Generative models

Generative models

- Generative Moment Matching Networks

- Generative Adversarial Networks (GAN)

Generative Moment Matching Networks

- Black board

Generative Adversarial Networks (GAN)

- Original paper:
  - Generative Adversarial Nets
- Authors:
  - Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio (2014)
- Organization:
  - Université de Montréal
- URL:
  - https://arxiv.org/abs/1406.2661

Generative Adversarial Networks (GAN)

Generative Adversarial Networks (GAN)

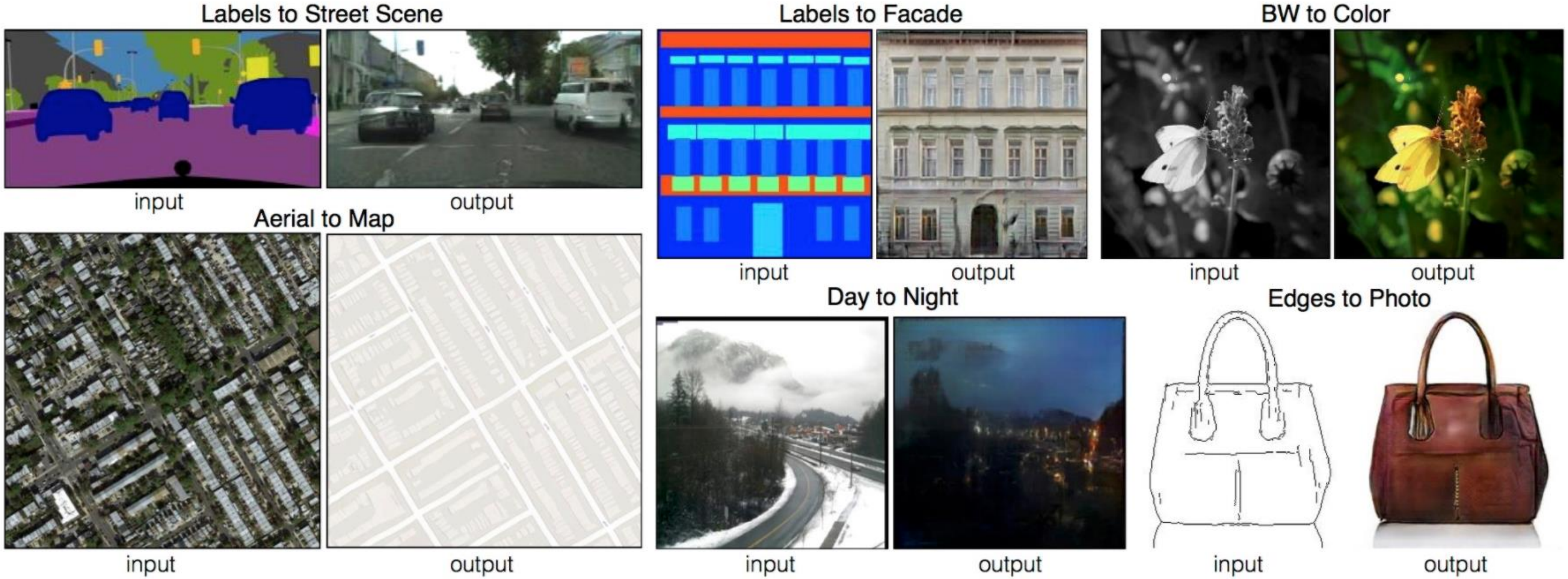
- Bengio: This may hold the key to making computers a lot more intelligent.

Generative Adversarial Networks (GAN)

- Bengio: This may hold the key to making computers a lot more intelligent.

- LeCun: The most important breakthrough, in my opinion, is adversarial training (also called GAN). This is the most interesting idea in the last 10 years in ML, in my opinion.

Different Applications

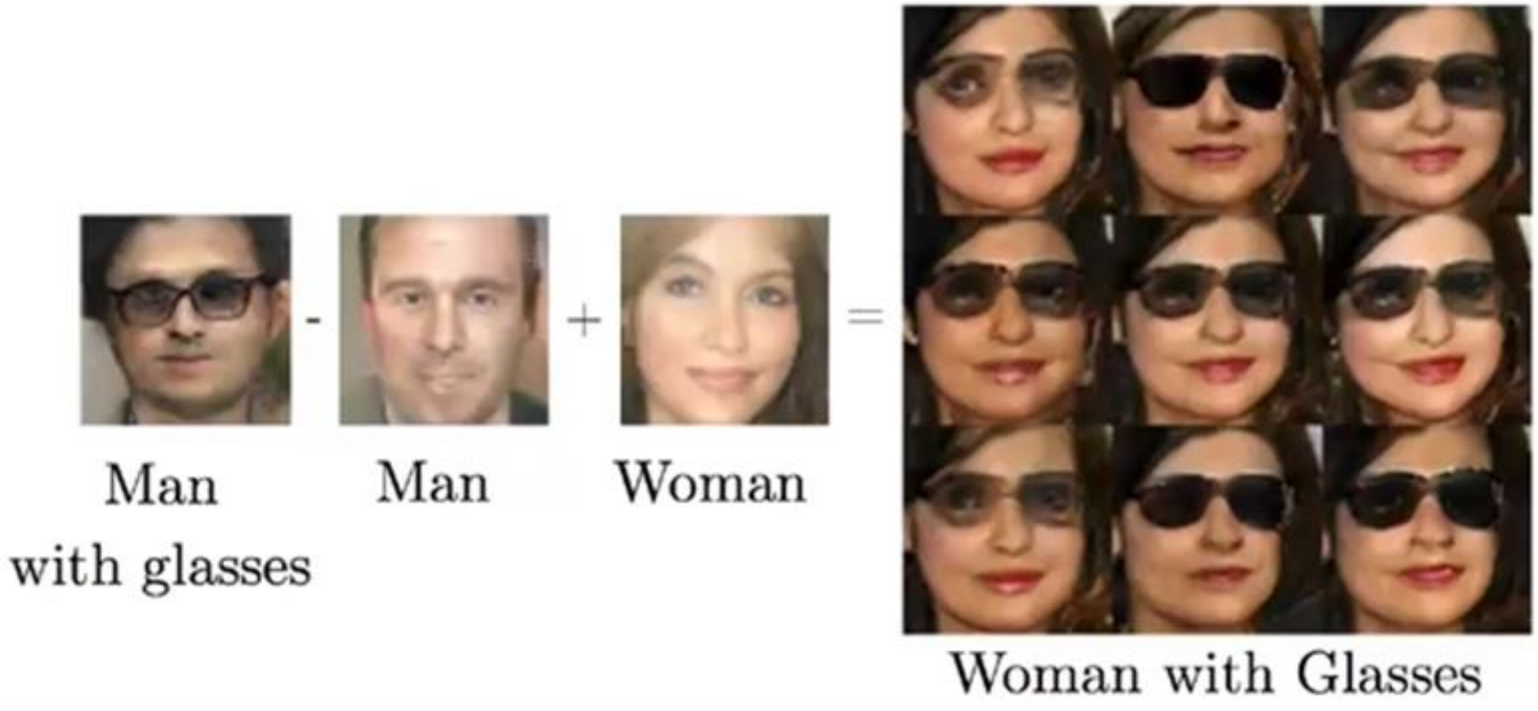


DCGANs for LSUN Bedrooms

(Radford et al 2015)

Vector Space Arithmetic

- Similar to word embedding (DCGAN paper)



(Radford et al 2015)

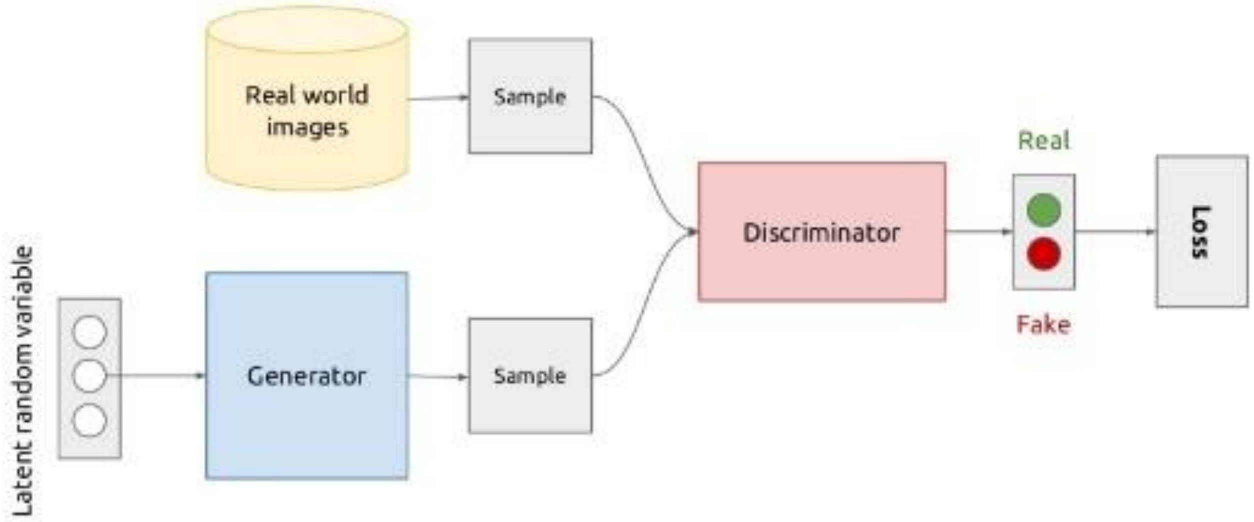
PPGN for caption to image

- From natural language to pictures

Oranges on a table next to liquor bottle

(Nguyen et al 2016)

Adversarial Learning

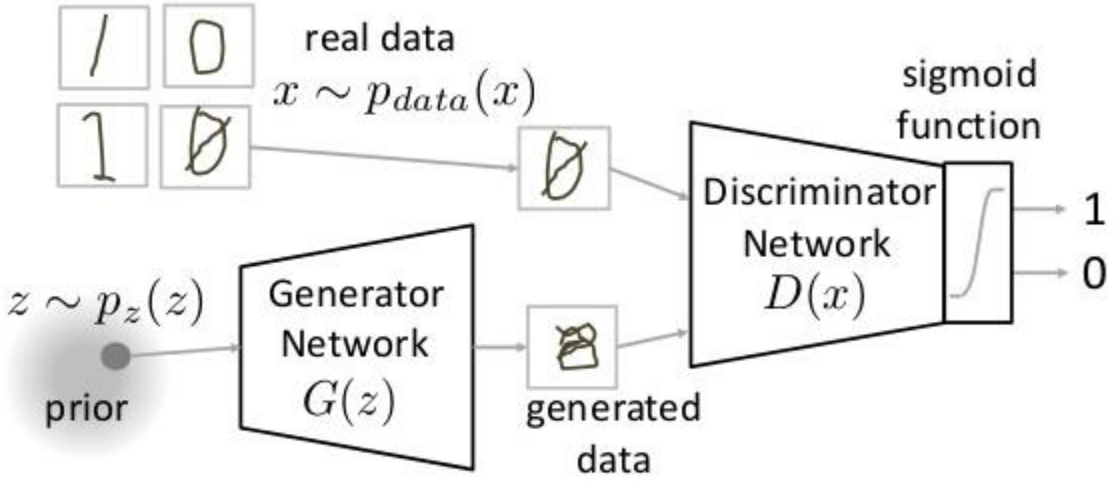


http://www.slideshare.net/xavigiro/deep-learning-for-computer-vision-generative-models-and-adversarial-training-upc-2016

Generative Adversarial Networks

$$\min_{G} \max_{D} V(D, G)$$

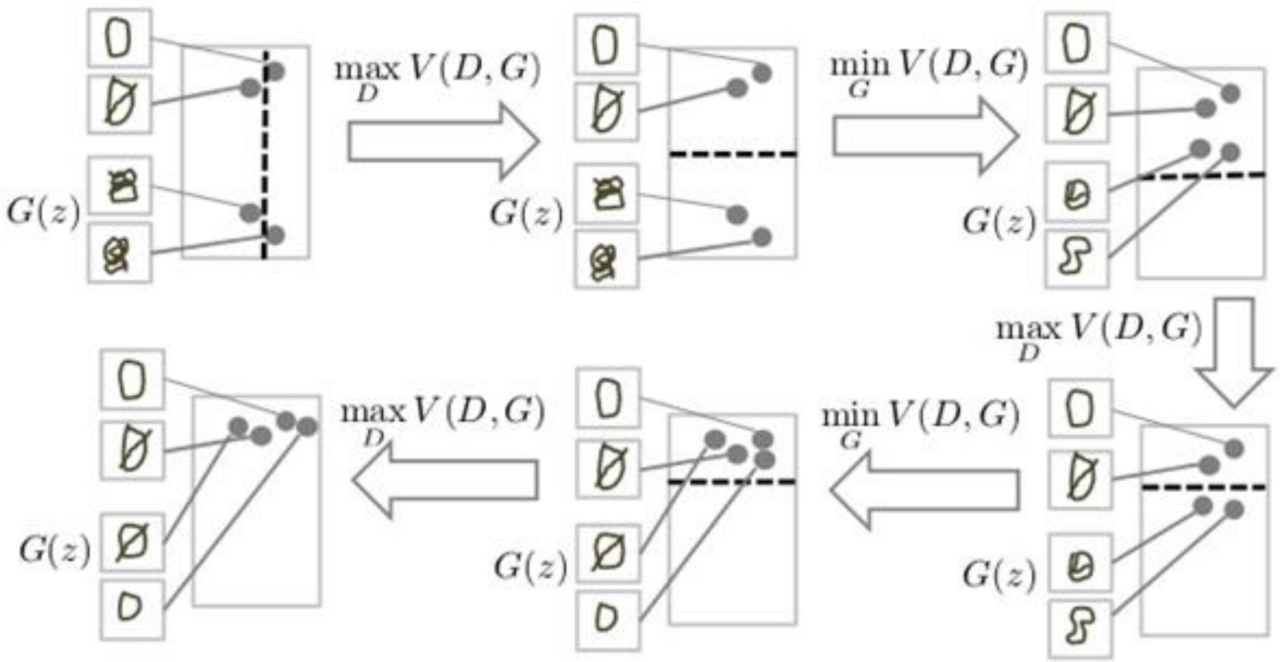
$$V(D, G) = \mathbb{E}_{x \sim P_{data}(x)}[logD(x)] + \mathbb{E}_{z \sim P_z(z)}[1 - logD(G(z))]$$



Credit: Mark Chang

Training
Generative Adversarial Networks

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



Credit: Mark Chang

$$V(D, G) = \mathbb{E}_{x \sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z \sim p_z(z)}[1 - logD(G(z))]$$

$$V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z\sim p_z(z)}[1 - logD(G(z))]$$

$$= \int_x p_{data}(x)log(D(x))dx + \int_z p_z(z)log(1 - D(G(z)))dz$$

$$V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z\sim p_z(z)}[1 - logD(G(z))]$$

$$= \int_x p_{data}(x)log(D(x))dx + \int_z p_z(z)log(1 - D(G(z)))dz$$

$$x = G(z) \Rightarrow z = G^{-1}(x) \Rightarrow dz = (G^{-1})'(x)dx$$

$$V(D, G) = \mathbb{E}_{x \sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z \sim p_z(z)}[1 - logD(G(z))]$$

$$= \int_x p_{data}(x) log(D(x))dx + \int_z p_z(z) log(1 - D(G(z)))dz$$

$$x = G(z) \Rightarrow z = G^{-1}(x) \Rightarrow dz = (G^{-1})'(x)dx$$

$$\Rightarrow p_g(x) = p_z(G^{-1}(x))(G^{-1})'(x)$$

$$V(D, G) = \mathbb{E}_{x \sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z \sim p_z(z)}[1 - logD(G(z))]$$

$$= \int_x p_{data}(x) log(D(x)) dx + \int_z p_z(z) log(1 - D(G(z))) dz$$

$$x = G(z) \Rightarrow z = G^{-1}(x) \Rightarrow dz = (G^{-1})'(x) dx$$

$$\Rightarrow p_g(x) = p_z(G^{-1}(x))(G^{-1})'(x)$$

$$= \int_x p_{data}(x) log(D(x)) dx + \int_x p_z(G^{-1}(x)) log(1 - D(x))(G^{-1})'(x) dx$$

$$V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z\sim p_z(z)}[1 - logD(G(z))]$$

$$= \int_x p_{data}(x)log(D(x))dx + \int_z p_z(z)log(1 - D(G(z)))dz$$

$$x = G(z) \Rightarrow z = G^{-1}(x) \Rightarrow dz = (G^{-1})'(x)dx$$

$$\Rightarrow p_g(x) = p_z(G^{-1}(x))(G^{-1})'(x)$$

$$= \int_x p_{data}(x)log(D(x))dx + \int_x p_z(G^{-1}(x))log(1 - D(x))(G^{-1})'(x)dx$$

$$= \int_x p_{data}(x)log(D(x))dx + \int_x p_g(x)log(1 - D(x))dx$$

$$V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z\sim p_z(z)}[1 - logD(G(z))]$$

$$= \int_x p_{data}(x)log(D(x))dx + \int_z p_z(z)log(1 - D(G(z)))dz$$

$$x = G(z) \Rightarrow z = G^{-1}(x) \Rightarrow dz = (G^{-1})'(x)dx$$

$$\Rightarrow p_g(x) = p_z(G^{-1}(x))(G^{-1})'(x)$$

$$= \int_x p_{data}(x)log(D(x))dx + \int_x p_z(G^{-1}(x))log(1 - D(x))(G^{-1})'(x)dx$$

$$= \int_x p_{data}(x)log(D(x))dx + \int_x p_g(x)log(1 - D(x))dx$$

$$= \int_x p_{data}(x)log(D(x)) + p_g(x)log(1 - D(x))dx$$

Understanding the objective function

$$\max_D V(D,G) = \max_D \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx$$

Understanding the objective function

$$\max_D V(D, G) = \max_D \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx$$

$$\frac{\partial}{\partial D(x)} (p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x))) = 0$$

Understanding the objective function

$$\max_D V(D,G) = \max_D \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx$$

$$\frac{\partial}{\partial D(x)}(p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x))) = 0$$

$$\Rightarrow \frac{p_{data}(x)}{D(x)} - \frac{p_g(x)}{1 - D(x)} = 0$$

Understanding the objective function

$$\max_D V(D, G) = \max_D \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx$$

$$\frac{\partial}{\partial D(x)} (p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x))) = 0$$

$$\Rightarrow \frac{p_{data}(x)}{D(x)} - \frac{p_g(x)}{1 - D(x)} = 0$$

$$\Rightarrow D(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$$

Understanding the objective function

Suppose the discriminator is optimal $D_G^*(x)$,
the optimal generator makes: $p_{data}(x) = p_g(x)$

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

Understanding the objective function

$$C(G) = \max_D V(G, D)$$

Understanding the objective function

$$C(G) = \max_D V(G, D)$$

$$= \max_D \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx$$

Understanding the objective function

$$C(G) = \max_D V(G, D)$$

$$= \max_D \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx$$

$$= \int_x p_{data}(x) log(D^*_G(x)) + p_g(x) log(1 - D^*_G(x)) dx$$

Understanding the objective function

$$C(G) = \max_D V(G, D)$$

$$= \max_D \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx$$

$$= \int_x p_{data}(x) log(D_G^*(x)) + p_g(x) log(1 - D_G^*(x)) dx$$

$$= \int_x p_{data}(x) log(\frac{p_{data}(x)}{p_{data}(x) + p_g(x)}) + p_g(x) log(\frac{p_g(x)}{p_{data}(x) + p_g(x)}) dx$$

Understanding the objective function

$$C(G) = \max_D V(G, D)$$

$$= \max_D \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx$$

$$= \int_x p_{data}(x) log(D_G^*(x)) + p_g(x) log(1 - D_G^*(x)) dx$$

$$= \int_x p_{data}(x) log(\frac{p_{data}(x)}{p_{data}(x) + p_g(x)}) + p_g(x) log(\frac{p_g(x)}{p_{data}(x) + p_g(x)}) dx$$

$$= \int_x p_{data}(x) log(\frac{p_{data}(x)}{\frac{p_{data}(x)+p_g(x)}{2}}) + p_g(x) log(\frac{p_g(x)}{\frac{p_{data}(x)+p_g(x)}{2}}) dx - log(4)$$

Understanding the objective function

$$C(G) = \max_{D} V(G, D)$$

$$= \max_{D} \int_x p_{data}(x) log(D(x)) + p_g(x) log(1 - D(x)) dx$$

$$= \int_x p_{data}(x) log(D_G^*(x)) + p_g(x) log(1 - D_G^*(x)) dx$$

$$= \int_x p_{data}(x) log(\frac{p_{data}(x)}{p_{data}(x) + p_g(x)}) + p_g(x) log(\frac{p_g(x)}{p_{data}(x) + p_g(x)}) dx$$

$$= \int_x p_{data}(x) log(\frac{p_{data}(x)}{\frac{p_{data}(x)+p_g(x)}{2}}) + p_g(x) log(\frac{p_g(x)}{\frac{p_{data}(x)+p_g(x)}{2}}) dx - log(4)$$

$$= KL[p_{data}(x) || \frac{p_{data}(x) + p_g(x)}{2}] + KL[p_g(x) || \frac{p_{data}(x) + p_g(x)}{2}] - log(4)$$

Understanding the objective function

$$C(G) = KL[p_{data}(x) || \frac{p_{data}(x)+p_g(x)}{2}] + KL[p_g(x) || \frac{p_{data}(x)+p_g(x)}{2}] - log(4)$$

$$\geq 0 \qquad\qquad \geq 0$$

Understanding the objective function

$$C(G) = \underbrace{KL[p_{data}(x) || \tfrac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} + \underbrace{KL[p_g(x) || \tfrac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} - log(4)$$

$$\min_G C(G) = 0 + 0 - log(4) = -log(4)$$

Understanding the objective function

$$C(G) = \underbrace{KL[p_{data}(x) || \tfrac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} + \underbrace{KL[p_g(x) || \tfrac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} - log(4)$$

$$\min_G C(G) = 0 + 0 - log(4) = -log(4)$$

$$KL[p_{data}(x) || \frac{p_{data}(x) + p_g(x)}{2}] = 0$$

Understanding the objective function

$$C(G) = \underbrace{KL[p_{data}(x)||\tfrac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} + \underbrace{KL[p_g(x)||\tfrac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} - log(4)$$

$$\min_G C(G) = 0 + 0 - log(4) = -log(4)$$

$$KL[p_{data}(x)||\frac{p_{data}(x)+p_g(x)}{2}] = 0$$

when $\quad p_{data}(x) = \frac{p_{data}(x)+p_g(x)}{2}$

Understanding the objective function

$$C(G) = \underbrace{KL[p_{data}(x)||\frac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} + \underbrace{KL[p_g(x)||\frac{p_{data}(x)+p_g(x)}{2}]}_{\geq 0} - log(4)$$

$$\min_G C(G) = 0 + 0 - log(4) = -log(4)$$

$$KL[p_{data}(x)||\frac{p_{data}(x)+p_g(x)}{2}] = 0$$

when $\quad p_{data}(x) = \frac{p_{data}(x)+p_g(x)}{2}$

$$\Rightarrow p_{data}(x) = p_g(x)$$

KL (Kullback-Leibler) divergence

- Jensen-Shannon Divergency (symmetric KL):

$$JSD(P||Q) = \frac{1}{2}D_{KL}(P||M) + \frac{1}{2}D_{KL}(Q||M),$$

$$M = \frac{1}{2}(P + Q)$$

Summary:

- Generator G, Discriminator D

$$V = \mathbb{E}_{x \sim P_{data}}[logD(x)] + \mathbb{E}_{x \sim P_G}[log(1 - D(x))]$$

Summary:

- Generator G, Discriminator D
- Looking for G^* such that

$$G^* = arg \min_G \max_D V(G, D)$$

$$V = \mathbb{E}_{x \sim P_{data}}[logD(x)] + \mathbb{E}_{x \sim P_G}[log(1 - D(x))]$$

Summary:

$$V = \mathbb{E}_{x \sim P_{data}}[logD(x)] + \mathbb{E}_{x \sim P_G}[log(1 - D(x))]$$

- Generator G, Discriminator D
- Looking for G^* such that

$$G^* = arg \min_G \max_D V(G, D)$$

- Given G, $\max_D V(G, D)$

$$= -2log(2) + 2JSD(P_{data}(x) || P_G(x))$$

Summary:

$$V = \mathbb{E}_{x \sim P_{data}}[logD(x)] + \mathbb{E}_{x \sim P_G}[log(1 - D(x))]$$

- Generator G, Discriminator D
- Looking for G^* such that

$$G^* = arg \min_G \max_D V(G, D)$$

- Given $G, \max_D V(G, D)$

$$= -2log(2) + 2JSD(P_{data}(x) || P_G(x))$$

- What is the optimal G? It is G that makes JSD smallest $= 0$:

$$P_G(x) = P_{data}(x)$$

Text to Image, by conditional GAN

Caption	Image
a pitcher is about to throw the ball to the batter	
a group of people on skis stand in the snow	
a man in a wet suit riding a surfboard on a wave	

Text to Image
- Results

From CY Lee lecture

Caption	Image
this flower has white petals and a yellow stamen	
the center is yellow surrounded by wavy dark purple petals	
this flower has lots of small round pink petals	

Project topic: Code and data are all on web, many possibilities!

Text to Image - Results

"red flower with black center"

From CY Lee lecture

Caption	Image
this flower has white petals and a yellow stamen	
the center is yellow surrounded by wavy dark purple petals	
this flower has lots of small round pink petals	

Project topic: Code and data are all on web, many possibilities!

Text to Image - Results

"red flower with black center"

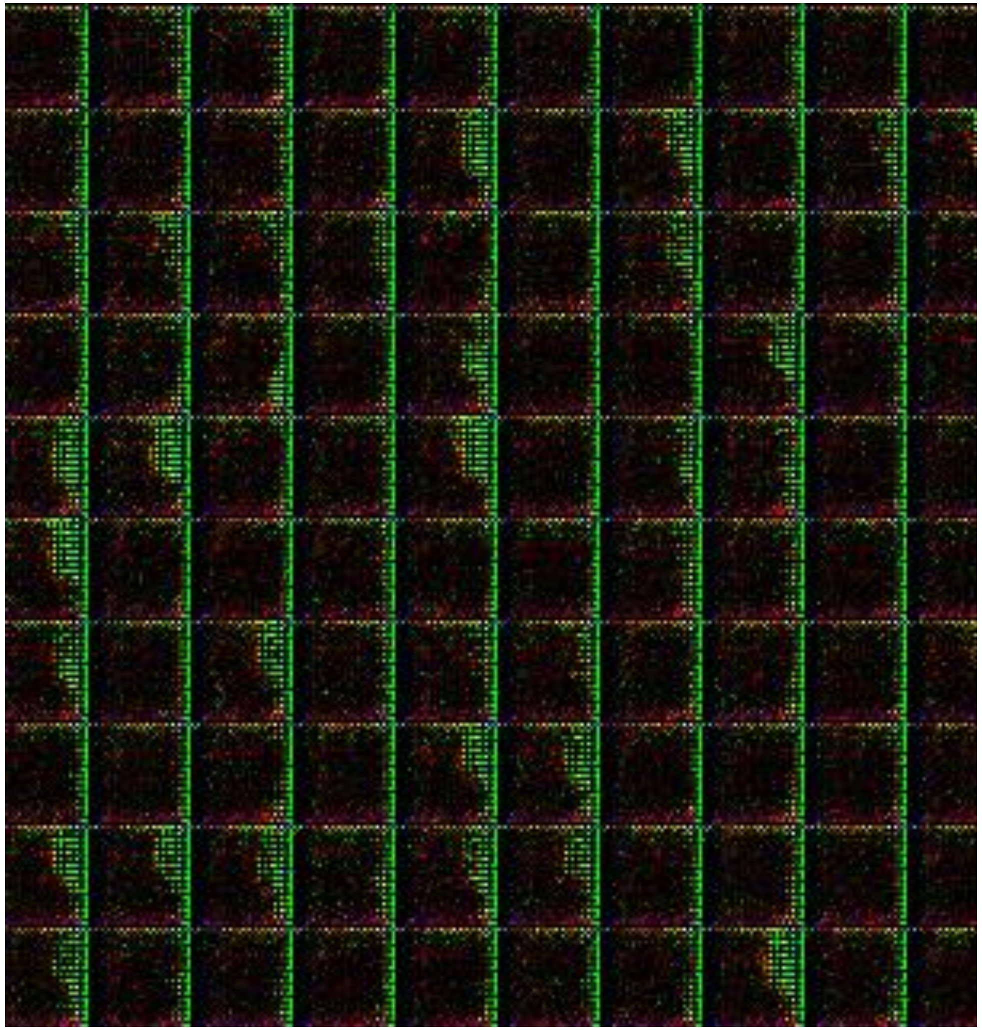
From CY Lee lecture

Caption	Image
this flower has white petals and a yellow stamen	
the center is yellow surrounded by wavy dark purple petals	
this flower has lots of small round pink petals	

Project topic: Code and data are all on web, many possibilities!

VAE

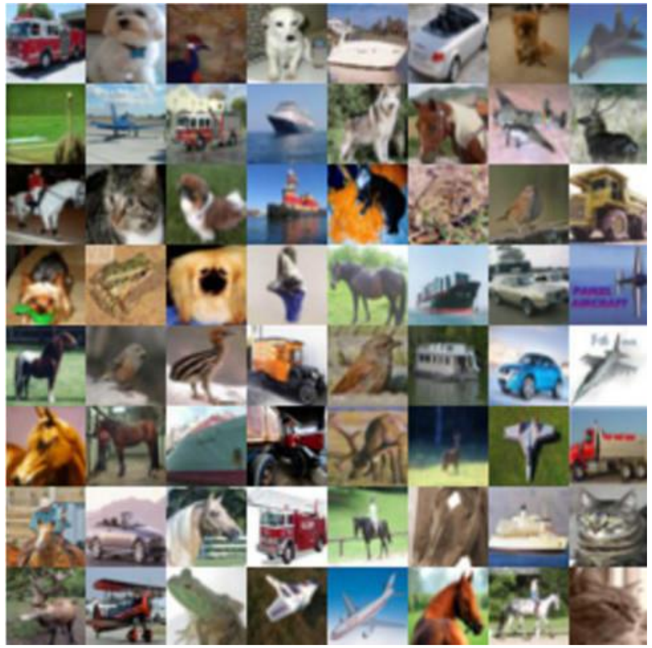
GAN



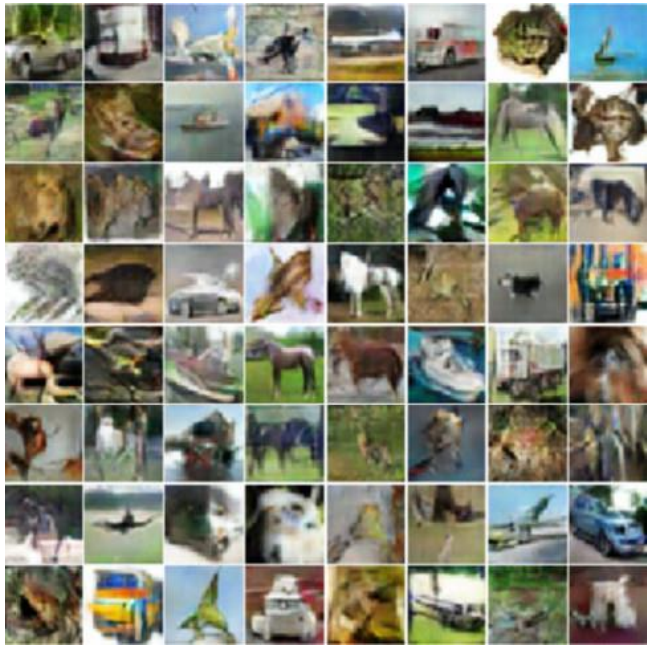
VAE

GAN

Real images (CIFAR-10)



Generated images



Source Code

- Original paper (theano):
  - https://github.com/goodfeli/adversarial
- Tensorflow implementation:
  - https://github.com/ckmarkoh/GAN-tensorflow

In practice ...

$$V = \mathbb{E}_{x\sim P_{data}}[log D(x)] + \mathbb{E}_{x\sim P_G}[log(1 - D(x))]$$

- Given G, how to compute $\max_D V(G, D)$?
  - Sample $\{x^1, , x^m\}$ from P_{data}
  - Sample $\{x^{*1}, , x^{*m}\}$ from generator P_G

Maximize:

$$V' = \frac{1}{m\sum_{i=1}^{m} logD(x^i)} + \frac{1}{m\sum_{i=1}^{m} log(1-D(x^{*i}))}$$

Credit: Mark Chang

In practice …

$$V = \mathbb{E}_{x \sim P_{data}}[logD(x)] + \mathbb{E}_{x \sim P_G}[log(1 - D(x))]$$

- Given G, how to compute $\max_D V(G,D)$?
  - Sample $\{x^1, , x^m\}$ from P_{data}
  - Sample $\{x^{*1}, , x^{*m}\}$ from generator P_G

Maximize:

$$V' = \frac{1}{m \sum_{i=1}^{m} logD(x^i)} + \frac{1}{m \sum_{i=1}^{m} log(1 - D(x^{*i}))}$$

This is what a Binary Classifier do

Output is $D(x)$ Minimize Cross-entropy

If x is a positive example → Minimize –log D(x)

If x is a negative example → Minimize –log(1-D(x))

In practice …

$$V = \mathbb{E}_{x \sim P_{data}}[log D(x)] + \mathbb{E}_{x \sim P_G}[log(1 - D(x))]$$

- Given G, how to compute $\max_D V(G, D)$?
  - Sample $\{x^1, , x^m\}$ from P_{data}
  - Sample $\{x^{*1}, , x^{*m}\}$ from generator P_G

Positive example

D must accept

Maximize:

$$V' = \frac{1}{m \sum_{i=1}^{m} log D(x^i)} + \frac{1}{m \sum_{i=1}^{m} log(1 - D(x^{*i}))}$$

This is what a Binary Classifier do

Output is $D(x)$ Minimize Cross-entropy

If x is a positive example → Minimize –log D(x)

If x is a negative example → Minimize –log(1-D(x))

In practice ...

$$V = \mathbb{E}_{x \sim P_{data}}[logD(x)] + \mathbb{E}_{x \sim P_G}[log(1 - D(x))]$$

- Given G, how to compute $\max_D V(G, D)$?
  - Sample $\{x^1, , x^m\}$ from P_{data}
  - Sample $\{x^{*1}, , x^{*m}\}$ from generator P_G

Maximize:

$$V' = \frac{1}{m \sum_{i=1}^{m} logD(x^i)} + \frac{1}{m \sum_{i=1}^{m} log(1 - D(x^{*i}))}$$

Positive example
D must accept

Negative example
D must reject

This is what a Binary Classifier do

Output is $D(x)$ Minimize Cross-entropy

If x is a positive example → Minimize –log D(x)

If x is a negative example → Minimize –log(1-D(x))