



글로벌미디어학부

미디어 GAN

Media GAN(Generative Adversarial Network)

13주차: DCGAN

박재완

학습 목표

- GAN 수식의 이해
- DCGAN의 이해

[사전 녹화]

- 전이학습의 이해
- Data Augmentation의 방법

1. 기대값 (Expectation)

- 기대값은 단순한 평균 그 이상으로 일반화된 개념임
 - 예측/추정하려는 어떤 특정값이 아닌, 기대되는 예측치들의 평균값
- 즉, 확률적 분포 개념이 고려된 평균임
 - 확률분포의 성격을 결정짓는 확률적 평균치
 - 확률변수가 나타내는 확률분포에서, 중심 경향/위치 (즉, 중심으로 기대되는 대표 값)
 - 확률변수에 취해지는 확률적 가중평균 : $E[X] = \sum_{i=1}^n x_i P(x_i)$

2. 기대값, 평균 비교

- 기대값(Expected Value)
 - 이따금, 평균(Mean Value)과 같은 의미로 잘못 사용됨
 - 기대값은 모집단에 대한 통계적 파라미터 중 하나로서 기대되는 예측치들의 평균값을 의미
 . 즉, 모집단 확률분포의 모수에 대한 무게중심으로 기대되는 값
- 결국,
 - 기대값은 매 표본 집단 마다 얻어지는 표본 평균들에 대한 일반화임
 . 시행횟수 $n \rightarrow \infty$ 이면, 표본평균은 모집단(확률변수)의 기대값에 수렴 \Rightarrow 대수의 법칙
- (기대값, 평균 표기 관계)
 - 확률변수 X 에 대한 기대값(Expectation) 표기 : $E[X]$ 또는 $\langle X \rangle$
 - 확률변수 X 에 대한 평균(Mean) 표기 : \bar{X} 또는 μ_X

Discriminator의 Back Propagation:

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

D는 최대화하는 방향으로, G는 이를 최소화하는 방향으로 가게하는 minmax Problems

$\mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})]$: D가 $V(D, G)$ 를 최대화하는 관점

$\mathbf{z} \sim p_{\text{data}}(\mathbf{x})$ 는 실제 데이터의 확률분포이고, \mathbf{x} 는 그 중 샘플링한 데이터이다. 판별자 D 는 출력이 실제 데이터가 들어오면 1에 가깝게 확률을 추정한다. $\log 1$, 최대값인 0에 가까운 값이 나온다.

$\mathbb{E}_{\mathbf{z} \sim p_z(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))]$: G 가 $V(D, G)$ 를 최소화하는 관점

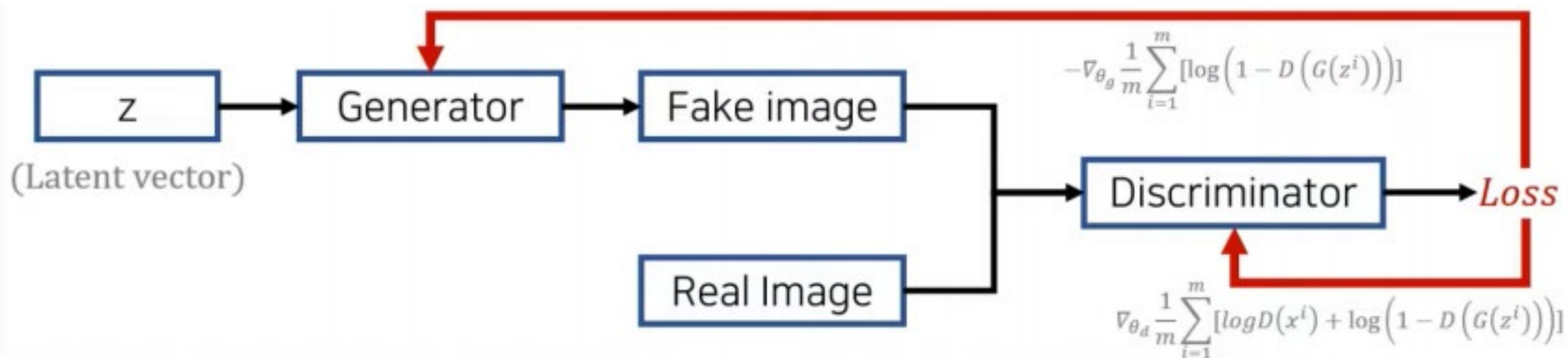
$\mathbf{z} \sim p_z(\mathbf{z})$ 는 보통 정규분포로 사용하는 임의의 노이즈 분포이고, \mathbf{z} 는 노이즈 분포에서 샘플링한 임의의 코드이다. 이 입력을 생성자 G 에 넣어 만든 가짜 데이터를 판별자 D 가 속아서 진짜로 판별된다면, $\log(1 - D(G(\mathbf{z})))$ 식에서 $\log(1 - D(G(\mathbf{z})))=1$ 값이 들어간다. $\log 0$ 이 되어 $-\infty$ 된다.

Discriminator loss

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

Generator loss

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



```

for e in range(epochs):
    for i in range(len(X_train) // batch_size):

        discriminator.trainable = True

        idx = np.random.randint(0, X_train.shape[0], batch_size)
        imgs = X_train[idx]
        d_loss_real = discriminator.train_on_batch(imgs, real)

        noise = np.random.normal(loc=0, scale=1, size=(batch_size, latent_dim))
        fake_imgs = generator.predict_on_batch(noise)
        d_loss_fake = discriminator.train_on_batch(fake_imgs, fake)

        d_loss_batch = 0.5 * (d_loss_real[0] + d_loss_fake[0])

        discriminator.trainable = False
        g_loss_batch = gan.train_on_batch(noise, real)

    d_loss.append(d_loss_batch)
    g_loss.append(g_loss_batch[0])
    print("epoch = %d/%d, d_loss=%.3f, g_loss=%.3f" % (e + 1, epochs, d_loss[-1], g_loss[-1]))

    if e % 5 == 0:
        samples = 10
        fake_imgs = generator.predict(np.random.normal(loc=0, scale=1, size=(samples, latent_dim)))

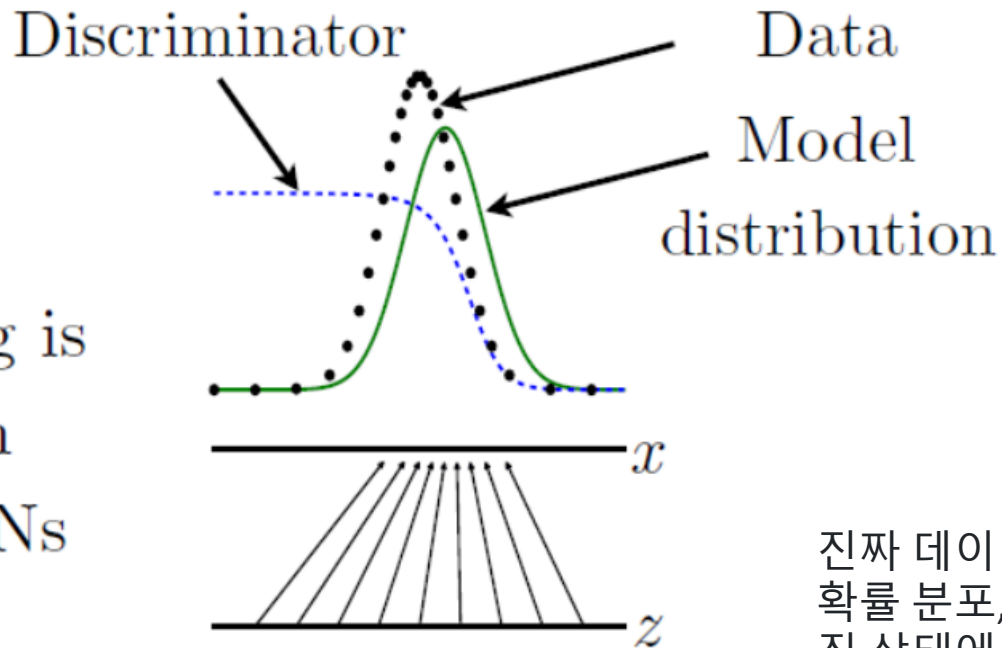
        for k in range(samples):
            plt.subplot(2, 5, k + 1)
            plt.imshow(fake_imgs[k].reshape(28, 28), cmap="gray")
            plt.xticks([])
            plt.yticks([])
        plt.tight_layout()
        plt.show()

```

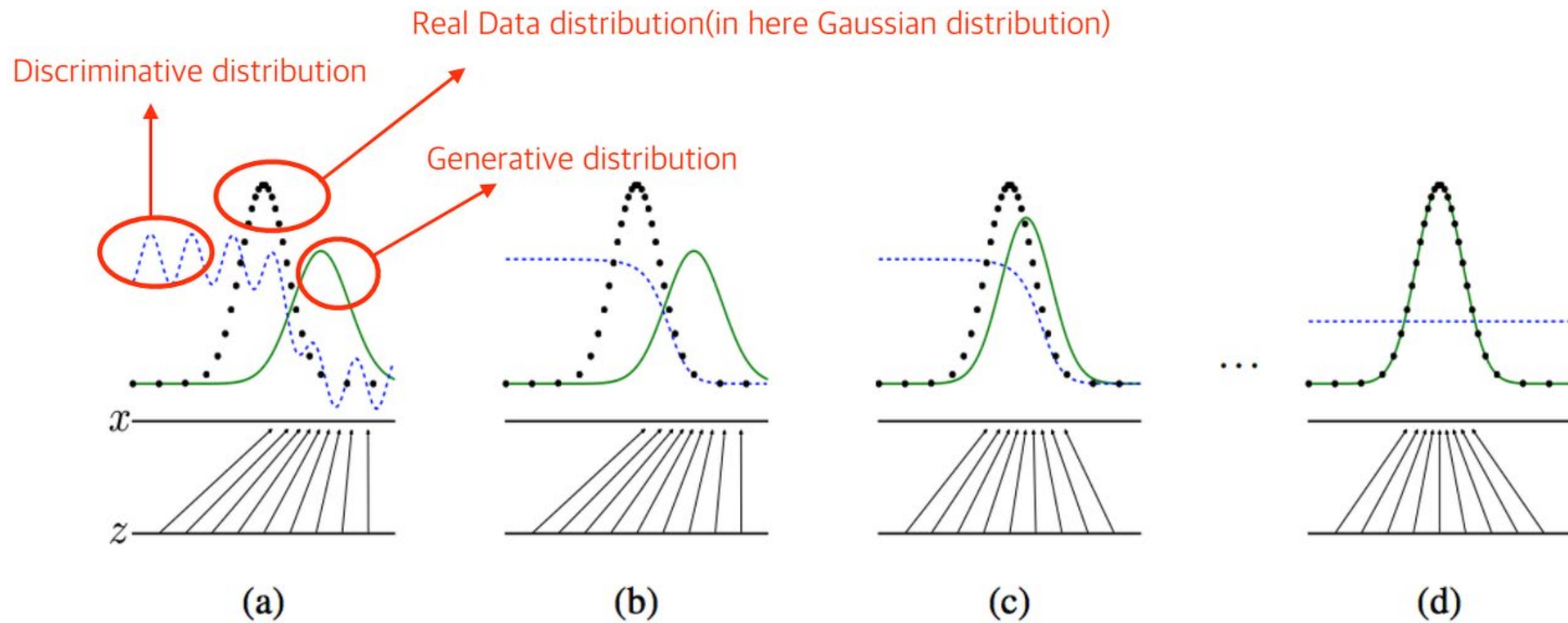

- Optimal discriminator

$$D(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_{\text{model}}(x)}$$

Estimating this ratio
using supervised learning is
the key approximation
mechanism used by GANs



진짜 데이터의 확률 분포, 생성기의 확률 분포, 판별기의 확률 분포를 가진 상태에서 판별기는 생성기가 진짜 데이터의 확률 분포가 얼마나 다른지를 판단



기말고사: 2022년 12월 14일 수

- 객관식: 20%
- LMS 이용: 노트북 준비

과제(프로젝트): DCGAN 구축

- 배점: 과제 10% + 기말고사 실기(10%)
- 마감: 2022년 12월 19일 월 23:59분
- 늦은 제출: 하루 당 1점 감점 (23일 이후 제출 할 필요가 없음)

- 방법

- (1) 주제를 자유롭게 선정
- (2) 데이터셋을 확보
- (3) 모델 구축 및 학습
- (4) 문제 해결

- 주요 평가

- (1) 주제의 창의성
- (2) 기술적용의 독창성
- (2) 문제해결 능력
- (3) 결과 퀄리티