



**Generative Adversarial Nets**

**GAN**

# INDEX

1. 배경
2. GAN이란?
3. Adversarial Nets
4. Theoretical Results
5. Experiments
6. Advantages and disadvantages
7. Conclusions and future work

# 1. 배경

## 기존의 생성모델

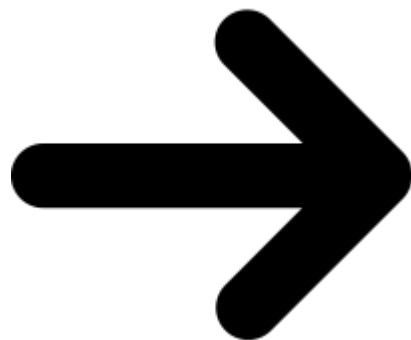
Markov chain or unrolled approximate inference network

**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** **NEW!**

## 2. GAN이란?

GAN

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

오직 Neural Network만 사용



위조지폐 판별 감별사

**Discriminator**

VS



위조지폐 위조범

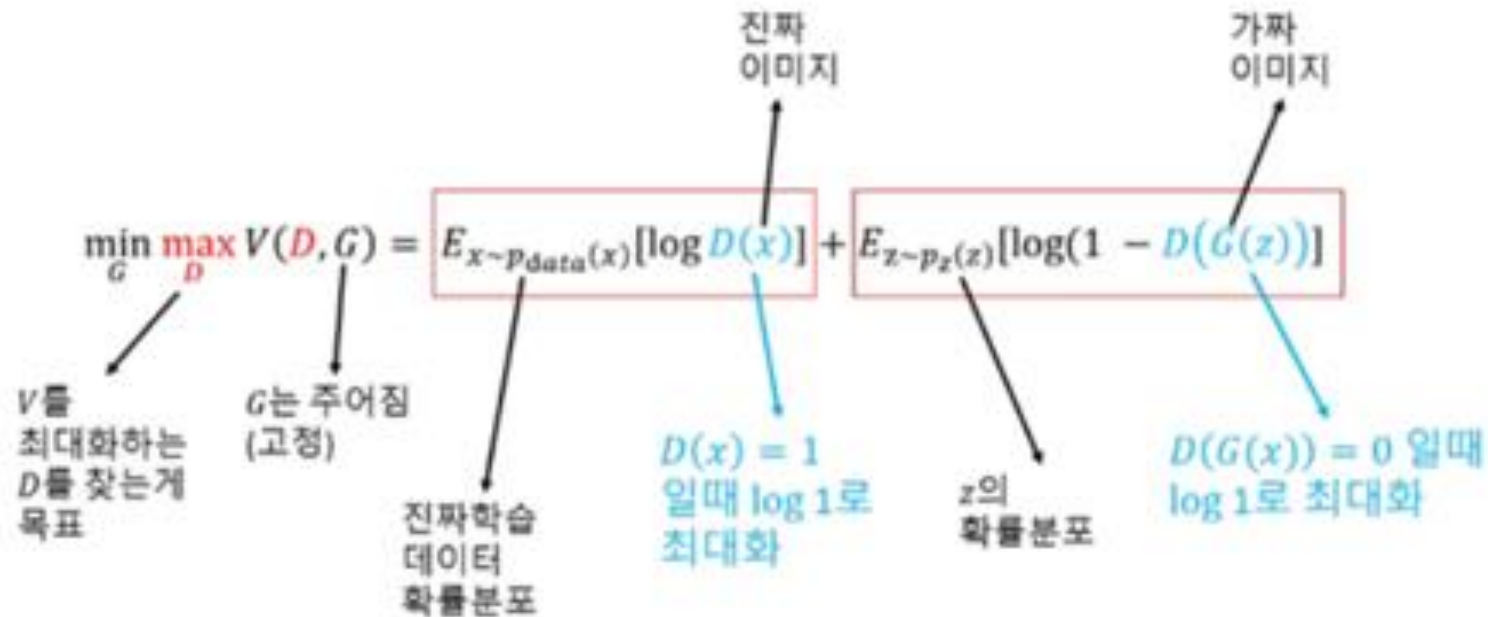
**Generator**

# 3. Adversarial Nets

## 목적함수

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

$G = \text{Generator}, D = \text{Discriminator}$



**V Maximize 조건**

$$D(\mathbf{x}) = 1$$

**Fake=0, Real = 1**

# 3. Adversarial Nets

## 목적함수

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

$G = \text{Generator}, D = \text{Discriminator}$

**V Minimize 조건**

$$D(G(\mathbf{z})) = 1$$

The diagram illustrates the minimization condition for the generator's loss. It shows the equation  $\min_G \max_D V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_z(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))]$ . Annotations include: 

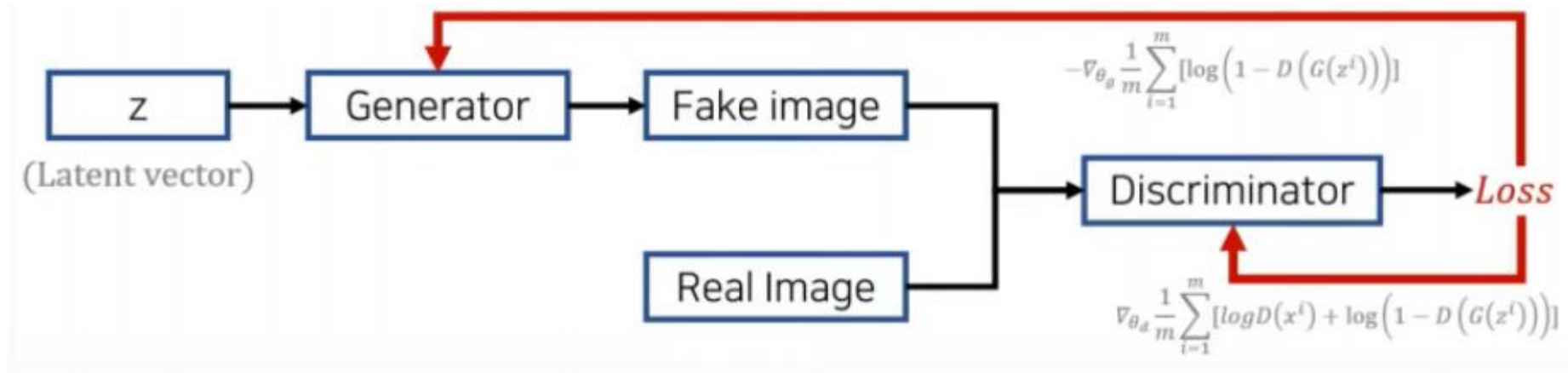
- A red arrow pointing to  $\min_G$  with the text "V를 최대화하는 G를 찾는게 목표" (Goal is to find G that maximizes V).
- A black arrow pointing to  $\max_D$  with the text "D는 주어짐 (고정)" (D is given, fixed).
- A blue line under the first term  $\mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})]$  with the text "학습하지 않음" (Do not learn).
- A red box around the second term  $\mathbb{E}_{\mathbf{z} \sim p_z(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))]$ .
- A black arrow pointing to  $D(G(\mathbf{z}))$  with the text "가짜 이미지" (Fake image).
- A blue arrow pointing from the boxed term to the text "D(G(x)) = 1 일때 log 0 으로 최소화" (Minimize to log 0 when D(G(x)) = 1).

# 3. Adversarial Nets

## 목적함수

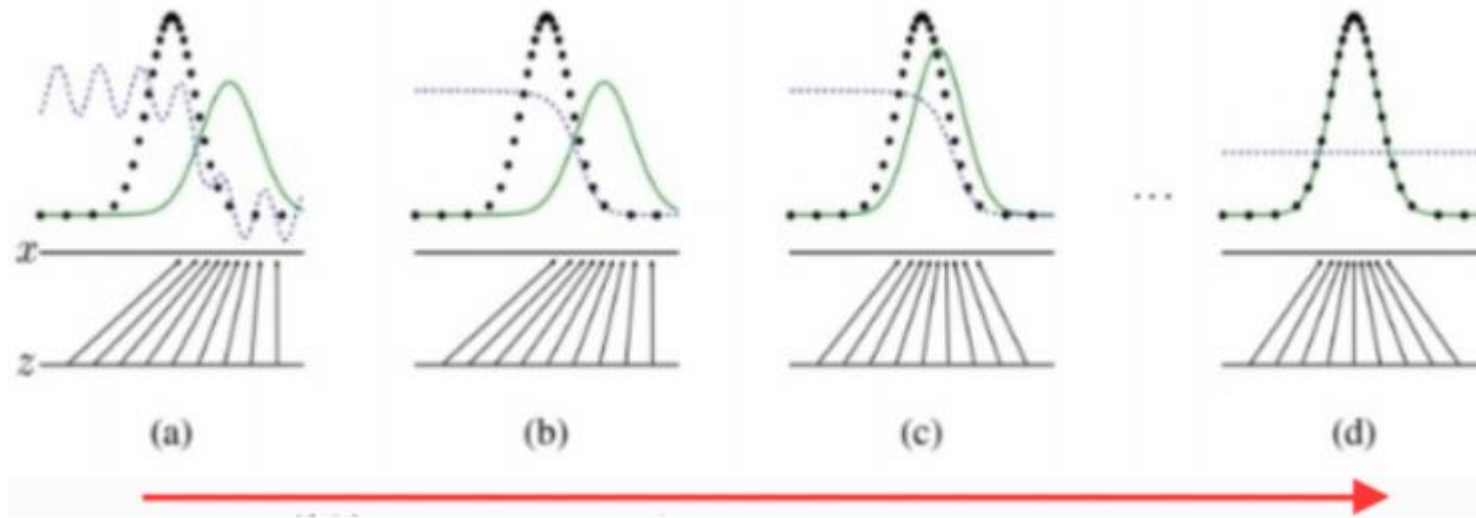
$$\min_G \max_D V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))].$$

$G = \text{Generator}, D = \text{Discriminator}$



# 3. Adversarial Nets

## GAN 수렴 과정



**Generator Distribution**      **Real image Distribution**      **Discriminator Distribution**




## 4. Theoretical Results

Global Optimality

$$p_g = p_{data}$$

Proposition 1

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

$$\begin{aligned} V(G, D) &= E_{x \sim p_{data}(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))] \\ &= \int_x p_{data}(x) \log(D(x)) dx + \int_z p_z(z) \log(1 - D(g(z))) dz \\ &= \int_x p_{data}(x) \log(D(x)) + p_g(x) \log(1 - D(x)) dx \end{aligned}$$

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

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

# 4. Theoretical Results

Theorem 1

$$p_g = p_{data}$$

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

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

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

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

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

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

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

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

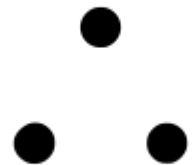
## 4. Theoretical Results

if  $p_g = p_{data}$

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

$\parallel \qquad \parallel$   
 $0 \qquad \qquad 0$

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



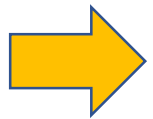
**Global Mimimize point :  $p_g = p_{data}$     Value :  $-\log(4)$**

## 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  $p_{data}$

$$\begin{aligned} V(G, D) &= E_{x \sim p_{data}(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))] \\ &= \int_x p_{data}(x) \log(D(x)) dx + \int_z p_z(z) \log(1 - D(g(z))) dz \\ &= \int_x \underbrace{p_{data}(x) \log(D(x))}_{\text{blue}} + \underbrace{p_g(x) \log(1 - D(x))}_{\text{orange}} dx \end{aligned}$$



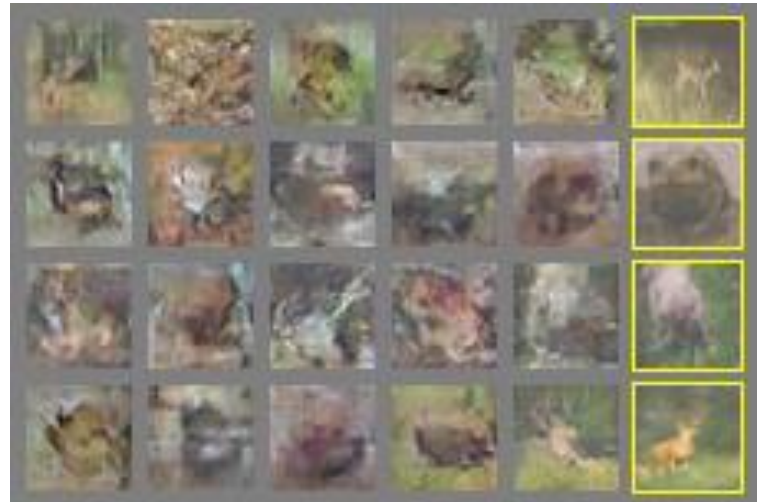
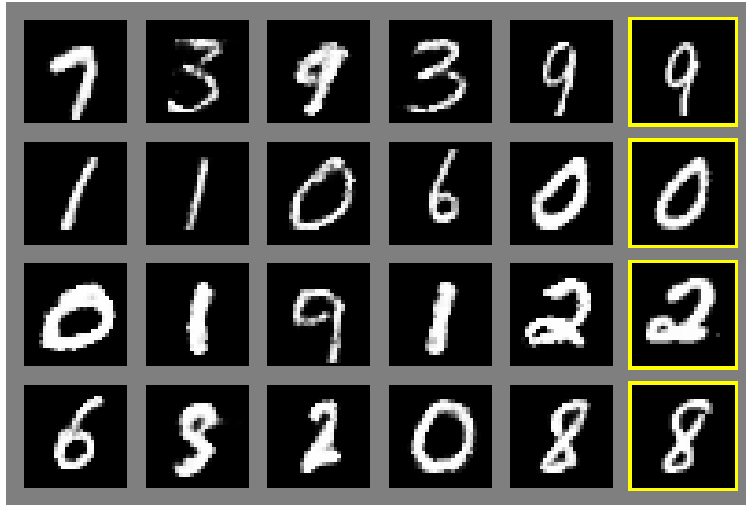
$V(G, D)$  is convex in  $p_g$

## 5. Experiments

- 데이터 셋: MNIST, 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]	$138 \pm 2$	$1909 \pm 66$
Stacked CAE [3]	$121 \pm 1.6$	<b><math>2110 \pm 50</math></b>
Deep GSN [6]	$214 \pm 1.1$	$1890 \pm 29$
Adversarial nets	<b><math>225 \pm 2</math></b>	<b><math>2057 \pm 26</math></b>

# 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  $P(x|c)$ 는  $c$ 를  $G$ 와  $D$ 에 모두 입력함으로써 얻어질 수 있다.
2. 학습된 근사 추론은  $x$ 가 주어진  $z$ 를 예측하도록 보조 네트워크를 학습하여 수행할 수 있다.
3. 모든 조건부  $P(x_s|x_{\setminus s})$ 를 근사 모델링 할 수 있다.
4. 반-감독 학습
5. 효율성 향상



감사합니다 ^^