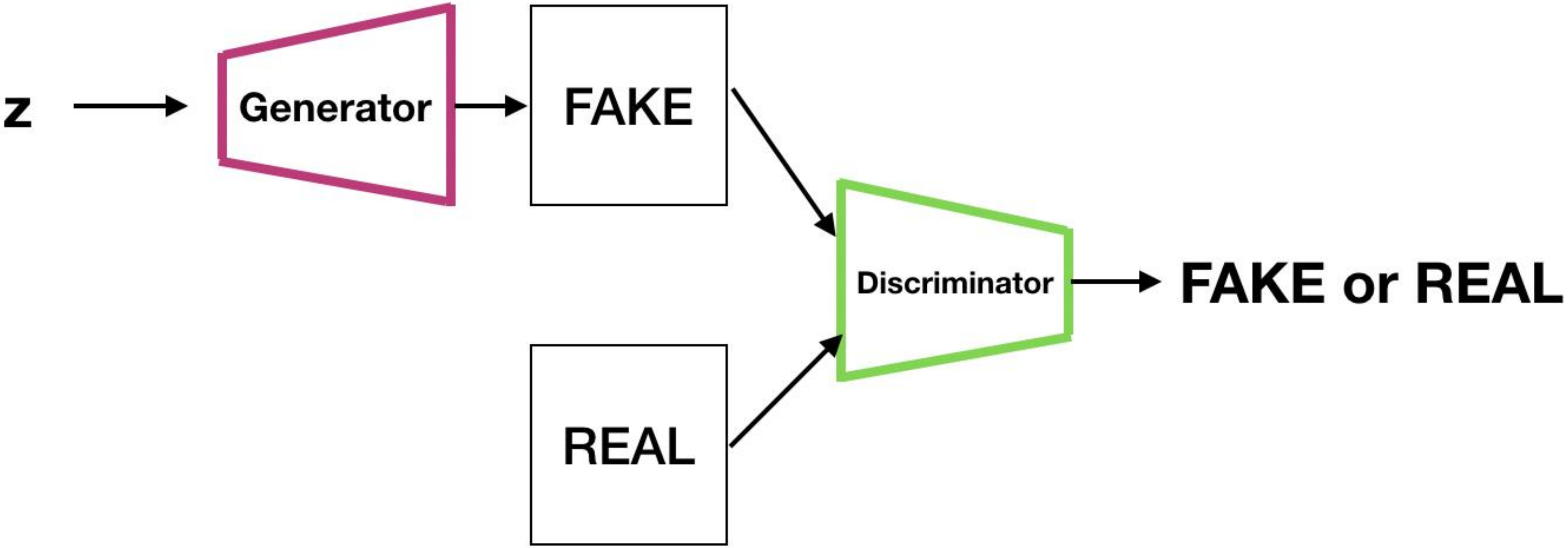
Generative Adversarial Nets 논문 리뷰

GAN?



0 Abstract

- 경쟁하는 프로세스를 통해 generative model을 추정하는 새로운 프레임워크를 제안
- 생성모델인 G (generative model)는 training data의 분포를 모사 -> discriminative model이 구별하지 못하도록
- 판별모델인 D (discriminative model)는 sample 데이터가 G로부터 나온 데이터가 아닌 training data로부터 나온 데이터일 확률을 추정
- -> G는 실제 training data의 분포를 모사하며 그와 비슷한 데이터를 생성하려하고, D는 실제 데이터와 G가 생성해낸 데이터를 구별하려하는 경쟁적인 과정

1 Introduction

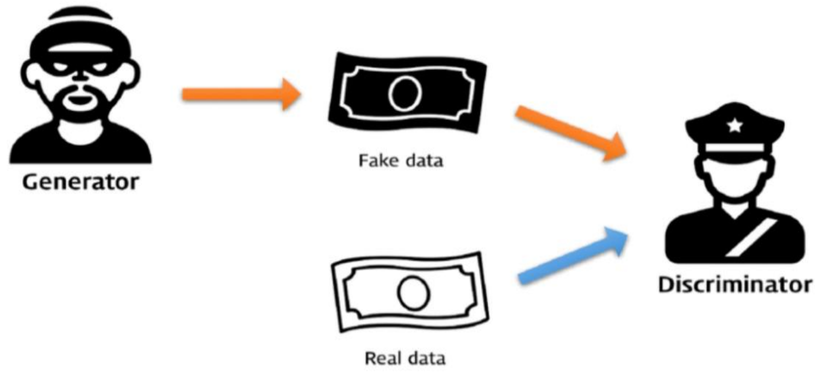
- Deep generative model은 maximum likelihood estimation 관련 전략에서 발생하는 많은 확률론적 계산을 근사화하는 것이 어렵고 generative context에서는, 앞서 모델 사용의 큰 성공을 이끌었던 선형 units의 이점을 활용하는 것이 어렵기 때문에 impact가 덜했음

- -> 이 논문에서 소개하는 generative model 은 이러한 어려움을 회피

1 Introduction

- generative model은 위조지폐를 제작하여 사용하려는 위조지폐범과 유사하다고 생각할 수 있는 반면, discriminative model은 위조지폐를 탐지하려는 경찰과 유사 -> 이러한 경쟁하는 과정의 반복은 어느 순간 위조지폐범이 진짜 같은 위조지폐를 만들 수 있고 경찰이 위조지폐를 구별할 수 있는 확률 역시 1/2가 됨

- -> **결국 GAN의 핵심 컨셉은 각각의 역할을 가진 두 모델을 통해 적대적 학습을 하면서 가짜와 진짜의 것을 구분 못할 때 까지 각 모델의 능력을 키워주는 것**



3 Adversarial nets

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

- 첫번째 항: (real data x)를 discriminator 에 넣었을 때 나오는 결과를 log취했을 때 얻는 기댓값
- 두번째 항: (z를 generator에 넣었을 때 나오는 결과를) discriminator에 넣었을 때 그 결과를 log(1-결과)했을 때 얻는 기댓값

3 Adversarial nets

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

- 학습 초반에 G가 생성해내는 이미지는 D가 G가 생성해낸 가짜 샘플인지 실제 데이터의 샘플인지 바로 구별할 수 있을만큼 학습이 잘 안되어서, D(G(z))의 결과가 0에 가까움

- 학습이 진행될수록, G는 실제 데이터의 분포를 모사하면서 D(G(z))의 값이 1이 되도록 발전

3 Adversarial nets

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

D의 입장

- V(D,G)의 이상적인 결과를 생각해보면, D가 매우 뛰어난 성능으로 판별을 잘 해낸다고 했을 때, D가 판별하려는 데이터가 실제 데이터에서 온 샘플일 경우에는 D(x)가 1이 되어 첫번째 항은 0이 되어 사라짐
- 두번째 항은 G(z)가 생성해낸 가짜 이미지를 구별해낼 수 있으므로 D(G(z))는 0이 되어 log(1-0)=log1=0이 되어 전체 식 V(D,G) = 0이 되어 D의 입장에서 얻을 수 있는 이상적인 결과인 최댓값은 0임

3 Adversarial nets

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

G의 입장

- V(D,G)의 이상적인 결과를 생각해보면, G가 D가 구별못할만큼 진짜와 같은 데이터를 잘 생성해낸다고 했을 때, 첫번째 항은 D가 구별해내는 것에 대한 항으로 G의 성능에 의해 결정될 수 있는 항이 아니므로 패스
- 두번째 항을 살펴보면 G가 생성해낸 데이터는 D를 속일 수 있는 성능이라고 한다면 D가 G가 생성해낸 이미지를 가짜라고 인식하지 못하고 진짜라고 결정해서 D(G(z)) =1이 되고 log(1-1)=log0=마이너스무한대가 되어 G의 입장에서 얻을 수 있는 이상적인 결과인 최솟값은 마이너스무한대임

3 Adversarial nets

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

-> **D입장에서는 V(D,G)를 최대화시키려고,**

G입장에서는 V(D,G)를 최소화시키려고 하고,

논문에서는 D와 G를 V(G,D)를 갖는 two-player minmax game 으로 표현

3 Adversarial nets

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

위의 수식으로 표현되는 minimax게임은

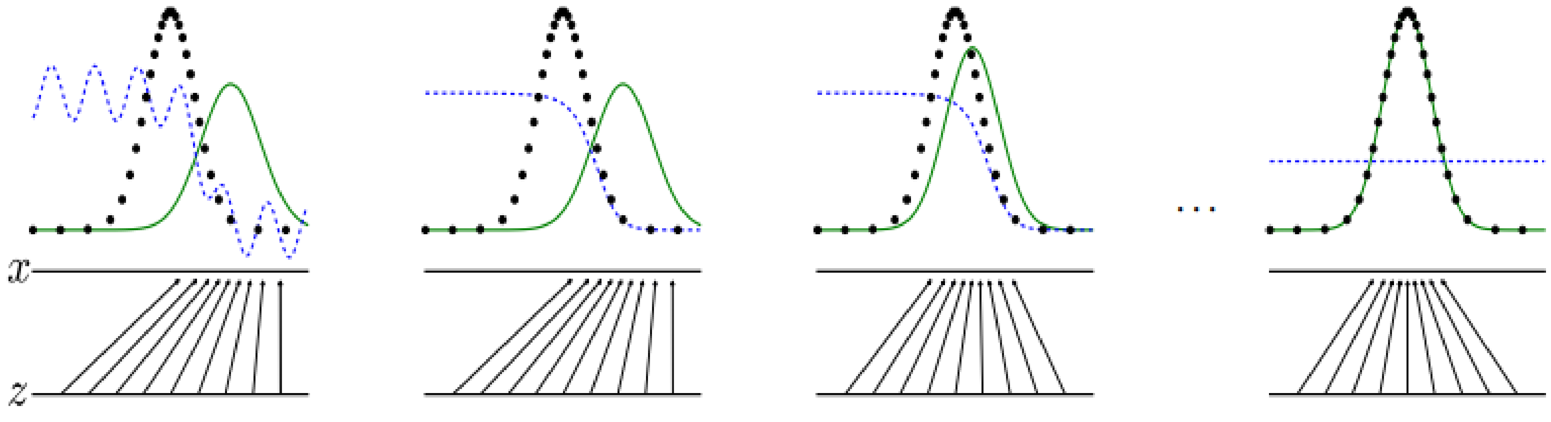
실용적인 측면에서 반드시 iterative하고 numerical한 접근으로 구현되어야 함

-> D를 먼저 k step 훈련시키고 G를 1 step 훈련시키는 방식으로 하나의 훈련 루프를 구축하는게 G를 천천히 변화시키는동안 D를 최적점에 가깝게 유지하는 결과를 줌

-> G가 학습 초기에는 log(1-D(G(z))의 gradient를 계산했을 때 너무 작은 값이 나오므로 학습이 느리기 때문에, log(1-D(G(z))를 최소화하려고 하는 것보 log(D(G(z))를 최대화되게끔 학습하는 것이 더 좋음

3 Adversarial nets

파란색 점선: discriminative distribution
검은색 점선: data generating distribution(real)
녹색 실선: generative distribution(fake)



(a) (b) (c) (d)

(a): 학습초기에는 real과 fake의 분포가 전혀 다르고, D의 성능도 좋지 않음
(b): D가 (a)처럼 들쑥날쑥하게 확률을 판단하지 않고, 흔들리지 않고 real과 fake를 분명하게 판별해내고 있음을 확인할 수 있다. 이는 D가 성능이 올라갔음을 확인 가능 (k step으로 먼저 진행)
(c): 어느정도 D가 학습이 이루어지면, G는 실제 데이터의 분포를 모사하며 D가 구별하기 힘든 방향으로 학습을 함
(d): 이 과정의 반복의 결과로 real과 fake의 분포가 거의 비슷해져 구분할 수 없을 만큼 G가 학습을 하게되고 D가 이 둘을 구분할 수 없게 되어 확률을 1/2로 계산

4 Theoretical Results

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

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:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

  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 descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for
The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

4.1 Global Optimality of Pg=Pdata

- Proposition 1) G가 고정된 경우, 최적의 discriminator D

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

증명)

$$\begin{aligned} V(G, D) &= \int_{\boldsymbol{x}} p_{\text{data}}(\boldsymbol{x}) \log(D(\boldsymbol{x}))dx + \int_{z} p_{\boldsymbol{z}}(\boldsymbol{z}) \log(1 - D(g(\boldsymbol{z})))dz \\ &= \int_{\boldsymbol{x}} p_{\text{data}}(\boldsymbol{x}) \log(D(\boldsymbol{x})) + p_g(\boldsymbol{x}) \log(1 - D(\boldsymbol{x}))dx \end{aligned}$$

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

4.1 Global Optimality of Pg=Pdata

- Proposition 1) G가 고정된 경우, 최적의 discriminator D

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

증명)

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

$$\min_G \max_D V(D, G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

$$\begin{aligned} C(G) &= \max_D V(G, D) \\ &= \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}}[\log D_G^*(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}}[\log(1 - D_G^*(G(\boldsymbol{z})))] \\ &= \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}}[\log D_G^*(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x} \sim p_g}[\log(1 - D_G^*(\boldsymbol{x}))] \\ &= \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}}\left[\log \frac{p_{\text{data}}(\boldsymbol{x})}{P_{\text{data}}(\boldsymbol{x}) + p_g(\boldsymbol{x})}\right] + \mathbb{E}_{\boldsymbol{x} \sim p_g}\left[\log \frac{p_g(\boldsymbol{x})}{p_{\text{data}}(\boldsymbol{x}) + p_g(\boldsymbol{x})}\right] \end{aligned}$$

4.1 Global Optimality of Pg=Pdata

- Theorem 1) The global minimum of the virtual training criterion C(G) is achieved if and only if Pg = Pdata.

At that point, C(G) achieves the value − log 4.

증명) $\mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}}[-\log 2] + \mathbb{E}_{\boldsymbol{x} \sim p_g}[-\log 2] = -\log 4$

$C(G) = \quad C(G) \quad$ +log4-log4

$C(G) =$ -log4+ $\mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}}\left[\log \frac{p_{\text{data}}(\boldsymbol{x})}{P_{\text{data}}(\boldsymbol{x}) + p_g(\boldsymbol{x})}\right] + \mathbb{E}_{\boldsymbol{x} \sim p_g}\left[\log \frac{p_g(\boldsymbol{x})}{p_{\text{data}}(\boldsymbol{x}) + p_g(\boldsymbol{x})}\right]$ +log2+log2

$$C(G) = -\log(4) + KL\left(p_{\text{data}} \middle\| \frac{p_{\text{data}} + p_g}{2}\right) + KL\left(p_g \middle\| \frac{p_{\text{data}} + p_g}{2}\right)$$

$$C(G) = -\log(4) + 2 \cdot JSD\left(p_{\text{data}} \| p_g\right)$$

*KL=kullback-Leibler divergence
KL(P||Q)=
-> P라는 분포 있을때 Q와 P가 얼마나 다른지 측정하는 값

*JSD=jenson-Shannon divergence
JSD(P||Q)=1/2KL(P||M)+1/2KL(Q||M)
-> 두 분포가 완전히 일치할 때만 0

따라서, C(G)의 global minimum은 –log4 이고 그 유일한 해는 Pg=Pdata

4.2 Convergence of Algorithm 1

- 이 알고리즘이 문제를 얼마나 잘 풀어주는지 증명

• Proposition 2) G,D가 충분한 용량을 갖고있고, Algorithm1 단계에서 각 step 에서 discriminator 가 주어진 G에 대해 optimum 에 도달하도록 허용하고, Pg가 업데이트 되어 기준을 개선한다면

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

then p_g *converges to* p_{data}

4.2 Convergence of Algorithm 1

증명) $V(G, D) = U(p_g, D)$

$U(p_g, D)$ is convex in p_g

if f(x) = $\sup_{\alpha \in \mathcal{A}} f_\alpha(x)$ and $f_\alpha(x)$ is convex in x for every α, then $\partial f_\beta(x) \in \partial f$ if $\beta = \arg\sup_{\alpha \in \mathcal{A}} f_\alpha(x)$.

= 대응하는 G가 있을 때 최적의 D에서의 Pg를 만드는데 사용될 gradient의 계산이 U(Pg, D)에 들어있다

-> Pg에 대한 적고 충분한 update만으로도 Pg->Pdata 수렴해서 thm1과 같은 맥락

5 Experiments

- MNIST, Toronto Face Database(TFD), CIFAR-10에 대해 학습
- G는 rectifier linear activations, sigmoid 혼합하여 사용, D는 maxout activation사용

- D를 학습시킬 때 Dropout사용

- 이론적인 프레임워크에서는 generator의 중간층에 dropout과 noise를 허용하지 않지만, 실험에서는 generator net 맨 하위계층에 input으로 noise 사용함

-> G가 생성해낸 sample이 기존의 존재하는 방법으로 생성된 sample보다 좋다고 주장할 수 는 없지만, 더 나은 생성모델과 경쟁할 수 있다 생각하며, adversarial framework의 잠재력 강조

6 Advantages and disadvantages

- 단점

Pg(x)가 명시적으로 존재하지 않음

D와 G가 균형을 잘 맞춰 성능이 향상되어야 함

- 장점

Markov chains이 전혀 필요 없고 gradients를 얻기 위해 back-propagation만이 사용됨

학습 중 어떠한 inference가 필요 없음

다양한 함수들이 모델이 접목될 수 있음

generator network가 데이터로부터 직접적으로 업데이트 되지 않고 오직 discriminator로부터 흘러들어오는 gradient만을 이용해 학습됨(이는 input의 요소들이 직접적으로 생성기의 파라미터에 복사되지 않는다는 걸 의미)

Markov chains을 쓸 때보다 훨씬 선명한 이미지를 얻을 수 있음

7 Conclusions and future work

- conditional generative model로 발전시킬 수 있음 (CGAN)
- x가 주어졌을때 z를 예측하는 보조 네트워크를 학습시켜 Learned approximate inference 할 수 있음
- parameters를 공유하는 conditionals model를 학습함으로써 다른 conditionals models을 근사적으로 모델링할 수 있음
- Semi-supervised learning: discriminator로 얻어지는 feature를 제한된 레이블이 있는 데이터의 classifier성능 향상시킬 수 있음
- 효율성 개선: G,D를 조정하는 더 나은 방법이나 학습하는 동안 sample z에 대한 더 나은 분포를 결정함으로써 학습을 가속화 할 수 있음