

Generative Basic

Generative Adversarial Nets

들어가기 전

생성모델(generative model)이란?

- 생성 모델은 주어진 학습 데이터를 학습하고 학습 데이터의 분포를 따르는 (있을 법한) 유사 데이터를 생성하는 모델 → training set의 distribution을 얻고자 함

Training samples ~ **p_data(x)**

Generated samples ~ **p_model(x)**

p_data : Real Word 데이터 분포

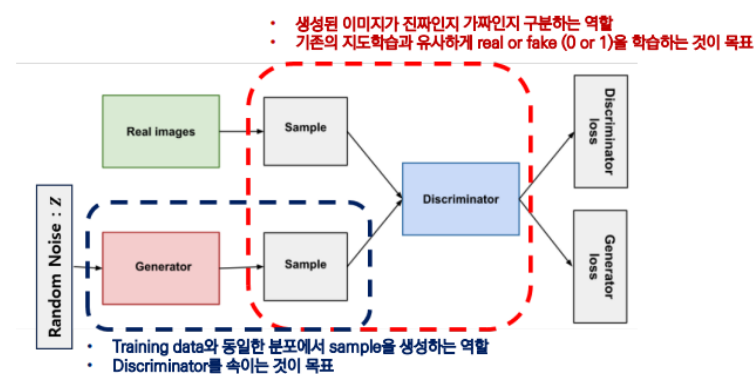
p_model : generated sample 분포

✓ p_data(x)와 p_model(x)가 유사하도록 학습한다.

- p_model을 확실하게 정의하는 **Explicit Density Estimation**(ex. VAE)
- model을 정의하지 않고 p_model에서 sample을 생성하는 **Implicit Density**(ex. GAN)

GAN

- Generative Adversarial Network**
 - 생산적 적대적인(적, 상대가 있음) 신경망



- 훈련 샘플과 생성 샘플을 **적대적으로** 학습시키며, 실제 데이터와 비슷한 데이터를 **생성**한다.
- 생성된 데이터는 label이 없기 때문에 Un-supervised learning 모델이다.
- 생성자와 판별자 두 개의 네트워크를 사용한다.

generator(생성자) : 이미지를 만들어낸다.

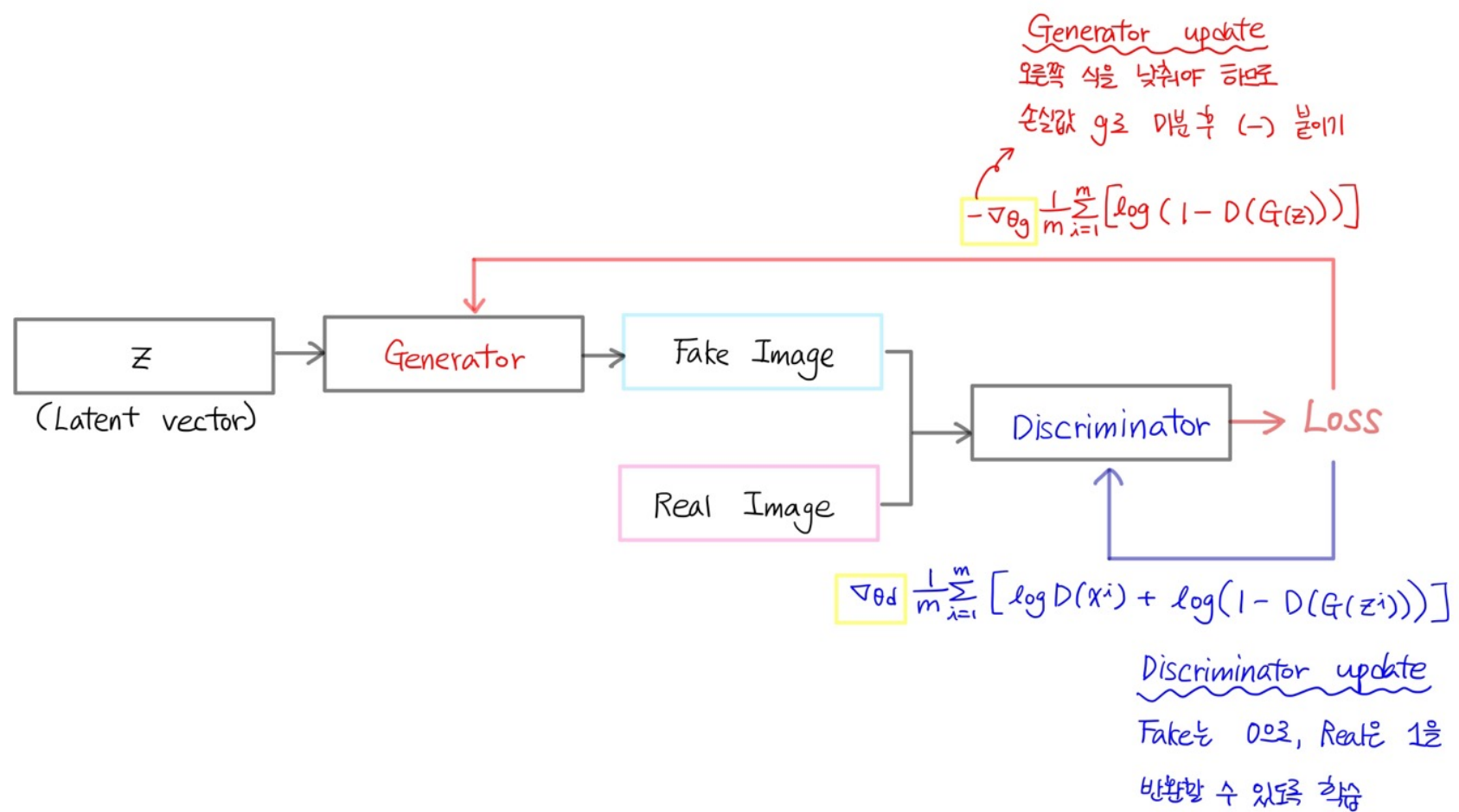
discriminator(판별자) : generator를 평가한다.

- objective function**

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

- function V** : D는 최대로, G는 최소로 하게 하려는 함수
- Pdata** : 실제 데이터 분포
- Pz** : 노이즈 데이터 분포
- D(x)** = Probability 얼마나 진짜 같은지 (Real:1 ~ Fake:0)

- **$G(z)$** : 새로운 데이터(fake sample)
- 왼쪽 식) 원본 데이터(Pdata)에서 이미지 여러개(x)를 꺼내 $\log D(x)$ 를 취한 값의 기댓값(평균값)
- 오른쪽) 노이즈 데이터(Pz)에서 이미지 여러개(z)를 꺼내 $G(z)$ 에 넣어서 가짜 이미지를 만든 다음, D에 넣은 값을 1에서 빼고 그 값에 로그를 취해 구한 기댓값
- D를 maximization : 원본 데이터 x에 대해서는 1로 분류할 수 있도록, 즉 real로 분류할 수 있도록 학습한다. 확률 D가 1이 되어야 하므로 D는 최대화시킨다. log는 monolithic한 함수로 1이 되도록 한다.
- 가짜 이미지 $G(z)$ 가 real일수록 $D(G(z))$ 는 1에 가까워지고, 1에서 빠지므로 $\log(1-D(G(z)))$ 는 작은 값을 가진다. 즉 생성자는 자신이 만든 이미지가 실제와 비슷하도록 학습을 진행한다.



- 동일한 식에 대해서 G는 minimize 방향으로, D는 maximize 방향으로 학습하는 다른 목적을 가지고 있음.

• GAN의 목표

- $P_g \rightarrow P_{data}$; 노이즈 데이터가 실제 데이터를 잘 따를 수 있도록 학습한다.
- $D(G(z)) \rightarrow 1/2$; 가짜 데이터와 이미지를 더이상 구분할 수 없도록 학습한다. (가짜 이미지를 D가 구분할 수 없음)

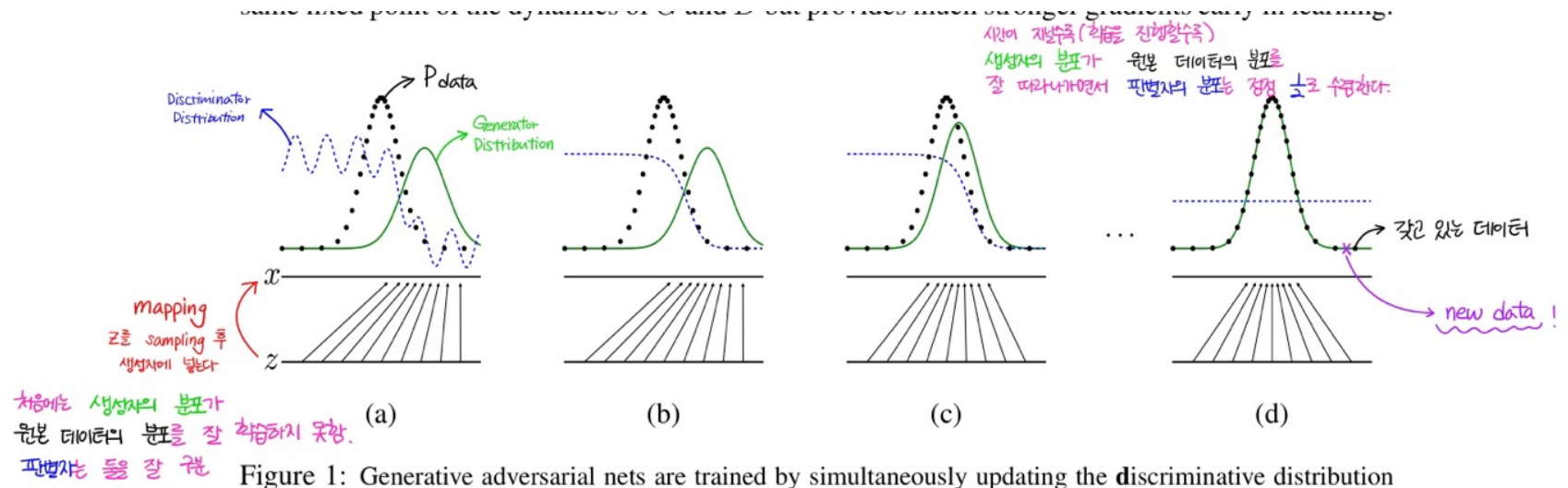


Figure 1: Generative adversarial nets are trained by simultaneously updating the discriminative distribution (D , blue, dashed line) so that it discriminates between samples from the data generating distribution (black, dotted line) and samples from the discriminator distribution (G , green, solid line). The discriminator distribution (D) is updated to be more discriminative, and the generator distribution (G) is updated to be more like the data distribution (P_{data}).

- 목표 : 생성모델의 분포 P_g 가 원본 데이터의 분포 P_{data} 로 수렴하도록 한다. → 어떻게? 증명을 해보자.

• Global Optimality of $P_g = P_{data}$

- 명제) G 가 고정되어 있을때, D 의 optimal point는 다음과 같다.

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

- 증명)

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

이 식의 구조는 다음과 같다.

$$a \log(y) + b \log(1 - y) \text{ for } (a, b) \in \mathbb{R}^2$$

$$\text{maximum in } [0, 1] \text{ at } \frac{a}{a+b}$$

따라서 위의 식은 다음과 같을 때 최대를 가진다.

$$C(G) = \max_D V(G, D) \quad (1)$$

$$= E_{x \sim p_{data}}[\log D_G^*(x)] + E_{z \sim p_z(z)}[\log(1 - D^*(G(z)))] \quad (2)$$

$$= E_{x \sim p_{data}}[\log D_G^*(x)] + E_{z \sim p_z(z)}[\log(1 - D^*(x))] \quad (3)$$

$$= E_{x \sim p_{data}}[\log \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}] + E_{x \sim p_g(x)}[\log \frac{p_g(x)}{p_{data}(x) + p_g(x)}] \quad (4)$$

$$= E_{x \sim p_{data}}[\log \frac{2 * p_{data}(x)}{p_{data}(x) + p_g(x)}] + E_{2 * x \sim p_g(x)}[\log \frac{p_g(x)}{p_{data}(x) + p_g(x)}] - \log(4) \quad (5)$$

$$= -\log(4) + KL(p_{data} || \frac{p_{data} + p_g}{2}) + KL(p_g || \frac{p_{data}(x) + p_g(x)}{2}) \quad (6)$$

$$= -\log(4) + 2 * JSD(p_{data} || p_g) \quad (7)$$

• GAN 알고리즘

