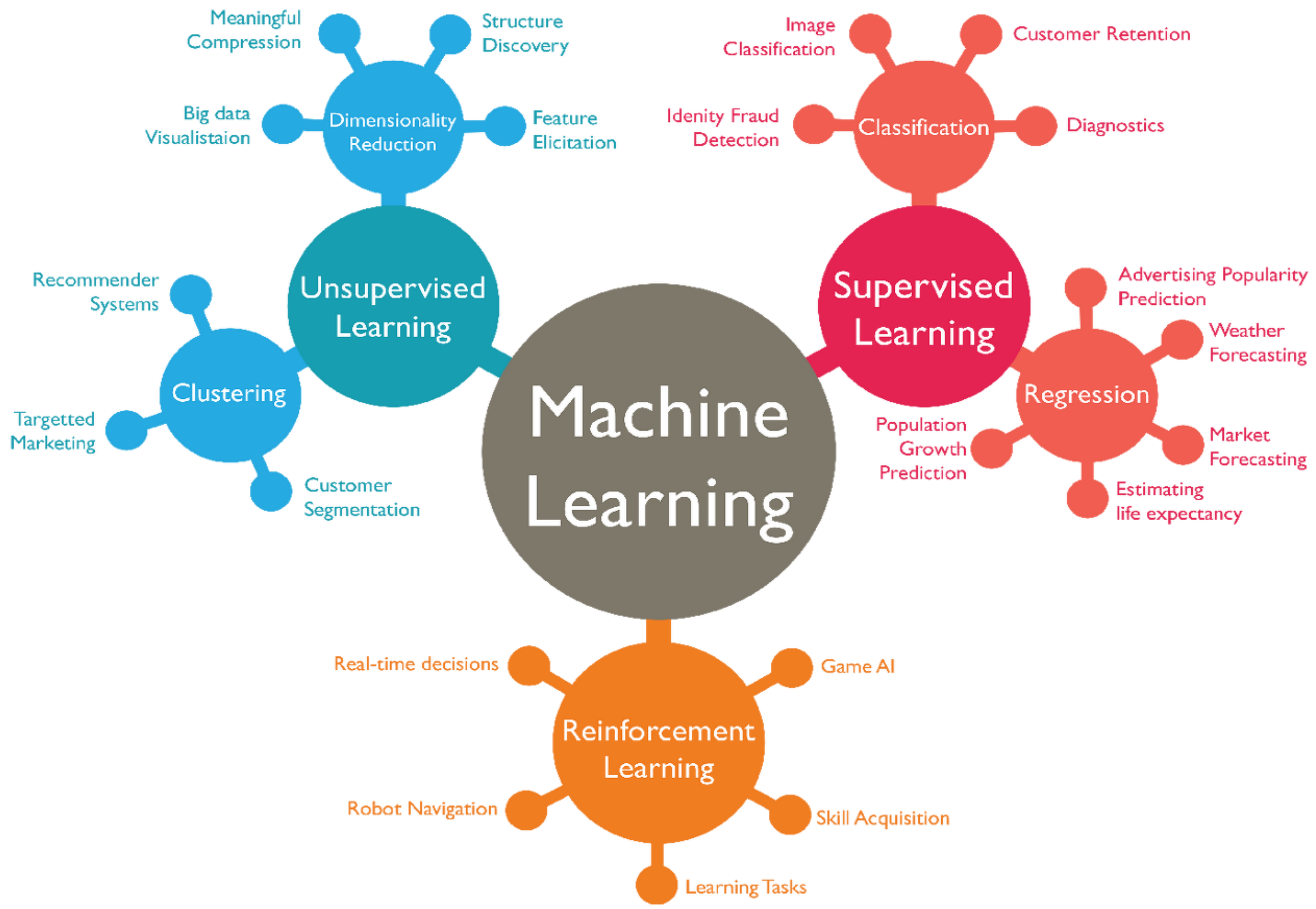
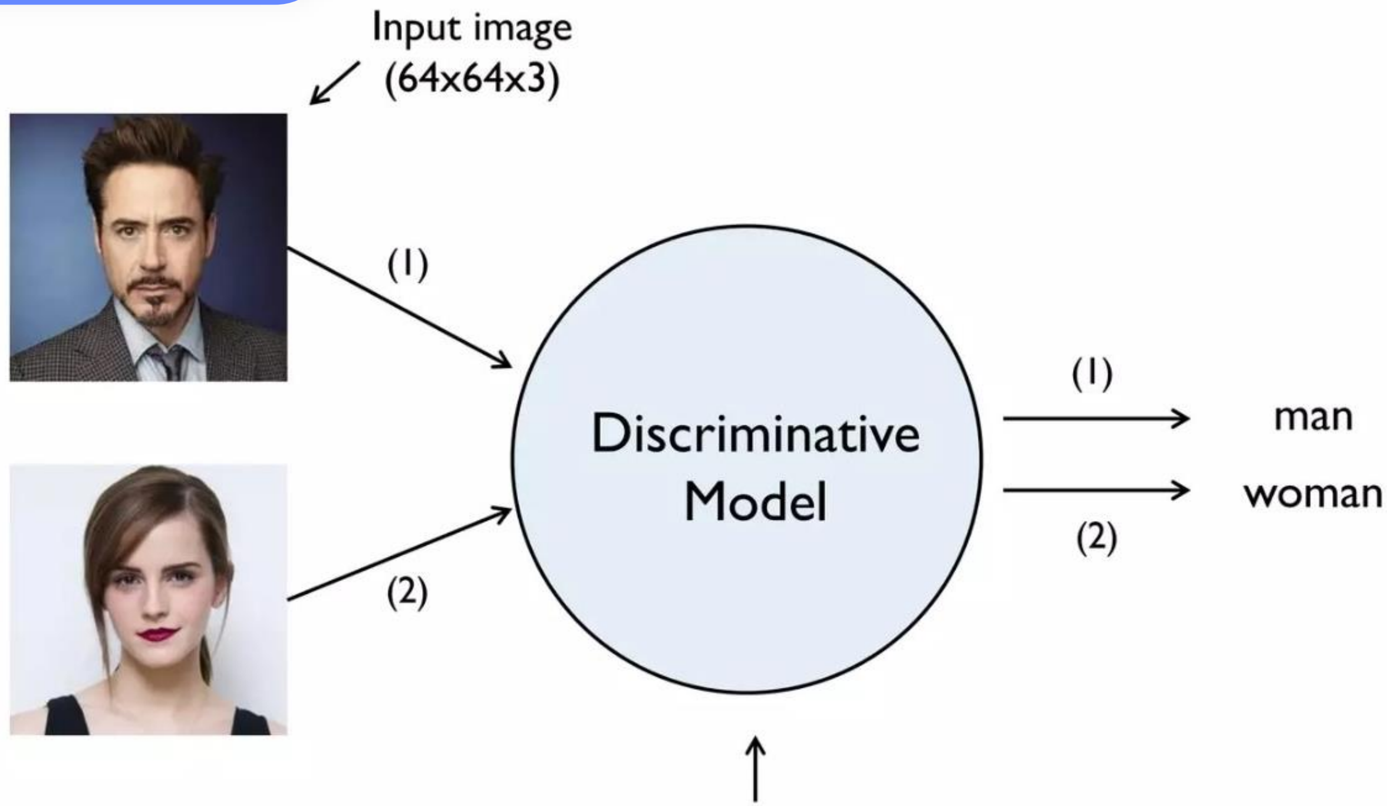
GAN

황주훈

Introduction



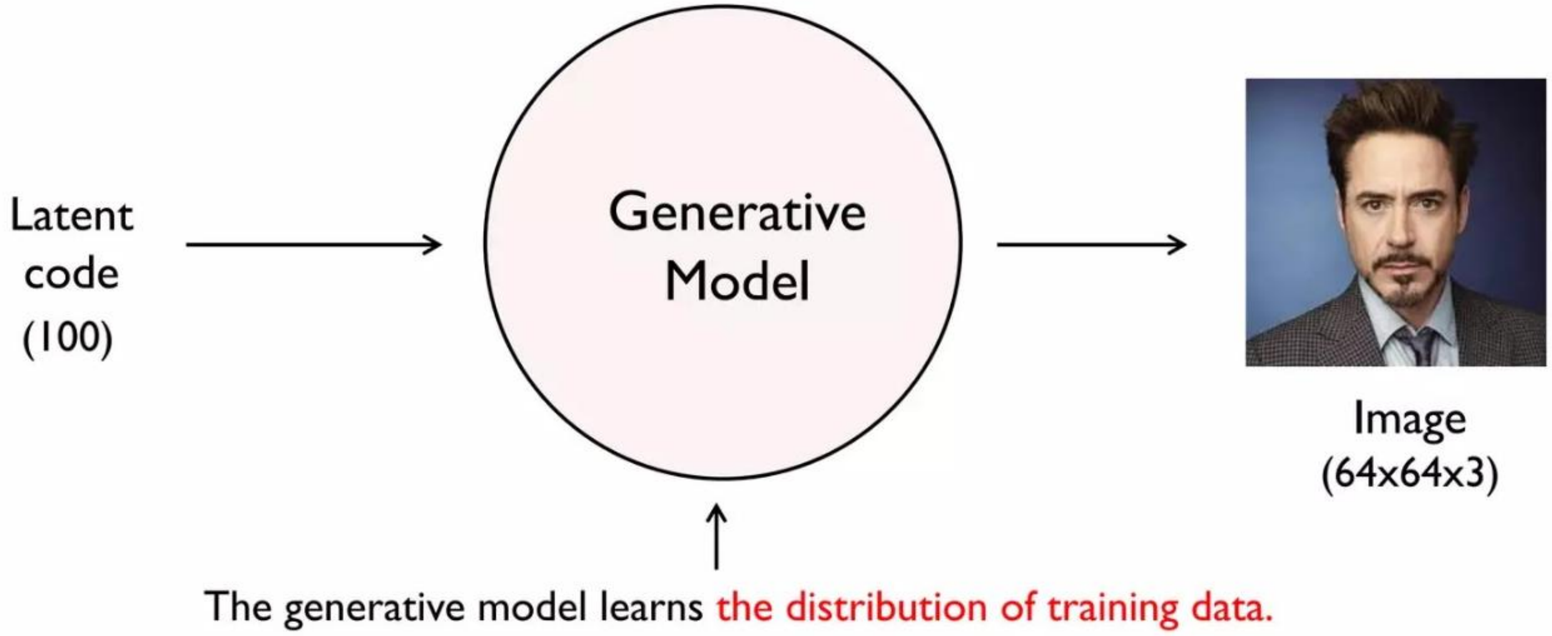
Supervised Learning



The discriminative model learns how to classify input to its class.

Introduction

Unsupervised Learning



확률분포

- **확률분포**는 확률 변수가 특정한 값을 가질 확률을 나타내는 함수
- 예를 들어 주사위를 던졌을때 나올 수 있는 수를 확률변수 X라고 합시다.
  - 확률변수 X는 1, 2, 3, 4, 5, 6
  - P(X = 1) = 1/6

$X=x_i$	1	2	3	4	5	6	합
$P(X=x_i)$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	1

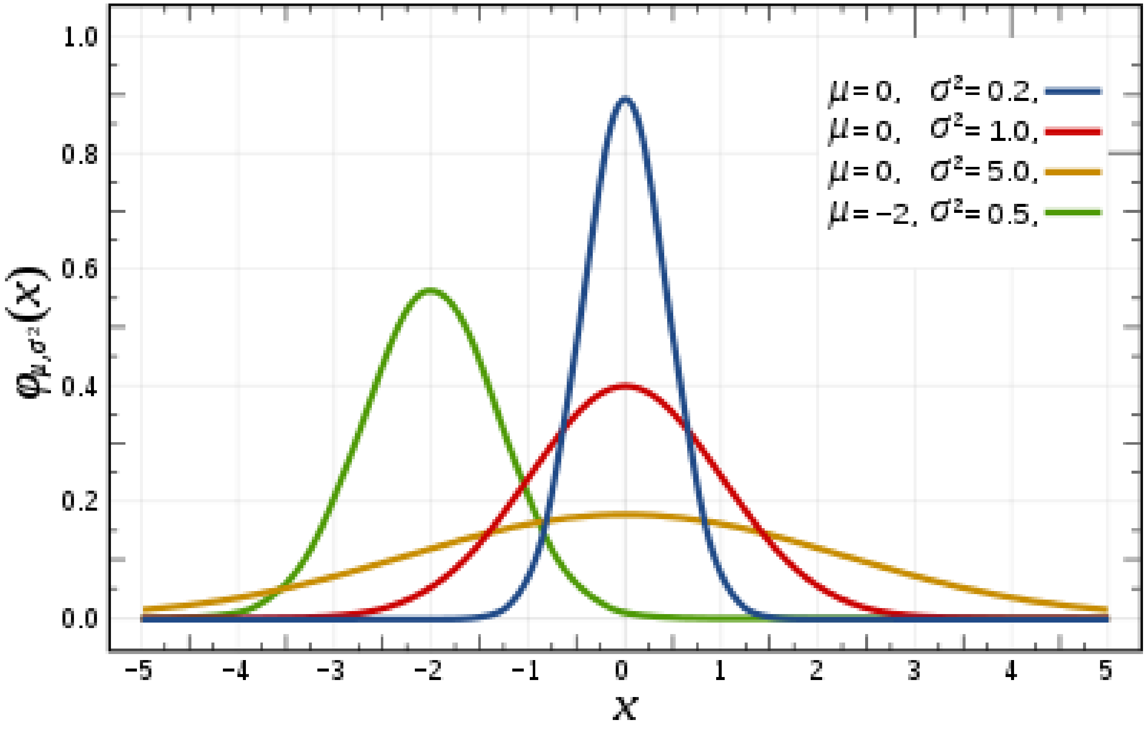
이산확률분포

- 확률변수 X의 는 개수는 정확히 셀 수 있을 때 **이산확률분포**라한다.

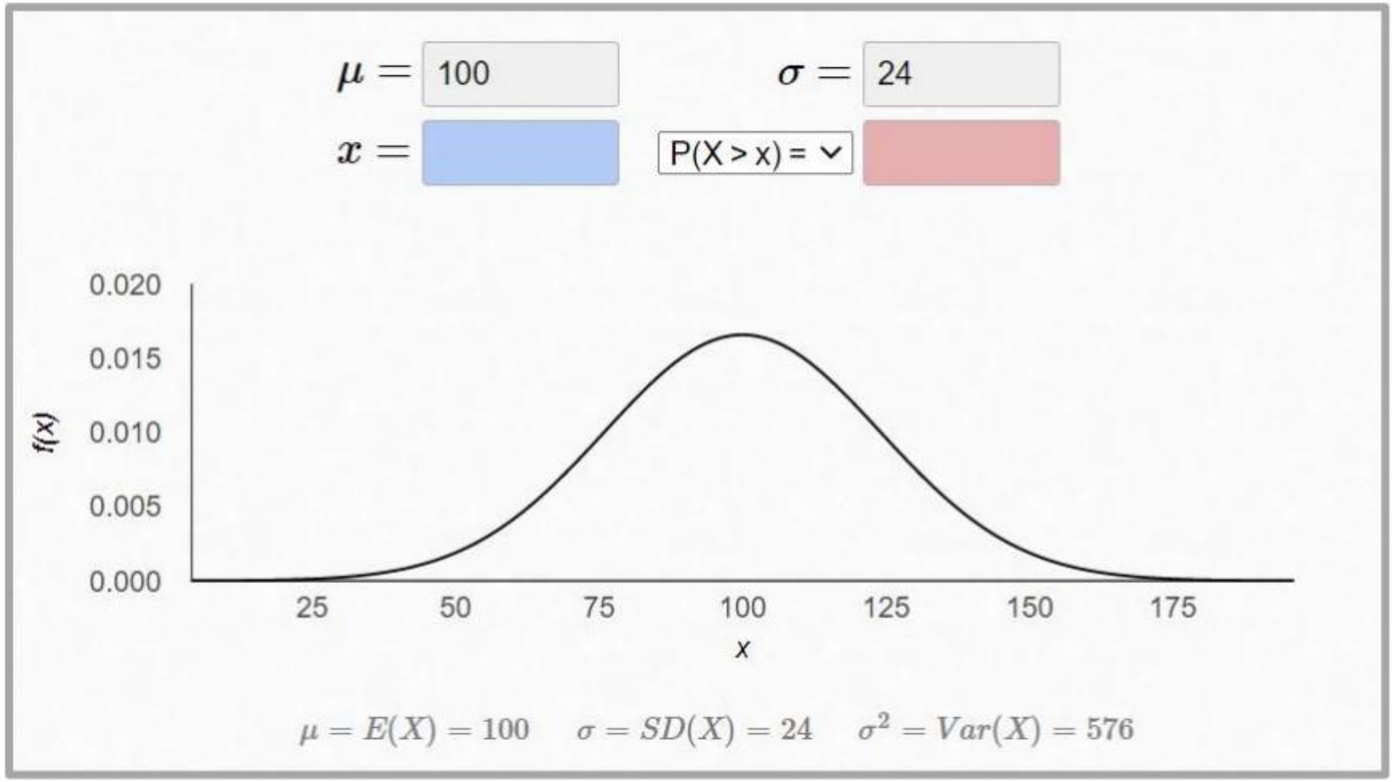
.

연속확률분포

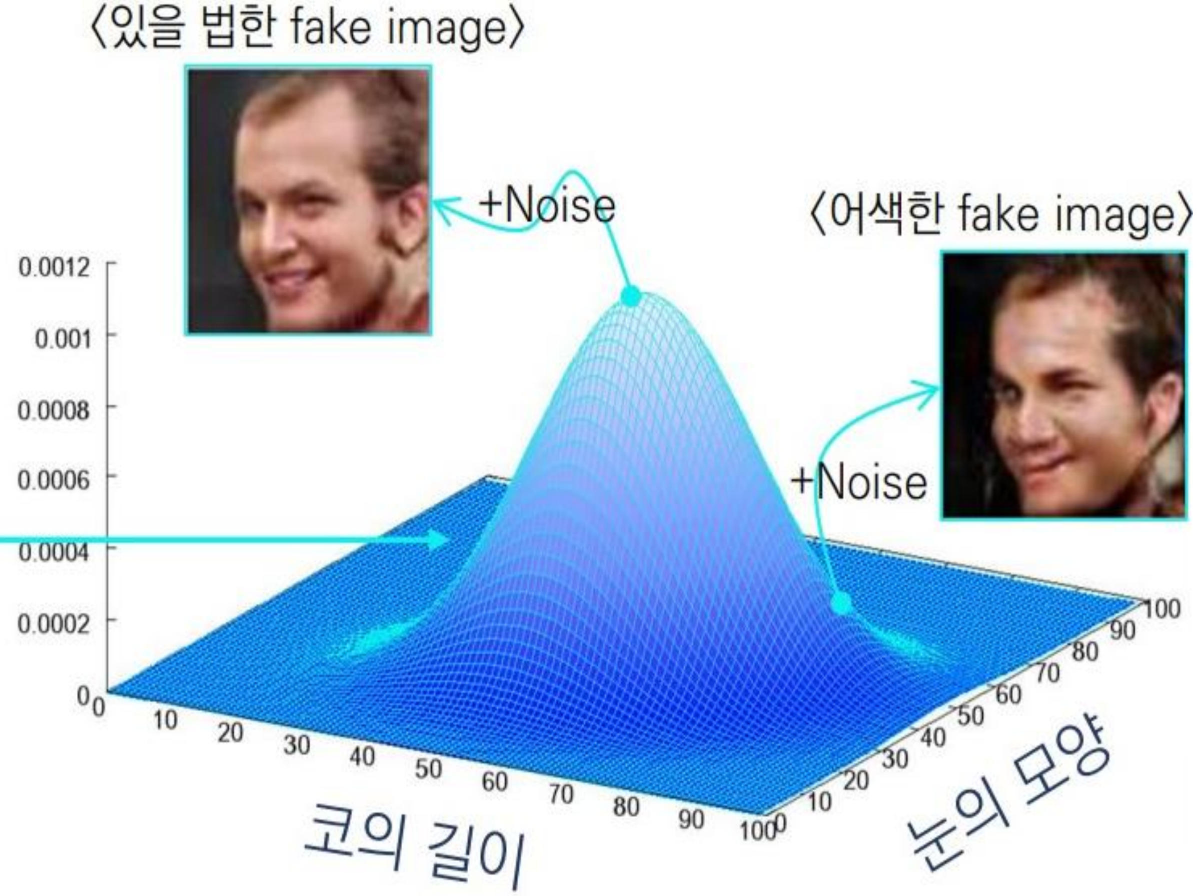
- 확률변수 X의 는 개수는 정확히 셀 수 없을 때 **연속확률분포**라한다
- 연속적인 값의 예시: 키, 달리기, 성적
- 정규분포 예시



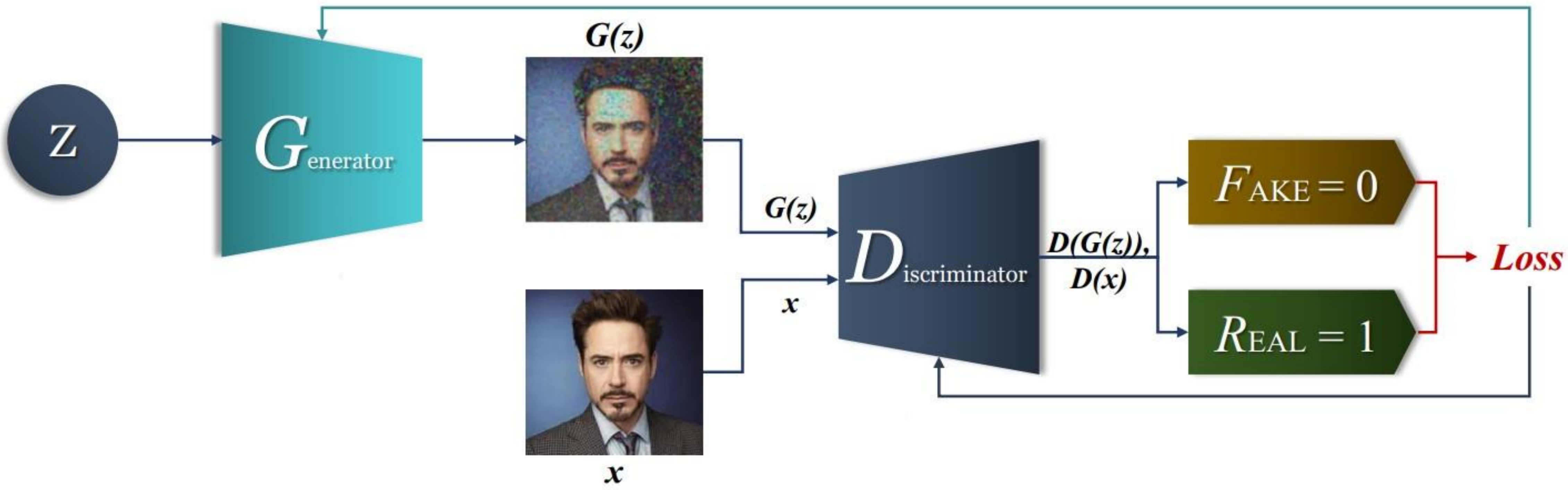
- 실제 많은 데이터들이 정규분포로 표현 가능하다.
- IQ에 대한 정규분포 예시 (표준편차 =24)



- **다변수 확률분포**
- 사람 얼굴도 정규분포 표현이 가능하다

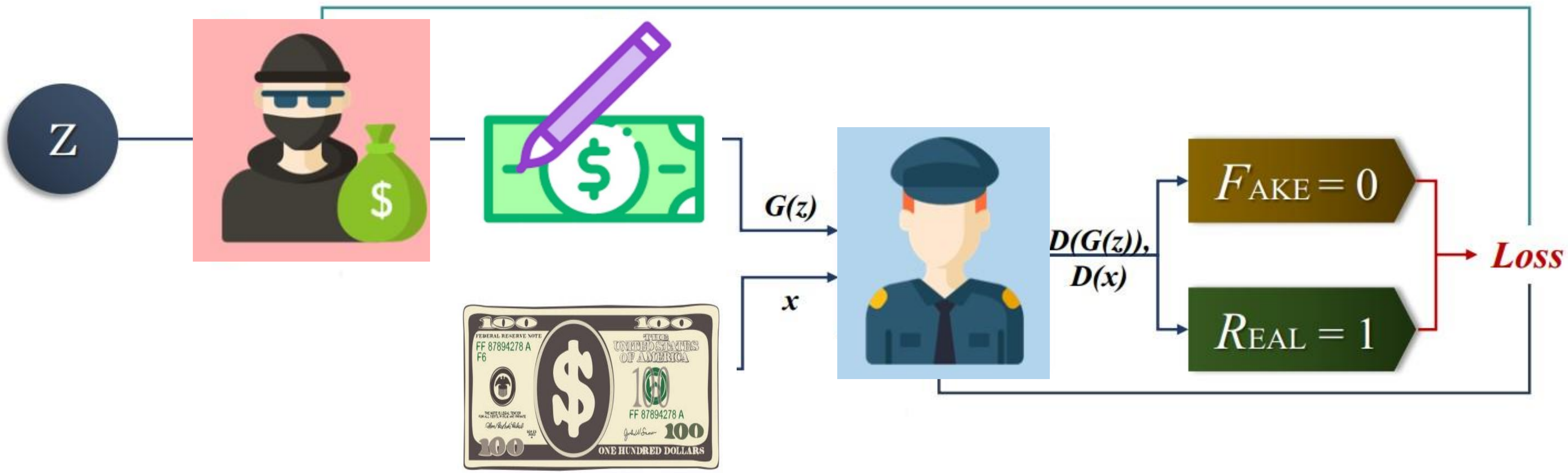


GAN

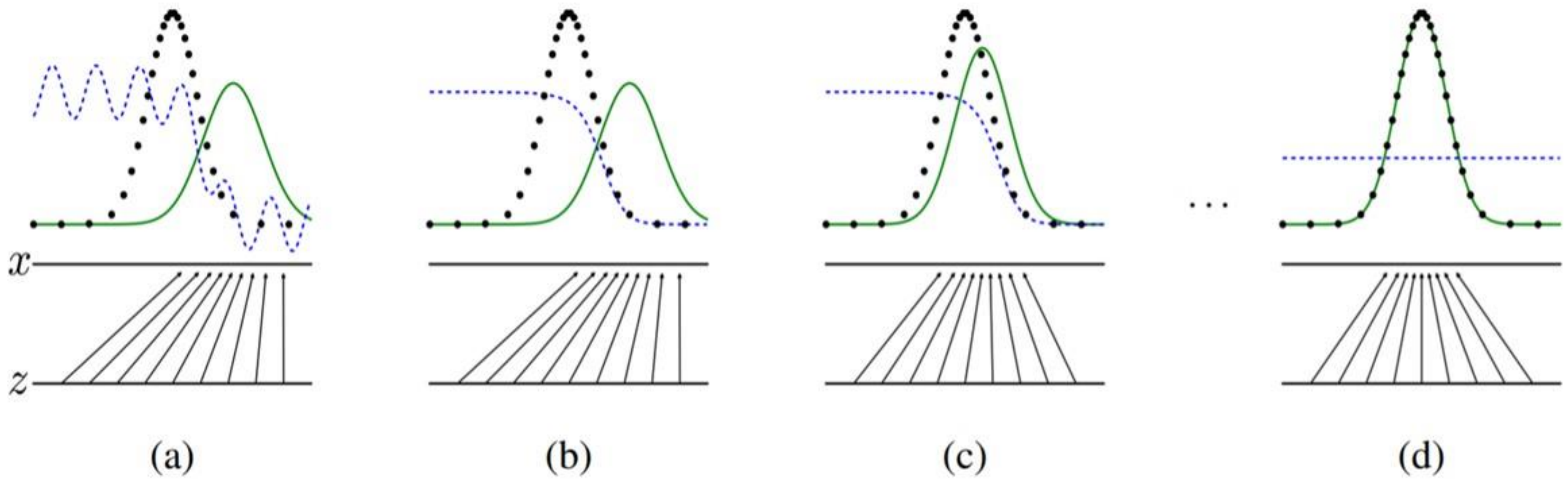


- **G : Unsupervised Learning**
  - G(z) : new data instance
- **D : Supervised Learning**
  - D(x) : probability (Real:1 ~ Fake:0)

GAN



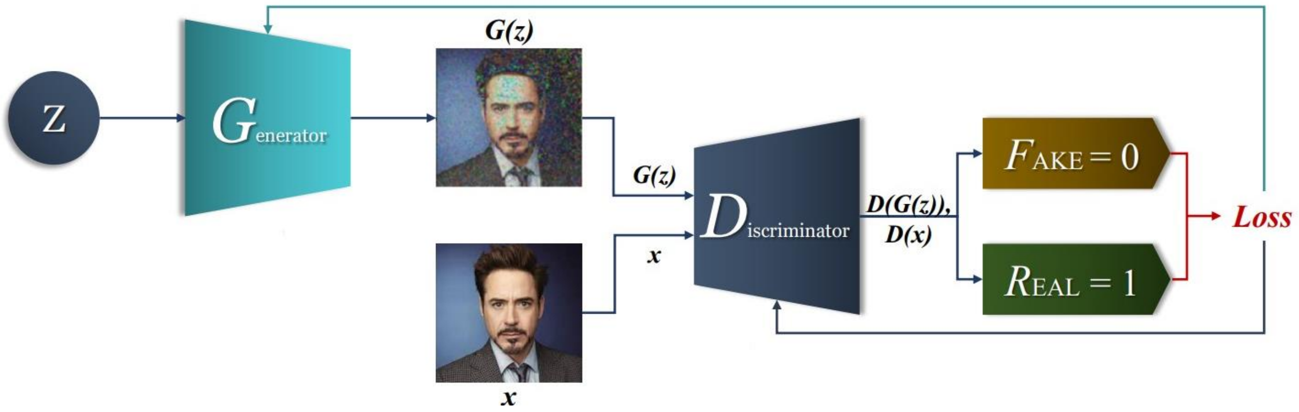
GAN



- — 원본 데이터의 분포(지폐)
- — 생성 모델의 분포(위조지폐)
- — 판별 모델의 분포.(경찰)

GAN

- 목적함수(Objective function)
- D 관점.
  - V함수 최대화

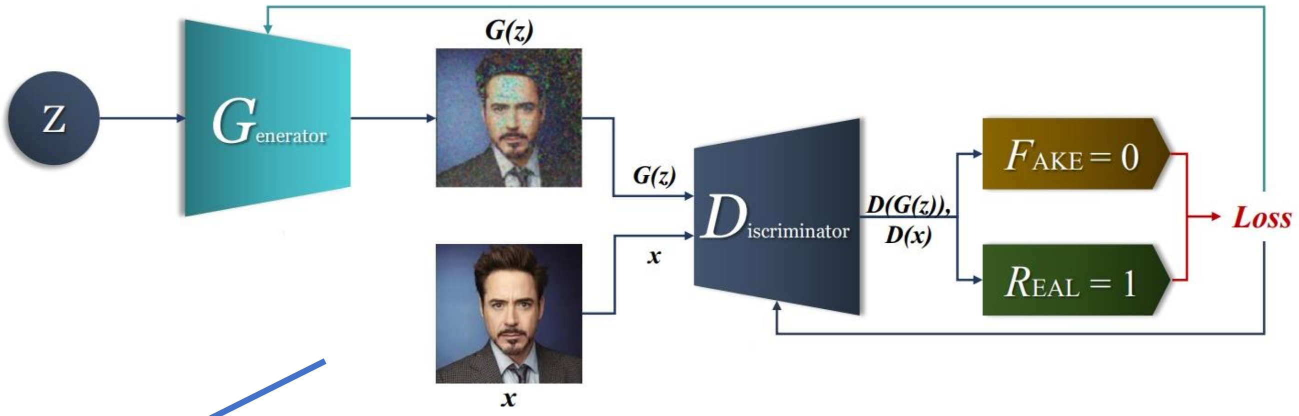


$$\min_G \max_D V(D,G) = E_{x \sim p_{data}(x)}[logD(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$

- Maximum when D(x) = 1.
- Maximum when D(G(x)) = 0.

GAN

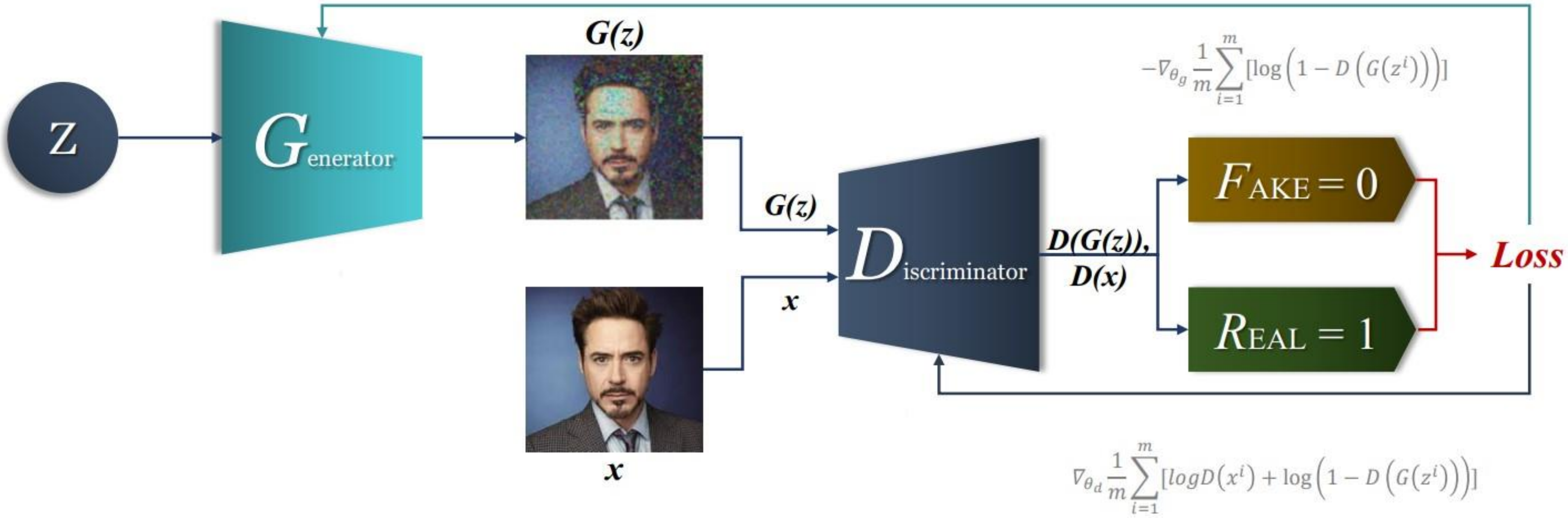
- 목적함수(Objective function)
- G 관점.
  - V함수 최소화



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

- 독립관계. 관여할수없음
- Minimum when D(G(x)) = 1.

GAN



Z
Generator
G(z)
x
G(z)
x
Discriminator
D(G(z)),
D(x)
FAKE = 0
REAL = 1
Loss
$-\nabla_{\theta_g}\frac{1}{m}\sum_{i=1}^{m}[\log\left(1-D\left(G(z^i)\right)\right)]$
$\nabla_{\theta_d}\frac{1}{m}\sum_{i=1}^{m}[logD(x^i)+\log\left(1-D\left(G(z^i)\right)\right)]$

- 기대값은 모든 사건에 대해 확률을 곱하면서 더하여 계산할 수 있습니다.

이산확률변수에 대한 기대값은 다음의 공식을 통해 계산할 수 있습니다.

$$E[X] = \sum_i x_i \cdot f(x_i)$$

연속확률변수에 대한 기대값은 다음의 공식을 통해 계산할 수 있습니다.

$$E[X] = \int x \cdot f(x)dx$$

X : 확률변수

x : 사건

f(x) : 확률 분포 함수

- Global Optimality 1

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

Proof: For G fixed,

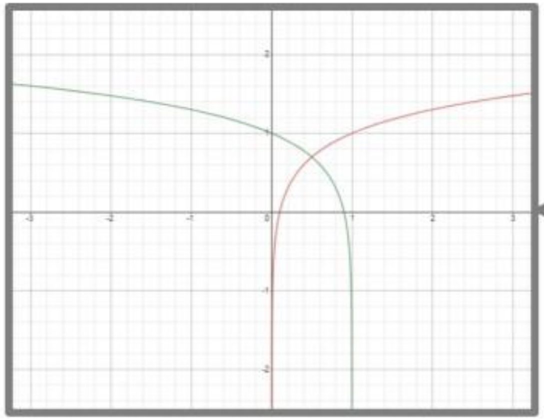
$$V(G,D) = E_{x\sim p_{data}(x)}[logD(x)] + E_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$

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

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

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

$$\boxed{function\ y \rightarrow alog(y) + blog(1-y)\ achieves\ its\ maximum\ in\ [0,1]\ at\ \frac{a}{a+b}}$$



same as ***optimal control***: $\frac{\delta V(G,D)}{\delta D}[D^*(x)] = 0$

- Global Optimality 2

Proposition: Global optimum point is $\boldsymbol{p_g = p_{data}}$

Proof:

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

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

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

removed when $p_g = p_{data}$

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

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

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

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

$$JSD(p||q) = \frac{1}{2}KL(p||\frac{p+q}{2}) + \frac{1}{2}KL(q||\frac{p+q}{2})$$

GAN

- 알고리즘

for the number of training iterations **do**

for k steps **do**

Sample minibatch of m noise samples $\{z^{(1)}, \dots, z^{(m)}\}$ from noise prior $p_g(z)$.

Sample minibatch of m examples $\{x^{(1)}, \dots, x^{(m)}\}$ from data generating distribution $p_{data}(x)$.

Update the discriminator by ascending its stochastic gradient:

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

end for

Sample minibatch of m noise samples $\{z^{(1)}, \dots, z^{(m)}\}$ from noise prior $p_g(z)$.

Update the generator by descending its stochastic gradient:

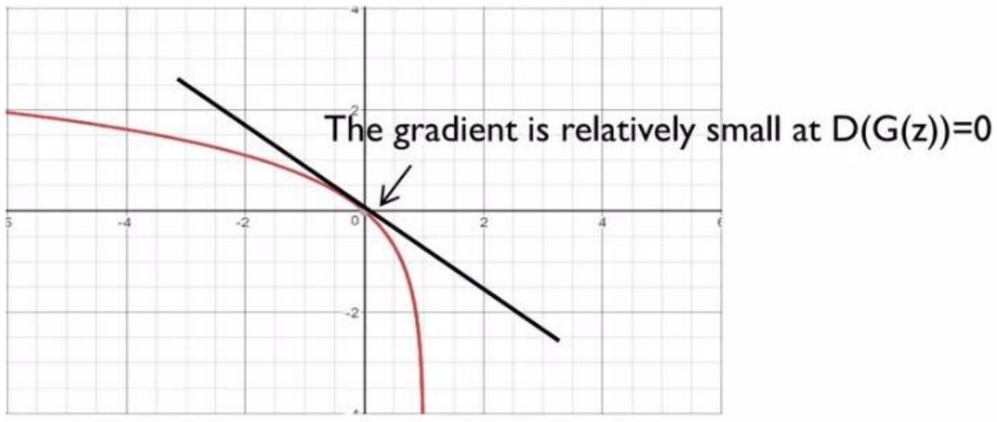
$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(z^i\right)\right)\right).$$

end for

The gradient-based updates can use any standard gradient-based learning rule. They used momentum.

GAN

$$\min_{G} E_{z \sim p_z(z)}[\log(1 - D(G(z))]$$



$$y = \log(1 - x)$$

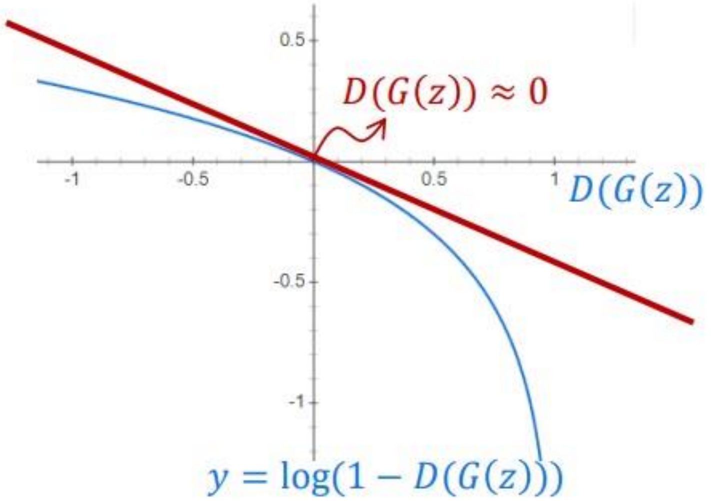
- 학습 초기 G의 성능이 좋지 않기 때문에 $D(G\ z\ ) \approx 0$일 확률 이 크다
- 따라서 $D(G\ z\ ) \approx 0$에서 Gradient가 이미 작아 학습이 **더딜 가능성이 많다**.

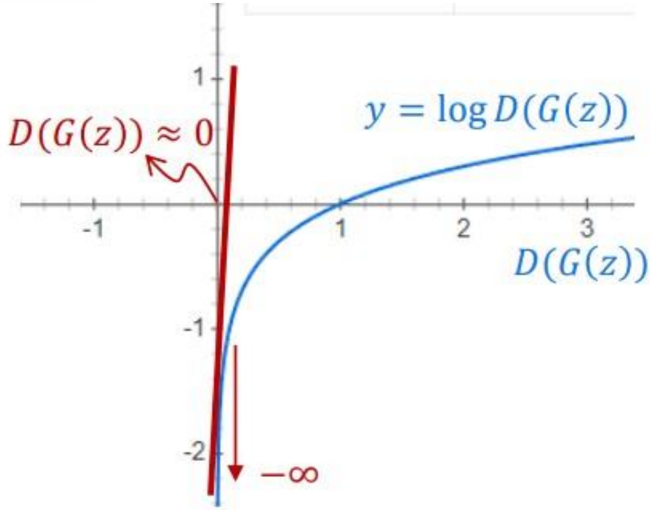
GAN

$$\cancel{\min_G E_{z\sim p_z(z)}[\log(1-D(G(z)))]}$$

$\downarrow$ Modification (heuristically motivated)

$$\max_G E_{z\sim p_z(z)}[\log D(G(z))]$$





- log$D(G(z))$ 형태로 변형하면 $D(G(z)) \approx 0$인 지점이 $-\infty$ 이 되 기 때문에 $D(G(z)) \approx 0$에서 Gradient가 매우 커지게 된다.
- **따라서 초기 학습속도를 높일 수 있다**

GAN

- 결과