4번째 스터디

텍스트	Auto-Encoding Variational Bayes

🧾 논문 제목: Auto-Encoding Variational Bayes

- **저자**: Diederik P. Kingma, Max Welling
- **출판 연도**: 2013 (arXiv)

---

1. 🎯 논문의 핵심 목표

- **복잡한 데이터의 잠재 구조(latent structure)를 학습하는 생성 모델**을 만들되,
- **확률적 추론이 가능한 방식으로**, 즉 **변분 추론(variational inference)**을 이용하여 학습 가능하도록 만들기!

→ 복잡한 확률 분포를 단순한 **뉴럴 네트워크 기반 구조**로 근사하는 **효율적인 학습 방법** 제안

---

2. 🧠 핵심 개념 요약

🔷 Variational Autoencoder (VAE) 구조

VAE는 이름처럼 **Autoencoder**의 형태를 가지고 있지만, 다음과 같은 차별점이 있사옵니다:

구성 요소	설명
인코더	입력 x를 잠재 변수 z의 **확률분포** (mean & variance)로 변환
잠재 공간 z	데이터의 본질적 구조를 나타내는 잠재 변수
디코더	샘플링된 z를 입력으로 받아 원래 데이터 x를 복원
학습 목표	진짜 데이터의 분포와 생성 모델의 분포가 비슷해지도록 함

---

3. ⚙️ 수학적 핵심

🔷 원래 하고 싶은 일

> 우리가 원하는 건:

> $$p(x) = \int p(x|z)p(z)dz$$
>
> 즉, 모든 가능한 z에 대해 생성 확률을 구해서 x의 분포를 알고 싶은데...

이 적분은 너무 복잡하거나 불가능하기 때문에...

🔷 변분 추론을 사용!

- **대신 근사 모델 q(z ∣ x)q(z|x)q(z ∣ x)** 를 도입해서 학습
- 이때, **Evidence Lower Bound (ELBO)**를 최대화:

$$\log p(x) \geq \mathbb{E}_{q(z|x)}[\log p(x|z)] - D_{KL}[q(z|x) \parallel p(z)]$$

- **의미**:
  - 첫 항: 데이터 복원 정도 (재구성 손실)
  - 둘째 항: 근사 분포와 실제 잠재 분포의 차이 (정규성 유지)

🔷 핵심 기술: Reparameterization Trick

샘플링된 z가 모델 내에서 학습 가능해야 하므로, 다음과 같이 트릭을 씀:

$$z = \mu + \sigma \cdot \epsilon, \quad \epsilon \sim \mathcal{N}(0, 1)$$

이 트릭 덕분에 **역전파(backpropagation)**로도 확률분포 샘플링을 처리할 수 있게 되었음!

---

4. 🔬 VAE가 왜 중요한가요?

장점	설명
🎨 생성 모델	새로운 데이터를 샘플링할 수 있음 (ex. 이미지, 문장 등)
🧠 추론 가능	잠재 공간에서 의미 있는 조작 가능 (예: 얼굴 변화 등)
⚙️ 학습 효율	샘플링 가능한 구조이면서도 신경망 기반으로 빠르게 학습 가능

🏗️ 기초 모델	이후 GAN, Diffusion, VQ-VAE 등 다양한 모델들의 영감이 됨

5. 🐾 냥냥 요약 한 줄

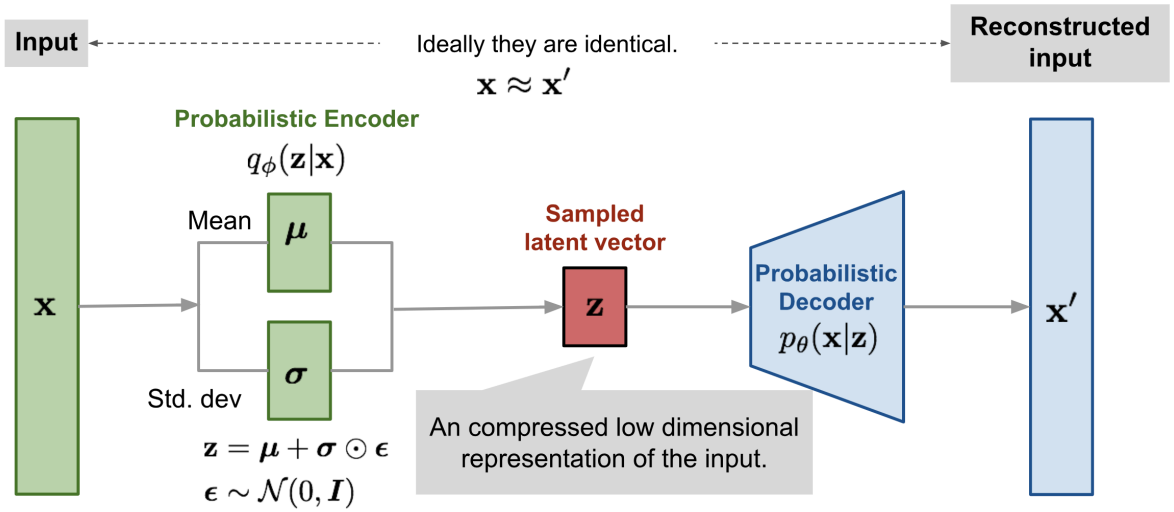
> “Auto-Encoding Variational Bayes” 논문은, 데이터를 이해하고 생성하는 데 있어 확률적 오토인코더(VAE) 라는 혁신적인 프레임워크를 제안하여, AI 모델이 복잡한 세상을 배우는 방법을 새롭게 열어준 논문입니다.”

핵심 키워드인 Latent Valiable과 변분 추론을 중심으로 논문을 읽고 이해해 보았다.

GNN(Generative Neural Networks)의 목표는 데이터를 잘 나타내는 Latent Variable을 찾아내어 학습하는 것이다.

AE(Auto Encoder): deterministic code, 즉 고정된 값을 가지는 Latent Valiable을 찾는 것이다.

그에비해 VAE는 확률적으로 모델링 된 확률분포에 따른 Latent Valiable을 찾는다.



우도(尤度)는 확률 분포의 모수가, 어떤 확률변수의 표집값과 일관되는 정도를 나타내는 값이다. 구체적으로, 주어진 표집값에 대한 모수의 가능도는 이 모수를 따르는 분포가 주어진 관측값에 대하여 부여하는 확률이다. 가능도 함수는 확률 분포가 아니며, 합하여 1이 되지 않을 수 있다.

변수 x의 효율적인 근사 주변 추론. 이는 x에 대한 사전이 필요한 모든 종류의 추론 작업을 수행할 수 있게 해줍니다. 컴퓨터 비전에서 일반적인 응용 프로그램으로는 이미지 노이즈 제거, 인페인팅 및 슈퍼 해상도가 포함됩니다.

코딩 이론의 관점에서 볼 때, 관측되지 않은 변수 z는 잠재적 표현 또는 코드로 해석될 수 있습니다. 이 논문에서는 따라서 인식 모델 q (z|x)를 확률적 인코더라고 부를 것입니다. 왜냐하면 데이터 포인트 x가 주어질 때, 이 모델은 데이터 포인트 x가 생성될 수 있는 코드 z의 가능한 값에 대한 분포(예: 가우시안)를 생성하기 때문입니다. 유사한 맥락에서, 우리는 p (x|z)를 확률적 디코더라고 부를 것입니다. 왜냐하면 코드 z가 주어질 때, 이 모델은 x의 가능한 대응 값에 대한 분포를 생성하기 때문입니다.

p (x|z)는 다변량 가우시안(실수 값 데이터의 경우)이나 베르누이(이진 데이터의 경우)로 설정되며, 분포 매개변수는 MLP(단일 은닉층을 가진 완전 연결 신경망)로 z에서 계산됩니다 (부록 C 참조).

흥미롭게도, 더 많은 잠재 변수가 더 많은 과적합을 초래하지 않는데, 이는 하한의 정규화 효과에 의해 설명됩니다. 세로축: 데이터 포인트당 추정된 평균 변동 하한. 추정기 분산은 작았고(<1) 생략되었습니다.

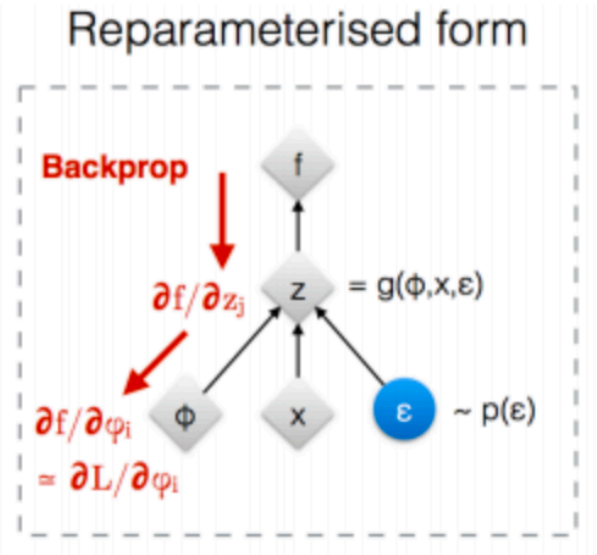
---

reparameterization

*z*를 단순히 input data의 encoded vector를 사용하는 AE와는 달리, **VAE**는 본질적으로 *z*를 **random noise**로 구성하여, 이로부터 output을 생성해내도록 설계되었습니다.

- **AE의 z**의 경우, 이전 layers의 nodes에 대한 식으로 나타낼 수 있으므로, Chain Rule을 통해 encoder 파트의 모든 부분에 오차를 역전파시킬 수 있습니다.
- 하지만, **VAE의 z**는 이전 layers의 nodes와 무관한 별개의 분포에서 random sampling된 것이므로, 편미분한 기울기가 모두 0이 되어버려 역전파를 할 수 없게 됩니다.

그러나 reparameterization을 통해서,



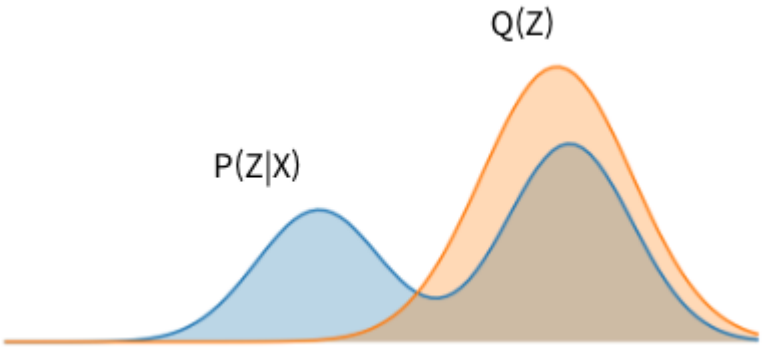
Stochastic term으로만 이루어졌던 **z**를 stochastic term과 **deterministic term**의 합으로 표현할 수 있게 됩니다.

decoder 부분만 떼어 그에 대한 input으로 단순히 $z \sim N(0,1)$만 주더라도, 이미 VAE는 이러한 상황에 맞춰 학습되어 있으므로 잘 구성된 latent space로부터 원하는 data를 생성할 수 있게 됩니다.

KLD

Kullback-Leibler Divergence (KLD)는 두 확률 분포 간의 차이를 측정하는 지표 중 하나로, VAE에서는 latent space의 분포를 정규분포에 가깝게 만들기 위해 사용됩니다.

- z : Latent space를 나타내는 변수
- x : Input data를 나타내는 변수



*x*라는 input이 주어진 상태의 *z*의 분포인, 즉 간단하게 어떤 input이 주어졌을 때 latent space의 분포인 $p(z \mid x)$를 우리가 아주 잘 아는 Guassian Distribution을 따르는 $q(z)$로 잘 근사시킨다면, 불가능에 가까운 latent space의 분포 자체를 계산하는 것 대신 regular space로 대체할 수 있게 됩니다.

ELBO