알기쉬운
Variational AutoEncoder

Sho Tatsuno

Univ. of Tokyo

번역 및 수정 : 김홍배

주요 내용

- Variational Auto-Encoder의 해설
  - 생성모델 자체에 대한 설명
  - Variational Auto-Encoder(VAE)에 대한 설명

- 설명하는 것/하지 않는 것
  - 설명하는 것
    - » 생성모델의 간단한 개요와 사례
    - » Variational AutoEncoder의 구조와 수학적・직관적 해석
  - 설명하지 않는 것
    - » LDA와 같은 다른 생성모델의 상세한 설명
    - » Deep Learning의 기초(Back Propagation・SGD등)
    - » 기존 최적화 기법에 대한 자세한 내용(MCMC・EM알고리즘 등)

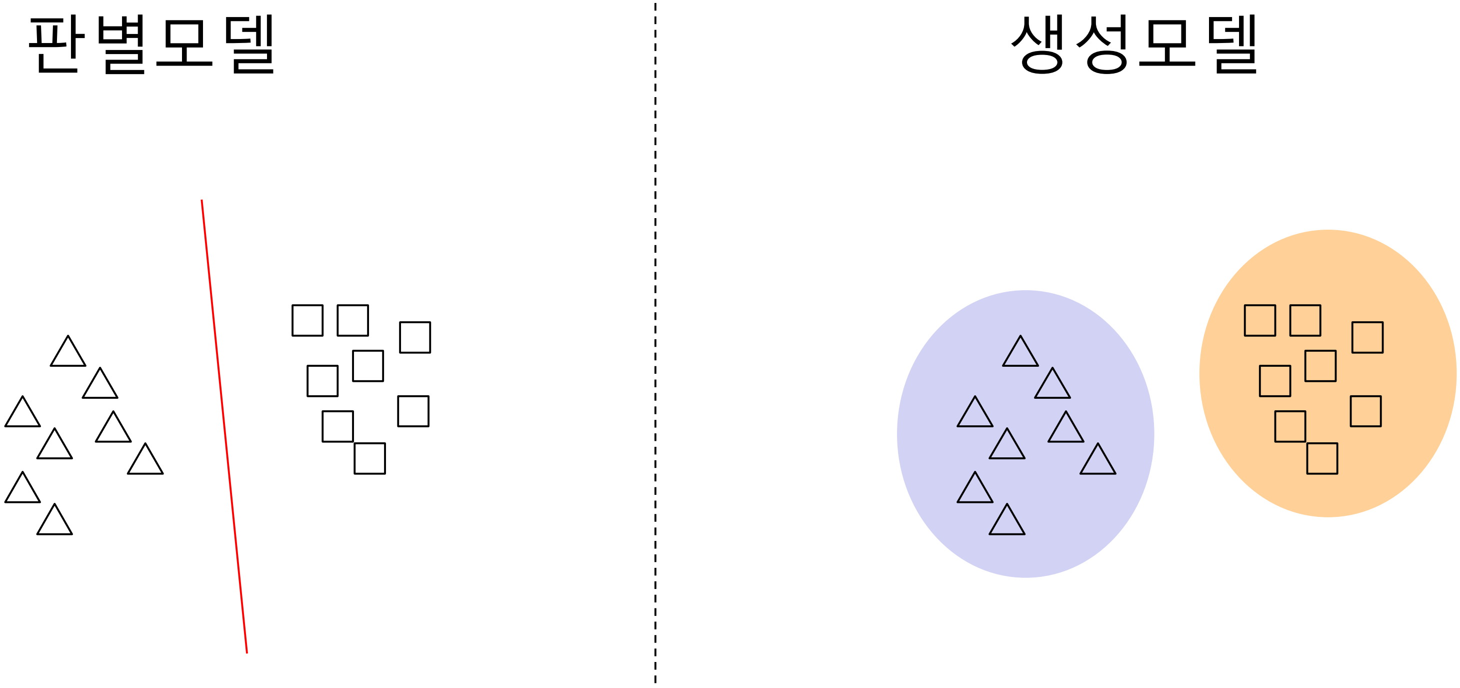
소개 논문들

- Auto-Encoding Variational Bayes
  - Author: D. P. Kingma, 2013
  - URL: https://arxiv.org/pdf/1312.6114.pdf
  - Variational Auto-Encoder를 최초로 제안한 논문

- Tutorial on Variational Autoencoders
  - Author: Carl Doersch, 2016
  - URL: https://arxiv.org/abs/1606.05908
  - 뉴럴넷에 의한 생성모델 Variational Autoencoder(VAE)의 소개
    - » Variational Bayes에 대한 사전지식이 필요없음
    - » 조건부 VAE인 Conditional Variational Autoencoder(CVAE)에 대한 소개

판별모델(discriminative model)과 생성모델(generative model)

- 일반적인 기계학습은 판별모델
  - 각각을 나누기 위해 선을 긋는다 !
- 생성모델은 판별하는 것이 아니라 범위를 고려



하고 싶은 것은 ?

- 이미지와 같은 고차원 데이터 X의 저차원 표현 z을 구할 수 있다면
- Z을 조정하여 training set에서 주어지지 않은 새로운 이미지 생성이 가능
- 카메라 각도, 조명 위치, 표정등의 조정이 가능

Other examples
- random faces
- MNIST
- Speech

These are not part of the training set !

https://www.youtube.com/watch?v=XNZIN7Jh3Sg

Manifold hypothesis

고차원 데이터를 저차원 데이터로 표현해낼 수 있을까 ?

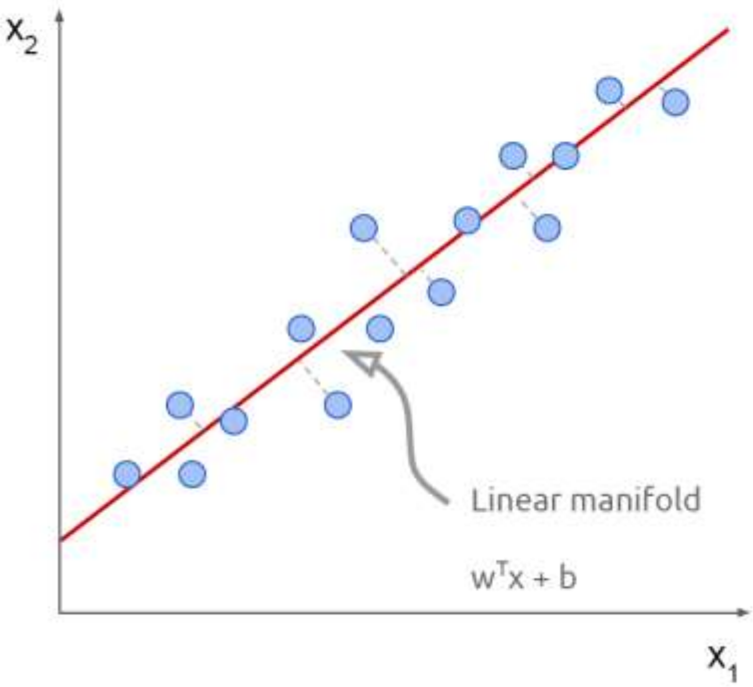
The data distribution lie close to a low-dimensional manifold

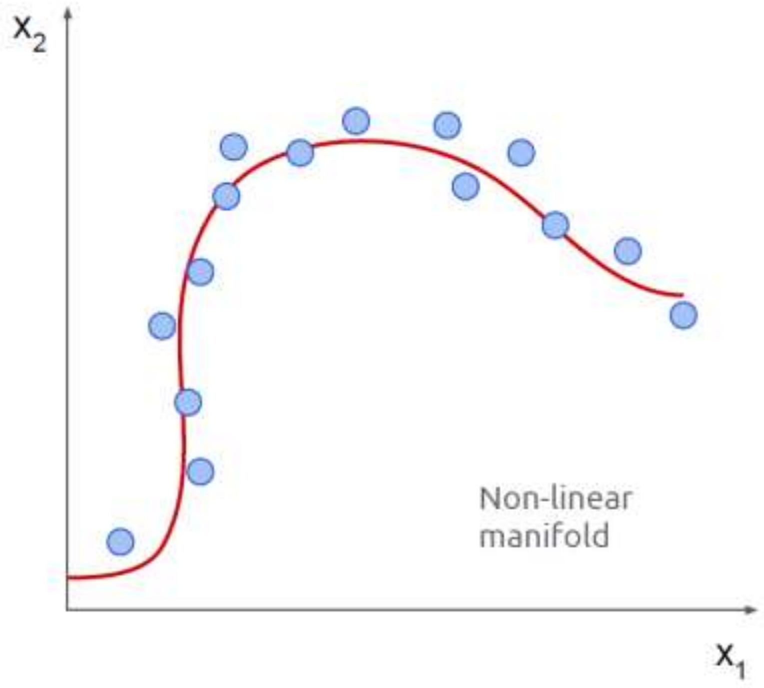
➔ **Manifold hypothesis**

Example: **consider image data**

- ✓ Very high dimensional (1,000,000D)
- ✓ A randomly generated image will almost certainly not look like any real world scene
  - The space of images that occur in nature is almost completely empty
- ✓ Hypothesis: real world images lie on a smooth, low-dimensional manifold
  - Manifold distance is a good measure of similarity

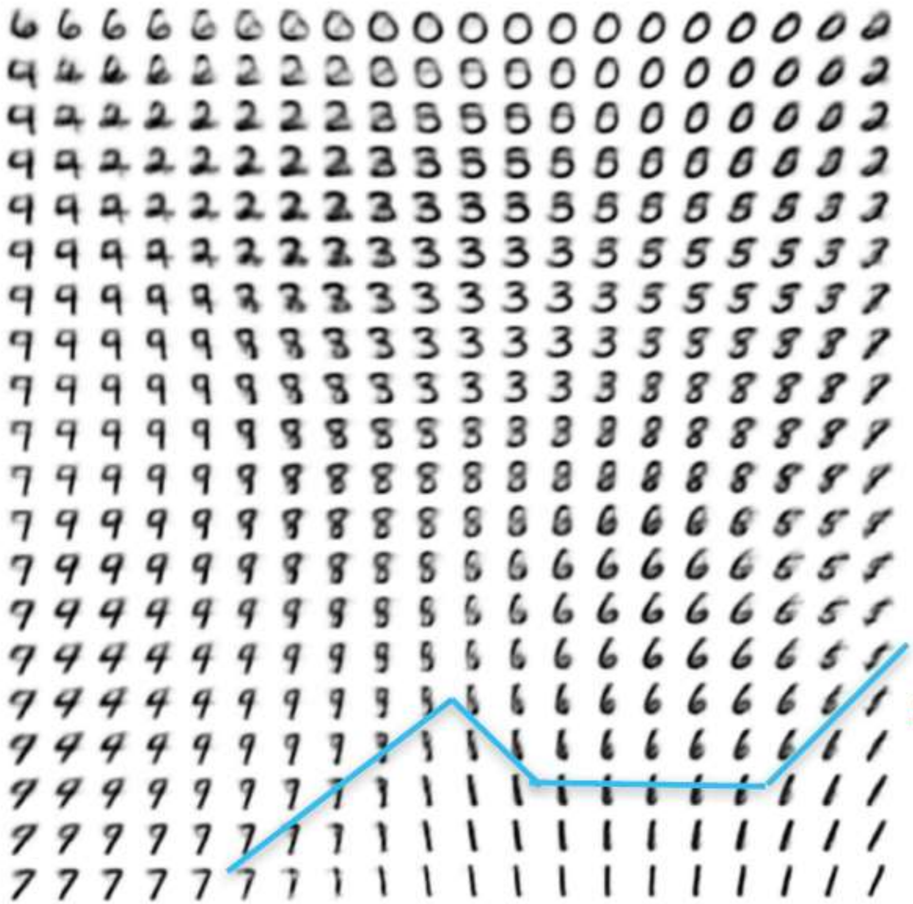
Manifold hypothesis





Manifold hypothesis

- 다음과 같은 손글씨 데이터는 784차원의 고차원이지만, 다음과 같이 저차원 표현이 가능



"1"을 분류하고 싶다면
그림과 같은 분류 경계선을
쉽게 그을 수 있다

왜 딥러닝은 성공적이나 ?

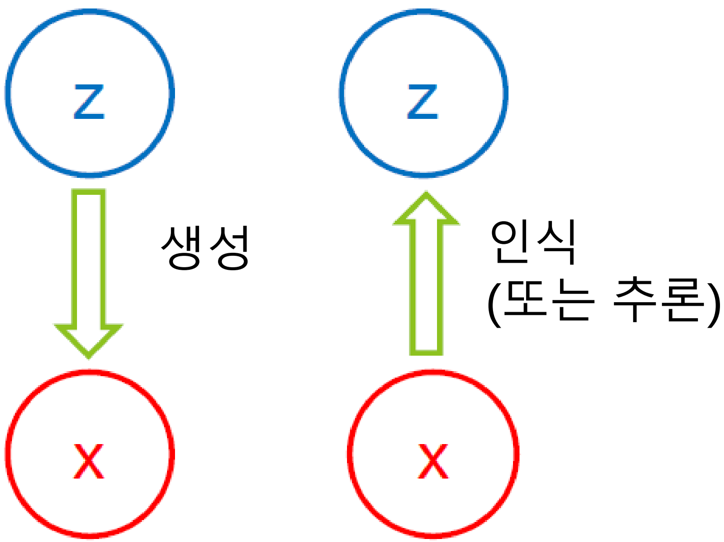
Lin의 가설[Lin+ 16]

- 왜 딥러닝이 다양한 문제에 잘 적용(특히 인식문제) 될까
- ⇒ 거의 모든 문제는 다음과 같은 특징이 있기 때문

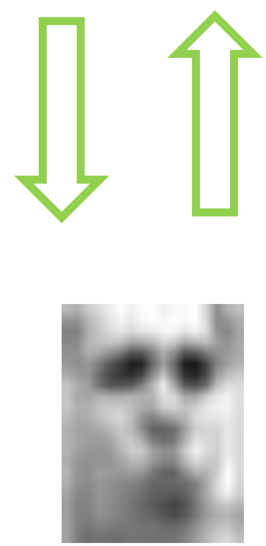
1. 저차성
   - ✓ 일반적인 물리현상의 <u>변수간 상호작용 차수는 2~4</u>
2. 국소 상호작용성
   - ✓ 상호작용 수는 변수의 개수에 대하여 선형적으로 증가
3. 대칭성
   - ✓ 이미지의 대칭성등에 의해 <u>변수의 자유도가 낮다</u>
4. 마코프성
   - ✓ 생성과정은 직전의 상태에만 의존한다.

잠재변수란 ?

- 다음과 같은 인식과 생성문제를 보면
- X : 이미지, z : 잠재변수



잠재변수 Z의 예
: 물체의 형상, 카메라 좌표, 광원의 정보
(남자, [10, 2,-4], white)

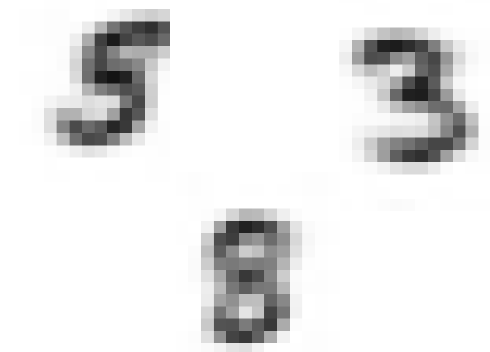


X : 이미지

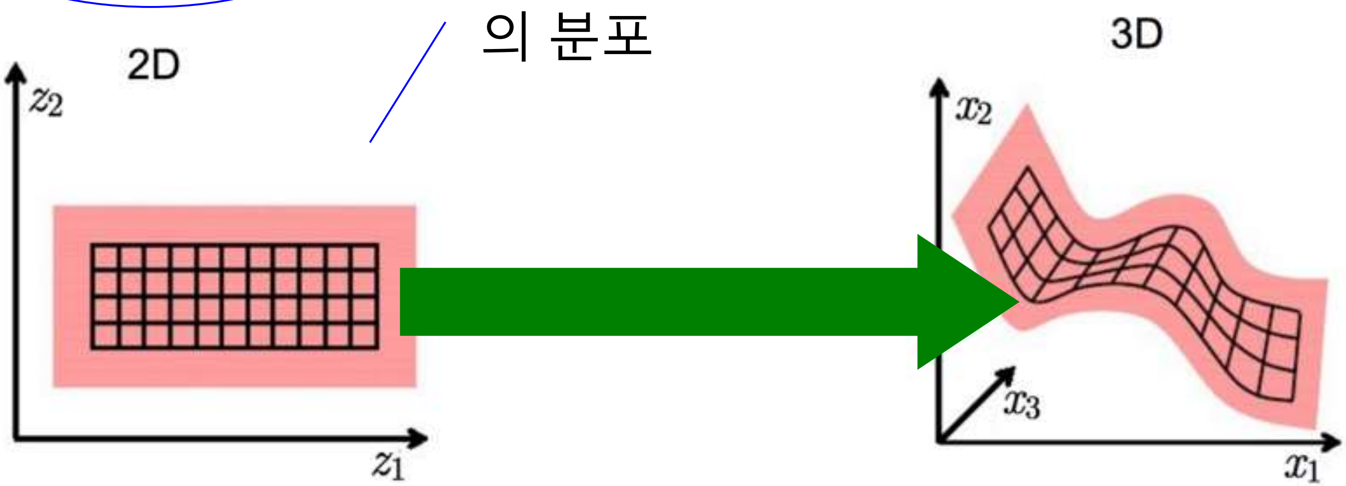
잠재변수란 ?

- 예를 들어 숫자의 잠재적인 의미를 생각하면
  - Digit(3인가 5인가)과 필체를 나타내는 잠재변수(각도, aspect ratio, 등)으로부터 숫자를 만들어낼 수 있다.

잠재변수의 분포



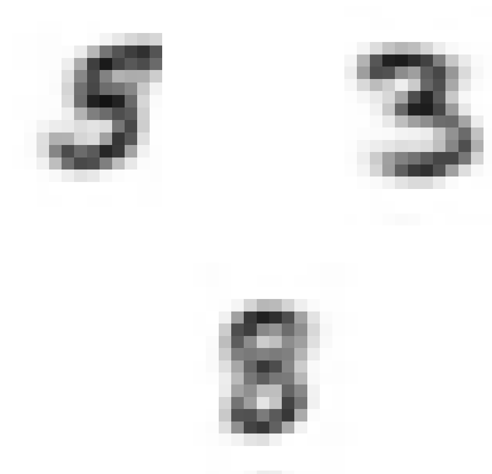
잠재변수
의 분포



잠재변수란 ?

- 이미지의 잠재공간의 확보와 이미지의 변형 생성

얼굴의 생성



숫자의 생성

표정・view point・
얼굴형태가 잠재변수 ?

digit・필체(기울기,
aspect ratio)가 잠재변수 ?

기존 생성모델의 문제점

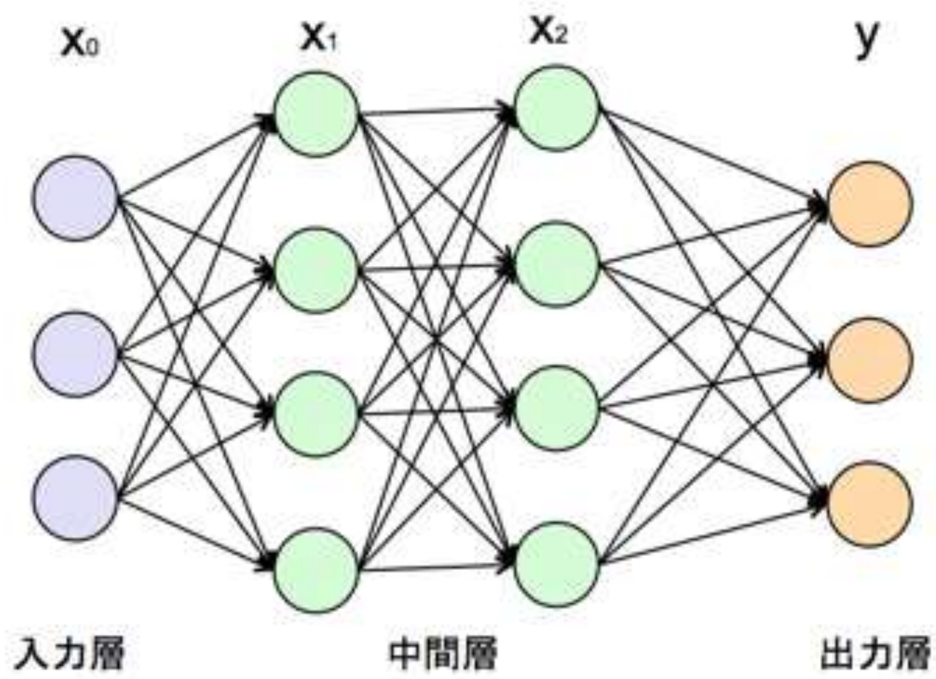
1. 데이터 구조의 가정과 모델의 근사가 필요
   – 여기서 어떠한 분포의 설정이 필요
   – 설정한 분포에 모델이 대응하여야 함

2. 시간이 소요되는 방법이 필요
   - MCMC등과같이 복수의 샘플링이 필요

자세한 것은 생략

뉴럴넷의 이용

- 단순한 뉴럴넷의 예



1. $y = f_2(w_2x_2+b_2)=f_2(w_2(f_1(w_1x_1+b_1))+b_2)=\cdots$

   → Convolution형에서는 임의의 함수표현이 가능 : 모델의 제약을 완화

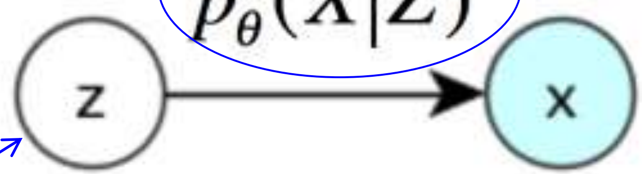
2. SGD를 사용하면 1샘플씩 최적화가 가능

생성모델 최적화의 전제

- 원래

요것을 구하고 싶다(Z을 바탕으로 X가 생성되는 확률분포)

$p_\theta(X|Z)$



잠재변수

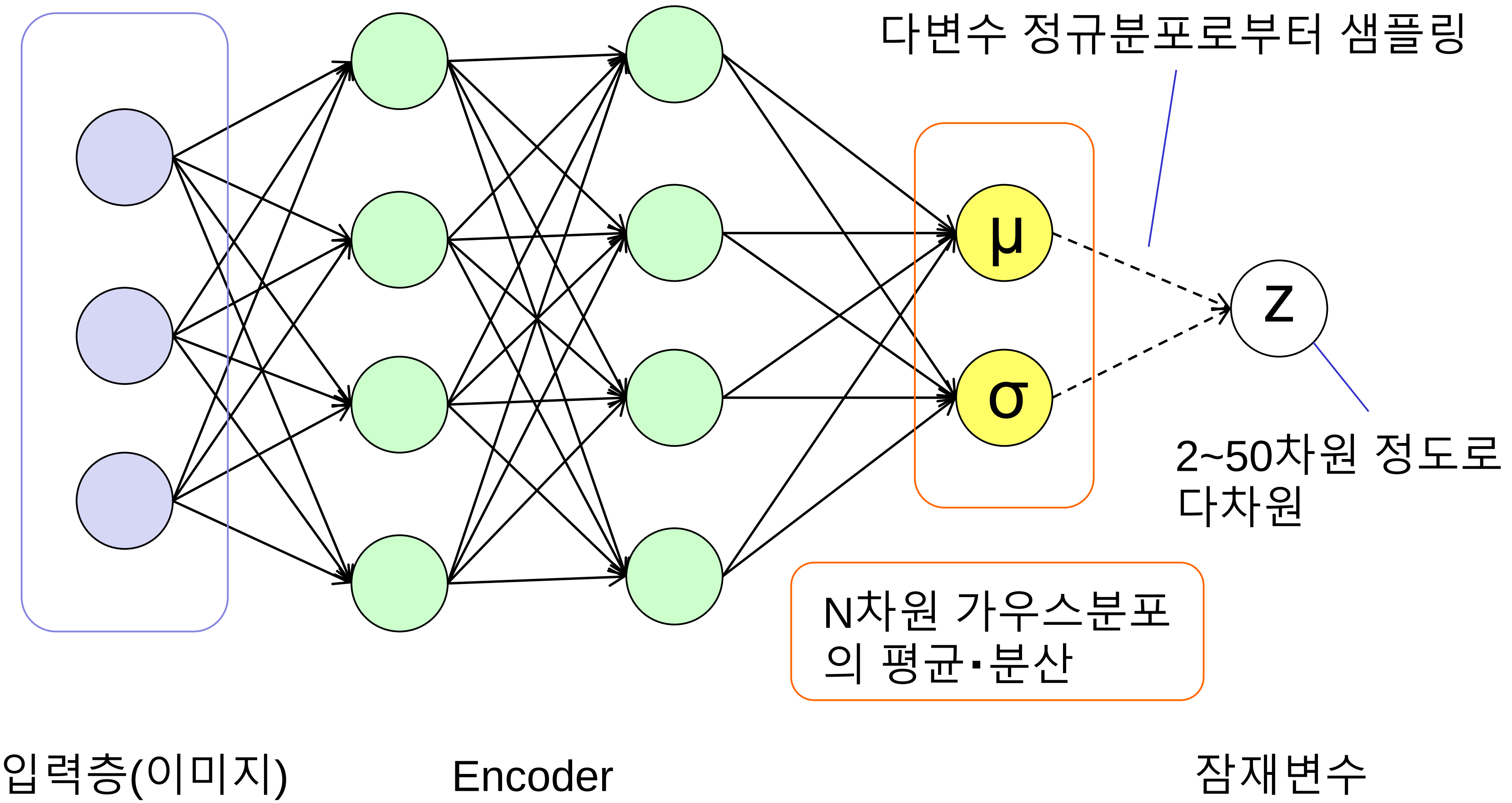
출력

  – 그러나, 이대로 p_θ를 구하는 것은 곤란함
    » 입력(잠재변수z)에 대응하는 답이 불명확함
      일반적으로 z은 저차원, X는 고차원임

입력에서 잠재변수 z으로의 분포를 가정

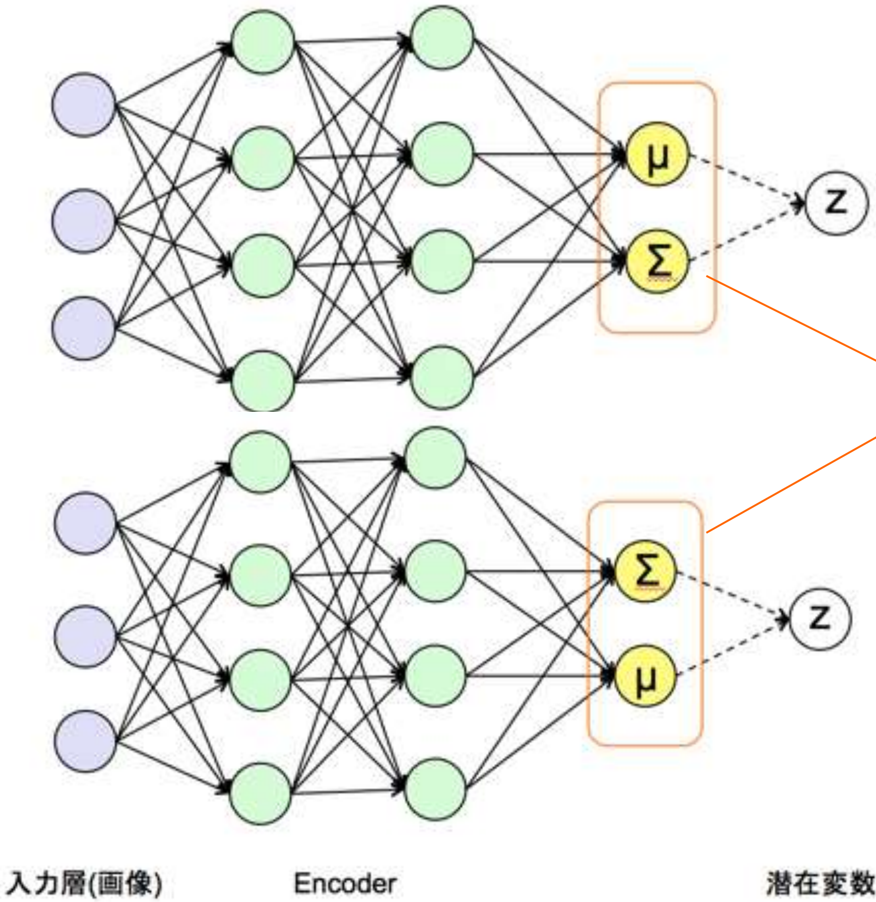
- Encoder
  - 잠재변수의 정규 분포를 가정

$$f(\mathbf{x}) = \frac{1}{(\sqrt{2\pi})^m \sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}} \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu})\right)$$



분포 parameter의 타당성

- μ 와 σ의 결정에 타당성은 있나 ?
  - 어느 쪽이 μ・ σ라도 좋다 : μ와 σ가 최적화되도록 NN을 최적화하면 됨

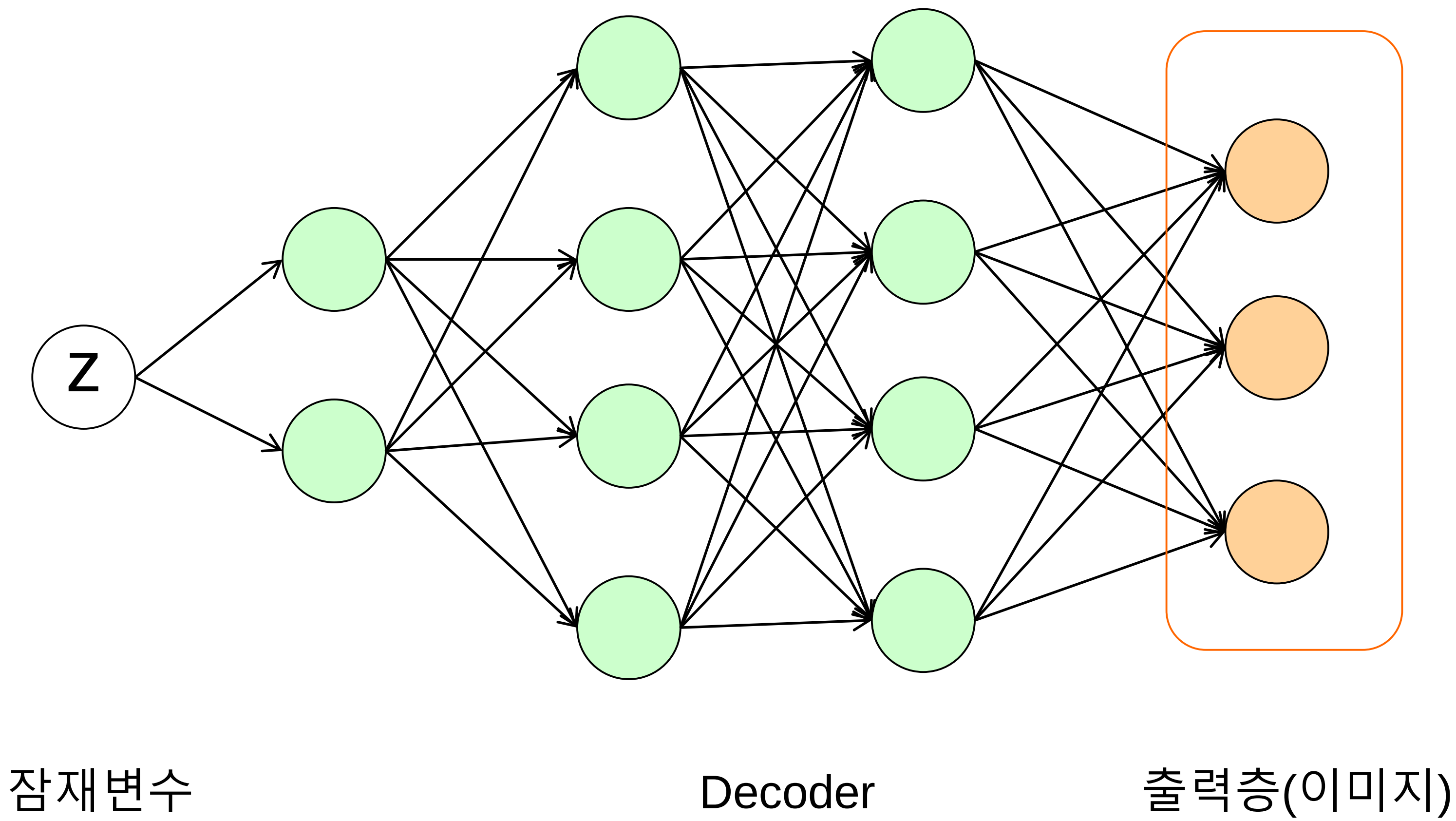


어떤 것이라도 괜찮다(미리μ・ σ가 정의되어있지 않음)

나중에 μ・ σ에 대응하도록 학습시킴

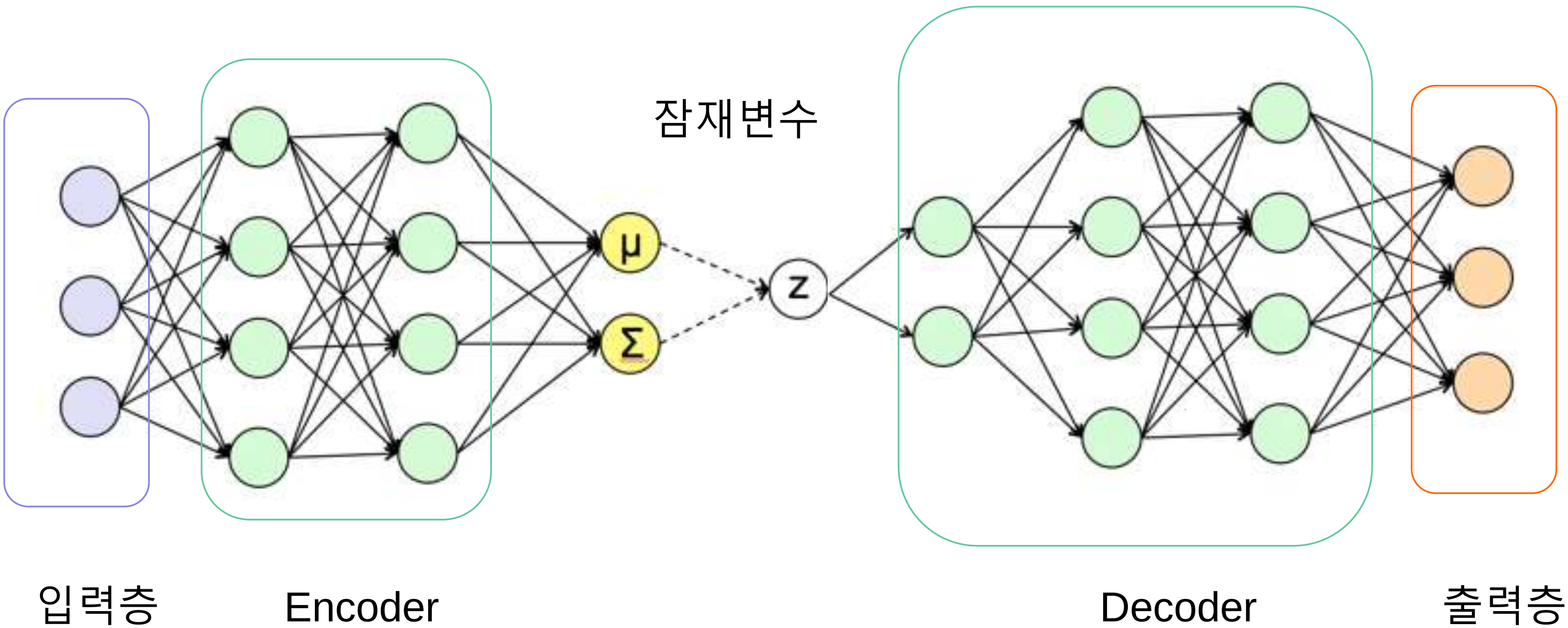
Variational AutoEncoder

- Decoder
  - 여기서는 z로부터 출력층까지에 NN을 만들면 됨.



Variational AutoEncoder

- Total Structure

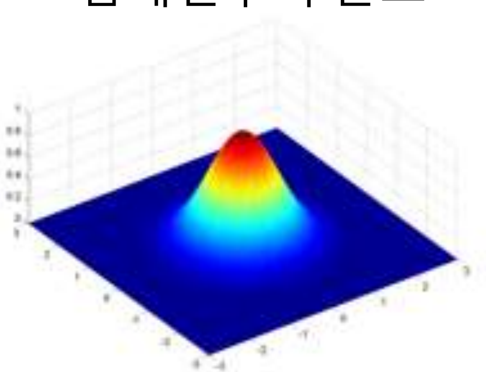


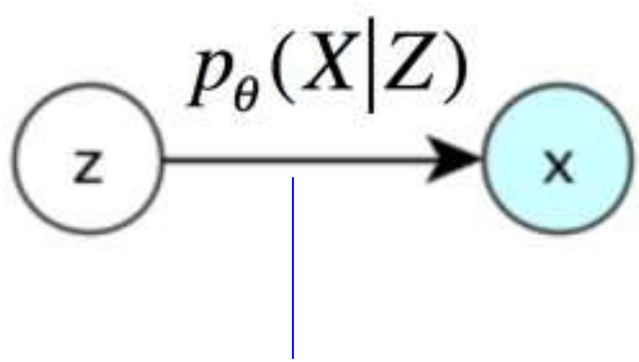
잠재변수의 가정

- 잠재변수는 다차원의 정규분포로 가정
  - 다루기 쉬움
  - 잠재변수로 문자의 필적과 형태를 가정 → 정규분포인가 ?

  • z ~ p(z) : p의 prior distribution으로 간단한 형태(다차원 표준 정규분포)

잠재변수의 분포



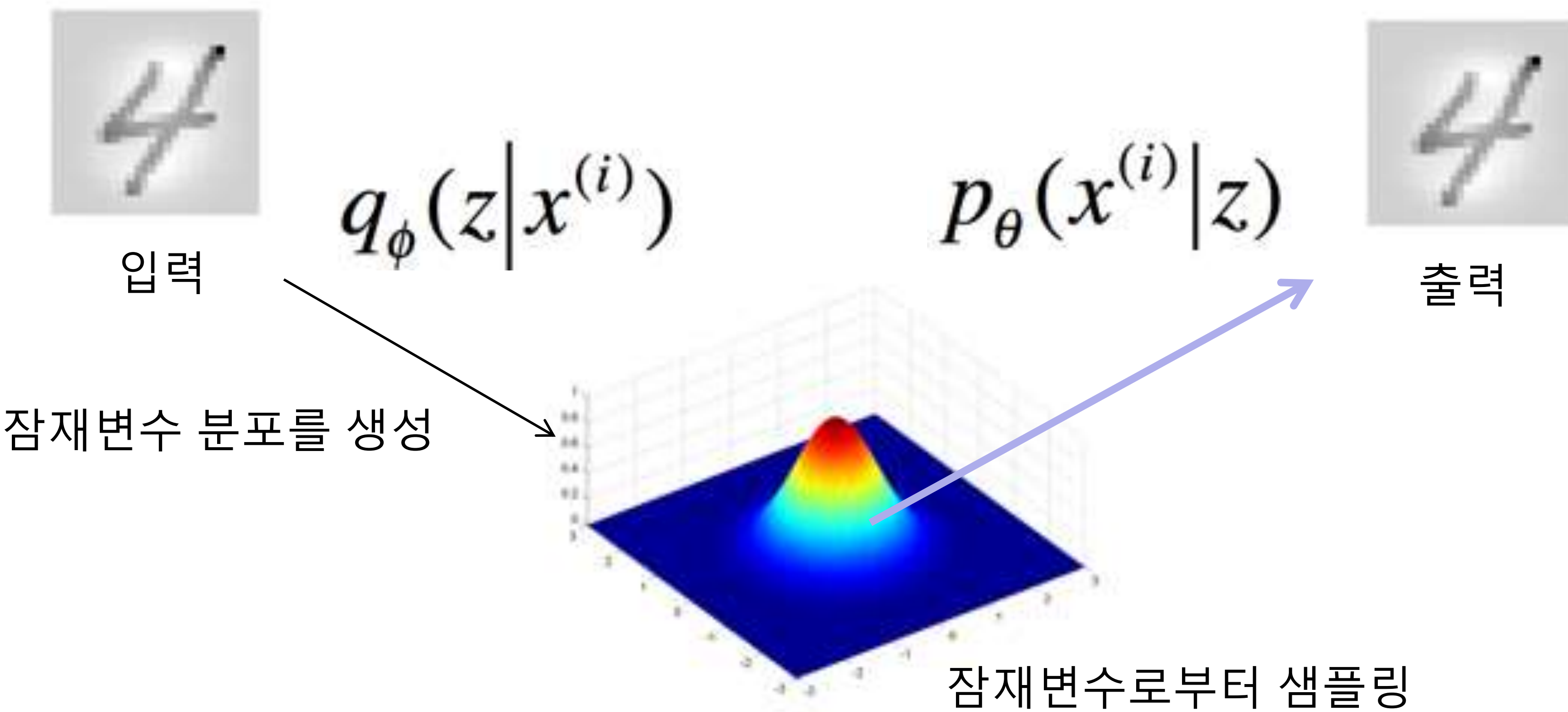


출력 이미지

잠재변수 z로 부터 이미지 X를 생성(θ는 parameter)

VAE의 도식적 이해

- 입력 → 잠재변수 분포를 생성 $q_\phi(z|x^{(i)})$
- 잠재변수로부터 샘플링 → 입력에 가까운 출력 생성 $p_\theta(x^{(i)}|z)$



최적화의 필요성

- 어떻게 최적화하여야 하나 ?
  - Maximum Likelihood Estimation : marginal likelihood log(p_θ(x))의 최대화 되도록

    Θ를 정할 때 취할 수있는 x의 Marginal probability가 가장 높도록
  - marginal likelihood log(p_θ(x))는 다음과 같이 나눌 수 있음

일반적인 variational lower limit에 있어서 수식전개

$$\log p_\theta(x) = D_{KL}(q_\phi(z|x)||p_\theta(z|x)) + \underline{\mathcal{L}(\theta,\phi,x)}$$

$$\geq \mathcal{L}(\theta,\phi,x)$$

variational lower limit : θ, φ의 함수

여기서 KL Divergence는

$$D_{KL}(q_\phi(z|x)||p_\theta(z)) \geq 0$$ (p=q시、등호성립)

➔ variational lower limit을 최대화하면 marginal likelihood도 커짐

Variational lower limit

- variational lower limit을 정리해보면
  - 아래와 같은 형태로 되며, 최적화 항이 도출됨

$$\mathcal{L}(\theta, \phi, x) = \mathbb{E}_{q_\phi(z|x)} [\log p_\phi(x, z) - \log q_\phi(z|x)]$$
$$= \mathbb{E}_{q_\phi(z|x)} [\log p_\phi(x|z) - p_\phi(z) - \log q_\phi(z|x)]$$
$$= -D_{KL}(q_\phi(z|x)||p_\phi(z)) + \mathbb{E}_{q_\phi(z|x)} [\log p_\theta(x|z)]$$

정규화 항 : KL Divergence
(Regularization Parameter)

복원오차
(Reconstruction Error)

이 두개의 합을 최대화하면 좋음

정규화 항 : KL Divergence

- KL Divergence의 계산

$$D_{KL}(\underbrace{q_\phi(z|x)}_{\sim N(\mu,\, \sigma)}||\underbrace{p_\theta(z)}_{\sim N(0,\, I)})$$

$$=D_{KL}(N(\mu,\Sigma)||N(0,I))$$

$$=-\frac{1}{2}\sum_{j=1}^{J}(1+\log(\sigma_j^2)-\mu_j^2+\sigma_j^2)$$

복원오차 : Reconstruction Error

- Reconstruction Error는 아래와 같이 근사화

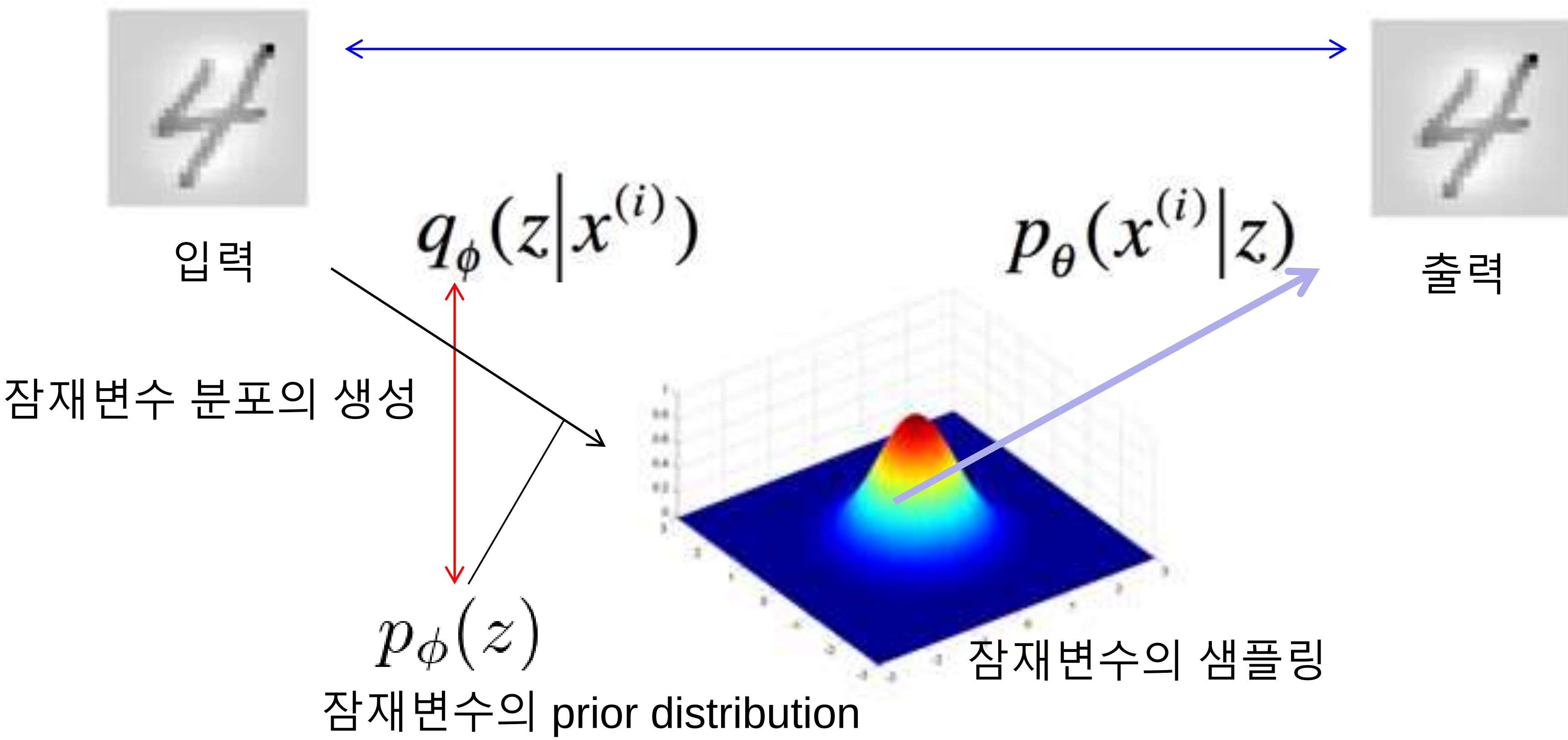
$$\mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right] \simeq \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x|z)$$

- 이미지 픽셀값을 0~1로 Normalize한 경우, Bernoulli분포로 가정하면 logp(x|z)는 아래와 같이 나타낼 수 있음
  (y는 잠재변수 z를 Fully Connected Layer를 통과한 Final layer의 변수)

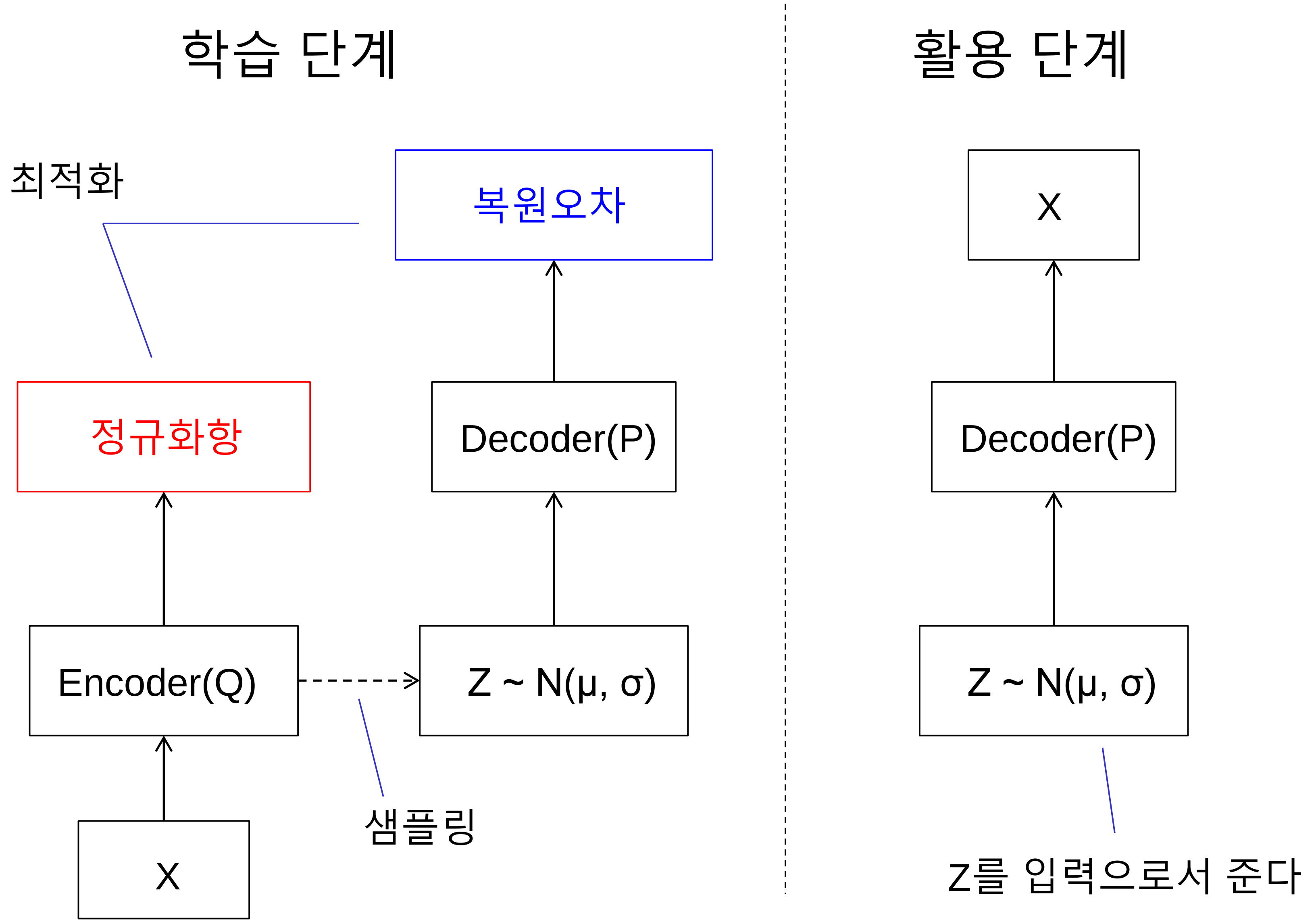
$$\log p(x|z) = \sum_{i=1}^{D} x_i \log y_i + (1 - x_i) \cdot \log(1 - y_i)$$

VAE의 도식적 이해

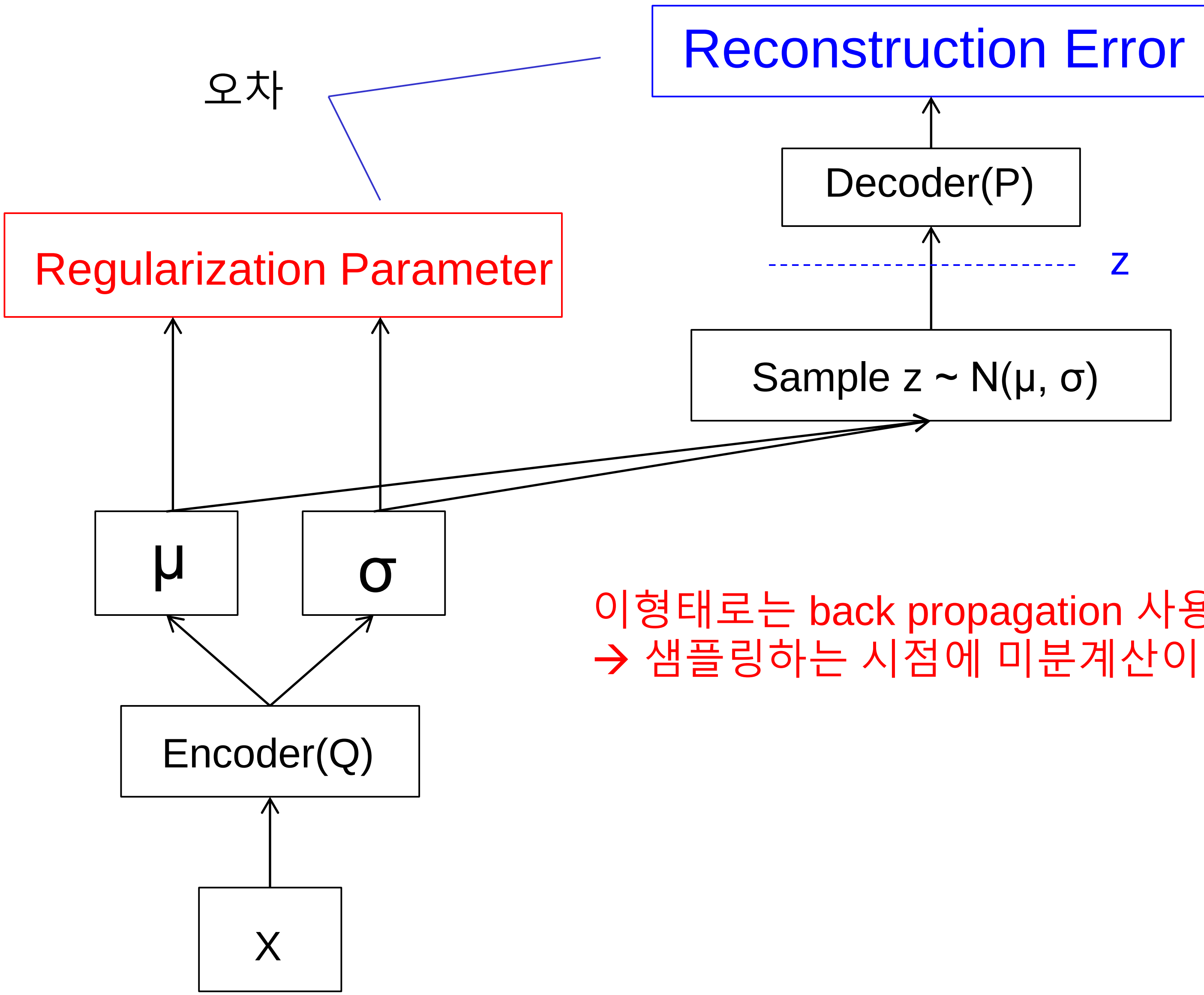
- 최적화 함수
  - KL Divergence: p(z)와 q(z|x)의 정보적 거리・정규화항 :
  - Reconstruction error: 입출력 오차 :



VAE 전체 block diagram

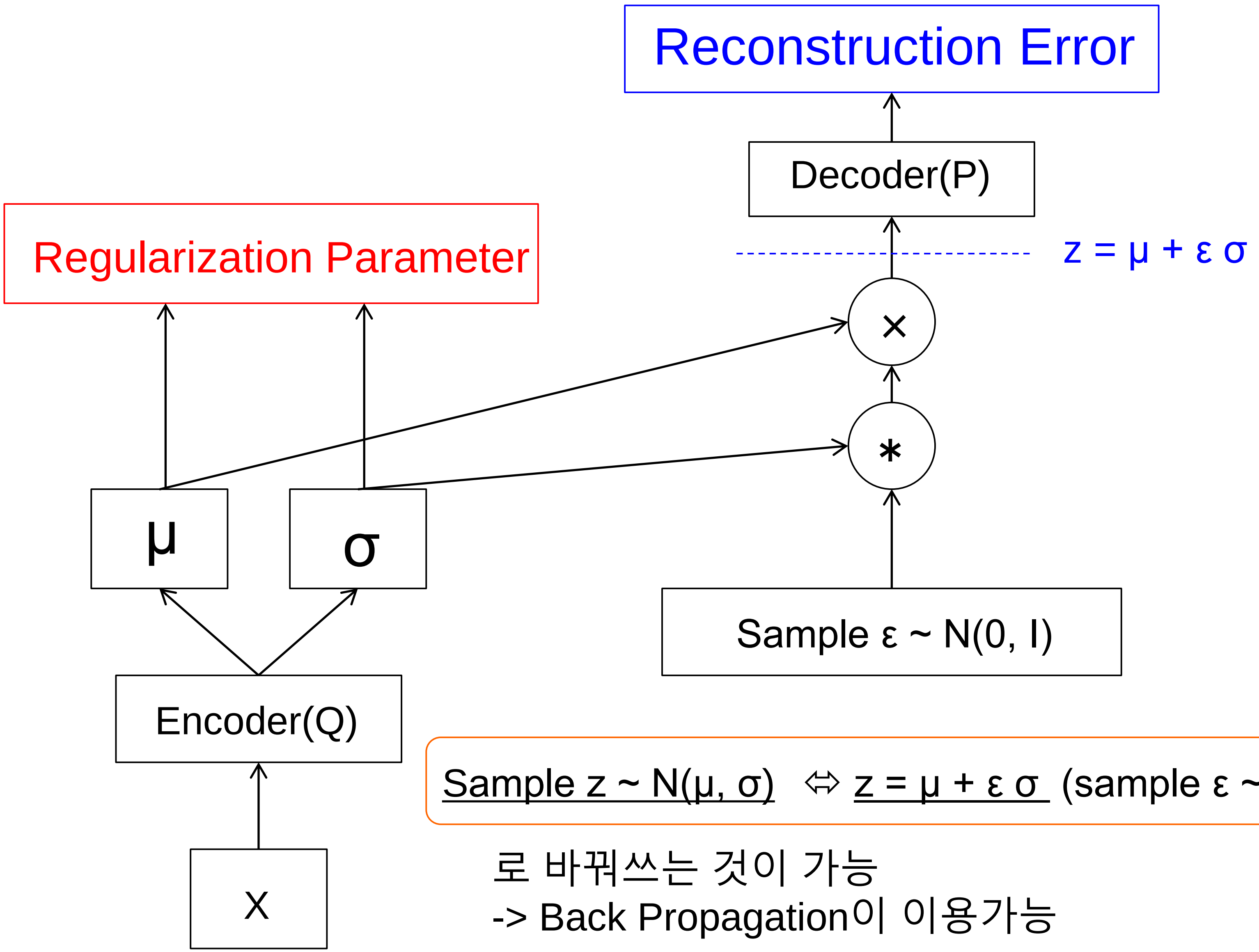


학습 단계의 보다 상세한 구조



이형태로는 back propagation 사용이 불가능 !
→ 샘플링하는 시점에 미분계산이 불가능

Reparametrization Trick



Sample z ~ N(μ, σ) ⇔ z = μ + ε σ (sample ε ~ N(0, I))

로 바꿔쓰는 것이 가능
-> Back Propagation이 이용가능

Z의 변환에 대하여

- 일차원 경우의 간단한 증명

Sample z ~ N(μ, σ) ⇔ z = μ + ε σ (sample ε ~ N(0, 1))

ε 는 표준 정규분포이므로 확률밀도함수는 $f(\epsilon) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{\epsilon^2}{2})$

$z = \mu + \epsilon \cdot \sigma \Leftrightarrow \epsilon = \frac{z-\mu}{\sigma}$ 로 변환가능하므로 대입하면

$$f(z) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp(-\frac{(z-\mu)^2}{2\sigma^2})$$

정규분포 z~(N(μ, σ))로부터 샘플링과 동일

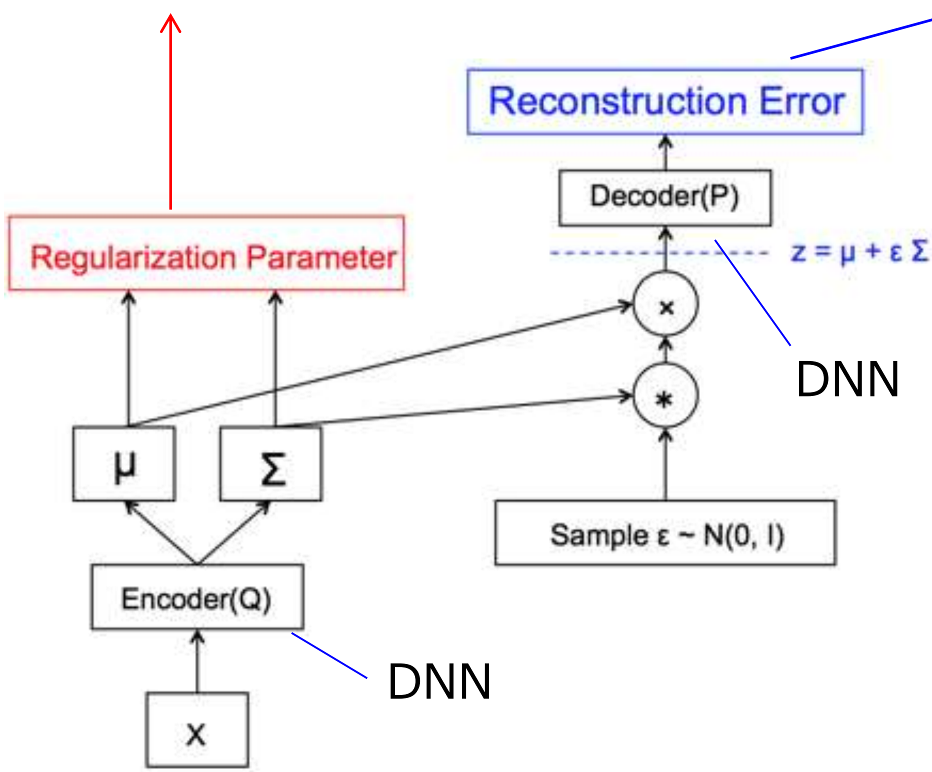
차수가 2차 이상인 경우도 동일

VAE의 최적화 정리

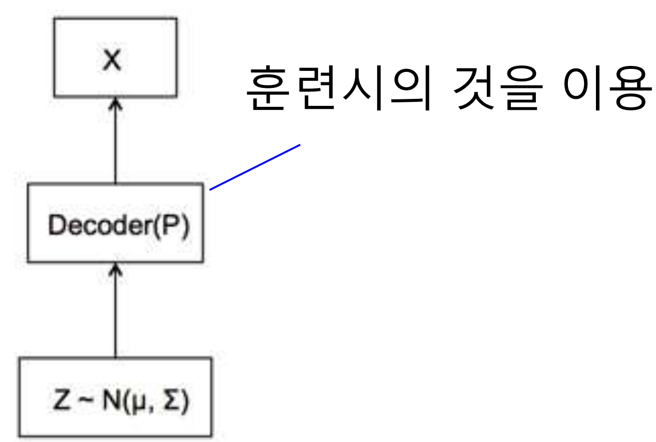
• 훈련시

$$\mathcal{L}(\theta, \phi, x)$$

$$= -\frac{1}{2}\sum_{j=1}^{J}(1 + \log(\sigma_j^2) - \mu_j^2 + \sigma_j^2) + \mathbb{E}\left[\sum_{i=1}^{D}(x_i \log y_i + (1 - x_i) \cdot \log(1 - y_i))\right]$$



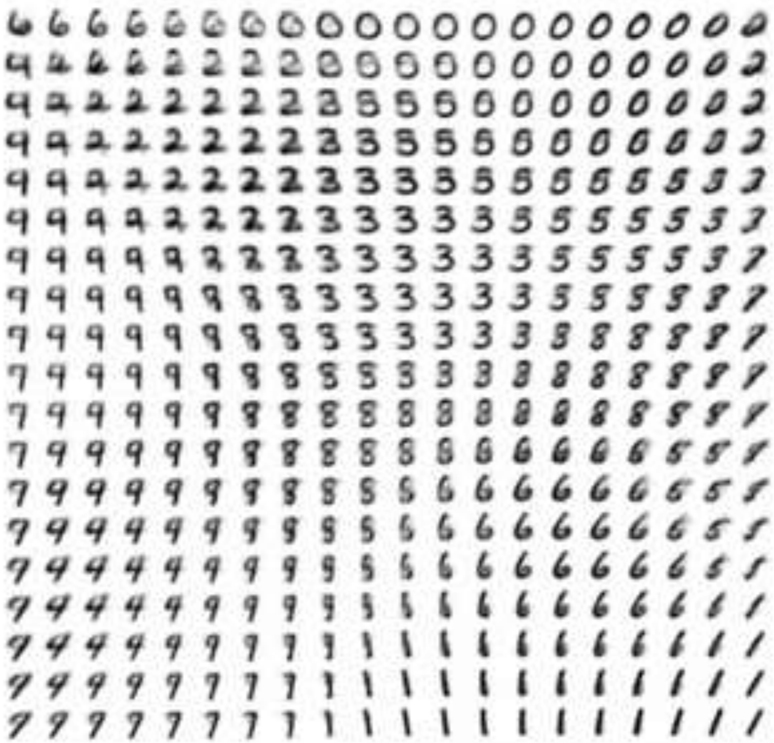
• 적용시



Result

- 잠재공간에 대응하는 이미지 생성

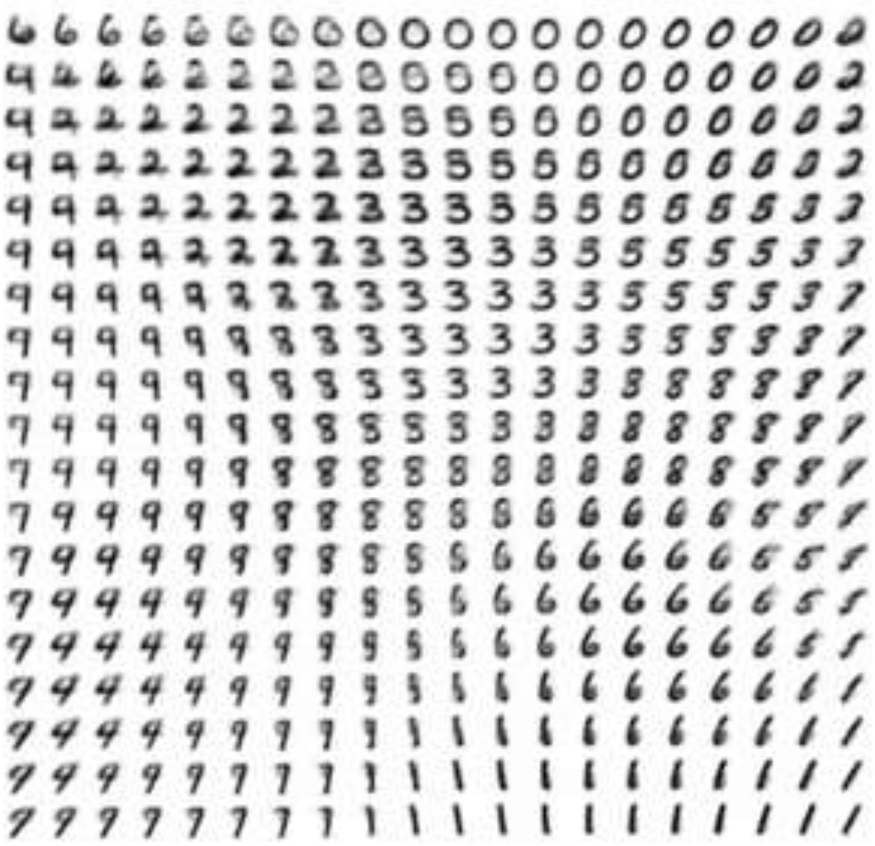
표정의 생성



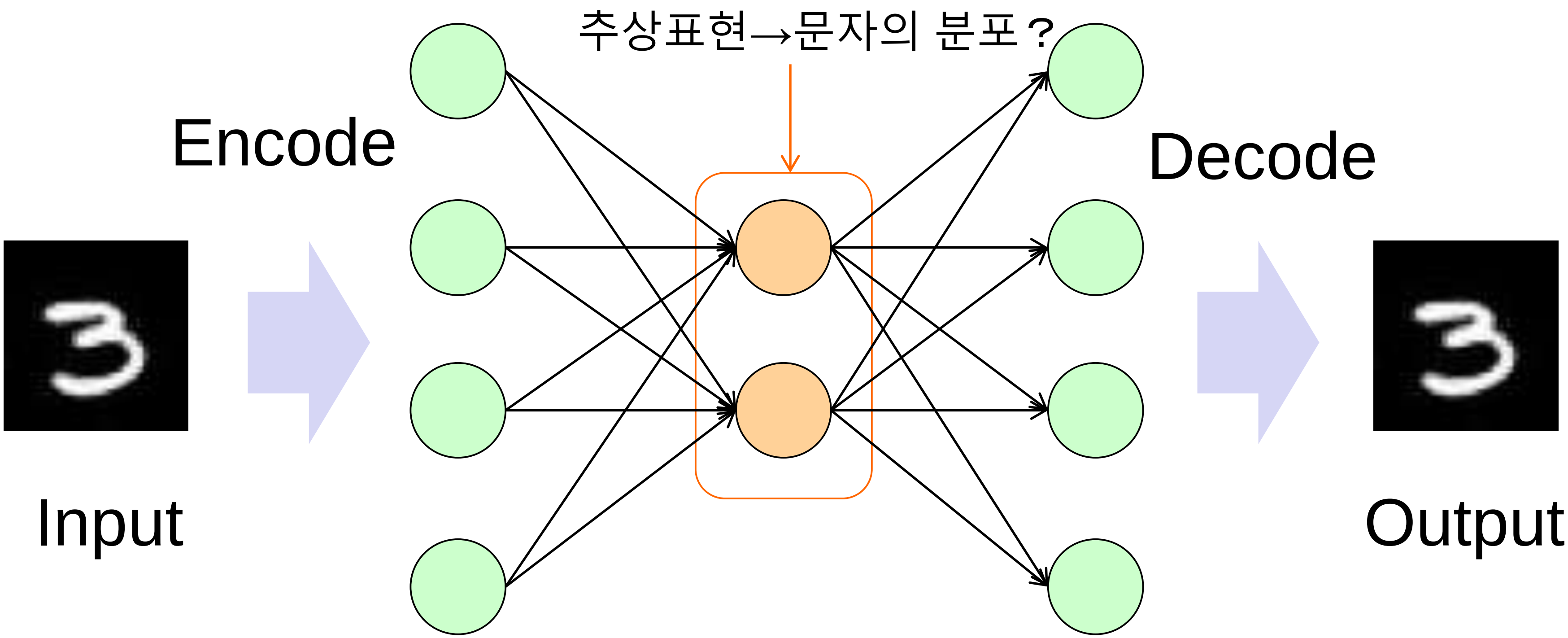
숫자의 생성

VAE의 결론

- Deep Learning을 생성모델에 적용
  - 손글씨나 얼굴표정에 존재하는 잠재변수의 분포를 찾아내고 데이터 셋에 존재하지 않는 자연적인 이미지 생성이 가능

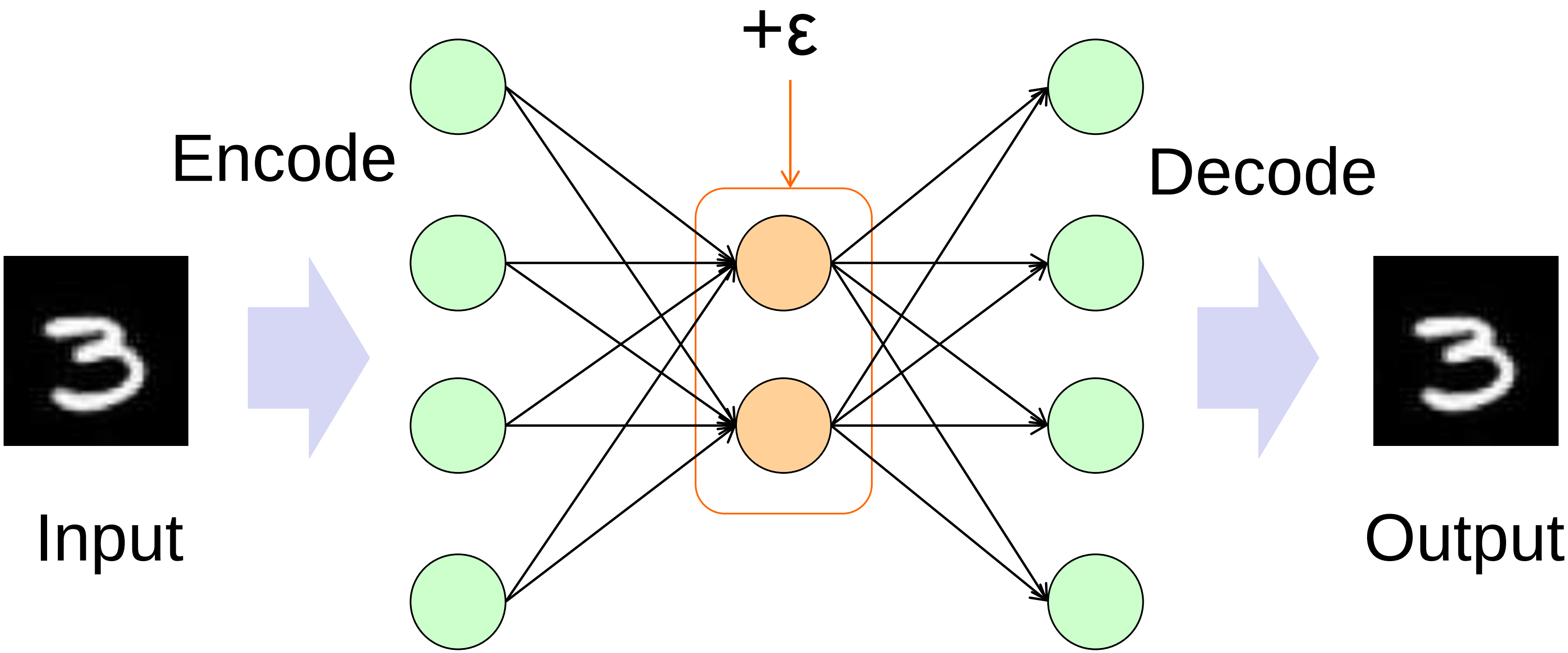


추가자료 : AutoEncoder



- AutoEncoder에 의한 이미지의 압축・재구성
- 중간층에서 이미지의 추상표현이 획득

Valuational AutoEncoder는 ?



- 구조는 AutoEncoder의 중간층에 노이즈를 넣는 것뿐
- loss함수에 정규화 항을 추가
- 구조, 이름은 상당히 유사하나 유래는 다름

참고문헌

- Introduction to variational autoencoders
  - URL: https://home.zhaw.ch/~dueo/bbs/files/vae.pdf

- Deep Advances in Generative Modeling
  - URL: https://www.youtube.com/watch?v=KeJINHjyzOU

- Digit Fantasies by a Deep Generative Model
  - URL: http://www.dpkingma.com/sgvb_mnist_demo/demo.html

- LAPGAN 해설
  - URL: http://www.slideshare.net/hamadakoichi/laplacian-pyramid-of-generative-adversarial-networks-lapgan-nips2015-reading-nipsyomi