생성적 적대적 네트워크

J. Goodfellow* Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair ‡
, , 아론 쿠르빌, 요슈아 벤지오 §
컴퓨터 과학 및 운영 연구부 Universite de Montreal Montreal, QC H3C 3J7

추상적인

우리는 적대적 과정을 통해 생성 모델을 추정하기 위한 새로운 프레임워크를 제안합니다. 이 프레임워크에서는 데이터 분포를 캡처하는 생성 모델 G와 샘플이 훈련 데이터에서 나올 확률을 추정하는 판별 모델 D의 두 모델을 동시에 훈련합니다. G에 대한 훈련 절차는 D가 실수할 확률을 최대화하는 것입니다. 이 프레임워크는 minimax 2인용 게임에 해당합니다. 임의 함수 G와 D의 공간에서 G는 훈련 데이터 분포를 복구하고 D는 모든 곳에서 동일한 고유한 솔루션이 존재합니다. G와 D가 다층 퍼셉트론으로 정의되는 경우 전체 시스템을 역전파로 훈련할 수 있습니다.

1 2

샘플을 훈련하거나 생성하는 동안 Markov 체인이나 풀린 근사 추론 네트워크가 필요하지 않습니다. 실험은 생성된 샘플의 정성적 및 정량적 평가를 통해 프레임워크의 잠재력을 입증합니다.

1. 소개

딥 러닝의 약속은 자연 이미지, 음성을 포함하는 오디오 파형, 자연어 말뭉치의 기호와 같은 인공 지능 응용 프로그램에서 발생하는 데이터 종류에 대한 확률 분포를 나타내는 풍부한 계층적 모델[2]을 발견하는 것입니다. 지금까지 딥 러닝에서 가장 눈에 띄는 성공은 식별 모델과 관련이 있으며 일반적으로 고차원의 풍부한 감각 입력을 클래스 레이블에 매핑하는 모델입니다[14, 20]. 이러한 눈에 띄는 성공은 특히 잘 작동하는 그래디언트가 있는 조각별 선형 단위[17, 8, 9]를 사용하는 역전파 및 드롭아웃 알고리즘에 주로 기반을 두고 있습니다. 심층 생성 모델은 최대 우도 추정 및 관련 전략에서 발생하는 많은 다루기 힘든 확률 계산을 근사화하는 것이 어렵고 생성 컨텍스트에서 조각별 선형 단위의 이점을 활용하기 어렵기 때문에 영향이 적습니다. 우리는 이러한 어려움을 피하는 새로운 생성 모델 추정 절차를 제안합니다.

1

제안된 적대적 네트워크 프레임워크에서 생성 모델은 적과 맞서게 됩니다. 샘플이 모델 분포에서 나온 것인지 아니면 데이터 분포에서 나온 것인지를 결정하는 방법을 배우는 차별적 모델입니다. 생성 모델은 위조 화폐를 만들어 발각되지 않고 사용하려는 위조 팀과 유사하다고 생각할 수 있으며, 판별 모델은 위조 화폐를 탐지하려는 경찰과 유사합니다. 이 게임의 경쟁은 위조품이 진품과 구별될 수 없을 때까지 양 팀이 방법을 개선하도록 유도합니다.

Ian Goodfellow는 현재 Google의 연구 과학자이지만 이전에 UdeM 학생일 때 이 작업을 수행했습니다. † Jean Pouget-Abadie는 Ecole Polytechnique에서 Universite de Montreal을 방문하는 동안 이 작업을 수행했습니다.

Sherjil Ozair는 Indian Institute of Technology Delhi에서 Universite de Montreal을 방문합니다. § Yoshua Bengio는 CIFAR 선임 연구원입니다.

1모든 코드 및 하이퍼파라미터는 http://www.github.com/goodfeli/adversarial에서 사용 가능

이 프레임워크는 다양한 종류의 모델 및 최적화 알고리즘에 대한 특정 학습 알고리즘을 생성할 수 있습니다. 이 글에서는 생성 모델이 다층 퍼셉트론에 랜덤 노이즈를 통과시켜 샘플을 생성하는 특수한 경우와 판별 모델도 다층 퍼셉트론인 경우를 살펴봅니다. 우리는 이 특별한 경우를 adversarial net이라고 부릅니다. 이 경우 매우 성공적인 역전파 및 드롭아웃 알고리즘[16]만 사용하여 두 모델을 훈련하고 순방향 전파만 사용하여 생성 모델에서 샘플링할 수 있습니다. 근사 추론이나 Markov 체인이 필요하지 않습니다.

2 관련 업무

최근까지 대부분의 심층 생성 모델은 확률 분포 함수의 매개변수 사양을 제공하는 모델에 초점을 맞췄습니다. 그런 다음 로그 우도를 최대화하여 모델을 훈련할 수 있습니다. 이 모델군에서 아마도 가장 성공적인 모델은 deep Boltzmann 머신일 것입니다[25]. 이러한 모델은 일반적으로 다루기 힘든 우도 함수를 가지고 있으므로 우도 구배에 대한 수많은 근사치가 필요합니다. 이러한 어려움은 가능성을 명시적으로 나타내지 않지만 원하는 분포에서 샘플을 생성할 수 있는 "생성 기계" 모델의 개발에 동기를 부여했습니다. 생성 확률 네트워크[4]는 Boltzmann 기계에 필요한 수많은 근사값이 아닌 정확한 역전파로 훈련할 수 있는 생성 기계의 예입니다. 이 작업은 생성 확률 네트워크에서 사용되는 Markov 체인을 제거하여 생성 기계의 아이디어를 확장합니다.

우리의 작업은 다음과 같은 관찰을 사용하여 생성 프로세스를 통해 파생 상품을 역전파합니다.

$$\lim_{\sigma \to 0} \nabla_x \mathbb{E}_{\epsilon \sim N(0,\sigma^2 I)} f(x+\epsilon) = \nabla_x f(x).$$

Kingma and Welling [18] 및 Rezende et al. [23]은 보다 일반적인 확률적 역전파 규칙을 개발하여 유한 분산이 있는 가우시안 분포를 통해 역전파하고 평균뿐만 아니라 공분산 매개변수로 역전파할 수 있도록 했습니다. 이러한 역전파 규칙을 통해 우리는 이 작업에서 하이퍼 매개변수로 취급한 생성기의 조건부 분산을 학습할 수 있습니다. Kingma 및 Welling [18] 및 Rezende et al. [23] VAE(variational autoen coder)를 훈련하기 위해 확률적 역전파를 사용합니다. 생성적 적대 신경망과 마찬가지로 변이 자동 인코더는 미분 가능한 생성기 신경망과 두 번째 신경망을 연결합니다. 생성적 적대 네트워크와 달리 VAE의 두 번째 네트워크는 근사 추론을 수행하는 인식 모델입니다. GAN은 눈에 보이는 단위를 통한 구분이 필요하므로 이산 데이터를 모델링할 수 없으며, VAE는 숨겨진 단위를 통한 구분이 필요하므로 이산 잠재 변수를 가질 수 없습니다. 접근 방식과 같은 다른 VAE가 존재하지만 [12, 22] 우리 방법과 덜 밀접하게 관련되어 있습니다.

이전 작업도 생성 모델을 훈련하기 위해 판별 기준을 사용하는 접근 방식을 취했습니다[29, 13]. 이러한 접근 방식은 심층 생성 모델에서 다루기 힘든 기준을 사용합니다. 이러한 방법은 확률을 낮추는 변동 근사를 사용하여 근사할 수 없는 확률의 비율을 포함하기 때문에 심층 모델에 대해 근사하기조차 어렵습니다. Noise-contrastive estimation (NCE) [13]은 모델이 고정된 노이즈 분포에서 데이터를 구별하는 데 유용한 가중치를 학습하여 생성 모델을 훈련하는 것과 관련됩니다. 이전에 훈련된 모델을 노이즈 분포로 사용하면 품질이 향상되는 일련의 모델을 훈련할 수 있습니다. 이는 적대적 네트워크 게임에서 사용되는 공식 경쟁과 정신이 유사한 비공식 경쟁 메커니즘으로 볼 수 있습니다. NCE의 주요 제한 사항은 "판별자"가 노이즈 분포와 모델 분포의 확률 밀도 비율로 정의되므로 두 밀도를 통해 평가하고 역전파할 수 있는 기능이 필요하다는 것입니다.

일부 이전 작업에서는 두 개의 신경망이 경쟁하는 일반적인 개념을 사용했습니다. 가장 관련성이 높은 작업은 예측 가능성 최소화[26]입니다. 예측 가능성 최소화에서 신경망의 각 숨겨진 유닛은 다른 모든 숨겨진 유닛의 값이 주어진 숨겨진 유닛의 값을 예측하는 두 번째 네트워크의 출력과 다르게 훈련됩니다. 이 작업은 세 가지 중요한 방식에서 예측 가능성 최소화와 다릅니다. 1) 이 작업에서 네트워크 간의 경쟁은 유일한 교육 기준이며 자체적으로 네트워크를 교육하기에 충분합니다. 예측 가능성 최소화는 신경망의 숨겨진 단위가 다른 작업을 수행하는 동안 통계적으로 독립적이도록 권장하는 정규화 장치일 뿐입니다. 기본 교육 기준이 아닙니다.

2) 경쟁의 성격이 다릅니다. 예측 가능성 최소화에서 두 네트워크의 출력을 비교합니다. 한 네트워크는 출력을 유사하게 만들고 다른 네트워크는 유사하게 만들려고 합니다.

출력이 다릅니다. 문제의 출력은 단일 스칼라입니다. GAN에서 한 네트워크는 다른 네트워크의 입력으로 사용되는 풍부한 고차원 벡터를 생성하고 다른 네트워크가 처리 방법을 모르는 입력을 선택하려고 시도합니다. 3) 학습 과정의 사양이 다릅니다. 예측 가능성 최소화는 목적 함수를 최소화하는 최적화 문제로 설명되며 학습은 목적 함수의 최소값에 접근합니다. GAN은 최적화 문제가 아닌 미니맥스 게임을 기반으로 하며, 한 에이전트는 최대화하고 다른 에이전트는 최소화하려는 가치 함수를 가지고 있습니다. 게임은 한 플레이어의 전략과 관련하여 최소이고 다른 플레이어의 전략과 관련하여 최대인 안장 지점에서 종료됩니다.

Generative adversarial network는 때때로 "adversar ial examples"[28]의 관련 개념과 혼동되었습니다. 적대적인 예는 아직 잘못 분류된 데이터와 유사한 예를 찾기 위해 분류 네트워크에 대한 입력에서 직접 그래디언트 기반 최적화를 사용하여 찾은 예입니다. 이것은 적대적 예제가 생성 모델을 훈련시키는 메커니즘이 아니기 때문에 현재 작업과 다릅니다. 대신, 적대적인 예는 주로 신경망이 흥미로운 방식으로 동작한다는 것을 보여주기 위한 분석 도구이며, 인간 관찰자가 두 이미지 사이의 차이를 인지할 수 없을지라도 높은 신뢰도를 가지고 두 이미지를 자신 있게 다르게 분류하는 경우가 많습니다. 그러한 적대적 사례의 존재는 생성적 적대적 네트워크 훈련이 비효율적일 수 있음을 시사합니다. 왜냐하면 현대의 차별적 네트워크가 해당 클래스의 인간이 인지할 수 있는 속성을 에뮬레이트하지 않고 자신 있게 클래스를 인식하도록 만드는 것이 가능함을 보여주기 때문입니다.

3 적대적 네트워크

적대적 모델링 프레임워크는 모델이 둘 다 다층 퍼셉트론일 때 적용하기에 가장 간단합니다. 데이터 x에 대한 생성기의 분포 pg를 학습하기 위해 입력 노이즈 변수 pz(z)에 대한 사전을 정의한 다음 데이터 공간에 대한 매핑을 G(z; θg)로 나타냅니다. 여기서 G는 다층 퍼셉트론으로 표현되는 미분 가능 함수입니다. 파라미터 θg. 단일 스칼라를 출력하는 두 번째 다층 퍼셉트론 D(x; θd) 도 정의합니다. D(x)는 x가 pg가 아닌 데이터에서 나온 확률을 나타냅니다 . 교육 예제와 G의 샘플 모두에 올바른 레이블을 할당할 확률을 최대화하도록 D를 교육합니다. 동시에 log(1 D(G(z)))를 최소화하도록 G를 교육합니다. 즉, D와 G는 가치 함수 V(G, D)를 사용하여 다음과 같은 2인 미니맥스 게임을 합니다. min G

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

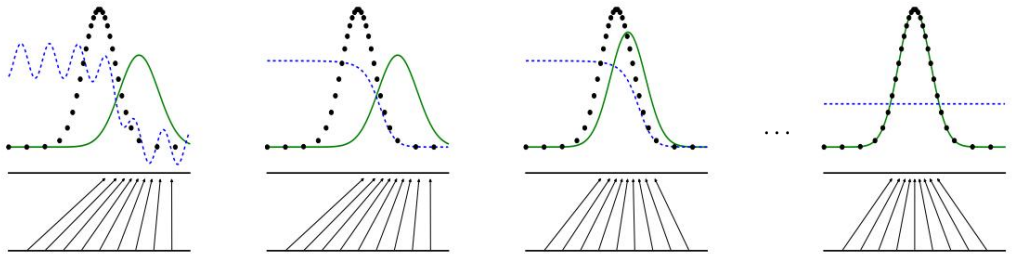
다음 섹션에서는 G와 D에 충분한 용량, 즉 비모수적 한계가 주어졌을 때 훈련 기준이 데이터 생성 분포를 복구할 수 있음을 본질적으로 보여주는 적대적 네트워크의 이론적 분석을 제시합니다. 접근 방식에 대한 덜 공식적이고 교육적인 설명은 그림 1을 참조하십시오. 실제로는 반복적이고 수치적인 접근 방식을 사용하여 게임을 구현해야 합니다. 교육의 내부 루프에서 D를 완료하도록 최적화하는 것은 계산적으로 금지되어 있으며 유한 데이터 세트에서는 과적합이 발생합니다. 대신, D를 최적화하는 k 단계와 G를 최적화하는 한 단계 사이를 번갈아 가며 수행합니다. 그 결과 G가 충분히 느리게 변경되는 한 D가 최적 솔루션에 가깝게 유지됩니다. 절차는 공식적으로 알고리즘 1에 제시되어 있습니다.

실제로 방정식 1은 G가 잘 학습하기에 충분한 기울기를 제공하지 않을 수 있습니다. 학습 초기에 G가 좋지 않을 때 D는 훈련 데이터와 분명히 다르기 때문에 높은 신뢰도를 가지고 샘플을 거부할 수 있습니다. 이 경우 log(1 D(G(z)))는 포화됩니다. log(1 D(G(z)))를 최소화하도록 G를 교육하는 대신 log D(G(z))를 최대화하도록 G를 교육할 수 있습니다. 이 목적 함수는 G와 D 동역학의 동일한 고정점을 생성하지만 학습 초기에 훨씬 더 강한 기울기를 제공합니다.

4 이론적 결과

생성기 G는 암시적으로 확률 분포 pg를 z pz일 때 얻은 샘플 G(z)의 분포로 정의합니다 . 따라서 충분한 용량과 훈련 시간이 주어지면 알고리즘 1이 pdata 의 좋은 추정기로 수렴되기를 바랍니다 . 이 섹션의 결과는 비모수적 설정에서 수행됩니다. 예를 들어 확률 밀도 함수 공간에서 수렴을 연구하여 무한한 용량을 가진 모델을 나타냅니다.

섹션 4.1에서 이 minimax 게임이 pg = pdata 에 대한 전역 최적임을 보여줍니다 . 그런 다음 섹션 4.2에서 알고리즘 1이 방정식 1을 최적화하여 원하는 결과를 얻는 것을 보여줍니다.



(ㅏ) (비) (씨) (디)

그림 1: Generative adversarial net 은 판별 분포(D, 파란색, 파선)를 동시에 업데이트하여 데이터 생성 분포(검은색, 점선) px의 샘플과 생성 분포 pg (G) 의 샘플을 구별하도록 학습됩니다. (녹색, 실선). 아래쪽 수평선은 z 가 샘플링되는 도메인입니다 (이 경우 균일하게). 위의 수평선은 x 도메인의 일부입니다. 위쪽 화살표는 매핑 x = G(z)가 변환된 샘플에 대해 불균일 분포 pg를 부과하는 방법을 보여줍니다 . G는 고밀도 영역에서 수축하고 저밀도 pg 영역에서 확장합니다. (ㅏ)

수렴에 가까운 적대적 쌍을 고려하십시오. pg 는 pdata 와 유사 하고 D는 부분적으로 정확한 분류기입니다. (x) = (b) 알고리 부 루프에서 D는 D pdata(x)+pg(x) 로 수렴하여 데이터에서 샘플을 구별하도록 훈련됩니다 . (c) G에 대한 업데이트 후 경사 pdata(x) 디언트는 G(z)가 데이터로 분류될 가능성이 더 높은 영역으로 흐르도록 안내했습니다 . (d) 여러 단계의 훈련 후 G와 D에 충분한 용량이 있으면 pg = pdata 이기 때문에 둘 다 개선할 수 없는 지점에 도달합니다 . 판별자는 두 분포를 구별할 수 없습니다. 즉, D(x) =

T2.

알고리즘 1 생성적 적대 신경망의 Minibatch 확률적 경사 하강 훈련. 판별자에 적용할 단계 수 k는 하이퍼 매개변수입니다. 실험에서 가장 저렴한 옵션인 k = 1을 사용했습니다. 교육 반복 횟수에 대해 k 단계 에 대해 수행 • m 노이즈 샘플의 샘플 미니배치 {z (1) • m 예의 샘플 미니배치 {x (1) pdata(x). • 확률적 기울기를 상승시켜 판별자를 업데이트합니다.

, . . . , pg(z) 이전의 잡음에서 z (m)} . 데이
, . . . , 터 생성 분포에서 x (m)}

$$\nabla_{\theta_d} \frac{1}{중} \sum_{i=1}^{중} \log D\, x^{(i)} + \log\left(1 - D(G(z^{(i)}))\right) .$$

끝

• m 노이즈 샘플의 샘플 미니 배치 {z (1) • 확률적 기울기 , . . . , pg(z) 이전의 잡음에서 z (m)} . 를 내림으로써 생성기를 업데이트합니다.

$$\nabla_{\theta_g} \frac{1}{나} \sum_{는=1}^{중} \log\left(1 - D(G(z^{(i)}))\right) .$$

end for

그래디언트 기반 업데이트는 모든 표준 그래디언트 기반 학습 규칙을 사용할 수 있습니다. 우리는 실험에서 운동량을 사용했습니다.

4.1 pg = pdata 의 전역 최적성

먼저 주어진 생성기 G에 대해 최적의 판별기 D를 고려합니다.

명제 1. G가 고정된 경우 최적 판별기 D는 다음과 같습니다.

$$D^*_G(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)} \tag{2}$$

증거. 생성자 G가 주어졌을 때 판별자 D에 대한 훈련 기준은 양 V(G, D)를 최대화하는 것입니다.

$$V(G, D) = \int_x pdata(x) \log(D(x))dx + \int_z pz(z) \log(1 - D(g(z)))dz$$

$$= \int_x pdata(x) \log(D(x)) + pg(x) \log(1 - D(x))dx \quad (3)$$

임의의 $(a, b) \in \mathbb{R}^2 \setminus \{0, 0\}$에 대해 함수 $y \to a \log(y) + b \log(1 - y)$는 $[0, 1]$에서 최대값을 달성합니다. $\frac{a}{a+b}$. 판별자는 Supp(pdata) ∪ Supp(pg) 외부에서 정의할 필요가 없으며 , 증명을 마칩니다. □

D에 대한 훈련 목표는 조건부 확률 P(Y = y|x)를 추정하기 위한 로그 우도를 최대화하는 것으로 해석될 수 있습니다. 여기서 Y는 x가 pdata에서 오는지(y = 1) 또는 pg 에서 오는지 (y = 0). 방정식의 미니맥스 게임. 1은 이제 다음과 같이 공식화할 수 있습니다. C(G) = 최대 V(G, D)

$$= Ex \sim pdata [\log D^*_G(x)] + Ez \sim pz [\log(1 - D^*_G(G(z)))] \quad (4)$$

$$= Ex \sim pdata [\log D^*_G(x)] + Ex \sim pg [\log(1 - D^*_G(x))]$$

$$= Ex \sim pdata \log \frac{pdata(x)}{pdata(x) + pg(x)} + Ex \sim pg \log \frac{pg(x)}{pdata(x) + pg(x)}$$

정리 1. 가상 훈련 기준 C(G)의 전역 최소값은 pg = pdata 인 경우에만 달성됩니다 . 그 시점에서 C(G)는 값 - log 4를 얻습니다.

증거. pg = pdata, $D^*_G(x) = \frac{1}{2}$, (식 2 고려). 따라서 Eq. 4 at $D^*_G(x) = \frac{1}{2}$ 우리 $-\log 4$. 이것이 가능한 최상의 pg = pdata인 경우에만 $-\log$ 를 찾으십시오 C(G) 값인지 확인하기 위해 도달

$$Ex \sim pdata [-\log 2] + Ex \sim pg [-\log 2] = -\log 4$$

C(G) = V(D^*_G, G)에서 이 식을 빼면 G, G), 우리는 다음을 얻는

$$C(G) = -\log(4) + KL\left(pdata \,\middle\|\, \frac{pdata + pg}{2}\right) + KL\left(pg \,\middle\|\, \frac{pdata + pg}{2}\right) \quad (5)$$

여기서 KL은 Kullback–Leibler 분기입니다. 이전 식에서 모델 분포와 데이터 생성 프로세스 간의 Jensen-Shannon 분기를 인식합니다.

$$C(G) = -\log(4) + 2 \cdot JSD(pdata \,\|\, pg) \quad (6)$$

두 분포 사이의 Jensen-Shannon 발산은 항상 음수가 아니고 동일하면 0이므로 $C^* = -\log(4)$가 C(G)의 전역 최소값이고 유일한 솔루션은 pg = pdata, 즉 데이터 분포를 완벽하게 복제하는 생성 모델입니다. □

4.2 알고리즘의 융합 1

명제 2. G와 D의 용량이 충분하고 알고리즘 1의 각 단계에서 판별기는 주어진 G에 최적값에 도달하도록 허용하고 기준을 개선하기 위해 pg 를 업데이트합니다.

$$Ex \sim pdata [\log D_G(x)] + Ex \sim pg [\log(1 - D_G(x))]$$

then pg 는 pdata 로 수렴

증거. V(G, D) = U(pg, D)를 위의 기준에서 수행된 pg 의 함수로 고려하십시오. U(pg, D)는 pg에서 볼록합니다. 볼록 함수의 상한의 하위 도함수는 최대값에 도달하는 지점에서 함수의 도함수를 포함합니다. 즉, $f(x) = \sup_{\alpha \in A} f_\alpha(x)$ 및 $f_\alpha(x)$가 모든 α에 대해 x에서 볼록한 경우 $\beta = \arg\sup_{\alpha \in A} f_\alpha(x)$ 인 경우 $\partial f_\beta(x) \in \partial f$.

이는 상응하는 G가 주어진 최적 D에서 pg 에 대한 경사 하강법 업데이트를 계산하는 것과 같습니다. supD U(pg, D) 는 Thm 1에서 입증된 고유한 전역 최적값으로 pg 에서 볼록 하므로 pg 의 충분히 작은 업데이트가 있습니다. , pg는 px 로 수렴하여 증명을 마칩니다. □

실제로 adversarial net은 함수 G(z; θg)를 통해 pg 분포의 제한된 계열을 나타내며 pg 자체 가 아닌 θg를 최적화하므로 증명이 적용되지 않습니다. 그러나 실제로 다층 퍼셉트론의 뛰어난 성능은 이론적 보장이 부족함에도 불구하고 사용하기에 합리적인 모델임을 시사합니다.

모델	MNIST	TFD
DBN [3]	138 ± 2	1909 ± 66
누적 CAE [3]	121 ± 1.6	2110 ± 50
적대적 네트워크	225 ± 2	2057 ± 26

표 1: Parzen 창 기반 로그 우도 추정치. MNIST에 보고된 숫자는 테스트 세트에 있는 샘플의 평균 로그 우도이며 예제 전체에서 계산된 평균의 표준 오차입니다. TFD에서 각 폴드의 검증 세트를 사용하여 다른 σ를 선택하여 데이터 세트의 폴드 전체에 걸쳐 표준 오차를 계산했습니다. TFD에서 각 접기에서 σ를 교차 검증하고 각 접기에서 평균 로그 우도를 계산했습니다.

MNIST의 경우 데이터 세트의 실제 값(이진이 아닌) 버전의 다른 모델과 비교합니다.

5 실험

우리는 MNIST[21], Toronto Face Database(TFD)[27], CIFAR-10[19]을 포함한 다양한 데이터 세트에서 적대적 네트워크를 훈련했습니다. 생성기 네트워크는 정류기 선형 활성화[17, 8]와 시그모이드 활성화의 혼합을 사용했으며 판별기 네트워크는 최대 활성화[9] 활성화를 사용했습니다. Dropout [16]은 discriminator net을 훈련하는데 적용되었습니다. 우리의 이론적 프레임워크는 생성기의 중간 레이어에서 드롭아웃 및 기타 노이즈의 사용을 허용하는 반면 생성기 네트워크의 최하위 레이어에 대한 입력으로만 노이즈를 사용했습니다.

G로 생성된 샘플에 Gaussian Parzen 창을 맞추고 이 분포에서 로그 우도를 보고하여 p_g 에서 테스트 세트 데이터의 확률을 추정합니다 . 가우시안의 σ 파라미터는 유효성 검사 세트에 대한 교차 유효성 검사를 통해 얻었습니다. 이 절차는 Breuleux et al. [7] 정확한 우도를 다루기 어려운 다양한 생성 모델에 사용됩니다[24, 3, 4]. 결과는 표 1에 보고되어 있습니다. 우도를 추정하는 이 방법은 분산이 다소 높고 고차원 공간에서 잘 수행되지 않지만 우리가 아는 한 최선의 방법입니다. 샘플링할 수 있지만 가능성을 추정할 수 없는 생성 모델의 발전은 그러한 모델을 평가하는 방법에 대한 추가 연구에 직접적인 동기를 부여합니다. 그림 2와 3에서는 훈련 후 생성기 네트워크에서 추출한 샘플을 보여줍니다. 우리는 이러한 샘플이 기존 방법으로 생성된 샘플보다 낫다고 주장하지는 않지만 이러한 샘플이 문헌의 더 나은 생성 모델과 적어도 경쟁적이며 적대적 프레임워크의 잠재력을 강조한다고 믿습니다.

6 장점과 단점

이 새로운 프레임워크는 이전 모델링 프레임 작업에 비해 장단점이 있습니다. 단점은 주로 $p_g(x)$의 명시적 표현이 없고 훈련 중에 D가 G와 잘 동기화되어야 한다는 것입니다(특히 "Helvetica 시나리오"를 피하기 위해 D를 업데이트하지 않고 G를 너무 많이 훈련해서는 안 됩니다) 여기서 G는 너무 많은 z 값을 동일한 x 값으로 축소하여 pdata를 모델링하기에 충분한 다양성을 갖 습니다. 이점은 Markov 체인이 전혀 필요하지 않고 기울기를 얻기 위해 역전파만 사용되며 학습 중에 추론이 필요하지 않으며 다양한 기능을 모델에 통합할 수 있다는 것입니다. 표 2는 Generative adversarial net과 다른 생성 모델링 접근법의 비교를 요약한 것입니다.

앞서 언급한 이점은 주로 계산적입니다. 적대적 모델은 데이터 예제로 직접 업데이트되지 않고 판별기를 통해 흐르는 기울기로만 업데이트되는 생성기 네트워크로부터 통계적 이점을 얻을 수도 있습니다. 이는 입력의 구성요소가 생성기의 매개변수에 직접 복사되지 않음을 의미합니다. 적대적 네트워크의 또 다른 장점은 매우 날카롭고 심지어 변질된 분포를 나타낼 수 있다는 것입니다. 반면 Markov 체인을 기반으로 하는 방법은 체인이 모드 간에 혼합될 수 있도록 분포가 다소 흐릿해야 합니다.

7 결론 및 향후 작업

이 프레임워크는 많은 간단한 확장을 허용합니다.

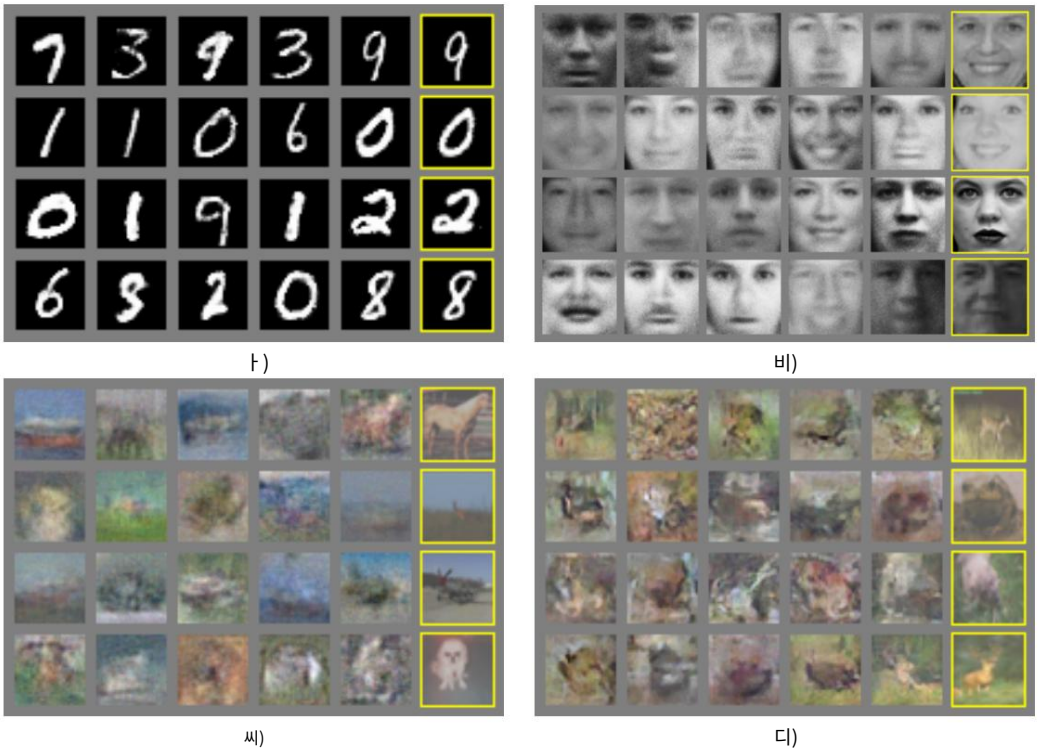


그림 2: 모델의 샘플 시각화. 가장 오른쪽 열은 모델이 학습 세트를 기억하지 않았음을 보여주기 위해 인접 샘플의 가장 가까운 학습 예를 보여줍니다. 샘플은 선별된 것이 아니라 공정한 무작위 추첨입니다. 심층 생성 모델의 대부분의 다른 시각화와 달리 이러한 이미지는 모델 분포의 실제 샘플을 표시하며 숨겨진 단위 샘플이 주어진 조건부 수단이 아닙니다.

또한 샘플링 프로세스가 Markov 체인 혼합에 의존하지 않기 때문에 이러한 샘플은 상관관계가 없습니다. a) MNIST b) TFD c) CIFAR-10(완전히 연결된 모델) d) CIFAR-10(컨볼루션 판별기 및 "디컨볼루션" 생성기)

그림 3: 전체 모델의 z 공간 에서 좌표 사이를 선형 보간하여 얻은 숫자 .

1. 조건부 생성 모델 p(x | c)는 G와 D 모두에 입력으로 c를 추가하여 얻을 수 있습니다.
2. 주어진 x를 예측하기 위해 보조 네트워크를 훈련함으로써 학습된 근사 추론을 수행할 수 있습니다. 이는 wake-sleep 알고리즘 [15]에 의해 훈련된 추론 네트워크와 유사하지만 생성기 네트워크가 훈련을 마친 후 고정된 발전기 네트워크에 대해 추론 네트워크가 훈련될 수 있다는 이점이 있습니다.

3. 파라미터를 공유하는 조건부 모델군을 교육함으로써 모든 조건부 p(xS | xS) 를 근사적으로 모델링할 수 있습니다. 여기서 S는 x 인덱스의 하위 집합입니다. 기본적으로 적대적 네트워크를 사용하여 결정론적 MP-DBM [10]의 확률적 확장을 구현할 수 있습니다.

4. Semi-supervised learning: discriminator 또는 inference net의 기능은 성능을 향상시킬 수 있습니다.
제한된 레이블이 지정된 데이터를 사용할 수 있는 경우 분류기의 필요성.
5. 효율성 향상: G와 D를 조정하는 더 나은 방법을 고안하거나 훈련 중에 샘플 z에 대한 더 나은 분포를 결정함으로써 훈련을 크게 가속화 할 수 있습니다.

이 논문은 적대적 모델링 프레임워크의 실행 가능성을 입증했으며, 이러한 연구 방향이 유용할 수 있음을 시사합니다.

	심층 그래픽 모델	심층 무방향 그래픽 모델	생성적 자동 인코더	적대적 모델
훈련	훈련 중에 필요한 추론.	필요한 추론 훈련 중. MCMC는 파티션 함수 기울기를 근사화하는 데 필요했습니다.	강제 절충 혼합과 재구성 생성의 힘 사이	Discriminator와 Generator를 동기화합니다. 헬베티카.
추론	학습된 대략적인 추론	변이 추론	MCMC 기반 추론	학습된 대략적인 추론
견본 추출	어려움 없음	Markov 체인 필요	Markov 체인 필요	어려움 없음
p(x) 평가	다루기 힘든, 다음과 같이 근사화될 수 있음 AIS	다루기 힘든, 다음과 같이 근사화될 수 있음 AIS	명시적으로 표현되지 않고 다음과 같이 근사화될 수 있습니다. Parzen 밀도 추정	명시적으로 표현되지 않고 다음과 같이 근사화될 수 있습니다. Parzen 밀도 추정
모델 디자인	모델은 원하는 추론 체계와 함께 작동하도록 설계 — 일부 추론 체계는 GAN과 유사한 모델군을 지원합니다.	여러 속성을 보장하기 위해 신중한 설계가 필요함	모든 미분 가능 함수는 이론적으로 허용됩니다.	모든 미분 가능 함수는 이론적으로 허용됩니다.

표 2: 제너레이티브 모델링의 과제: 모델과 관련된 각 주요 작업에 대한 심층 제너레이티브 모델링에 대한 다양한 접근 방식에서 직면하는 어려움에 대한 요약입니다.

감사의 말

유익한 토론을 해주신 Patrice Marcotte, Olivier Delaleau, 조경현, Guillaume Alain, Jason Yosinski에게 감사드립니다. Yann Dauphin이 자신의 Parzen 창 평가 코드를 공유했습니다. 우리는 Pylearn2[11]와 Theano[6, 1]의 개발자, 특히 이 프로젝트에 도움이 되도록 Theano 기능을 서두른 Fred' eric Bastien에게 감사드립니다. Arnaud Bergeron은 LATEX 조판에 절실히 필요한 지원을 제공했다. 또한 CIFAR와 캐나다 연구 위원장에게 자금을 지원하고 Compute Canada와 Calcul Quebec에 컴퓨팅 리소스를 제공한 것에 감사드립니다. Ian Goodfellow는 2013 Google Fellowship in Deep Learning의 지원을 받습니다. 마지막으로 우리의 창의성을 자극해 준 Les Trois Brasseurs에게 감사드립니다.

참조

[1] Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, IJ, Bergeron, A., Bouchard, N. 및 Bengio, Y. (2012). Theano: 새로운 기능 및 속도 개선. 딥 러닝 및 감독되지 않은 기능 학습 NIPS 2012 워크샵.

[2] Bengio, Y. (2009). AI를 위한 심층 아키텍처를 학습합니다. 이제 게시자.

[3] Bengio, Y., Mesnil, G., Dauphin, Y. 및 Rifai, S. (2013). 깊은 표현을 통한 더 나은 믹싱. ~ 안에 ICML'13.

[4] Bengio, Y., Thibodeau-Laufer, E. 및 Yosinski, J. (2014a). 학습 가능한 심층 생성 확률 네트워크 역전파로. ICML'14에서.

[5] Bengio, Y., Thibodeau-Laufer, E., Alain, G. 및 Yosinski, J. (2014b). 역전파로 학습 가능한 심층 생성 확률 네트워크. 기계 학습에 관한 30차 국제 회의(ICML'14) 절차에서.

[6] Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D. 및 Bengio, Y. .(2010). Theano: CPU 및 GPU 수학 표현식 컴파일러. Python for Scientific Computing Conference(SciPy) 절차에서. 구두 발표.

[7] Breuleux, O., Bengio, Y. 및 Vincent, P. (2011). 대표 샘플을 신속하게 생성 RBM 파생 프로세스. 신경계산, 23(8), 2053–2073.

[8] Glorot, X., Bordes, A. 및 Bengio, Y. (2011). 깊은 희소 정류기 신경망. AISTATS'2011에서.

[9] Goodfellow, IJ, Warde-Farley, D., Mirza, M., Courville, A. 및 Bengio, Y. (2013a). 맥스아웃 네트워크.
ICML'2013에서.

[10] Goodfellow, IJ, Mirza, M., Courville, A. 및 Bengio, Y. (2013b). 다중 예측 심층 볼츠만 머신. NIPS'2013에서.

[11] Goodfellow, IJ, Warde-Farley, D., Lamblin, P., Dumoulin, V., Mirza, M., Pascanu, R., Bergstra, J., Bastien, F. 및 Bengio, Y.
(2013c). Pylearn2: 기계 학습 연구 라이브러리. arXiv 프리프린트 arXiv:1308.4214.

[12] Gregor, K., Danihelka, I., Mnih, A., Blundell, C. 및 Wierstra, D. (2014). 심층 자기회귀 네트워크.
ICML'2014에서.

[13] Gutmann, M. 및 Hyvarinen, A. (2010). Noise-contrastive estimation: 비정규화 통계 모델을 위한 새로운 추정 원리. 제13차 인공 지능
및 통계에 관한 국제 회의(AISTATS'10) 진행 중.

[14] Hinton, G., Deng, L., Dahl, GE, Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. 및 Kingsbury, 나.
(2012a). 음성 인식의 음향 모델링을 위한 심층 신경망.
IEEE 신호 처리 잡지, 29(6), 82–97.

[15] Hinton, GE, Dayan, P., Frey, BJ 및 Neal, RM(1995). 비감독을 위한 깨우기-잠자기 알고리즘
신경망. 과학, 268, 1558–1161.

[16] Hinton, GE, Srivastava, N., Krizhevsky, A., Sutskever, I. 및 Salakhutdinov, R. (2012b). 특징 검출기의 공동 적응을 방지하여 신경망
을 개선합니다. 기술 보고서, arXiv:1207.0580.

[17] Jarrett, K., Kavukcuoglu, K., Ranzato, M. 및 LeCun, Y. (2009). 객체 인식을 위한 최고의 다단계 아키텍처는 무엇입니까? 프로세스에서
컴퓨터 비전에 관한 국제 회의(ICCV'09), 2146–2153페이지.
IEEE.

[18] Kingma, DP 및 Welling, M.(2014). 자동 인코딩 변형 베이. Interna 절차에서
ICLR(학습 표현에 관한 회의).

[19] Krizhevsky, A. 및 Hinton, G. (2009). 작은 이미지에서 여러 계층의 기능을 학습합니다. 인위적인
보고서, 토론토 대학.

[20] Krizhevsky, A., Sutskever, I. 및 Hinton, G. (2012). 깊은 컨벌루션 신경망을 사용한 ImageNet 분류. NIPS'2012에서.

[21] LeCun, Y., Bottou, L., Bengio, Y. 및 Haffner, P. (1998). 문서에 적용된 기울기 기반 학습
인식. IEEE 절차, 86(11), 2278–2324.

[22] Mnih, A. 및 Gregor, K. (2014). 신념 네트워크에서 신경 변이 추론 및 학습. 인위적인
보고서, arXiv 프리프린트 arXiv:1402.0030.

[23] Rezende, DJ, Mohamed, S. 및 Wierstra, D. (2014). 확률적 역전파 및 근사
심층 생성 모델의 추론. 기술 보고서, arXiv:1401.4082.

[24] Rifai, S., Bengio, Y., Dauphin, Y. 및 Vincent, P. (2012). 계약 샘플링을 위한 생성 프로세스
자동 인코더. ICML'12에서.

[25] Salakhutdinov, R. 및 Hinton, GE(2009). 깊은 Boltzmann 기계. AISTATS'2009, 페이지 448–
455.

[26] Schmidhuber, J. (1992). 예측 가능성 최소화를 통한 계승 코드 학습. 신경계산,
4(6), 863–879.

[27] Susskind, J., Anderson, A. 및 Hinton, GE(2010). 토론토 얼굴 데이터셋. 기술 보고서 UTML TR 2010-001, U. Toronto.

[28] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, IJ 및 Fergus, R. (2014).
신경망의 흥미로운 속성. ICLR, abs/1312.6199.

[29] Tu, Z. (2007). 차별적 접근법을 통한 생성 모델 학습. 컴퓨터 비전 및 패턴 인식, 2007. CVPR'07. IEEE 컨퍼런스 온, 1-8페이지. IEEE.