

GAN: Generative Adversarial Networks

Tobigs20기 오수아

0. Introduction: a new framework

분류 모델은 backpropagation 방법으로 인해 더 deep한 NN을 만들고 성능이 좋아졌지만, 생성 모델은 아직 probabilistic 모델들을 근사하는 데 있어서 어려움이 있었고, RELU와 같은 함수의 장점을 최대화해서 이용하기에는 한계가 있었다. 그래서 이러한 어려움을 극복하고자 adversarial nets이라는 하나의 새로운 framework를 제안한다.

1. Adversarial Nets: competition

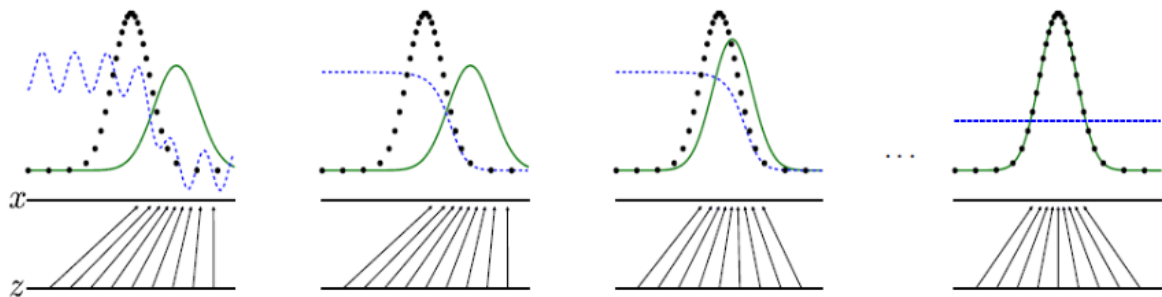
두 개의 모델을 적대적으로 경쟁시키며 발전하는 생성 모델이고, Generative model과 Discriminative model로 구성되어 있다. 이 모델은 비유하자면 위조지폐를 만드는 범죄자(Generator)와 위조지폐를 감별하는 경찰(Discriminator)의 싸움이다. 위조지폐범은 위조지폐를 진짜처럼 만드는 게 목표이고, 경찰은 진짜지폐와 위조지폐를 잘 구분하는 게 목표이다. 이렇듯, 위조지폐범과 경찰을 각각 Deep neural network로 두고 적절한 Loss로 서로 반대되는 목표를 학습하도록 만든 것이 GAN의 아이디어이다. 적절한 Loss는 우리가 익히 알고 있는 2개의 class를 구분하는 방법 즉, binary cross entropy를 이용하면 된다. Loss of GAN은 다음과 같다.

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

이 식에서는 D(): Discriminator, G(): Generator, x: real image, z: latent vector로 설정되어 있다. 위 방정식을 D의 입장, G의 입장에서 각각 이해해보면,

- D가 매우 뛰어난 성능으로 판별을 잘 해낸다고 했을 때, $D(\mathbf{x})=1$, $D(G(\mathbf{z}))=0$ 이 되어 전체 식 $V(D, G)=\log(1)+\log(1-0)=0+0=0$ 이 된다. 즉 D의 입장에서 얻을 수 있는 이상적인 결과, '최댓값'은 0임을 확인할 수 있다.
- G가 D가 구별하지 못할 만큼 진짜와 같은 데이터를 잘 생성해낸다고 했을 때, $D(G(\mathbf{z}))=1$ 이 되어 전체 식 $V(D, G)=\log(1-1)=\log(0)=(-\infty)$ 가 된다. 즉 G의 입장에서 얻을 수 있는 이상적인 결과, '최솟값'은 -무한대임을 확인할 수 있다.

다시 말해, D입장에서는 V를 최대화하려 하고, G입장에서는 V를 최소화하려 하고, 논문에서는 D와 G를 V를 갖는 two-player minmax game으로 표현했다. 또한, GAN의 학습과정을 그림을 통해 확인해보면 다음과 같다.



여기서 파란색 점선: discriminative distribution, 검은색 점선: real data distribution(P-data), 녹색 실선: generative distribution(P-g)이다. 이 그림을 설명해보자면 z에서 샘플링 과정을 거쳐서 생성자 x에 mapping 시키고 이를 위 목적 함수를 이용하여 점점 x의 이미지와 동일하게 만드는 과정이라고 할 수 있다. 이산적으로 뽑은 검은색 점선 즉, 원래 이미지 바로 옆에 연속적인 녹색 실선에서의 하나 값을 뽑으면 새로운 이미지가 생성되는 것이다. 그리고 위 그림에서 파란색 점선은 어느 한 값으로 수렴하게 되는데 이 값은 1/2이다. 밑에서 증명하겠지만 P-g와 P-data가 같아질 때의 값인데 이는 생성자가 내놓은 이미지가 실제 이미지와 구분할 수 없다는 것을 의미한다.

2. Theoretical Results

논문에서는 이 부분에 관하여 2가지를 증명하였다.

- Global Optimality of P-data

✓ G가 원본데이터를 학습할 수 있다는 것에 대한 명제와 그에 따른 증명이다.

명제, $D_G^*(x) = \frac{P_{data}(x)}{P_{data}(x) + P_g(x)}$	$E[x] = \int_x x f(x) dx$
증명, $V(D, G) = E_{x \sim P_{data}(x)}[\log D(x)] + E_{z \sim P_g(z)}[\log(1 - D(G(z)))]$: 앞서 정의하였던 목적함수
$= \int_x P_{data}(x) \log D(x) dx + \int_z P_g(z) \log(1 - D(G(z))) dz$: 연속확률분포의 기대값 공식
$= \int_x P_{data}(x) \log D(x) + P_g(x) \log(1 - D(x)) dx$: z는 x에 매핑되기 때문
이 식의 형태는 $f(y) = a \log(y) + b \log(1-y)$ 이고, 극대값은 $\frac{a}{a+b}$ 임.	
따라서 극대값 명제 $D_G^*(x)$ 를 가질 수 있게 됨.	

✓ 위에서 D의 극대값을 구했으니 D가 optimal한 포인트를 가질 때의 G의 값을 구하는 증명이다.

$$\text{증명, } C(G) = \max_D V(G, D)$$

$$= E_{x \sim P_{\text{data}}} [\log D_G^*(x)] + E_{z \sim P_z} [\log (1 - D_G^*(G(z)))]$$

$$= E_{x \sim P_{\text{data}}} [\log D_G^*(x)] + E_{x \sim P_g} [\log (1 - D_G^*(x))] \quad \because z \text{는 } x \text{에 매핑되기 때문}$$

$$= E_{x \sim P_{\text{data}}} \left[\log \frac{P_{\text{data}}(x)}{P_{\text{data}}(x) + P_g(x)} \right] + E_{x \sim P_g} \left[\log \frac{P_g(x)}{P_{\text{data}}(x) + P_g(x)} \right]$$

$$= E_{x \sim P_{\text{data}}} \left[\log \frac{2 \times P_{\text{data}}(x)}{P_{\text{data}}(x) + P_g(x)} \right] + E_{x \sim P_g} \left[\log \frac{2 \times P_g(x)}{P_{\text{data}}(x) + P_g(x)} \right] - \log(4)$$

$$= \text{KL} \left(P_{\text{data}} \parallel \frac{P_{\text{data}} + P_g}{2} \right) + \text{KL} \left(P_g \parallel \frac{P_{\text{data}} + P_g}{2} \right) - \log(4)$$

$$= 2 \times \text{JSD}(P_{\text{data}} \parallel P_g) - \log(4)$$

JSD로 변환해주는 이유는 distance matrix 형태

바꿔주기 위함이고, minimize가 되기 위한 조건은

$P_{\text{data}} = P_g$. 따라서 P_g 와 P_{data} 가 같을 때 $C(G)$ 의 global minimize를 구할 수 있고 이 값은 $-\log(4)$ 가 됨.

$$\text{KL}(P_{\text{data}} \parallel P_g) = \int_{-\infty}^{\infty} P_{\text{data}}(x) \log \left(\frac{P_{\text{data}}(x)}{P_g(x)} \right) dx$$

$$\text{JSD}(P \parallel Q) = \frac{1}{2} \text{KL} \left(P \parallel \frac{P+Q}{2} \right) + \frac{1}{2} \text{KL} \left(Q \parallel \frac{P+Q}{2} \right)$$

- Convergence of Algorithm

✓ P_g 가 G 로 수렴할 수 있는지에 대한 증명이다.

증명, $V(G, D) = U(P_g, D)$: V 의 함수는 앞에서 봤듯이, convex function이기 때문에 미분이 가능함.

: 함수 V 는 P_g 에 있어서 선형함수이기 때문에 P_g 는 G 로 수렴할 수 있음.

3. Experiments

평가지표는 kernel method 중 대표적인 방법인 Parzen window-based log-likelihood estimate를 사용하였고, 이전의 모델들과 비교했을 때의 결과는 오른쪽과 같다. Adversarial nets가 충분히 competitive한 성능을 보임을 확인할 수 있다.

Model	MNIST	TFD
DBN [3]	138 ± 2	1909 ± 66
Stacked CAE [3]	121 ± 1.6	2110 ± 50
Deep GSN [6]	214 ± 1.1	1890 ± 29
Adversarial nets	225 ± 2	2057 ± 26



왼쪽은 GAN 결과이다. 맨 끝 줄에 노란색 박스로 쳐진 부분은 실제 이미지이고 나머지 부분이 GAN으로 생성된 이미지이다.

4. How to evaluate GANs?

GAN(Quality, Diversity)의 몇 가지 평가지표를 검색하고 이를 리뷰하고자 한다.

- Frechet Inception Distance (FID)

FID는 생성된 이미지의 분포와 원래 이미지의 분포가 얼마나 비슷한지를 측정하는 지표이다. 이미지는 고차원 공간이므로 간단하게 분포의 거리를 계측하는 것은 힘들다. 따라서, 이미지 인식 정밀도를 가진 모델을 사용하여 이미지를 저차원 공간으로 변경한 뒤, 그 공간에서 분포의 거리를 구하는 것이 FID의 기본 컨셉이다. 정의식은 다음과 같다.

$$\|m - m_w\|_2^2 + \text{Tr}(C + C_w - 2(CC_w)^{1/2})$$

이 식에서 m : 평균 벡터, c : 공분산 행렬, w 첨자: 생성 이미지, w 첨자 x : 실제 이미지로 설정되어 있다. Inception V3라는 네트워크를 사용하여 이미지를 벡터로 변경한 뒤, Wasserstein-2 거리를 계산한다. 이는 분산의 거리이므로, 값이 작을수록 실제 이미지에 가까운 가짜 이미지가 생성되었다고 할 수 있다. 즉, 값이 작아야 Generator 성능이 좋다고 할 수 있다.

- Perceptual Path Length (PPL)

PPL은 지각적으로 즉 우리가 감각적으로 봤을 때 잠재공간 상에 이미지가 부드럽게 변화하고 있는지를 측정하는 지표이다. FID와 동일하게 학습이 끝난 모델로 생성된 이미지의 거리를 사용한다. 정의식은 다음과 같다.

$$l_w = \mathbb{E} \left[\frac{1}{\epsilon^2} d(g(\text{lerp}(f(\mathbf{z}_1), f(\mathbf{z}_2); t)), g(\text{lerp}(f(\mathbf{z}_1), f(\mathbf{z}_2); t + \epsilon))) \right]$$

이 식에서 g : Generator, f : StyleGAN에 있어서 잠재변수 z 를 스타일 w 로 한 함수, lerp : 선형 보안을 의미한다. 위 식을 설명해보자면 지각적인 거리를 수치화한 것으로 2개의 잠재공간 z_1, z_2 를 비율 t 로 섞은 잠재변수로 생성한 이미지와 비율 $t+\epsilon$ 으로 섞은 잠재변수로 생성한 이미지 거리의 기대치라고 할 수 있다. t 로 섞은 데이터와 $t+\epsilon$ 으로 섞은 데이터가 지각적으로 가까우면 작은 값이 도출된다. 즉, 값이 작아야 Generator가 잘 작동하고 있다고 말할 수 있다.

5. Conclusion

본 논문은 Adversarial Learning이라는 개념을 생성모델에 처음 적용한 새로운 framework를 제안했다. Markov Chain method나 Approximate Inference등에 의존하는 이전 연구들과 달리, generator와 discriminator 두 모델을 경쟁적으로 학습시켜 둘 모두를 동시에 최적화하는 방법을 사용하여 데이터의 분포에 대한 명시적인 가정없이 데이터 생성이 가능하도록 했다. 이러한 전략은 다양한 함수들이 모델에 접목될 수 있도록 했으며 매우 많은 후속 연구들을 유발하고, 많은 변형 모델들을 통해 이미지 생성 task에서 SOTA로 군림한다.

[Reference]

- <https://arxiv.org/pdf/1406.2661.pdf>
- <https://www.youtube.com/watch?v=AVvIDmhHgC4>
- <https://csm-kr.tistory.com/31>
- <https://rpubs.com/Statdoc/204928>
- <https://www.puzzledata.com/blog190423/>
- <https://www.youtube.com/watch?v=cBgX-8lxRUM>
- <https://bigdata-analyst.tistory.com/264>
- <https://process-mining.tistory.com/95>
- <https://engineer-mole.tistory.com/269>