AUTO-ENCODING VARIATIONAL BAYES
&
VARIATIONAL GRAPH AUTO-ENCODERS

by
Steffen Rendle

Presented by Sungwon Kim

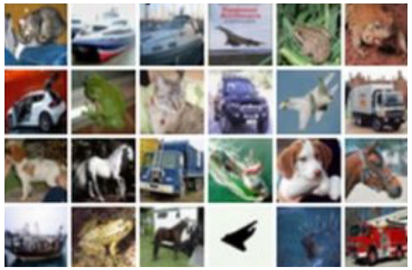
Contents

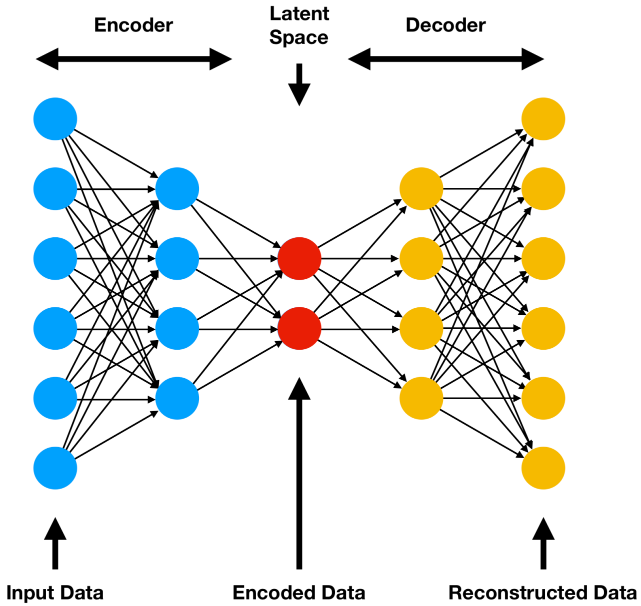
AUTO-ENCODING VARIATIONAL BAYES

Preliminary

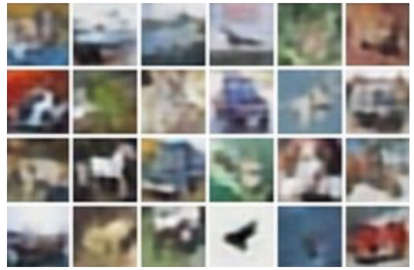
Traditional Auto-Encoders

Input Data





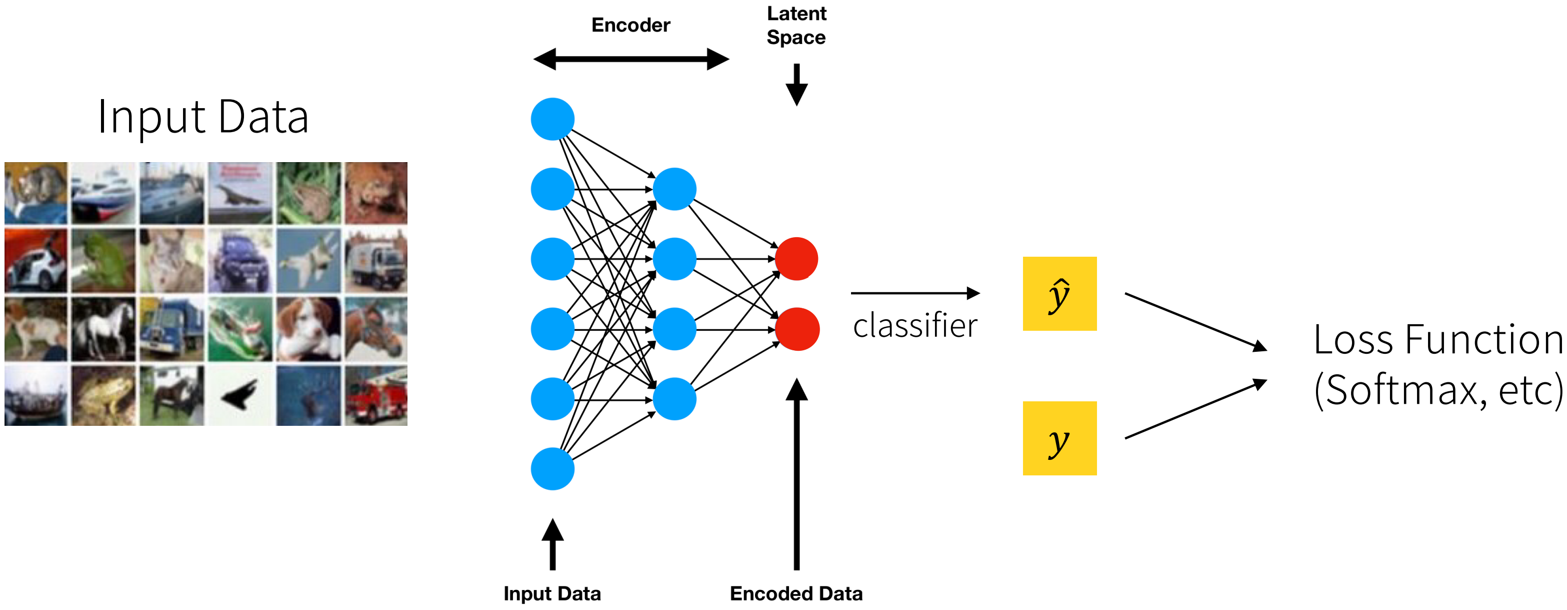
Reconstructed Data



Goal : Learn some underlying hidden structure of the data ➡ **feature learning**

Preliminary

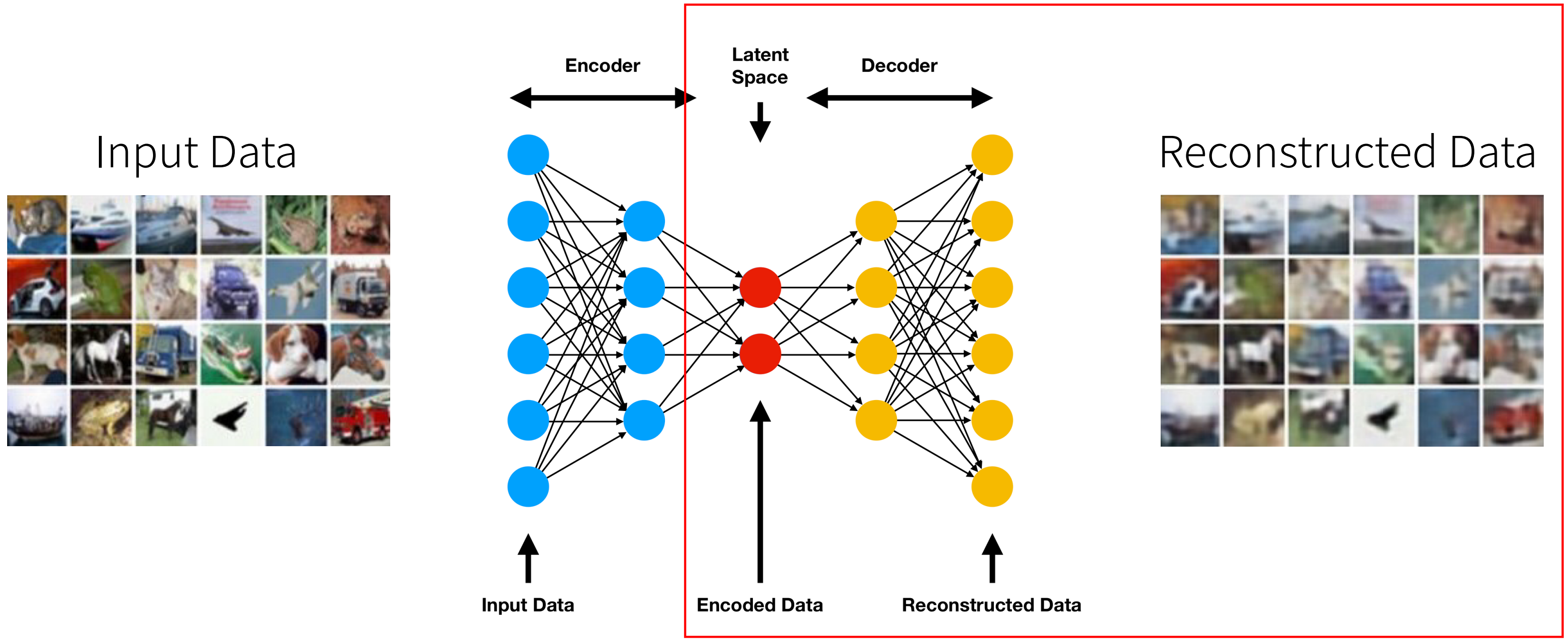
Traditional Auto-Encoders



Fine-tune encoder jointly with supervised model

Preliminary

How to generate new images from an AE?

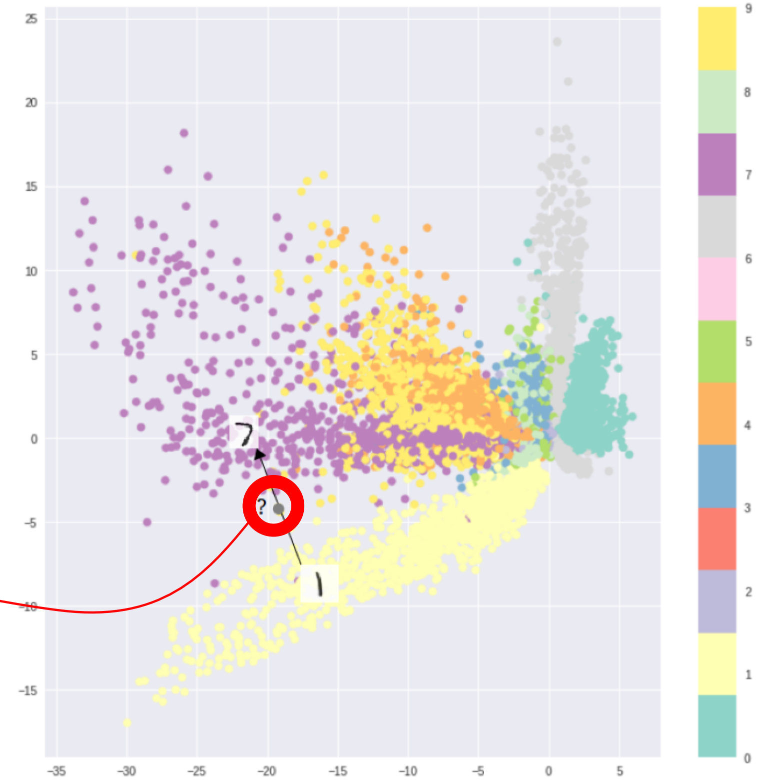


Latent Features ➡ Generate new samples

Preliminary

How to generate new images from an AE?

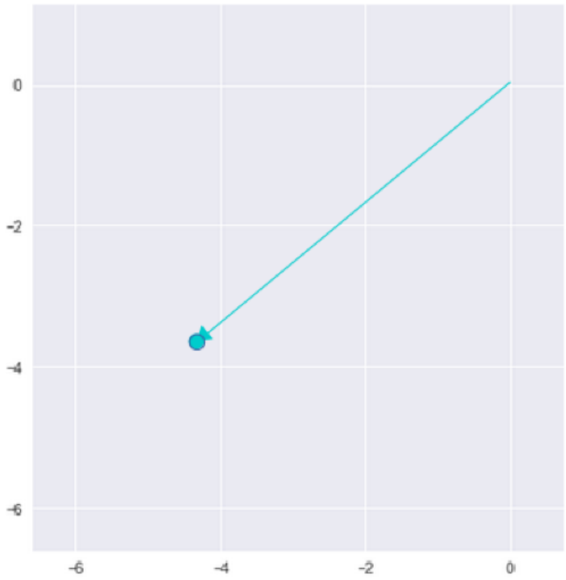
Latent Distribution by Label for AE



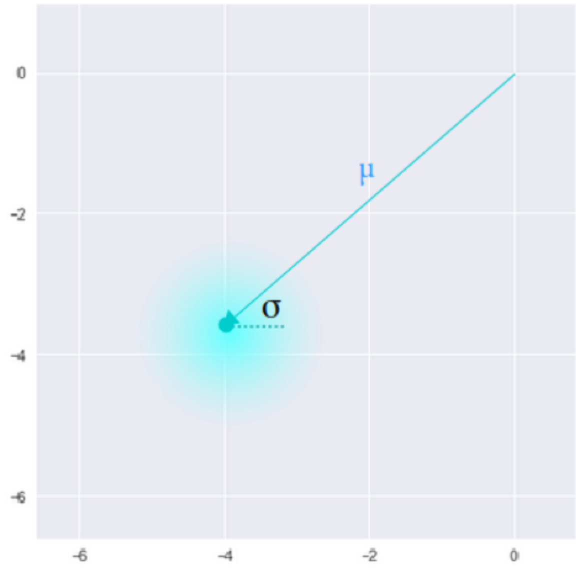
If the space has discontinuities and we sample from there, **the decoder will simply generate unrealistic output.**

Preliminary

Generative Model



Standard Autoencoder
(direct encoding coordinates)



Variational Autoencoder
(μ and σ initialize a probability distribution)

Generative Model

training data

↓

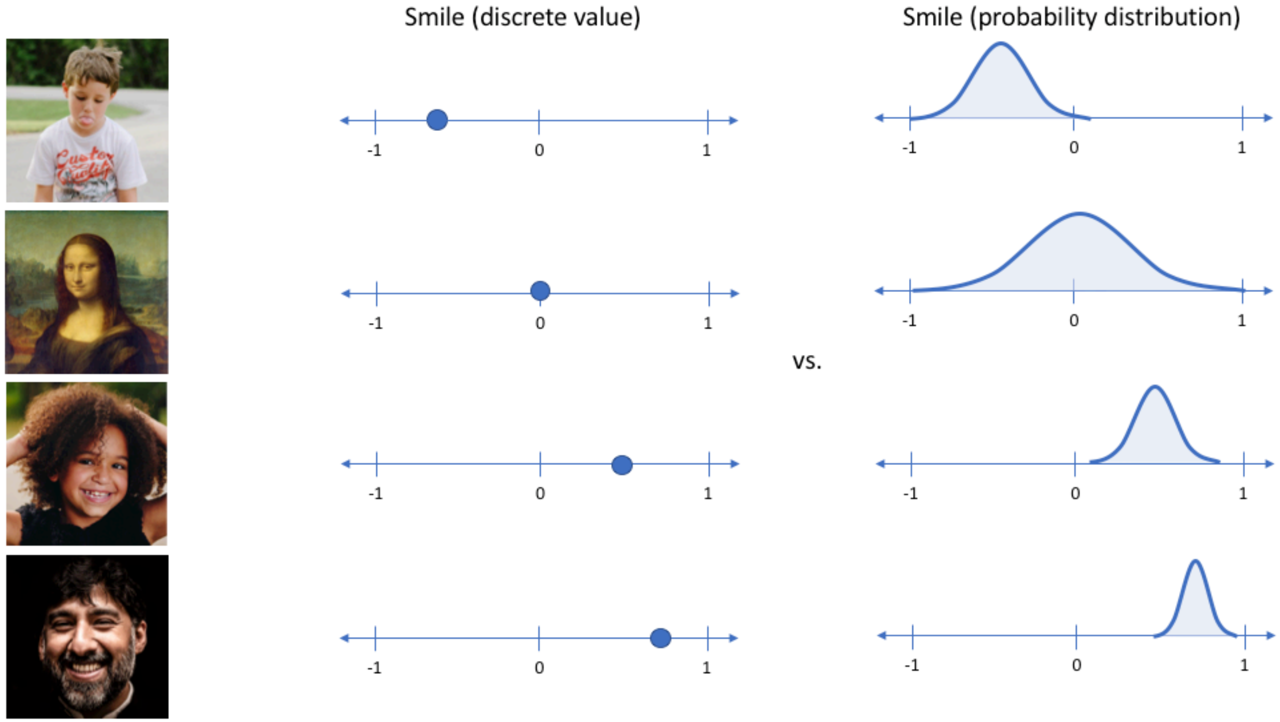
probability distribution

↓

sample

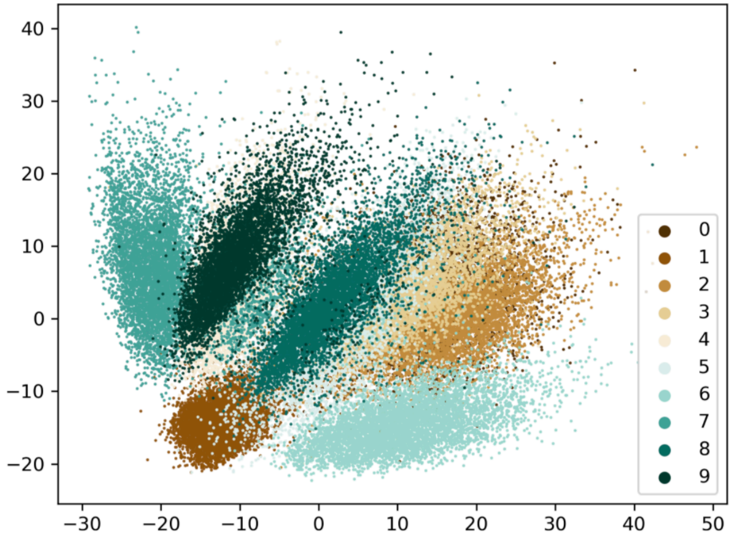
Preliminary

Generative Model

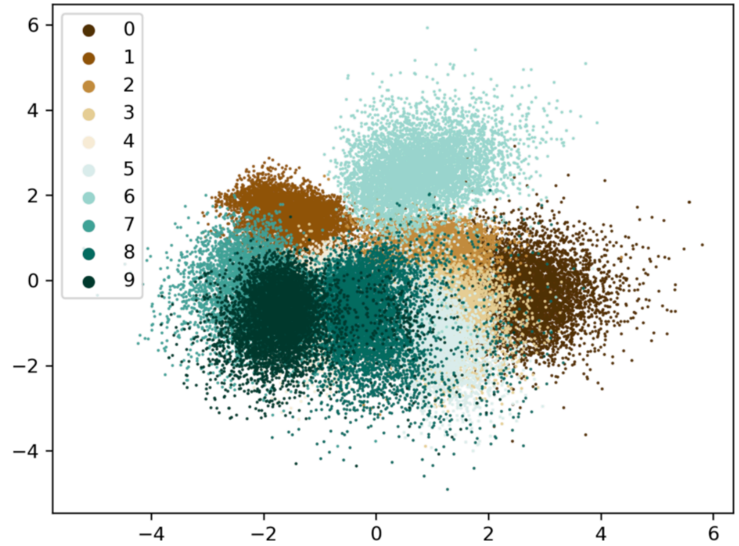


Preliminary

Generative Model



(a) *Latent Distribution by Label for AE*



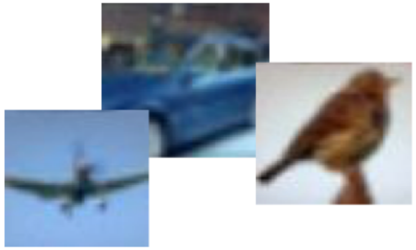
(b) *Latent Distribution by Label for VAE*

VAE build up smaller, spherical clusters which are contiguous and share a similar value range.

Preliminary

Probablistic spin on Autoencoders

Given training data, **generate new samples** from same distribution



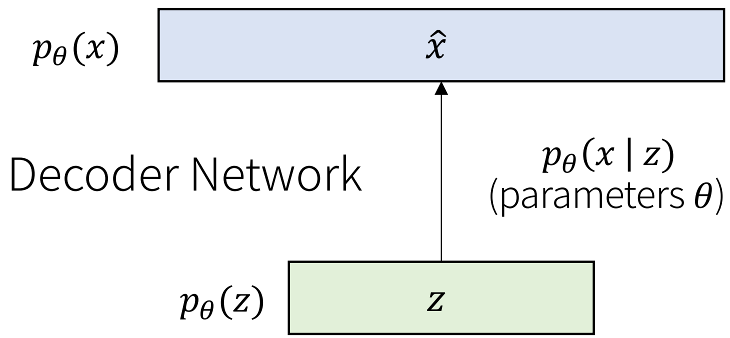
Training data ~ $p_{data}(x)$

Generated data ~ $p_{model}(x)$

Let's approximate the $p_{model}(x)$ distribution close to the $p_{data}(x)$ distribution.

Variational AutoEncoders

Intractable Density Function



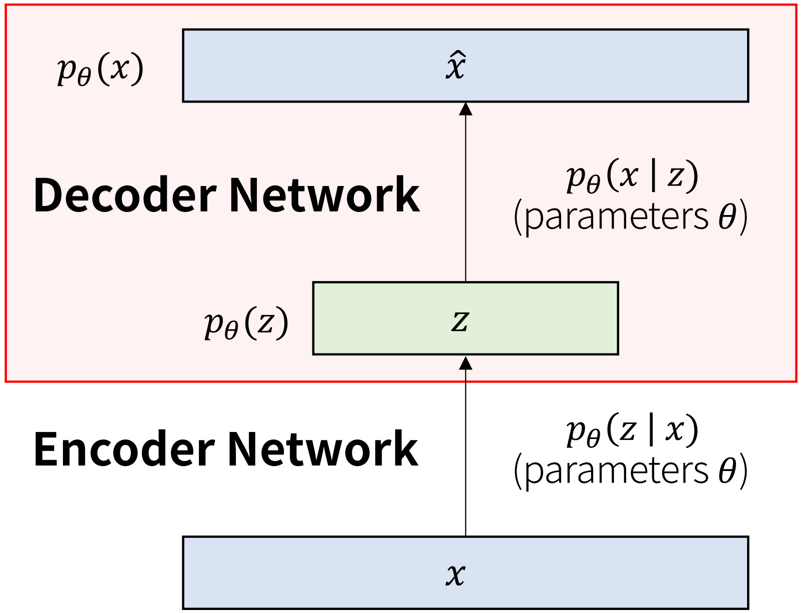
$$p_\theta(x) = \int p_\theta(x \mid z)\, p_\theta(z) dz$$

$= p_{model}(x)$ ➡ **approximate $\mathbf{p_{model}(x)}$ close to $\mathbf{p_{data}(x)}$**

= Maximize Likelihood

Variational AutoEncoders

Intractable Density Function



Data Likelihood

$p_\theta(x) = \int p_\theta(z)\, p_\theta(x \mid z) dz$ Maximize Likelihood!

$\mathbf{p_\theta(z)}$

Resonable한 latent attributes 있다고 가정

➡ Simple Gaussian prior로 가정

$\mathbf{p_\theta(x \mid z)}$

p_θ를 이용하여 $p_\theta(x \mid z)$ 식을 define하는 것은 굉장히 complex함

➡ NN를 이용하여 근사 (parameters θ)

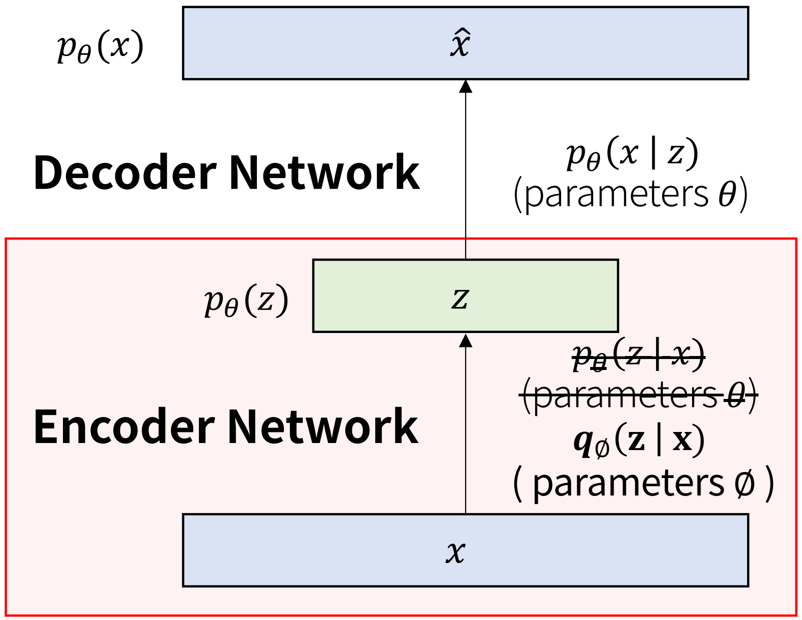
$\int$

Intractable to compute $p_\theta(x \mid z)$ for every z

➡ Reparameterization Trick

Variational AutoEncoders

Intractable Density Function



Decoder

$p_\theta(x) = \int p_\theta(z)\, p_\theta(x \mid z) dz$ Intractable !

Encoder

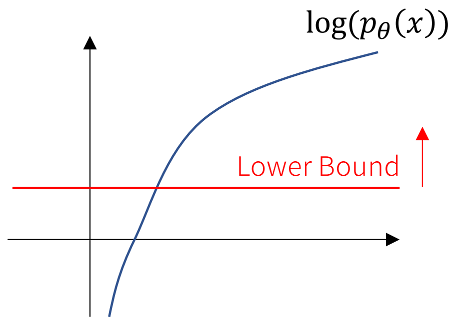
$p_\theta(z \mid x) = \frac{p_\theta(x \mid z)\, p_\theta(z)}{p_\theta(x)}$ (Bayesian Rule)

$\mathbf{p}_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x})$
Posterior density also intractable

➡ NN를 이용하여 근사
$\boldsymbol{q}_\emptyset(\mathbf{z} \mid \mathbf{x})$
(parameters ∅)

then how can we maximize $p_\theta(x)$?
Find the Lower Bound!

Variational AutoEncoders



Lower Bound

$\boxed{\text{Maximize } p_\theta(x)}$

$\log p_\theta(x^{(i)}) \longrightarrow$ monotonically increasing / Maximize 'Lower Bound' = Maximize $p_\theta(x)$

$$= \log\{p_\theta\} \cdot 1 = \log\{p_\theta(x)\} \cdot \int_z q_\phi(z \mid x)\, dz \longrightarrow \text{z에 대한 어떤 확률분포의 모든 합} = 1$$

$$= \int_z \log\{p_\theta(x)\}\, q_\phi(z \mid x)\, dz = \int_z \log\left\{\frac{p_\theta(z,x)}{p_\theta(z|x)}\right\} q_\phi(z \mid x)\, dz = \int_z \log\left\{\frac{p_\theta(z,x)}{q_\phi(z|x)} \frac{q_\phi(z|x)}{p_\theta(z|x)}\right\} q_\phi(z \mid x)\, dz$$

$$= \int_z \log\left\{\frac{p_\theta(z,x)}{q_\phi(z|x)}\right\} q_\phi(z \mid x)\, dz + \int_z \log\left\{\frac{q_\phi(z|x)}{p_\theta(z|x)}\right\} q_\phi(z \mid x)\, dz$$

$$= \int_z \log\left\{\frac{p_\theta(z,x)}{q_\phi(z|x)}\right\} q_\phi(z \mid x)\, dz + D_{KL}(q_\phi(z \mid x)\, ||\, p_\theta(z \mid x)) \longrightarrow KL\ Divergence = \int a \log\frac{a}{b}$$

Variational AutoEncoders

Lower Bound

$\boxed{\text{Maximize } p_\theta(x)}$

$\log p_\theta(x^{(i)})$

$$= \int_z \log\left\{\frac{p_\theta(z,x)}{q_\emptyset(z|x)}\right\} q_\emptyset(z \mid x)\, dz + D_{KL}(q_\emptyset(z \mid x) \,||\, p_\theta(z \mid x))$$

$$= \int_z \log\left\{\frac{p_\theta(z)p_\theta(x|z)}{q_\emptyset(z|x)}\right\} q_\emptyset(z \mid x)\, dz + D_{KL}(q_\emptyset(z \mid x) \,||\, p_\theta(z \mid x))$$

$$= \int_z \log\{p_\theta(x|z)\}\, q_\emptyset(z \mid x)\, dz - \int_z \log\left\{\frac{q_\emptyset(z \mid x)}{p_\theta(z)}\right\} q_\emptyset(z \mid x) dz + D_{KL}(q_\emptyset(z \mid x) \,||\, p_\theta(z \mid x))$$

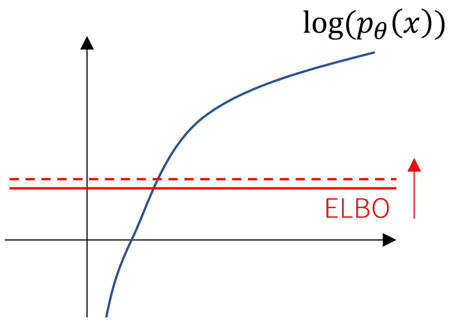
$$= \underbrace{E_{q_\emptyset(z \mid x)}[\log\{p_\theta(x \mid z)\}] - D_{KL}(q_\emptyset(z \mid x) \,||\, p_\theta(z))}_{\text{ELBO (Evidence Lower Bound)}} + \underbrace{D_{KL}(q_\emptyset(z \mid x) \,||\, p_\theta(z \mid x))}_{\geq 0}$$

ELBO (Evidence Lower Bound)

likelihood (decoder)

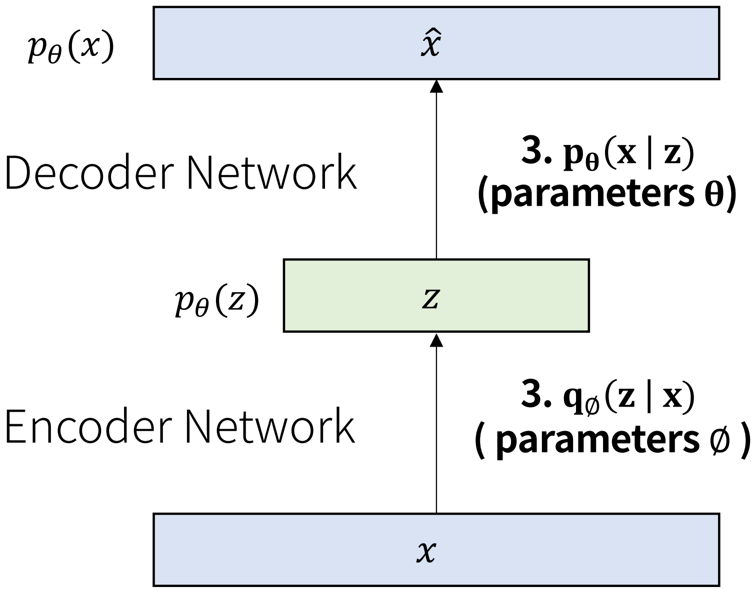
KL term between Gaussian and Encoder

≥ 0



Variational AutoEncoders

Interim Check

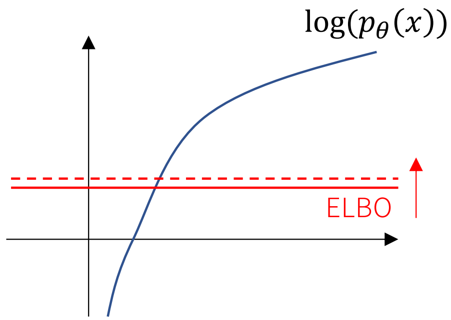


1. approximate $p_{model}(x)$ close to $p_{data}(x)$

2. Maximize $p_\theta(x)$

3. $p_\theta(z)$ ➡ Simple Gaussian prior로 가정

4.



Variational AutoEncoders

Optimization Function

$$\underbrace{E_{q_\emptyset(z\mid x)}[\log\{p_\theta(x\mid z)\}] - D_{KL}(q_\emptyset(z\mid x)\,||\,p_\theta(z))}$$

ELBO ➡ Gradient Ascent ➡ $\boxed{\text{Maximize } p_\theta(x)}$

Final Optimization Function

$$\boxed{\underset{\emptyset,\theta}{\arg\min}(\underbrace{-E_{q_\emptyset(z\mid x)}[\log\{p_\theta(x\mid z)\}]}_{\text{Reconstruction Error}} + \underbrace{D_{KL}(q_\emptyset(z\mid x)\,||\,p_\theta(z))}_{\text{Regularization Error}})}$$

Reconstruction Error
x 의 복원 오차

Regularization Error
샘플링용 함수에 대한 추가 조건

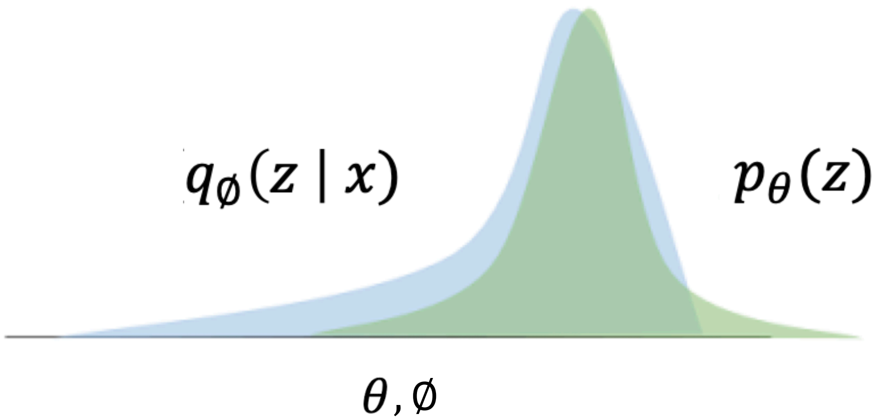
Variational AutoEncoders

Variational Inference

Final Optimization Function

$$\boxed{\underset{\emptyset,\theta}{\arg\min}\left(-E_{q_\emptyset(z\mid x)}[\log\{p_\theta(x\mid z)\}] + \underbrace{D_{KL}(q_\emptyset(z\mid x)\,||\,p_\theta(z))}_{\text{Regularization Error}}\right)}$$

샘플링용 함수에 대한 추가 조건

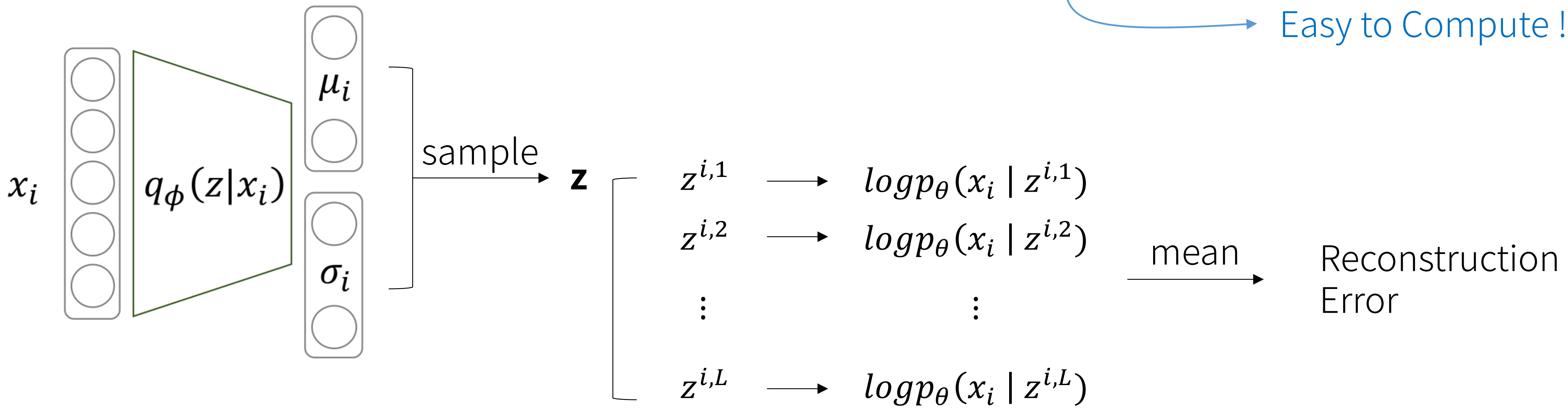


Variational AutoEncoders

Optimization Function

$$\arg\min_{\emptyset,\theta}(\underbrace{-E_{q_\emptyset(z\,|\,x)}[\,\log\{p_\theta(x\mid z)\}\,]}_{\text{Reconstruction Error}} + \underbrace{D_{KL}(q_\emptyset(z\mid x)\,||\,p_\theta(z))}_{\text{Regularization}})$$

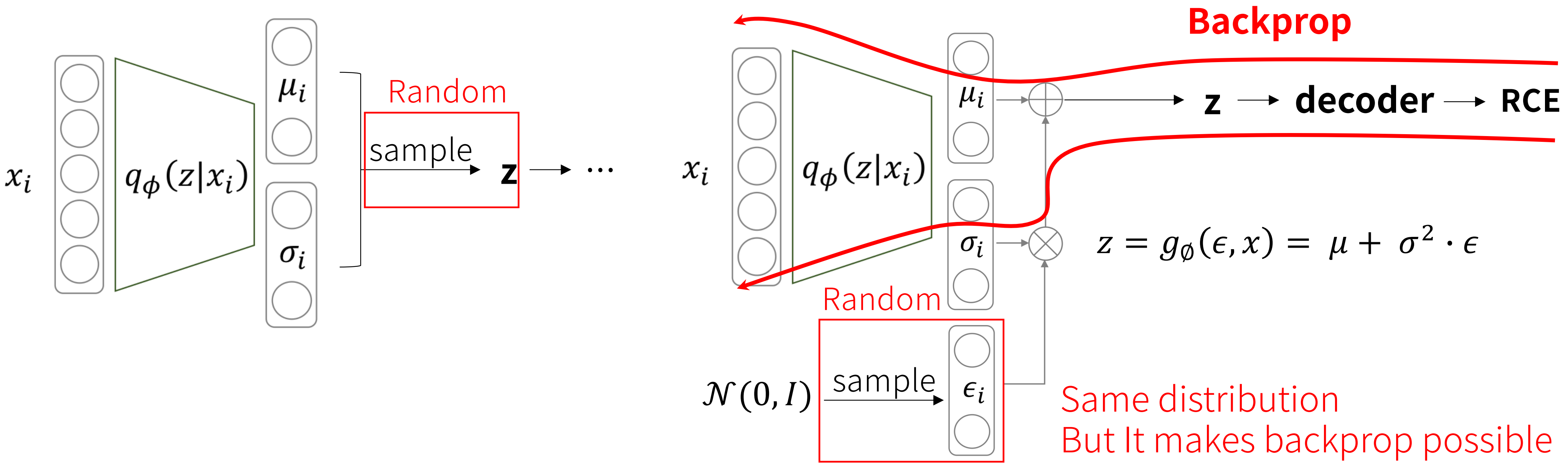
Easy to Compute !



Variational AutoEncoders

Reparameterization Trick

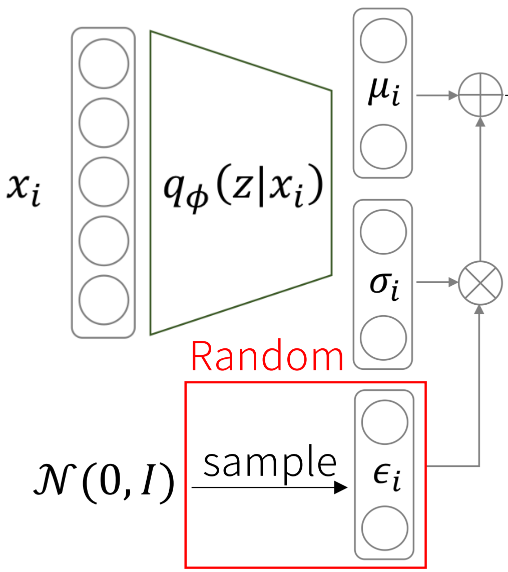
Original Form **Reparameterised Form**

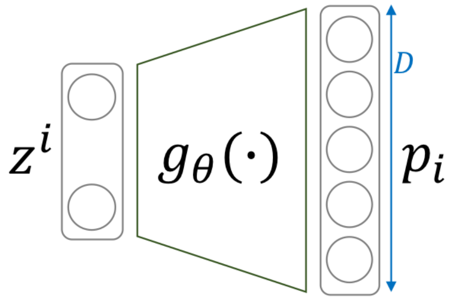


Variational AutoEncoders

Reconstruction Error

Case 1 : Decoder's output = Bernoulli





Reconstruction Error = Cross Entropy

$p_\theta(x_i|z^i) \sim Bernoulli(p_i)$

$$\log\left(p_\theta(x_i|z^i)\right) = \log \prod_{j=1}^{D} p_\theta(x_{i,j}|z^i) = \sum_{j=1}^{D} \log p_\theta(x_{i,j}|z^i)$$

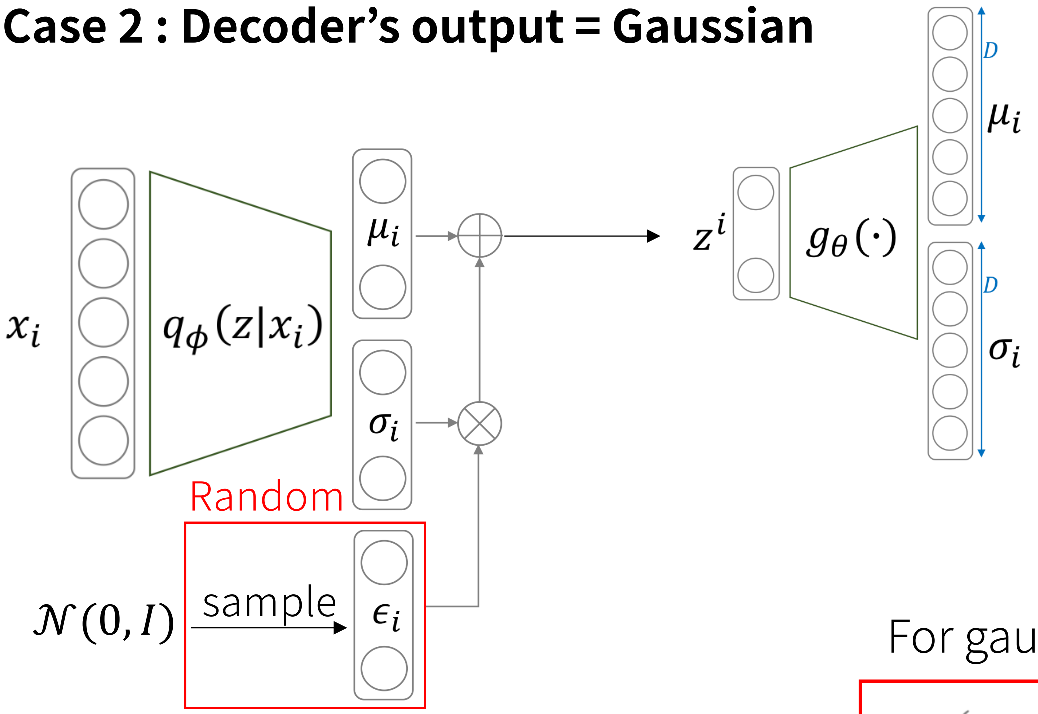
$$= \sum_{j=1}^{D} \log p_{i,j}^{x_{i,j}} (1 - p_{i,j})^{1-x_{i,j}} \quad \leftarrow p_{i,j} \triangleq network\ output$$

$$\boxed{= \sum_{j=1}^{D} x_{i,j} \log p_{i,j} + (1 - x_{i,j}) \log(1 - p_{i,j}) \quad \leftarrow \text{Cross entropy}}$$

Variational AutoEncoders

Reconstruction Error

Case 2 : Decoder's output = Gaussian



Reconstruction Error = Mean Squared Error

$$\log\left(p_\theta\left(x_i|z^i\right)\right) = \log\left(N\left(x_i; \mu_i, \sigma_i^2 I\right)\ \right)$$

$$= -\sum_{j=1}^{D} \frac{1}{2}\log\left(\sigma_{i,j}^2\right) + \frac{\left(x_{i,j} - \mu_{i,j}\right)^2}{2\sigma_{i,j}^2}$$

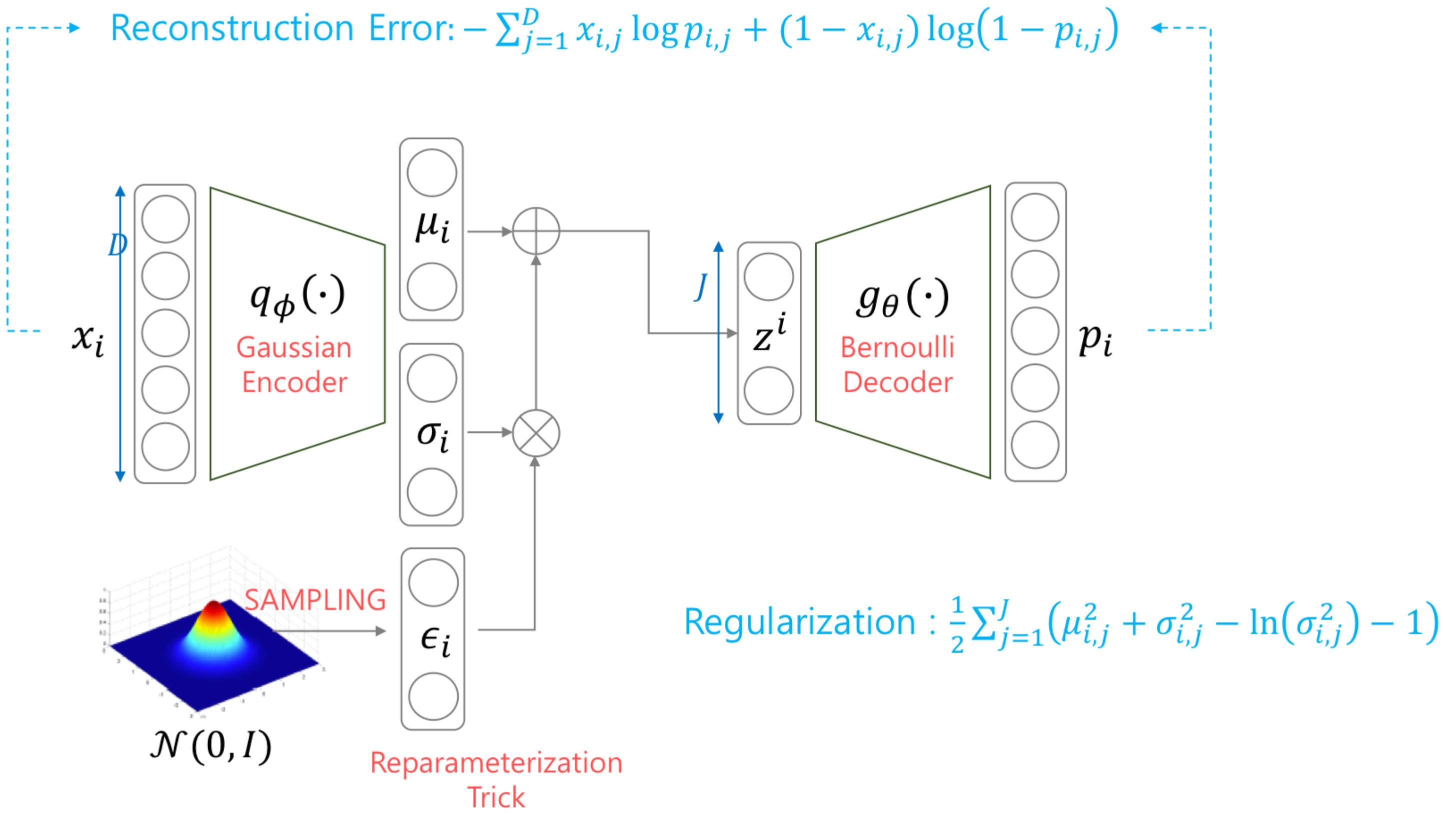
For gaussian distribution with identity covariance

$$\boxed{\log\left(p_\theta\left(x_i|z^i\right)\right) \propto -\sum_{j=1}^{D} \left(x_{i,j} - \mu_{i,j}\right)^2 \quad \leftarrow \text{Squared Error}}$$

Variational AutoEncoders

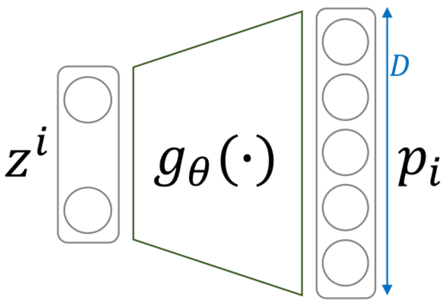
Structure

$$\underset{\emptyset,\theta}{\arg\min}\Big(\underbrace{-E_{q_\emptyset(z\mid x)}[\,\log\{p_\theta(x\mid z)\}\,]}_{\text{Reconstruction Error}} + \underbrace{D_{KL}(q_\emptyset(z\mid x)\,||\,p_\theta(z))}_{\text{Regularization}}\Big)$$



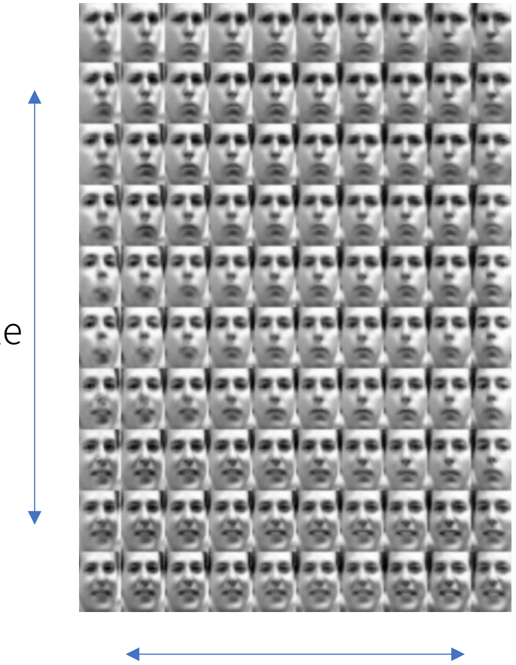
Variational AutoEncoders

Result



$p_\theta(x_i|z^i) \sim Bernoulli(p_i)$

vary z_1
Degree of smile



vary z_2
Head Pose

```
model.eval()

with torch.no_grad():
    # (0,1) Gaussian sampling
    z = torch.randn(args.num_predict,n_output).to(device)
    sample = model.bernoulli_decoder(z)
    save_image(sample.view(args.num_predict,1,28,28), f'./sample/{args.fname}')
```

Variational AutoEncoders

Summary

1. Probablistic spin on autoencoders - Generative Models
2. Intractable density → NN, optimize a lower bound
3. Reparameterization Trick for back propagation

- Useful latent representation, inference queries.
- Samples blurrier and lower quality compared to GANs

VARIATIONAL GRAPH AUTO-ENCODERS

VARIATIONAL GRAPH AUTO-ENCODERS

Graph AutoEncoder (GAE)

AutoEncoder (AE)

↓

Latent Representation
목적 : **Encoder** > Decoder

Variational AutoEncoder (VAE)

↓

Generative Model
목적 : **Decoder** > Encoder

Trick

Graph AutoEncoder (GAE)

↓

Latent Representation
목적 : **Encoder** > Decoder

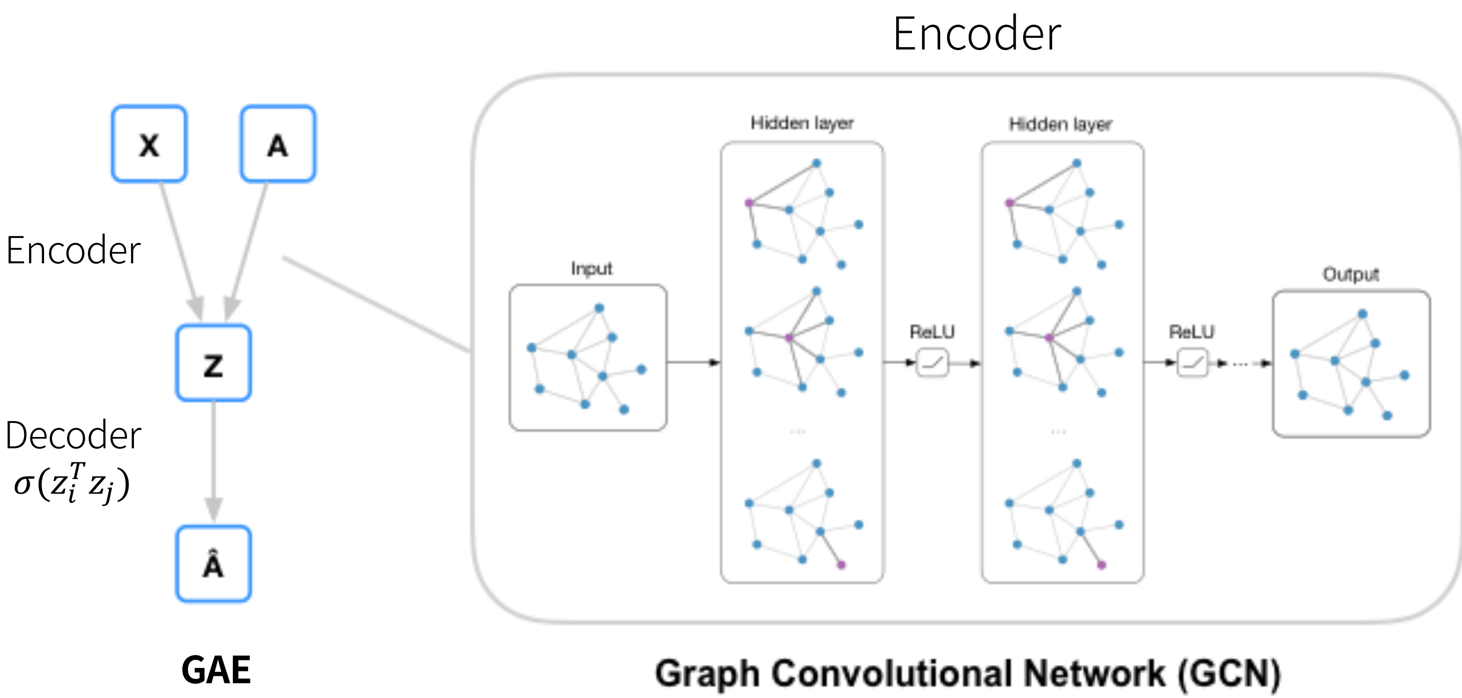
Variational Graph AutoEncoder (VGAE)

↓

Latent Representation > Generative Model
목적 : **Encoder** > Decoder

VARIATIONAL GRAPH AUTO-ENCODERS

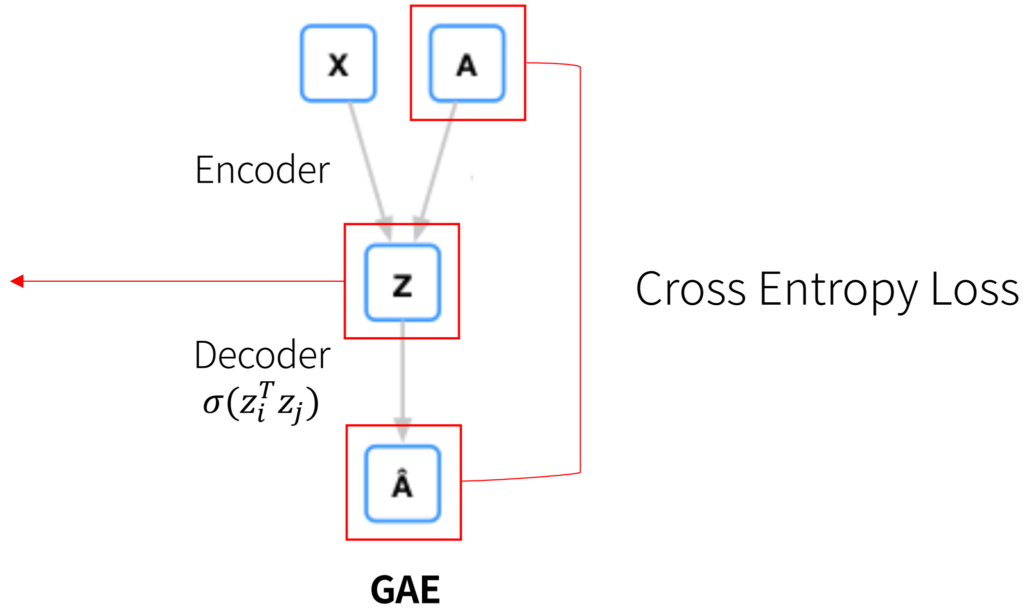
Graph AutoEncoder (GAE)



VARIATIONAL GRAPH AUTO-ENCODERS

Graph AutoEncoder (GAE)

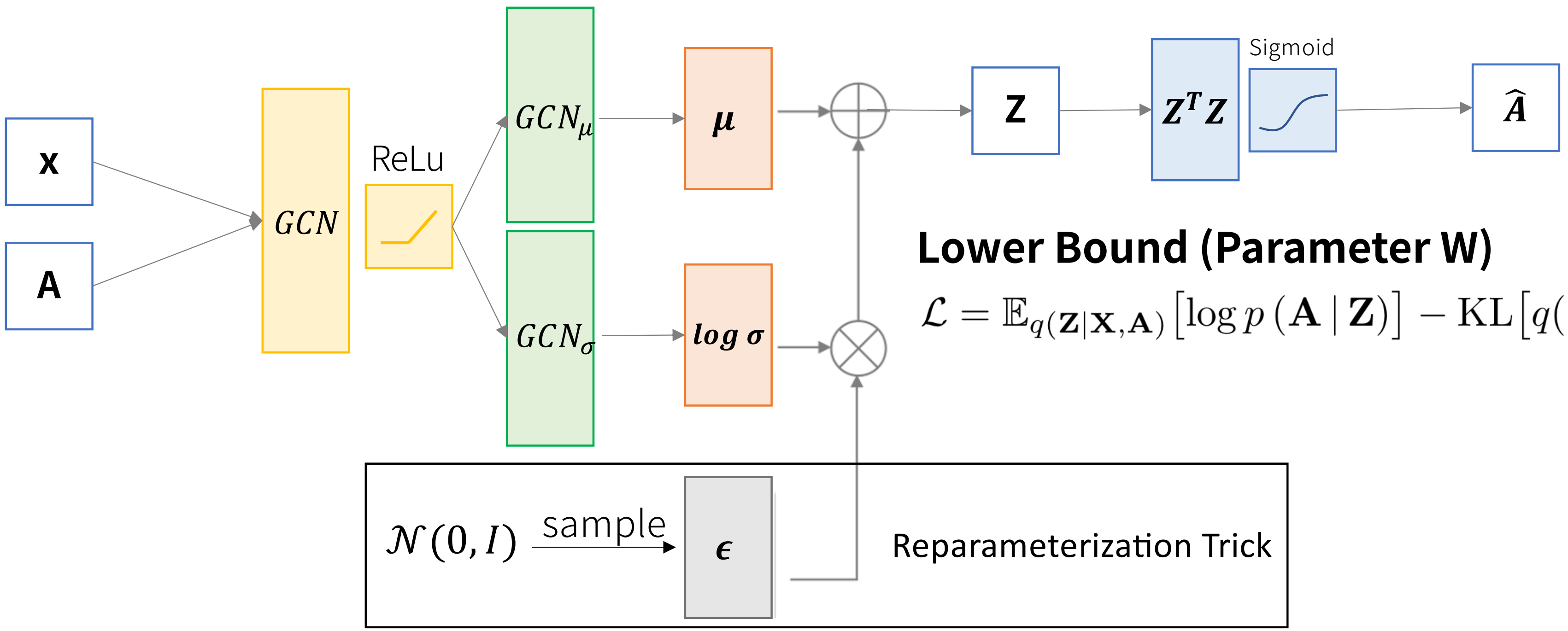
Latent Feature
- Clusterting
- Link Prediction
- ...



VARIATIONAL GRAPH AUTO-ENCODERS

VGAE Structure

$$GCN(X, A) = \tilde{A}\, ReLu(\tilde{A} X W_0)\, W_1$$



Lower Bound (Parameter W)

$$\mathcal{L} = \mathbb{E}_{q(\mathbf{Z}|\mathbf{X},\mathbf{A})}\left[\log p\left(\mathbf{A} \mid \mathbf{Z}\right)\right] - \mathrm{KL}\left[q(\mathbf{Z} \mid \mathbf{X}, \mathbf{A}) \,||\, p(\mathbf{Z})\right]$$

VARIATIONAL GRAPH AUTO-ENCODERS

Experiments

Table 1: Link prediction task in citation networks. See [1] for dataset details.

Method	**Cora**		**Citeseer**		**Pubmed**	
	AUC	AP	AUC	AP	AUC	AP
SC [5]	84.6 ± 0.01	88.5 ± 0.00	80.5 ± 0.01	85.0 ± 0.01	84.2 ± 0.02	87.8 ± 0.01
DW [6]	83.1 ± 0.01	85.0 ± 0.00	80.5 ± 0.02	83.6 ± 0.01	84.4 ± 0.00	84.1 ± 0.00
GAE*	84.3 ± 0.02	88.1 ± 0.01	78.7 ± 0.02	84.1 ± 0.02	82.2 ± 0.01	87.4 ± 0.00
VGAE*	84.0 ± 0.02	87.7 ± 0.01	78.9 ± 0.03	84.1 ± 0.02	82.7 ± 0.01	87.5 ± 0.01
GAE	91.0 ± 0.02	92.0 ± 0.03	89.5 ± 0.04	89.9 ± 0.05	$\mathbf{96.4} \pm 0.00$	$\mathbf{96.5} \pm 0.00$
VGAE	$\mathbf{91.4} \pm 0.01$	$\mathbf{92.6} \pm 0.01$	$\mathbf{90.8} \pm 0.02$	$\mathbf{92.0} \pm 0.02$	94.4 ± 0.02	94.7 ± 0.02

SUMMARY

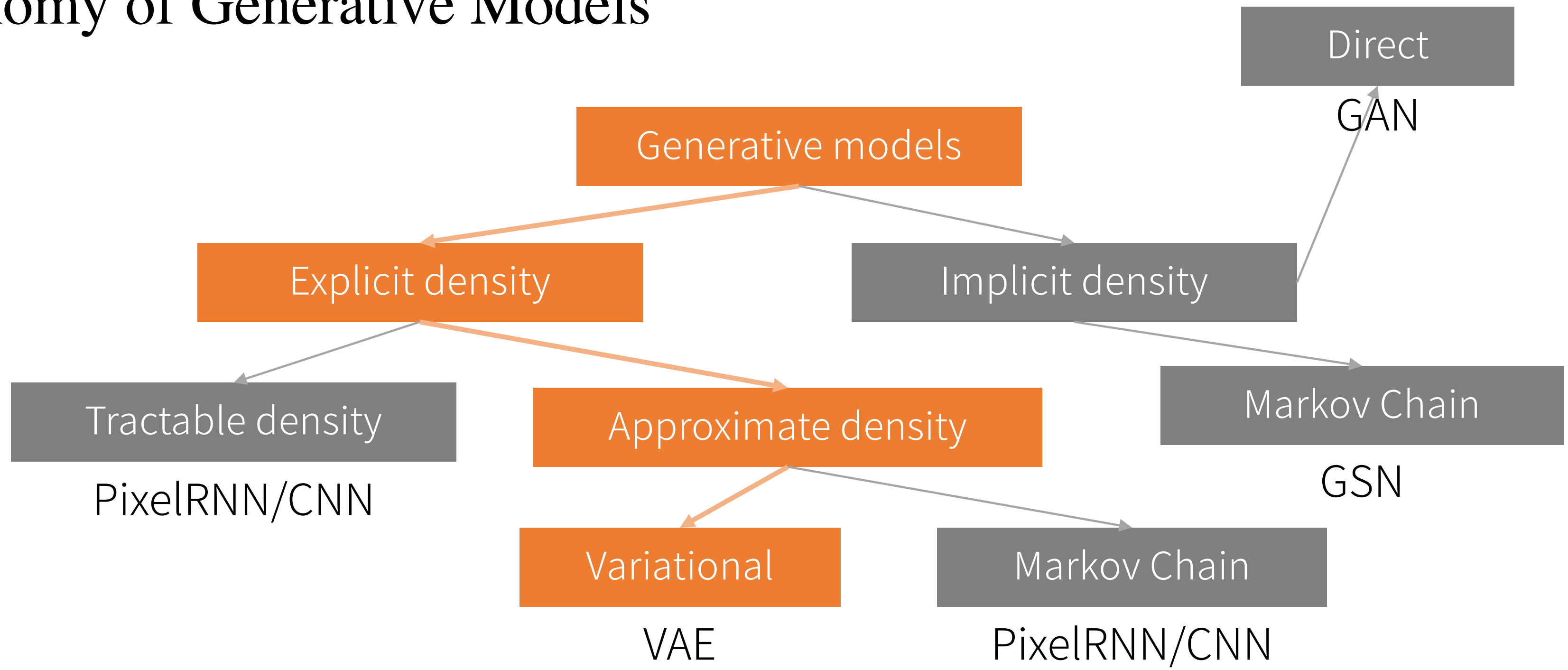
1. VGAE는 GAE에 확률분포를 적용시킨 모델
2. 더 좋은 Latent Representation 성능을 보임 – e.g. good at link prediction

DISCUSSION

1. Prior를 가우시안으로 가정하는 경우 decoder에서 inner product를 쓰면 조합이 좋지 않다.
2. For very sparse A, it can be beneficial to re-weight terms with Aij = 1 in L or alternatively sub-sample terms with Aij = 0?

APPENDIX

Taxonomy of Generative Models



VAE *does not* define the density function since it has some intractable terms.
➡ can be approximated using NN !