WGAN / WGAN-GP

Generative Deep Learning

GAN의 도전 과제와 개선방향

손실이 일정치 않고 진동하거나 모드 붕괴 현상 개선 필요

- **Mode Collapse**
  - 손실 함수의 그래디언트가 0에 가까운 값으로 무너진다(Collapse)
  - 생성자가 판별자를 속이는 적은 수의 샘플을 찾을 때 일어남
    >생성자가 다양한 출력을 만들지 않게 됨

- **WGAN**
  - Wasserstein거리에 의한 손실함수의 설계
    > 요소를 만족하기 위해 가중치를 클리핑
    > 학습이 불안정한 문제

- **WGAN-GP**
  - Grgdient penality를 도입

* 생성자는 판별자를 항상 속이는 하나의 샘플(이를 Mode라고 부른다)을 찾으려는 경향이 있다.

손실함수의 변화 (binary cross entropy -> Wasserstein loss)

•GAN 손실 함수

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

•Wasserstein 손실 함수

- Wasserstein loss는 target을 1과 0 대신 1과 –1 사용
- 마지막 층에서 시그모이드 활성화 함수를 제거하여 예측 범위가 [0, 1]로 국한되지 않고 [–∞, ∞] 범위의 어떤 숫자도 될 수 있음
- 손실함수에 log를 이용하지 않음
- Discriminator의 대신에 Critic(비평가)라 부름

$$L = \mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}}\sim\mathbb{P}_g}[D(\tilde{\boldsymbol{x}})] - \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathbb{P}_r}[D(\boldsymbol{x})]$$

Discriminator의 구조 변화 (Critic)

• Lipschitz 제약

- Wasserstein 손실은 제한이 없어 큰 값일 수 있기에 $[-\infty, \infty]$, Critic에 제약이 필요
- Lipschitz(립시츠) 함수

  > 임의의 두 지점의 기울기가 어떤 상수 값 이상 증가하지 않는 함수

  > 상수 값이 1인 경우 1-Lipschitz 함수라고 부름

$$\frac{|D(x_1) - D(x_2)|}{|x_1 - x_2|} \leq 1$$: 분모는 두 이미지의 픽셀의 차이, 분자는 Critic 예측간의 차이를 의미

• 가중치 클리핑

- Critic의 가중치를 $[-0.01, 0.01]$ 안에 놓이도록 가중치 클리핑을 통해 립시츠 제약을 부과함

WGAN의 개선 필요성

•기본 GAN과 차이점

- 기본 GAN은 gradient 소실을 피하기 위해 판별자가 너무 강해지지 않도록 하는 것이 중요함 (*Discriminator.Trainable = False)

  >Wasserstein 손실을 이용하면 이런 어려움을 제거 할 수 있음

  > 일반적으로 생성자를 업데이트 하는동안 Critic을 여러번 업데이트 함

•단점

- Critic에서 가중치를 클리핑했기 때문에 학습 속도가 크게 감소함

  (논문의 저자도 립시츠 제약을 두기 위해 가중치를 클리핑 하는 것은 좋지 않은 방법이다 언급)

- 강한 Critic은 성공의 핵심. 정확한 gradient가 없다면 생성자의 학습이 어려워짐

  –> 가중치 클리핑 보다 다른 방법이 필요

WGAN과 생성자는 동일하고, 비평자(Critic)가 차이가 있음
비평자에 립시츠 제약을 강제하는 다른 방법을 제안함

•WGAN-GP의 Critic

- 비평자 손실 함수에 gradient penalty항을 포함
- 비평자 가중치를 클리핑하지 않음
- 비평자에 배치 정규화 층을 사용하지 않음

•Gradient penalty loss

- Critic에서 가중치를 클리핑했기 때문에 학습 속도가 크게 감소함
  (논문의 저자도 립시츠 제약을 두기 위해 가중치를 클리핑 하는 것은 좋지 않은 방법이다 언급)
- 강한 Critic은 성공의 핵심. 정확한 gradient가 없다면 생성자의 학습이 어려워짐
  -> 가중치 클리핑 보다 다른 방법이 필요

Norm은 벡터의 길이 혹은 크기를 측정하는 방법(함수)

•L1 Norm

- 벡터의 요소에 대한 절댓값의 합

$$L_1 = (\sum_i^n |x_i|)$$
$$= |x_1| + |x_2| + |x_3| + \ldots + |x_n|$$

- L1 Loss : 실제 값과 예측치 사이의 차이(오차) 값의 절대값 구하고 그 오차들의 합

$$L = \sum_{i=1}^{n} |y_i - f(x_i)|$$

•L2 Norm

- L2 Norm은 n 차원 좌표평면(유클리드 공간)에서의 벡터의 크기
- 2차원 좌표 평면상의 최단 거리를 계산

$$L_2 = \sqrt{\sum_i^n x_i^2}$$
$$= \sqrt{x_1^2 + x_2^2 + x_3^2 + \ldots + x_n^2}$$

- L2 Loss : 오차의 제곱의 합

$$L = \sum_{i=1}^{n} (y_i - f(x_i))^2$$

* y_i 는 실제 값을, f(x_i)는 예측치를 의미

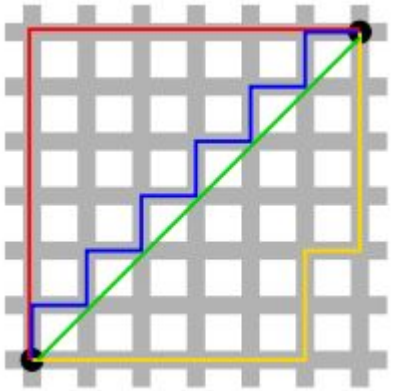
L1 Norm은 Feature selection이 가능하고, L2 Norm은 Unique shortest path를 가진다

•L1 & L2 Feature

- 검정색 두 점사이의 L1 Norm 은 빨간색, 파란색, 노란색 선으로 표현 될 수 있고, L2 Norm 은 오직 초록색 선으로만 표현될 수 있다
- L1 Norm 은 여러가지 path 를 가지지만 L2 Norm 은 Unique shortest path 를 가진다

  > 즉, L2 Norm 은 각각의 벡터에 대해 항상 Unique 한 값을 내지만,

  L1 Norm 은 경우에 따라 특정 Feature(벡터의 요소) 없이도 같은 값을 낼 수 있다



- L1 Norm 은 파란색 선 대신 빨간색 선을 사용하여 특정 Feature 를 0으로 처리하는 것이 가능하다고 이해할 수 있고, L1 Norm 은 Feature selection 이 가능하고 이런 특징이 L1 Regularization 에 동일하게 적용 될 수 있다

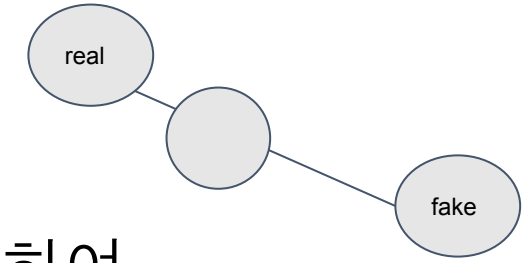
출처: https://light-tree.tistory.com/125

WGAN-GP - Gradient penalty loss

비평자에 1-Lipschitz 제약을 강제하는 다른 방식 : Gradient penalty loss

- 비평자 손실 함수에 gradient penalty항을 포함

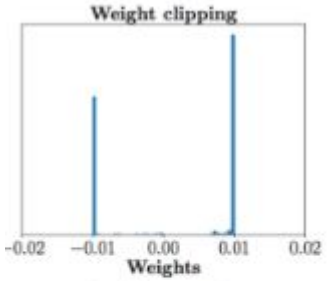
$$L = \underbrace{\mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}}\sim\mathbb{P}_g}[D(\tilde{\boldsymbol{x}})] - \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathbb{P}_r}[D(\boldsymbol{x})]}_{\text{Original critic loss}} + \underbrace{\lambda \mathop{\mathbb{E}}_{\hat{\boldsymbol{x}}\sim\mathbb{P}_{\hat{\boldsymbol{x}}}}\left[(\|\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}})\|_2 - 1)^2\right]}_{\text{Our gradient penalty}}.$$



- 훈련과정에서 모든 곳에서 gradient를 계산하기는 불가능하여, 일부 지점에서만 gradient를 계산한다

  >한쪽에 치우치지 않기 위해 진짜 이미지와 가짜 이미지쌍을 연결한 직선을 보간한 이미지를 사용

- 보간된 포인트에서 계산한 gradient와 1사이의 차이를 제곱하여 반환



- 미분가능한 함수는 모든 곳에서 gradients norm이 1이어야만 1-Lipschtiz이다.

출처: https://leechamin.tistory.com/232