NIPS 2014

Generative Adversarial Nets

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu,
David Warde-Farley,Sherjil Ozair, Aaron Courville, Yoshua Bengio

고민수

Summary

Background

- KL divergence (KL)
- Autoencoder (AE)
- Variabel Autoencoder (VAE)

GAN

- Idea
- Optimization
- Advanced GAN

KL Divergence

KL Divergence : Kullback Leibler Divergence

기준이 되는 확률분포의 Entropy + 상대적 Entropy = 비교할 확률분포의 Entropy

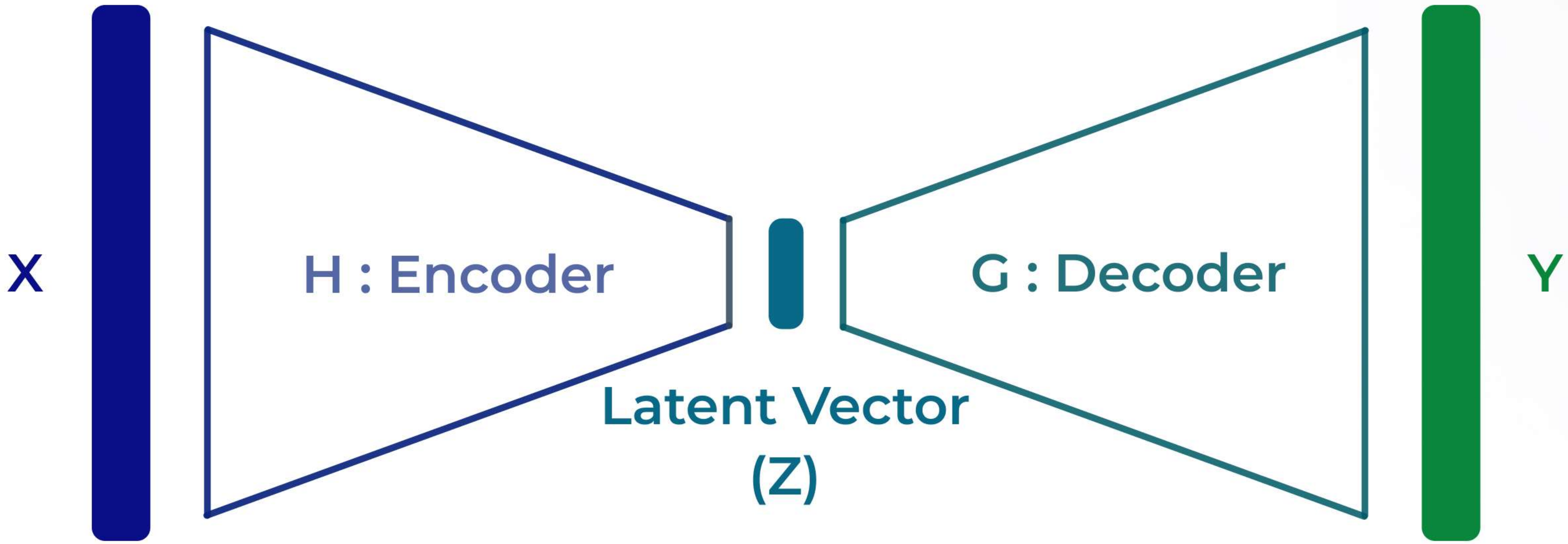
KL Divergence의 2가지 특징

- 항상 0보다 같거나 크다
- 비대칭적, 순서가 바뀌면 같지 않다

JSD(Jensen-Shannon Divergence)

- JSD(P||Q) = (KL(P||Q)+KL(Q||P))/2

Autoencoder



입출력이 동일한 네트워크

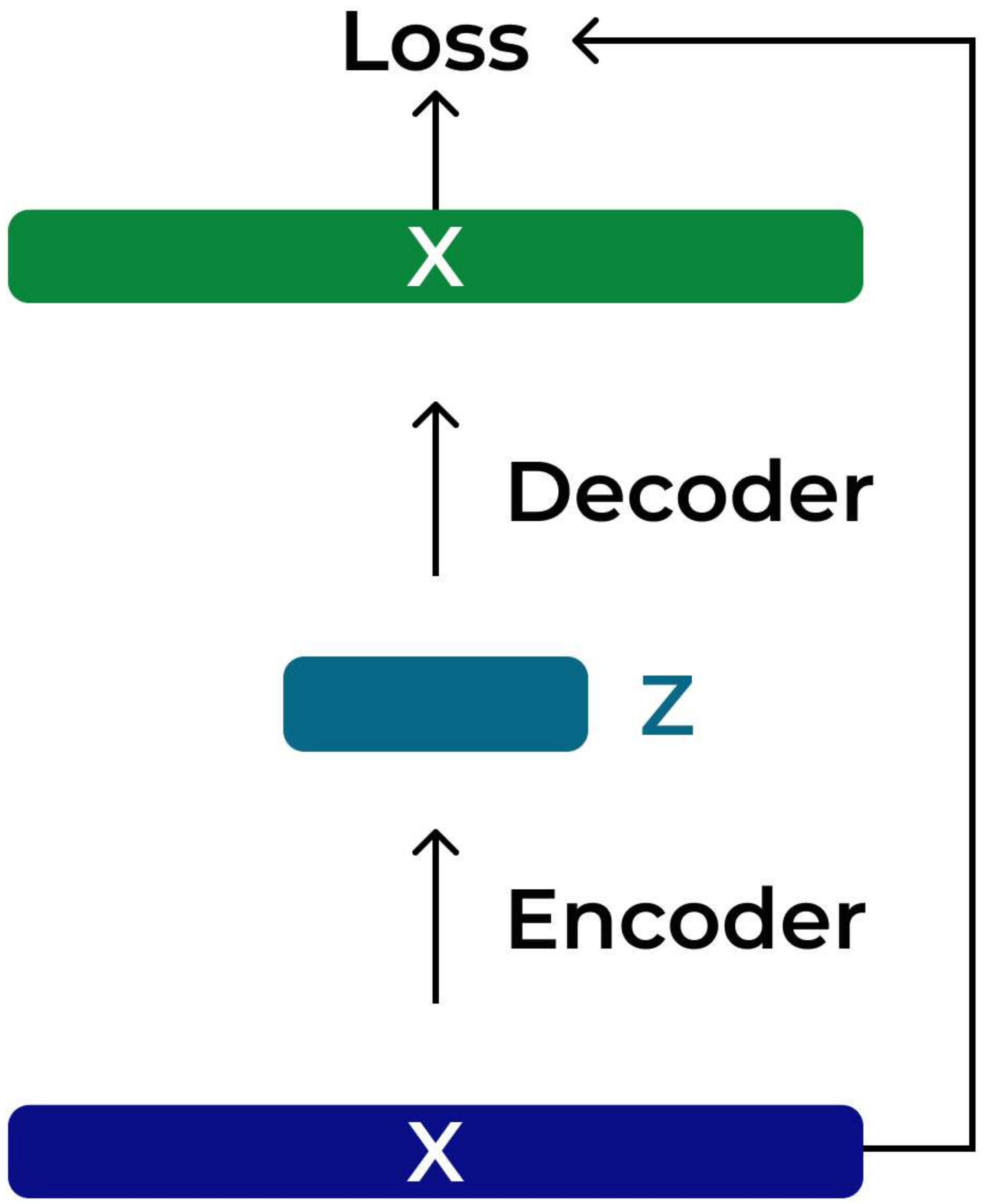
- Y가 X가 되도록 하는 네트워크
- 비지도학습문제를 지도학습문제로 전환
- Loss는 MSE 또는 cross-entropy를 사용

Advanced AE

- 분류모델의 새로운 학습 매커니즘
- CNN의 결합
- Anomaly Detection

Autoencoder

CNN 결합 - Convutional Autoencoder (CAE)



Decoder

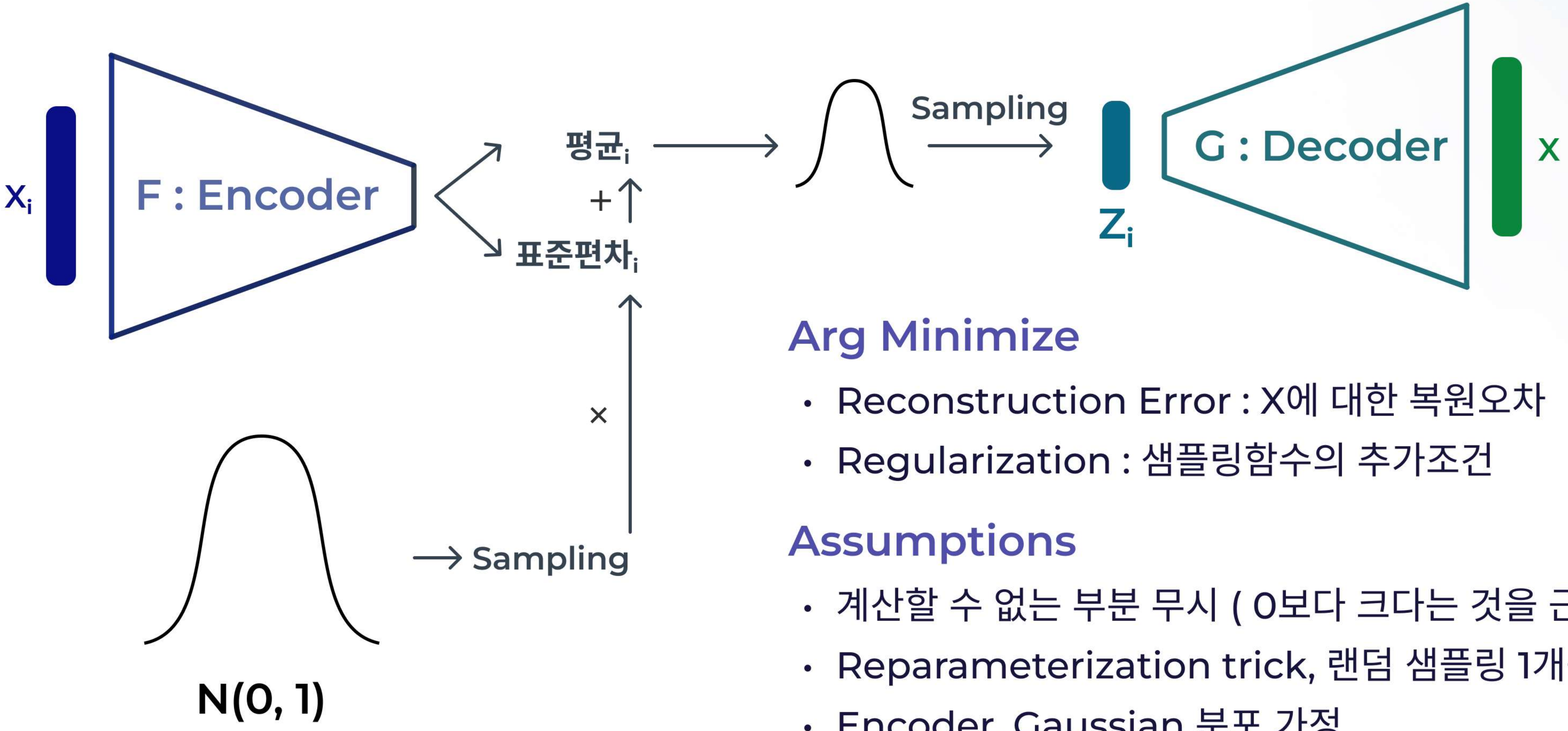
- ReLU, LeakyReLU 사용
- Up-sampling

Encoder

- ReLU, LeakyReLU 사용
- CNN
- 이미지의 추상화, 압축이 뛰어남

Variational Autoencoder

Autonecoder에 대한 수학적 접근 - X를 통해 X가 나올 확률이 가장 커지는 Distribution



Arg Minimize

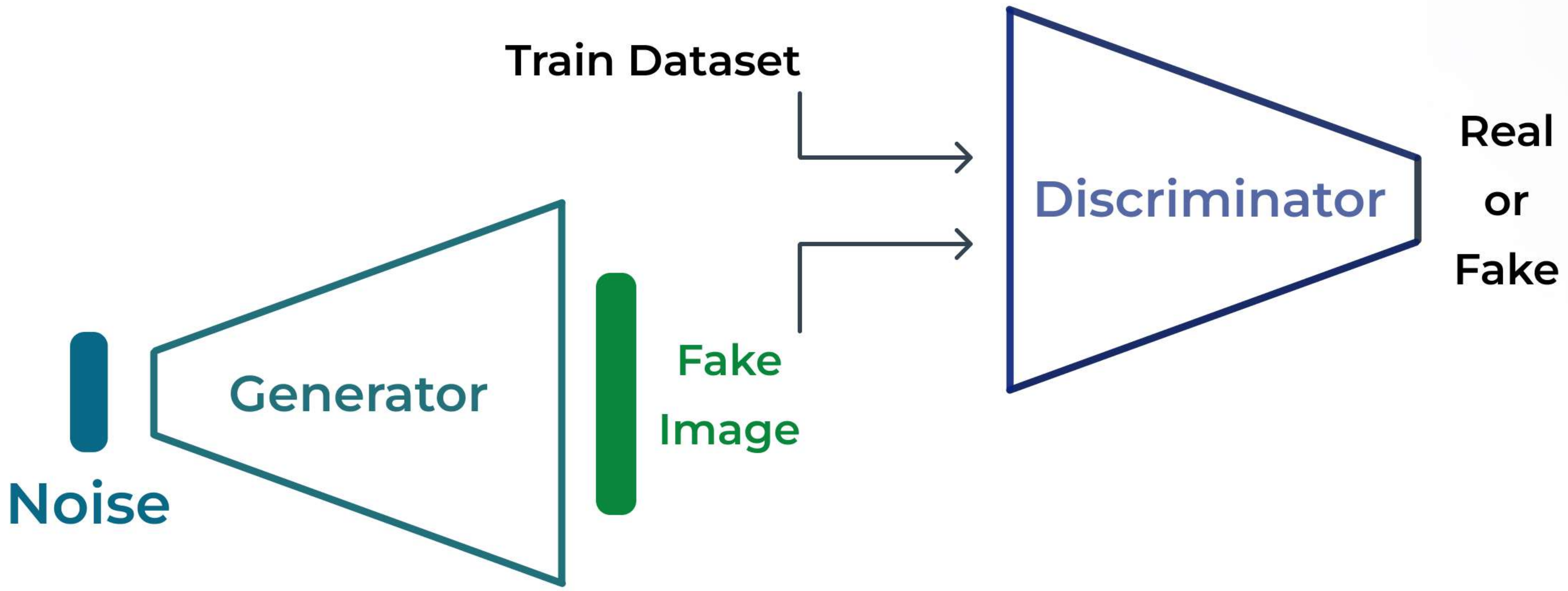
- Reconstruction Error : X에 대한 복원오차
- Regularization : 샘플링함수의 추가조건

Assumptions

- 계산할 수 없는 부분 무시 (0보다 크다는 것을 근거)
- Reparameterization trick, 랜덤 샘플링 1개를 대표값
- Encoder, Gaussian 분포 가정
- Decoder, Bernoulli 분포 가정

GAN

Generative Adversarial Nets



$$\underset{G}{MIN}\ \underset{D}{MAX}\ V(D,G) = E_{x\sim Pdata(x)}[\log(D(x))] + E_{z\sim Pz(z)}[\log(1-D(G(Z)))]$$

➞ **Two players game**

➞ **Real_data의 분포와 created_data의 분포의 차이를 최소화**

GAN

Discriminator

$$D^*(x) = \text{argMax}V(D) = E_{x\sim data(x)}[\log D(x)] + E_{z\sim p(z)}[\log(1-D(G(z)))]$$

Z 에 대한 식을 X로 전환

$$= E_{x\sim data(x)}[\log D(x)] + E_{z\sim Pg(x)}[\log(1-D(x))]$$

적분 표현

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

결합

$$= \int_x \boxed{P_{data(x)} \log D(x) + P_g(x) \log(1-D(x))} dx$$

최대화

$$D^*(x) = \text{argMax}V(D) = P_{data(x)} \log D(x) + P_g(x) \log(1-D(x))$$

$a = P_{data(x)}$, $b = P_g(x)$, $y = D(x)$

$$= a\log y + b\log(1-y)$$

$f'=0$

$y = a/(a+b)$ 일때 $D^*(x)$

$$\boxed{D^*(x) = \frac{P_{data(x)}}{P_{data(x)} + P_{g(x)}}}$$

GAN

Generator

$$\text{MinV}(G) = E_{x\sim data(x)}[\log D^*(x)] + E_{z\sim Pg(x)}[\log(1-D^*(x))]$$

$$= \int_x P_{data(x)} \log \frac{P_{data(x)}}{P_{data(x)} + P_{g(x)}} dx + \int_x P_g(x) \log(1 - \frac{P_{data(x)}}{P_{data(x)} + P_{g(x)}})dx$$

$$= \int_x P_{data(x)} \log \frac{2*P_{data(x)}}{P_{data(x)} + P_{g(x)}} dx + \int_x P_g(x) \log(1 - \frac{2*P_{data(x)}}{P_{data(x)} + P_{g(x)}})dx - \log 2$$

$$= KL(P_{data} || \frac{P_{data(x)} + P_{g(x)}}{2}) + KL(P_g || \frac{P_{data(x)} + P_{g(x)}}{2}) - \log 4$$

$$= 2*JSD(P_{data}||P_g) - \log 4$$

$$\boxed{\text{MinV}(G) = JSD(P_{data}||P_g)}$$

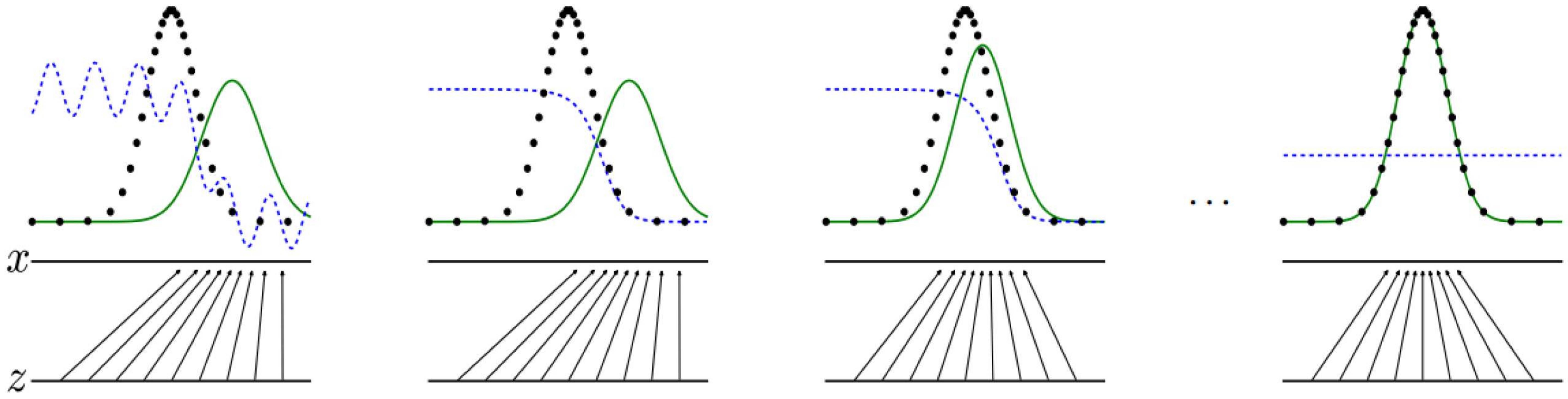
→ $P_{data} = P_g$ 가 되도록 한다 → $D^*(x) = 1/2$ 가 되도록 한다 (가짜를 구분할 수 없음)

GAN

Generator

$\boxed{\mathrm{MinV}(G) \quad = \mathrm{JSD}(P_{data} || P_g)}$

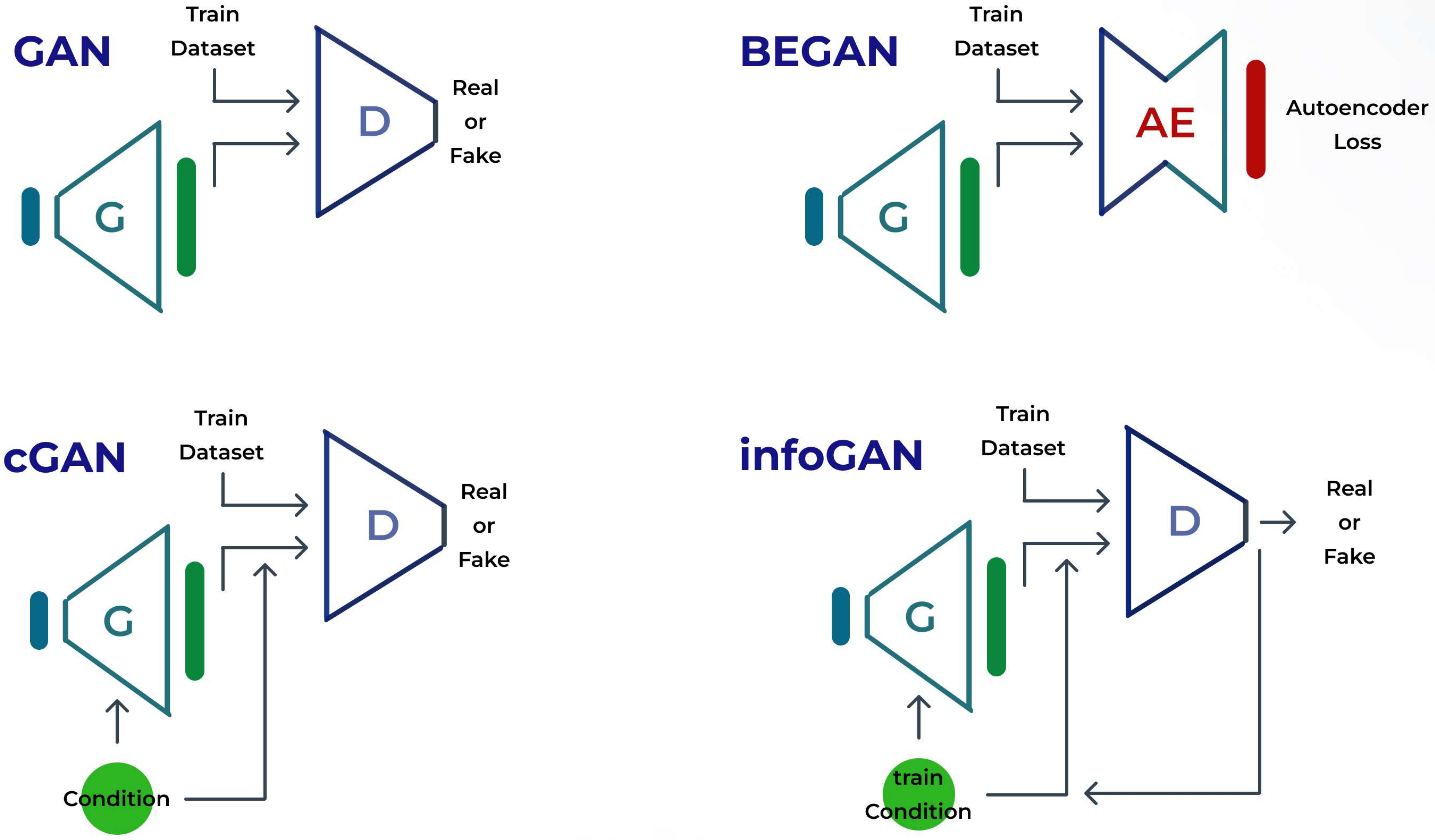
→ $P_{data} = P_g$ 가 되도록 한다 → $D^*(x) = 1/2$ 가 되도록 한다 (가짜를 구분할 수 없음)



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

Real_data의 분포와 created_data의 분포의 차이를 최소화

advanced_GAN



Thank you

Generative Adversarial Nets

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu,
David Warde-Farley,Sherjil Ozair, Aaron Courville, Yoshua Bengio