Generative Adversarial Nets

GAN

김영민 김지수 이다인

INDEX

1. 배경

기존의 생성모델

Markov chain or unrolled approximate inference network

별도의 테크닉 필요

기존의 생성모델 한계

- 확률론적인 모델들을 근사하는데 어려움
- ReLU와 같은 함수의 장점을 제대로 살리기 어려움

How do we get over it?

New Framework!!

GAN

2. GAN이란?

GAN **생성자(Generator)와 판별자(Discriminator)의 MinMax 게임**

오직 Neural Network만 사용

vs

위조지폐 판별 감별사

Discriminator

위조지폐 위조범

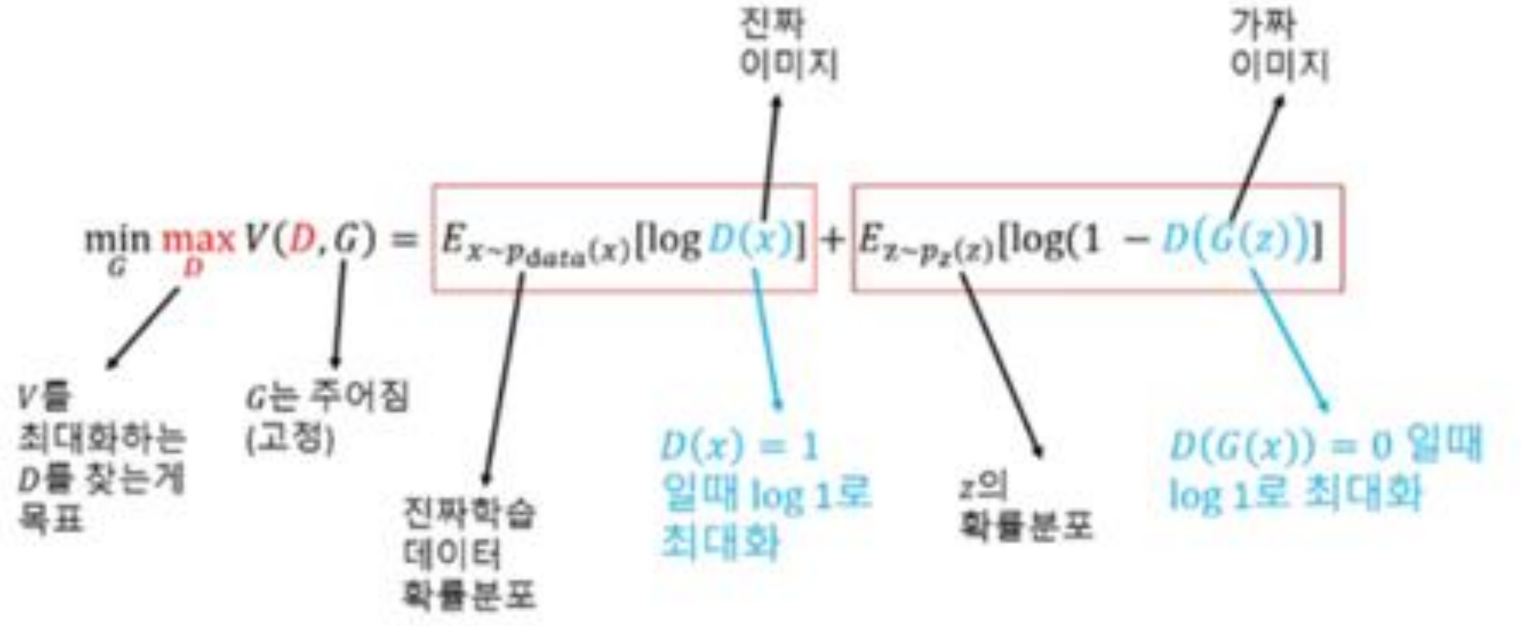
Generator

3. Adversarial Nets

목적함수

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

$G = Generator, D = Descriminator$



V Maximize 조건

D(x) = 1

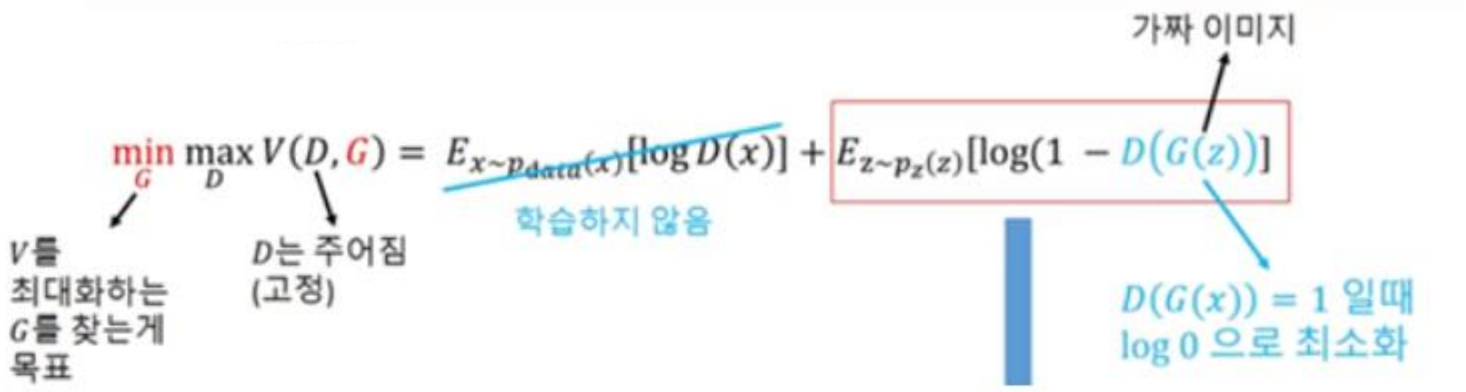
Fake=0, Real = 1

3. Adversarial Nets

목적함수

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

$G = Generator, D = Descriminator$



V Minimize 조건

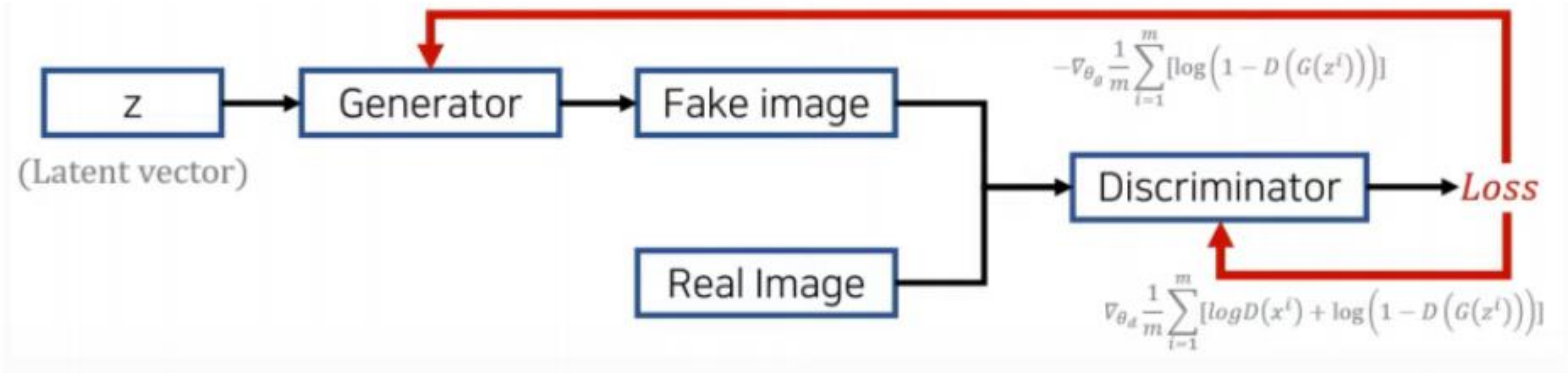
D(G(z))=1

3. Adversarial Nets

목적함수

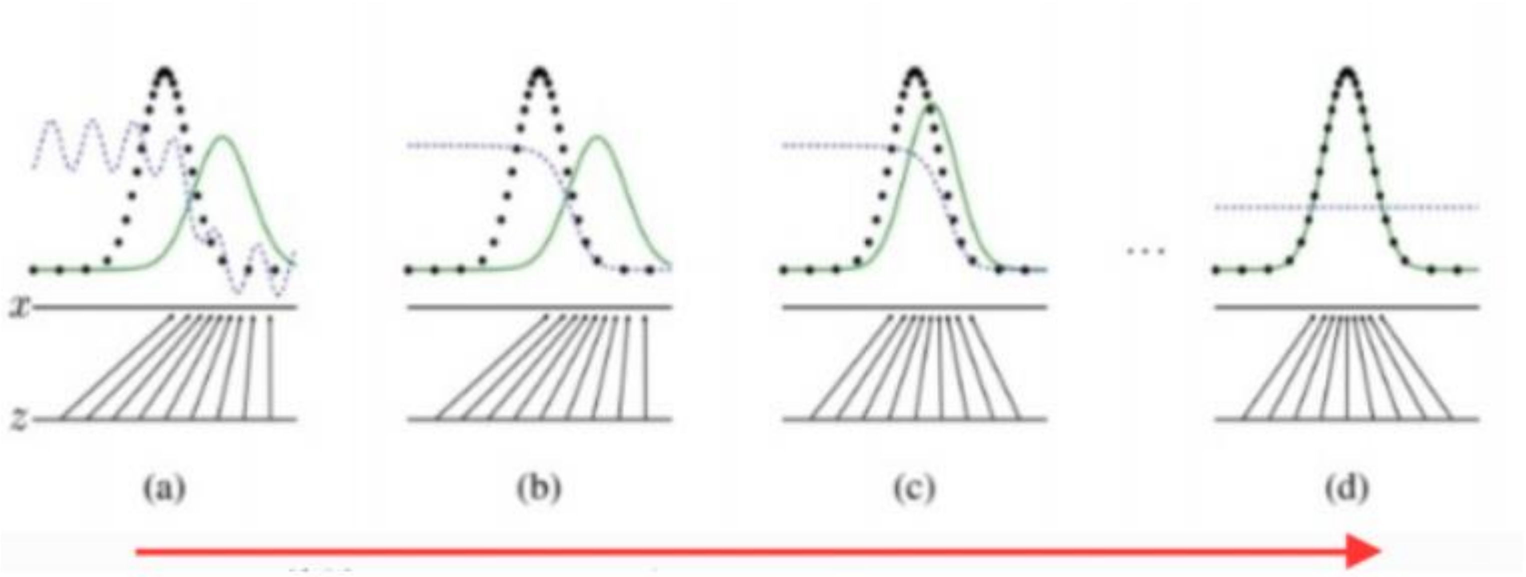
$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

$G = Generator, D = Descriminator$



3. Adversarial Nets

GAN 수렴 과정



Generator Distribution

Real image Distribution

Discriminator Distribution

4. Theoretical Results

Global Optimality $p_g = p_{data}$

Proposition 1 $D_G^*(\boldsymbol{x}) = \frac{p_{data}(\boldsymbol{x})}{p_{data}(\boldsymbol{x}) + p_g(\boldsymbol{x})}$ $G\ fixed$

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

$$E[X] = \int_{-\infty}^{\infty} x f(x) dx$$

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

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

$$function\ y = a\log(y) + b\ log(1-y) \qquad maximum = \frac{a}{a+b} \qquad in\ [0,1]$$

4. Theoretical Results

Theorem 1

$$p_g = p_{data}$$

KLD : $KL(p_{data}||p_g) = \int_{-\infty}^{\infty} p_{data}(x) \log\left(\frac{p_{data}(x)}{p_g(x)}\right) dx$

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

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

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

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

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

$$= E_{x\sim p_{data}(x)}\left[log \frac{2 * \mathrm{p_{data}}(\mathrm{x})}{\mathrm{p_{data}}(\mathrm{x}) + \mathrm{p_g}(\mathrm{x})}\right] + E_{x\sim p_g(x)}\left[log \frac{2 * \mathrm{p_g}(\mathrm{x})}{\mathrm{p_{data}}(\mathrm{x}) + \mathrm{p_g}(\mathrm{x})}\right] - \log(4)$$

KLD

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

$$= 2 * JSD(p_{data}||p_g) - \log(4) \leftarrow JSD(p||q) = \frac{1}{2}KL(p||\frac{p+q}{2}) + \frac{1}{2}KL(q||\frac{p+q}{2})$$

4. Theoretical Results

$if \quad p_g = p_{data}$

$$E_{x\sim p_{data}(x)}\left[\underbrace{\cancel{log \frac{2 * p_{data}(x)}{p_{data}(x) + p_g(x)}}}_{=\,0}\right] + E_{x\sim p_g(x)}\left[\underbrace{\cancel{log \frac{2 * p_g(x)}{p_{data}(x) + p_g(x)}}}_{=\,0}\right] - \log(4)$$

$$= \cancel{2 * JSD(p_{data} || p_g)} - \log(4) = -\log(4)$$

$\therefore$ **Global Mimize point :** $p_g = p_{data}$ **Value :** $-\log(4)$

4. Theoretical Results

Proposition 2

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

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

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

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

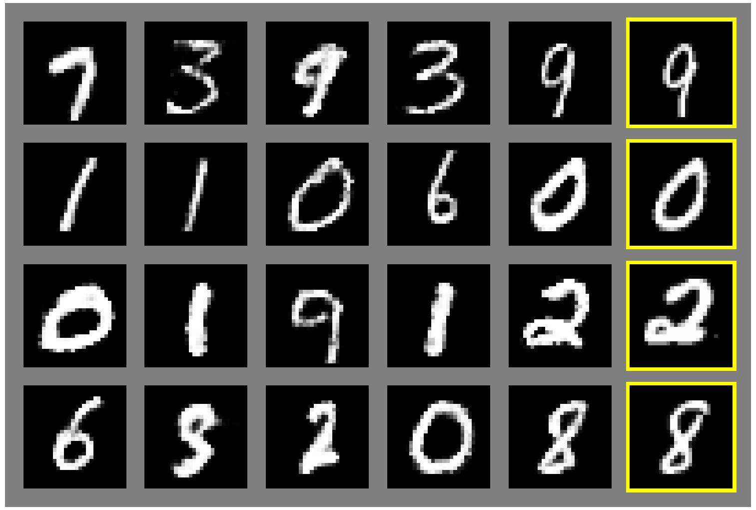
$\Rightarrow$ $V(G,D)\ is\ convex\ in\ p_g$

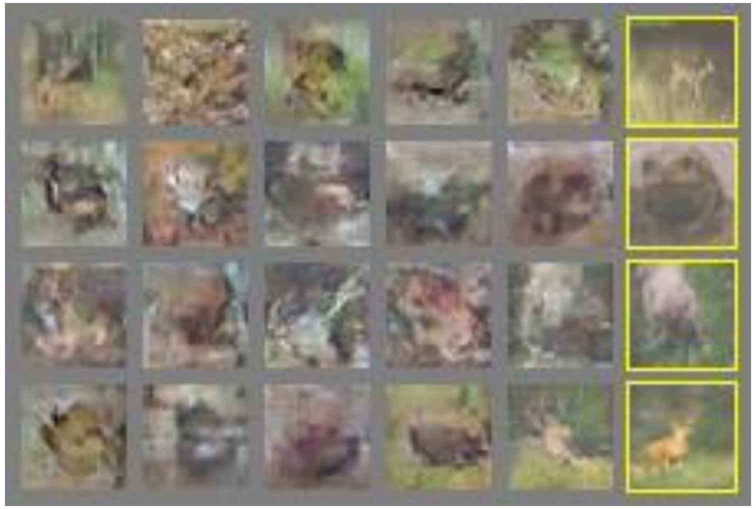
5. Experiments

- 데이터 셋: MINIST, Toronto Face Database, CIFAR-10
- Generator nets: rectifier linear activation + sigmoid activation
- Discriminator net: maxout activation
- Dropout: Discriminator net에 적용하여 훈련

Model	MNIST	TFD
DBN [3]	138 ± 2	1909 ± 66
Stacked CAE [3]	121 ± 1.6	$\mathbf{2110 \pm 50}$
Deep GSN [6]	214 ± 1.1	1890 ± 29
Adversarial nets	$\mathbf{225 \pm 2}$	$\mathbf{2057 \pm 26}$

5. Experiments





6. Advantages and Disadvantages

단점

- $P_g(x)$ 의 명시적인 표현이 없음
- D가 훈련 중 G와 동기화가 잘 되어야 함

장점

- Markov chain이 필요 없음
- Gradients를 얻기 위해서 back-propagation만 사용
- 학습 중에 추론이 필요 없음
- 다양한 기능을 모델에 통합 가능

7. Conclusions and future work

1. Conditional generative model $\mathrm{P}(x|c)$는 c를 G와 D에 모두 입력함으로써 얻어질 수 있다.
2. 학습된 근사 추론은 x가 주어진 z를 예측하도록 보조 네트워크를 학습하여 수행할 수 있다.
3. 모든 조건부 $\mathrm{P}(x_s|x_{\not s})$를 근사 모델링 할 수 있다.
4. 반-감독 학습
5. 효율성 향상

감사합니다 ^^