$

Proof Idea

$$KL(a\|b) \leq \lambda KL(a_1\|b_1) + (1-\lambda)KL(a_2\|b_2)$$

$$p = \lambda p_1 + (1-\lambda)p_2$$

Proof of $KL\left(p\|\frac{p_{\textbf{data}}+p}{2}\right) \leq \lambda KL\left(p_1\|\frac{p_{\textbf{data}}+p_1}{2}\right) + (1-\lambda)KL\left(p_2\|\frac{p_{\textbf{data}}+p_2}{2}\right)$

Proof.

Let $a_1 = p_1$, $a_2 = p_2$, $b_1 = \frac{p_{\text{data}}+p_1}{2}$, $b_2 = \frac{p_{\text{data}}+p_2}{2}$. Substituting these into the KL inequality, we get the conclusion. □

Proof Idea

Conclusion

$\mathcal{F}(p_g) = KL\left(p_{\text{data}} \| \frac{p_{\text{data}}+p_g}{2}\right) + KL\left(p_g \| \frac{p_{\text{data}}+p_g}{2}\right)$ is convex w.r.t. p_g:

- Global optima of $p_g = p_{\text{data}}$ can be obtained by optimizing $\mathcal{F}(p_g)$ with sub-gradient descent on the space of probability distributions:
  - Sub-gradient descent on the space of probability distributions corresponds to gradient descent on the parameter space of G.

Important

- Before optimizing p_g from $\mathcal{F}(p_g)$, we have assume the discriminator D is optimal given G, *i.e.*, $D^* = \frac{p_{\text{data}}(\mathbf{x})}{p_{\text{data}}(\mathbf{x})+p_g(\mathbf{x})}$.
- This is why we need to do SGD for several steps for discriminator in the algorithm in each round.

Proof Idea

Convexity of KL divergence

Let a_1, b_1 and a_2, b_2 be probability distributions over x, and $\lambda \in (0, 1)$. Define $a = \lambda a_1 + (1 - \lambda)a_2$, $b = \lambda b_1 + (1 - \lambda)b_2$. Then

$$KL(a \| b) \leq \lambda KL(a_1 \| b_1) + (1 - \lambda)KL(a_2 \| b_2)$$

Proof Idea

Convexity of KL divergence

Let a_1, b_1 and a_2, b_2 be probability distributions over x, and $\lambda \in (0, 1)$. Define $a = \lambda a_1 + (1 - \lambda)a_2$, $b = \lambda b_1 + (1 - \lambda)b_2$. Then

$$KL(a\|b) \leq \lambda KL(a_1\|b_1) + (1 - \lambda)KL(a_2\|b_2)$$

Lemma (Log sum inequality)

Let $a_1, \cdots, a_n$ and $b_1, \cdots, b_n$ be nonnegative numbers. The log sum inequality states that

$$\left(\sum_i a_i\right) \log \frac{\sum_i a_i}{\sum_i b_i} \leq \sum_i a_i \log \frac{a_i}{b_i}$$

Proof Idea

$$KL(a\|b) \leq \lambda KL(a_1\|b_1) + (1-\lambda) KL(a_2\|b_2)$$

$$\left(\sum_i a_i\right) \log \frac{\sum_i a_i}{\sum_i b_i} \leq \sum_i a_i \log \frac{a_i}{b_i}$$

Proof.

Let $p_1 = \lambda a_1$, $p_2 = (1-\lambda)a_2$, $q_1 = \lambda b_1$, $q_2 = (1-\lambda)b_2$.

$$\begin{aligned}
KL(a\|b) &= \int (\lambda a_1(x) + (1-\lambda)a_2(x)) \log \frac{\lambda a_1(x) + (1-\lambda)a_2(x)}{\lambda b_1(x) + (1-\lambda)b_2(x)} \mathrm{d}x \\
&= \int (p_1(x) + p_2(x)) \log \frac{p_1(x) + p_2(x)}{q_1(x) + q_2(x)} \mathrm{d}x \\
&\leq \int \left(p_1(x) \log \frac{p_1(x)}{q_1(x)} + p_2(x) \log \frac{p_2(x)}{q_2(x)}\right) \mathrm{d}x \\
&= \int \left(\lambda a_1(x) \log \frac{\lambda a_1(x)}{\lambda b_1(x)} + (1-\lambda)a_2(x) \log \frac{(1-\lambda)a_2(x)}{(1-\lambda)b_2(x)}\right) \mathrm{d}x \\
&= \lambda KL(a_1\|b_1) + (1-\lambda) KL(a_2\|b_2)
\end{aligned}$$

Announcement

Final exam

- 3:30-5:30PM, May 16, Filmor 355
- Semi-open-book: you are only allowed to bring in **one** A4 paper, write down whatever you want for your reference in the exam.

Project presentation

- Starting from May 2.
- Each group take turns to do a 10 minute project presentation, talk about what your project is, and what your results are.

  Even if you haven't finished your project, you should present whatever you got so far.
- Detailed schedule will be released later.