Auto-Encoding Variational Bayes

Kingma, Diederik P., and Max Welling

Arxiv

https://arxiv.org/abs/1312.6114

Presenter : Minho Park

Demo of VAE

- http://dpkingma.com/sgvb_mnist_demo/demo.html

Kingma MNIST demo: https://youtu.be/0QI3xgXuB-Q

Purpose of Generative Model

- Learning the true data distribution of the training set
- Sample x from $p(x)$

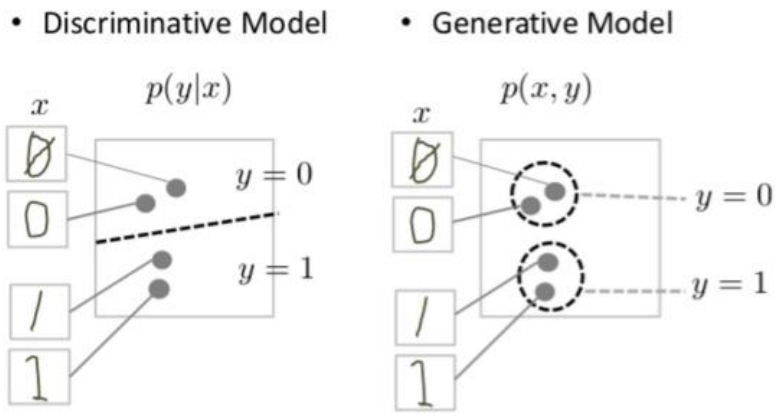


Figure 1: Discriminative and generative models of handwritten digits.

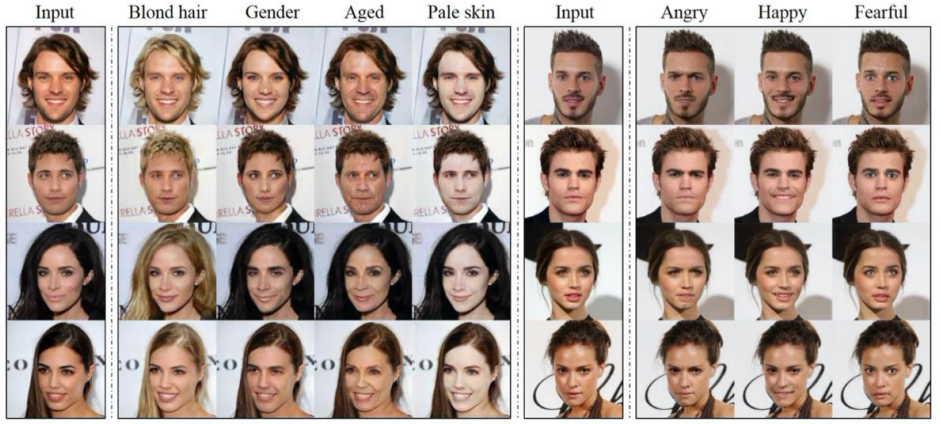
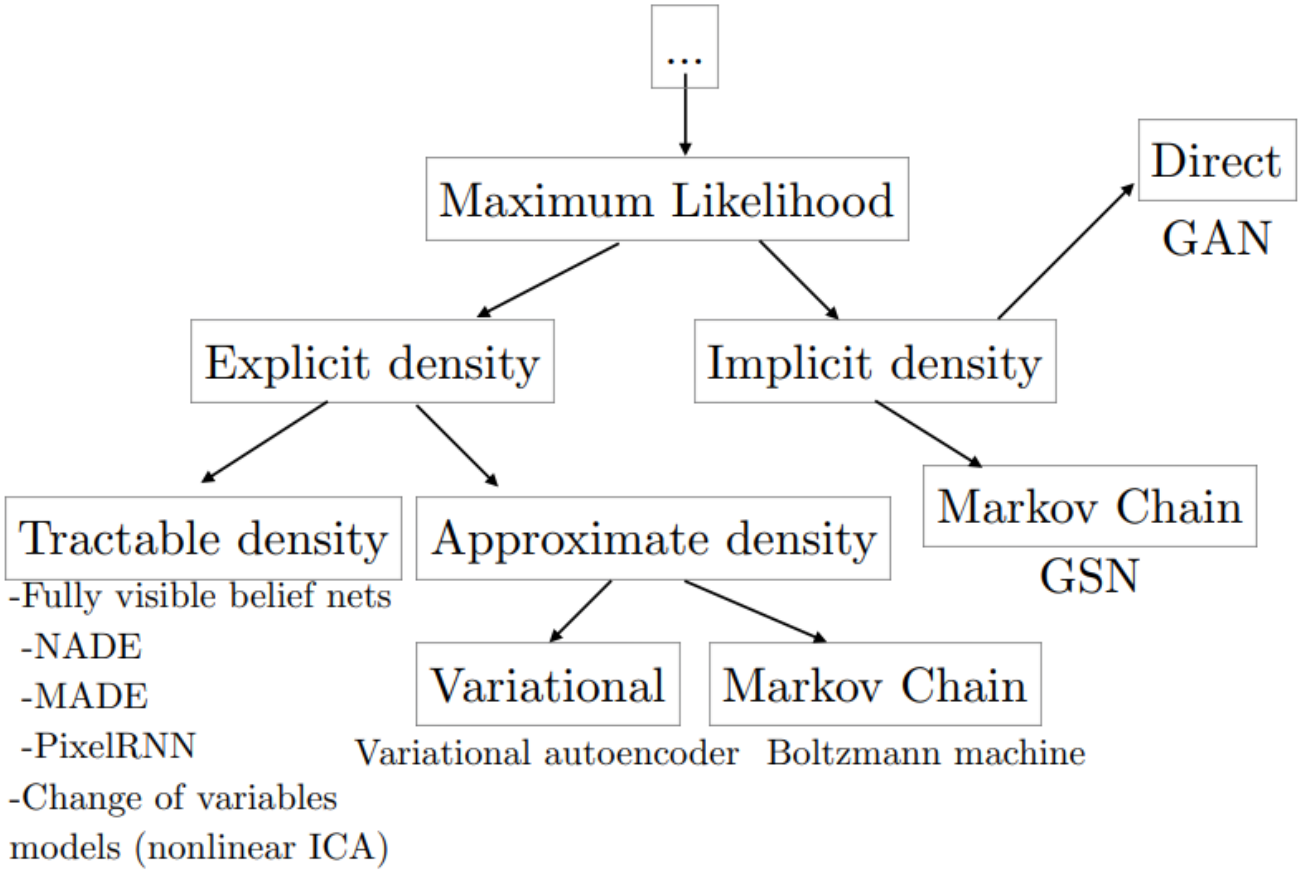


Figure 1. Multi-domain image-to-image translation results on the CelebA dataset via transferring knowledge learned from the RaFD dataset. The first and sixth columns show input images while the remaining columns are images generated by StarGAN. Note that the images are generated by a single generator network, and facial expression labels such as angry, happy, and fearful are from RaFD, not CelebA.

Background: What is a Generative Model?, https://developers.google.com/machine-learning/gan/generative
Yunjey Choi et al., StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation, https://arxiv.org/abs/1711.09020

Taxonomy

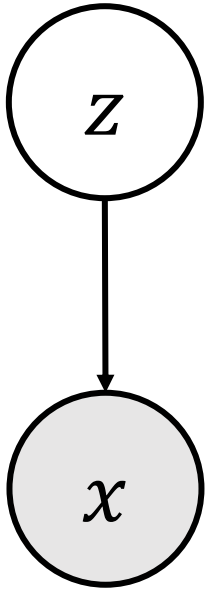


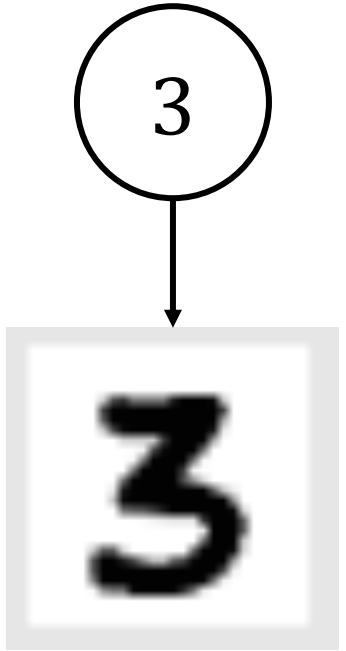
Ian Goodfellow, NIPS 2016 Tutorial: Generative Adversarial Networks, https://arxiv.org/abs/1701.00160

Latent Variable Model

- Generate handwritten characters example:

$$p(X) = \int p(x|z;\theta)p(z)dz$$





Yann LeCun et al., MNIST, http://yann.lecun.com/exdb/mnist/

Decoder of VAE

- In VAEs, output distribution is often Gaussian.
- We can use gradient descent to increase $p(x)$ by making $f(z;\theta)$ approach x for some z.

$$p(x) = \int \underbrace{p(x|z;\theta)}_{\text{In VAE, } \mathcal{N}(x|f(z;\theta),\sigma^2 * I)} p(z)dz$$

Two Problems

$$\boldsymbol{p(x) = \int p(x|z;\theta)p(z)dz}$$

- **How to define the latent variables z.**
  - It needs to choose not just the digit, but the angle that the digit is drawn, the stroke width, and also abstract stylistic properties. Worse, these properties may be correlated.
- **How to deal with the integral over z.**
  - Can we use Monte Carlo method? Sample a large number of z values and compute $p(x) = \frac{1}{n}\sum_{i=1}^{n} p(x|z_i)$.
  - In high dimensional spaces, n might need to be extremely large.

How to define the latent variables z

- Assume that there is no simple interpretation of the dimension of z.
- Instead assert the samples of z can be drawn from a simple distribution, i.e., $\mathcal{N}(0, I)$, where I is the identity matrix.

How to define the latent variables z

- How is this possible?

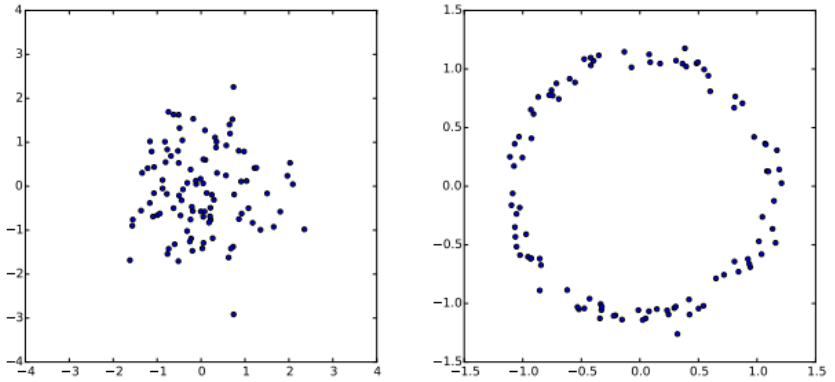
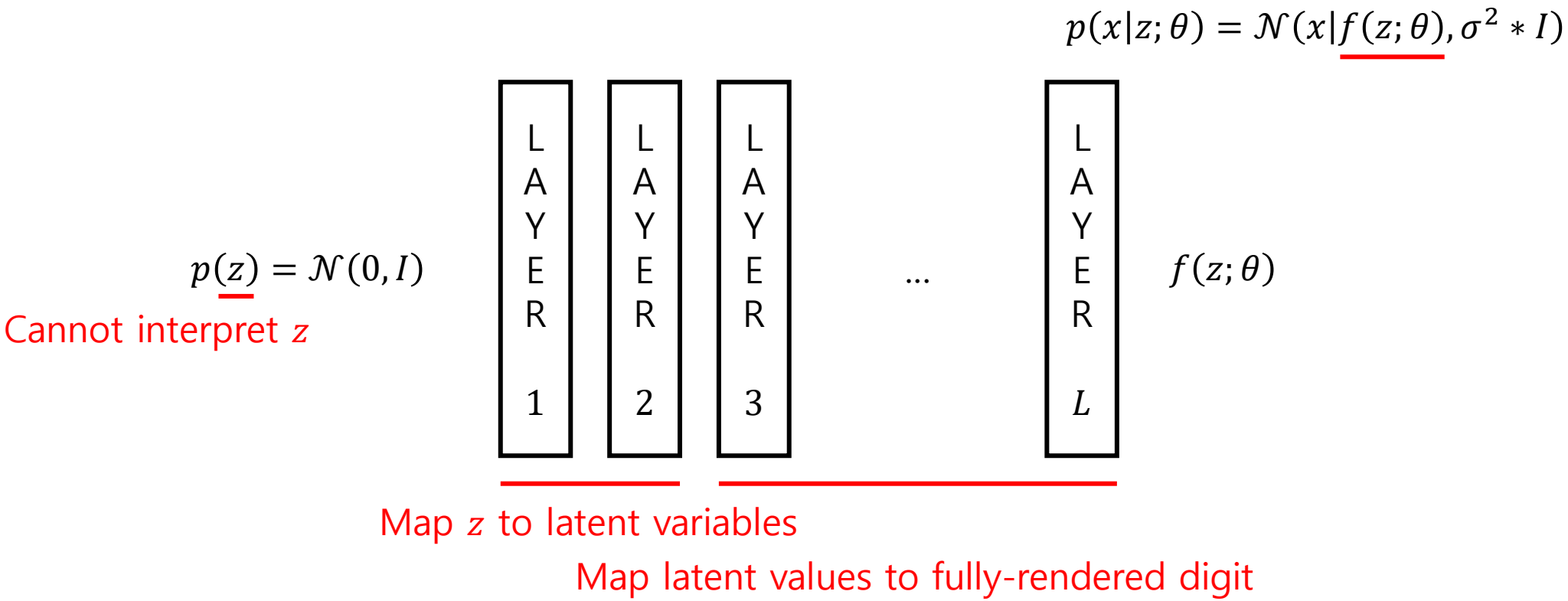


Figure 2: Given a random variable z with one distribution, we can create another random variable $X = g(z)$ with a completely different distribution. Left: samples from a gaussian distribution. Right: those same samples mapped through the function $g(z) = z/10 + z/||z||$ to form a ring. This is the strategy that VAEs use to create arbitrary distributions: the deterministic function g is learned from data.

Carl Doersch, Tutorial on Variational Autoencoders: https://arxiv.org/abs/1606.05908

How to define the latent variables z

- Imagine the network using MLP.



How to deal with the integral over z

- Can we use Monte Carlo method? Sample a large number of z values and compute $p(x) = \frac{1}{n}\sum_{i=1}^{n} p(x|z_i)$.
- In high dimensional spaces, n might need to be extremely large.

How to deal with the integral over z

- Set the σ hyperparameter such that erroneous digit (Figure 3(b)) does not contribute to $p(x)$.
- On the other hand, Figure 3(c) would contribute to $p(x)$.

(a) (b) .0387 (c) .2693

Figure 3: It's hard to measure the likelihood of images under a model using only sampling.

$$p(x|z;\theta) = \mathcal{N}(x|f(z;\theta), \sigma^2 * I)$$

Decrease σ,
Increase n extremely.
or using a better similarity metric

Carl Doersch, Tutorial on Variational Autoencoders: https://arxiv.org/abs/1606.05908

How to deal with the integral over z

- VAEs alter the sampling procedure without changing the similarity metric.

Sampling procedure

- For most z, $p(x|z) \approx 0$, and hence contribute almost nothing to our estimate of $p(x)$.
- Sample values of z that are likely to have produced x, and compute $p(x)$ just from those.

Importance Sampling

- Solving integral unknown distribution problem with known distribution

Integral problem $$m = \mathbb{E}_{p(z)}[f(z)] = \int f(z)p(z)dz$$

Identity trick $$m = \int f(z)p(z)\frac{q(z)}{q(z)}dz$$

Re-group/re-weight $$m = \int f(z)\frac{p(z)}{q(z)}q(z)dz = \mathbb{E}_{q(z)}\left[f(z)\frac{p(z)}{q(z)}\right]$$

Importance Sampling

- Monte Carlo Estimator
- + Markov Chain Sampling method ⇒ MCMC

$$m = \mathbb{E}_{q(z)}\left[f(z)\frac{p(z)}{q(z)}\right]$$

Sampling $\quad w^{(s)} = \frac{p(z)}{q(z)}, \qquad z^{(s)} \sim q(z)$

Monte Carlo Estimator $\quad m = \frac{1}{S}\sum_{s} w^{(s)} f\left(z^{(s)}\right)$

$$\log m = \log\left(\sum_{s} w^{(s)} f\left(z^{(s)}\right)\right) - \log S$$

Our Integral Problem

- $p(x) = \int p(x|z)p(z)dz$
- Needs Extremely large samples

$$m = \mathbb{E}_{q(z)}\left[p(x|z)\frac{p(z)}{q(z)}\right]$$

Sampling $\quad w^{(s)} = \frac{p(z)}{q(z)}, \qquad z^{(s)} \sim q(z)$

Monte Carlo Estimator $\quad m = \frac{1}{S}\sum_{s} w^{(s)} p\left(x\middle|z^{(s)}\right)$

$$\log m = \log\left(\sum_{s} w^{(s)} p\left(x\middle|z^{(s)}\right)\right) - \log S$$

Variational Inference

- Importance sampling to variational inference

$$p(x) = \int p(x|z)\frac{p(z)}{q(z)}q(z)dz$$

Jensen's inequality
$\log \mathbb{E}_{p(x)}[g(x)] \geq \mathbb{E}_{p(x)}[\log g(x)]$

$$\log p(x) \geq \int q(z) \log\left(p(x|z)\frac{p(z)}{q(z)}\right)dz$$

$$= \int q(z) \log p(x|z)\, dz - \int q(z) \log\frac{q(z)}{p(z)}\, dz$$

Variational Lower Bound
Evidence Lower Bound (ELBO)

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

Variational Lower Bound

$$\mathcal{F}(x, q) = \mathbb{E}_{q(z)}[\log p(x|z)] - KL[q(z) \parallel p(z)]$$
$$= \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - KL\left[q_\phi(z|x) \parallel p_\theta(z)\right]$$

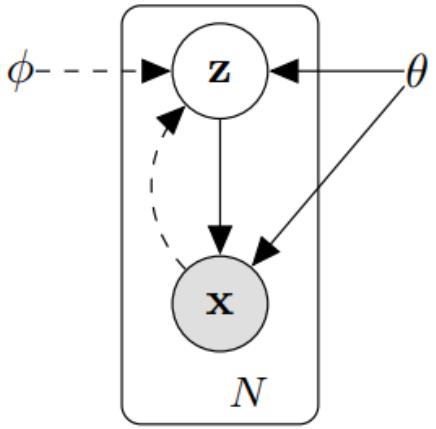


Figure 1: The type of directed graphical model under consideration.

Diederik P Kingma, Max Welling, Auto-Encoding Variational Bayes: https://arxiv.org/abs/1312.6114

Difference between ELBO and $\log p(x)$

$$\log p(x) - ELBO(\phi) = \log p(x) - \left(\int q_\phi(z|x) \log p(x|z)\, dz - \int q_\phi(z|x) \log \frac{q_\phi(z|x)}{p(z)}\, dz \right)$$

$$= \log p(x) - \int q_\phi(z|x) \log \frac{p(x|z)p(z)}{q_\phi(z|x)}\, dz$$

$$= \log p(x) - \int q_\phi(z|x) \log p(x) + q_\phi(z|x) \log \frac{p(z|x)}{q_\phi(z|x)}\, dz$$

$$= \int q_\phi(z|x) \log \frac{p(z|x)}{q_\phi(z|x)}\, dz$$

$$= KL\left(q_\phi(z|x) \parallel p(z|x) \right)$$

EM (Expectation-Maximization)

- EM algorithm is a general technique for finding the Maximum Likelihood solution of a latent variable model.
- For example, EM for GMM (Supplementary)

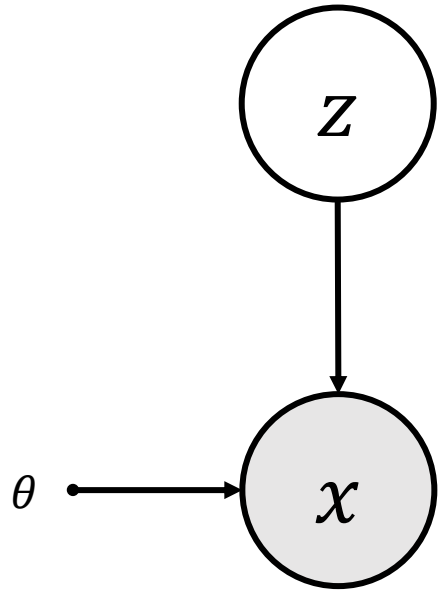
EM (Expectation-Maximization)

$$\log p(x|\theta) = \mathcal{L}(q,\theta) + KL(q \parallel p)$$

$$= \sum_z q(z) \log\left(\frac{p(x,z|\theta)}{q(z)}\right) + \sum_z q(z) \log\left(\frac{q(z)}{p(z|x,\theta)}\right)$$

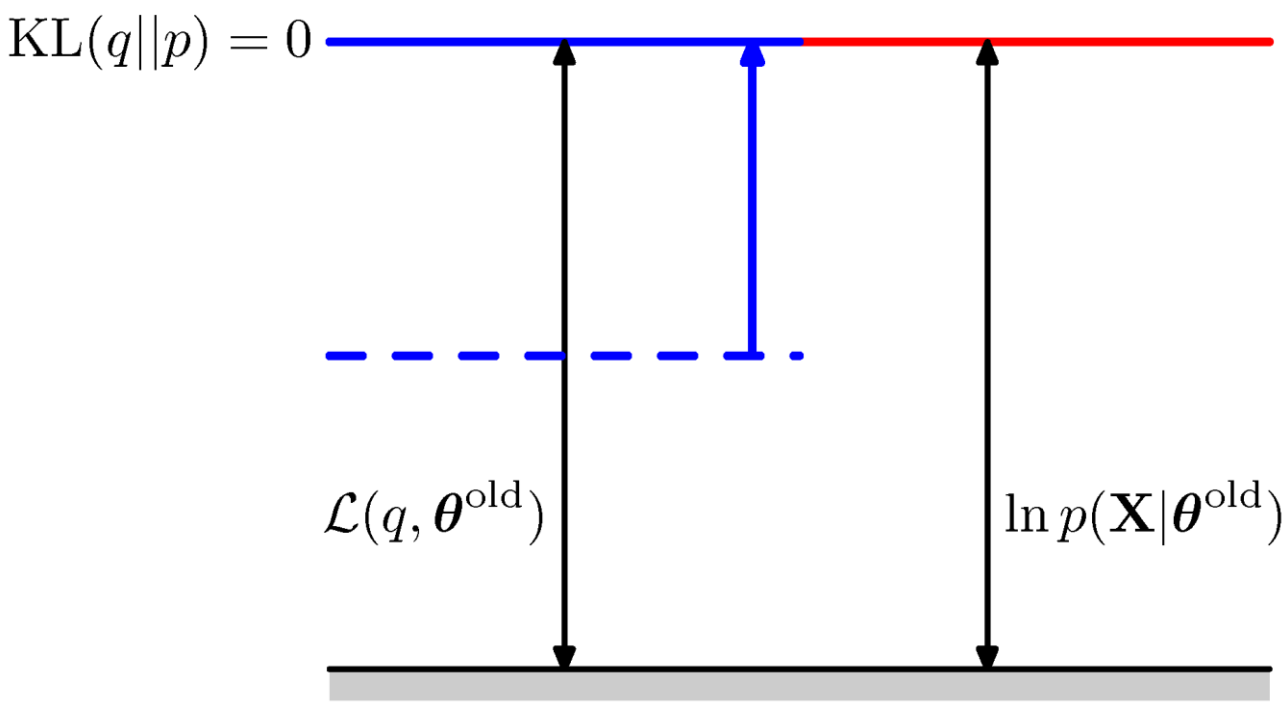
- **E-step:** Fix θ. Maximize expectation of posterior probability w.r.t. $q(z)$.
  In other words, fit $q(z)$ to $p(z)$.

- **M-step:** Fix $q(z)$, Maximize expectation of posterior probability w.r.t. θ.
  Then, $\log p(x|\theta)$ also increase.



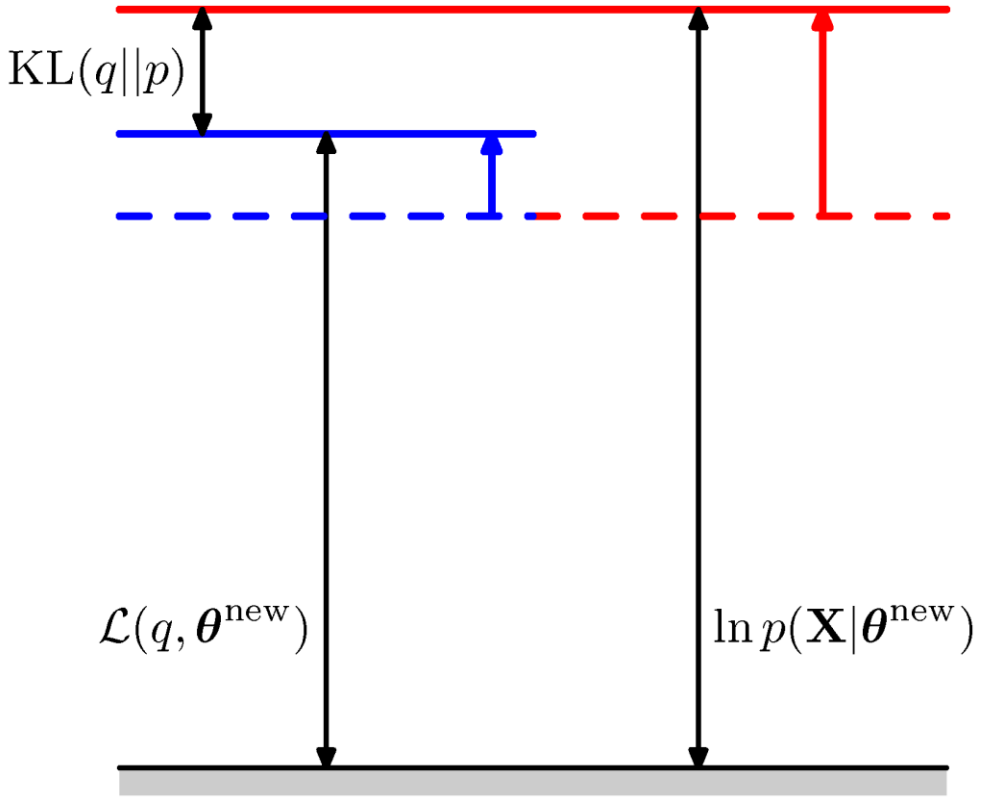
Christopher Bishop, Pattern Recognition & Machine Learning

E-step



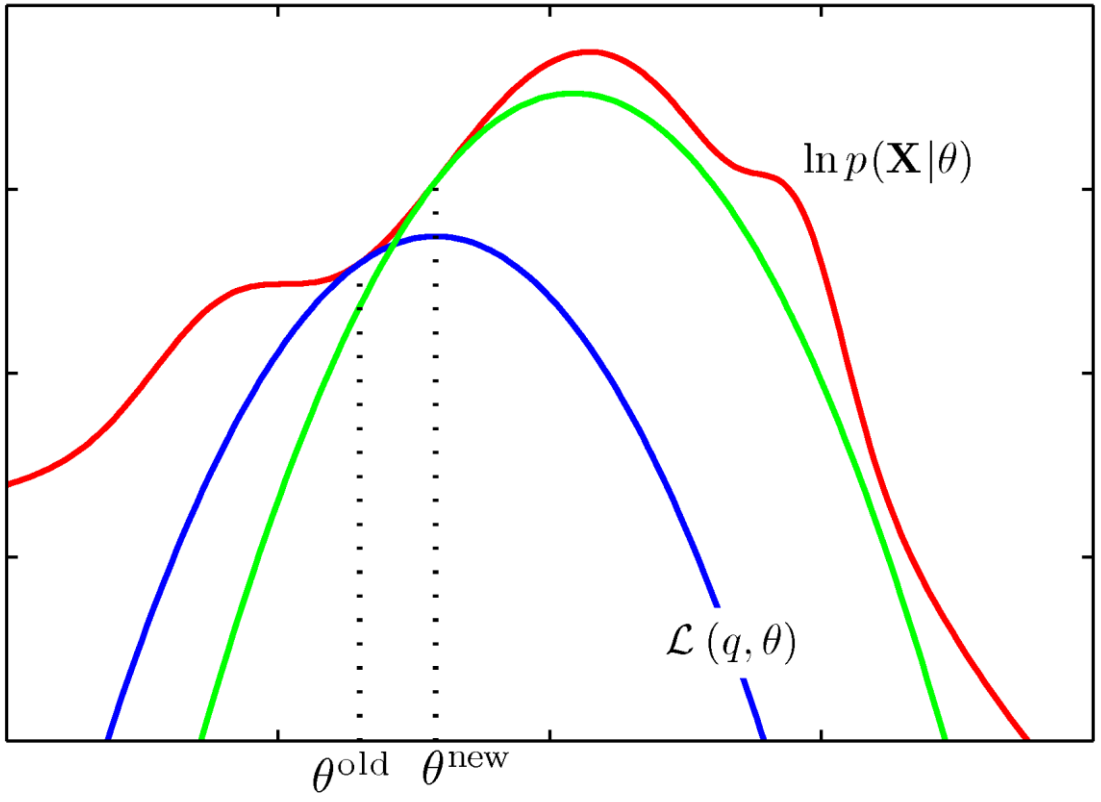
Christopher Bishop, Pattern Recognition & Machine Learning

M-step



Christopher Bishop, Pattern Recognition & Machine Learning

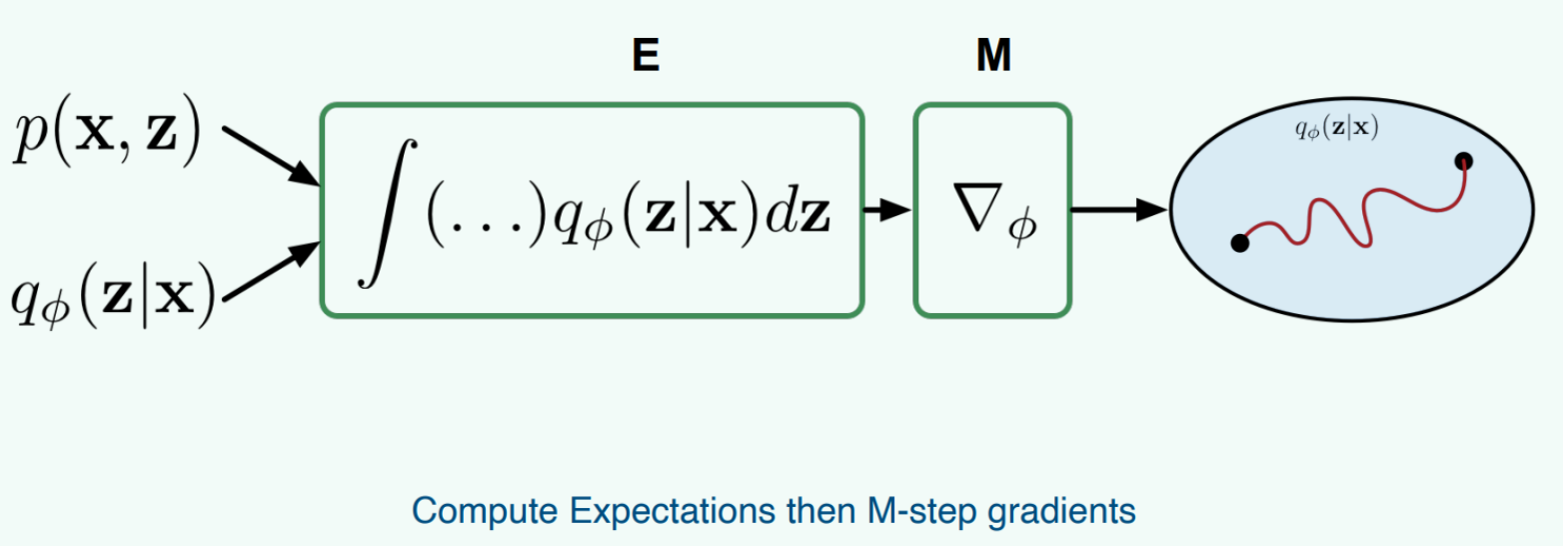
EM algorithm



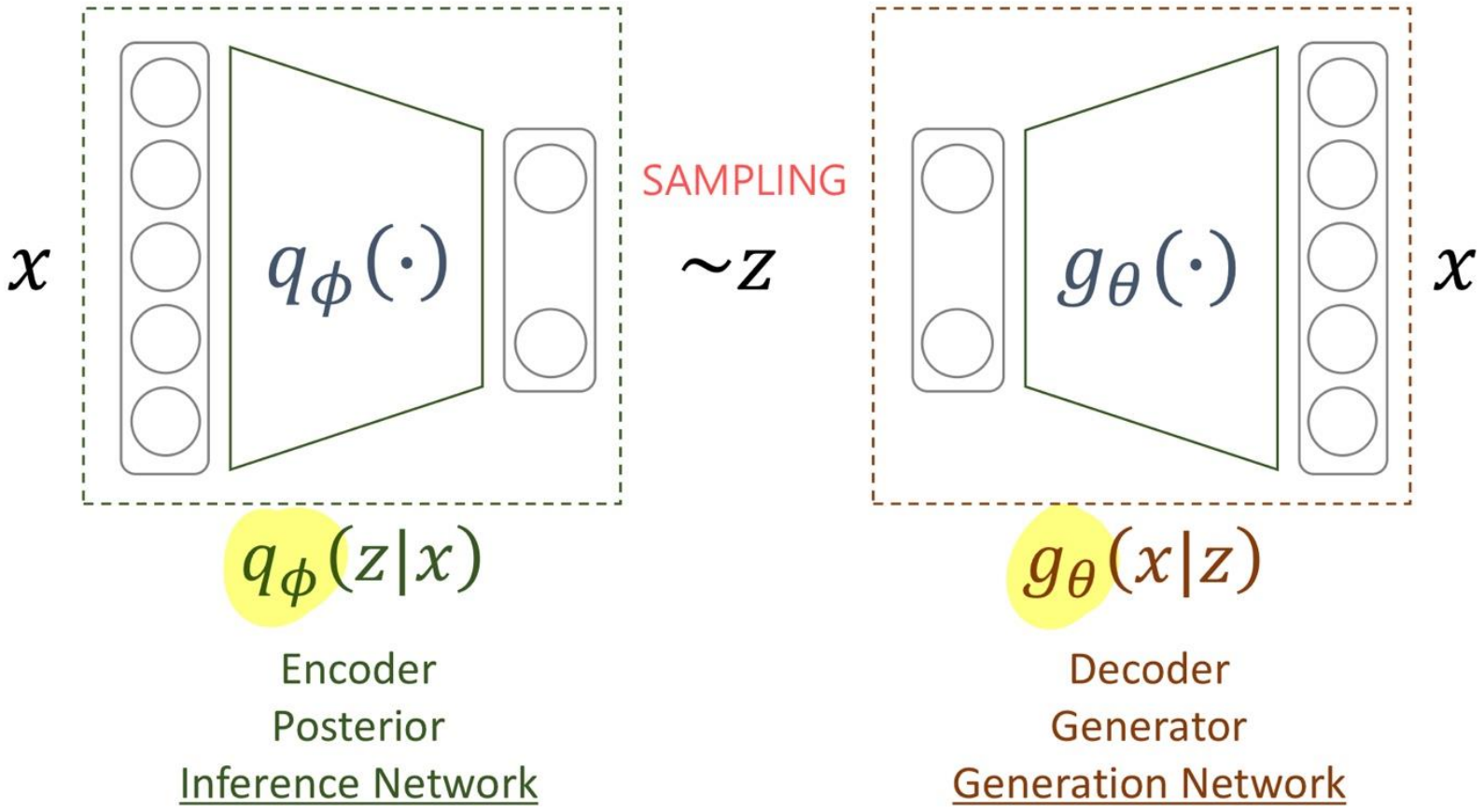
Christopher Bishop, Pattern Recognition & Machine Learning

Variational EM

- $\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - KL\left[q_\phi(z|x) \parallel p_\theta(z)\right]$
- E-step: $\phi \propto \nabla_\phi \mathcal{F}(x, q)$, M-step: $\theta \propto \nabla_\theta \mathcal{F}(x, q)$



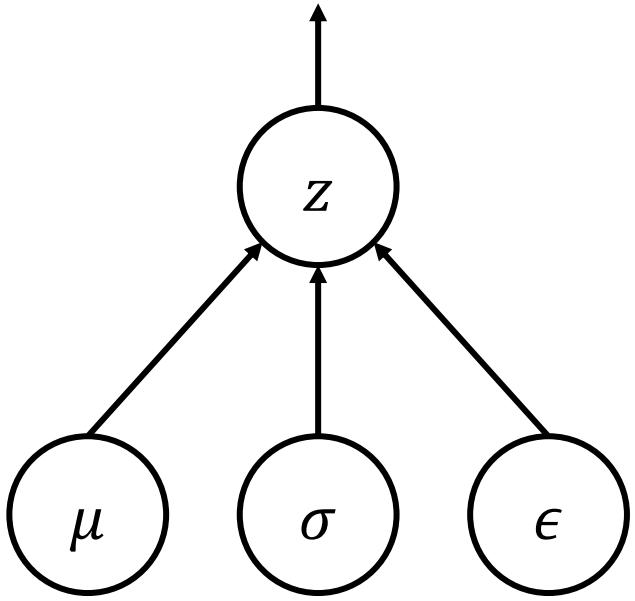
VAE



이활석, 오토인코더의 모든 것: https://www.youtube.com/watch?v=o_peo6U7IRM

Reparameterization Trick

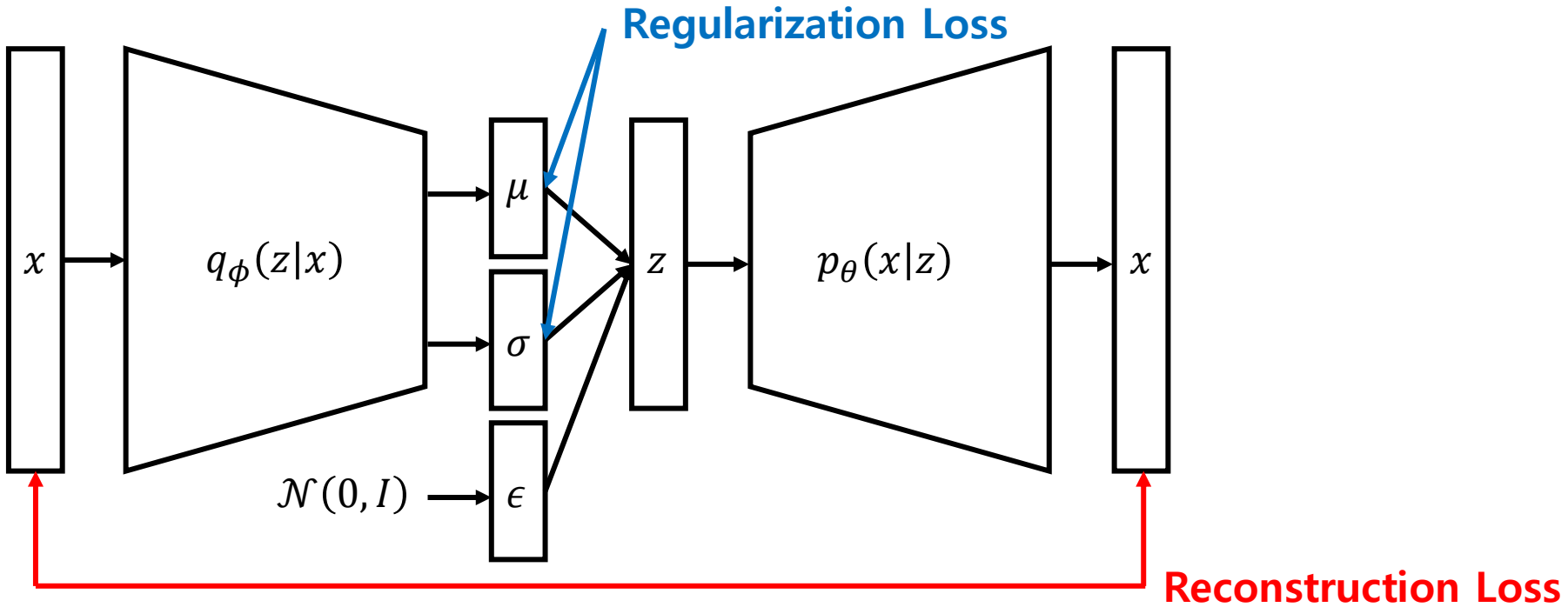
- How can we backpropagate to parameter of random variable?
  - $q_\phi(z|x)$



VAE

- Maximize $\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - KL[q_\phi(z|x) \| p_\theta(z)]$ w.r.t. ϕ, θ.

Apporx. posterior Reconstruction Regularization



Stochastic Gradient Variational Bayes

- Practical estimator of the lower bound and its derivatives w.r.t. the parameters.

Reference

- Carl Doersch, Tutorial on Variational Autoencoders: https://arxiv.org/abs/1606.05908
- Diederik P Kingma, Max Welling, Auto-Encoding Variational Bayes: https://arxiv.org/abs/1312.6114
- 이활석, 오토인코더의 모든 것: https://www.youtube.com/watch?v=o_peo6U7IRM
- Christopher Bishop, Pattern Recognition & Machine Learning
- Shakir Mohamed, MLSS 2020 Bayesian Inference: https://shakirm.com/mlss2020.html
- norman3, vi: https://github.com/norman3/vi

GMM (Gaussian Mixture Model)

- GMM distribution is followed by

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_{\mathrm{k}})$$

- Explain GMM with discrete latent variable **z**
- $p(z_k = 1) = \pi_k, \quad 0 \leq \pi_k \leq 1, \quad \sum_{k=1}^{K} \pi_k = 1$
- $p(x|z_k = 1) = \mathcal{N}(x|\mu_k, \Sigma_k)$
- $p(x) = \sum_z p(x, z) = \sum_z p(x|z)p(z) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_{\mathrm{k}})$

MLE for GMM

- We don't know $\pi_k, \mu_k, \Sigma_k, p(z) \Rightarrow$ parameterize
- $NLL = -\sum_{n=1}^{N} \ln p(x|\pi, \mu, \Sigma) = -\sum_{n=1}^{N} \ln \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k)$
- Use $\frac{dNLL}{d\mu_k} = 0, \frac{dNLL}{d\Sigma_k} = 0$, and $\frac{dNLL}{d\pi_k} = 0, with\ \sum_{k=1}^{K} \pi_k = 1.$
- $\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) x_n$
- $\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k)(x_n - \mu_k)^T$
- Using Lagrange Multiplier Method,
- $\pi_k = N_k / N$

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n|\mu_j, \Sigma_j)}$$

$$N_k = \sum_{n=1}^{N} \gamma(z_{nk})$$

EM for GMM

- Expectation
- $\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n|\mu_j, \Sigma_j)}$
- $N_k = \sum_{n=1}^{N} \gamma(z_{nk})$

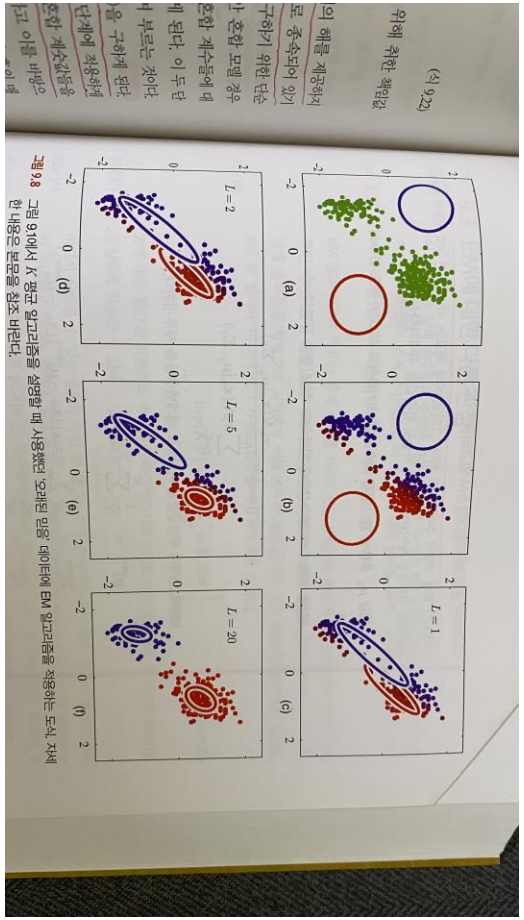
- Maximization
- $\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) x_n$
- $\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k)(x_n - \mu_k)^T$
- $\pi_k = N_k / N$

In general,

Calculate $p(Z|X, \theta^{old})$

Find new $\theta^{new} = \arg\max_{\theta} Q(\theta, \theta^{old})$

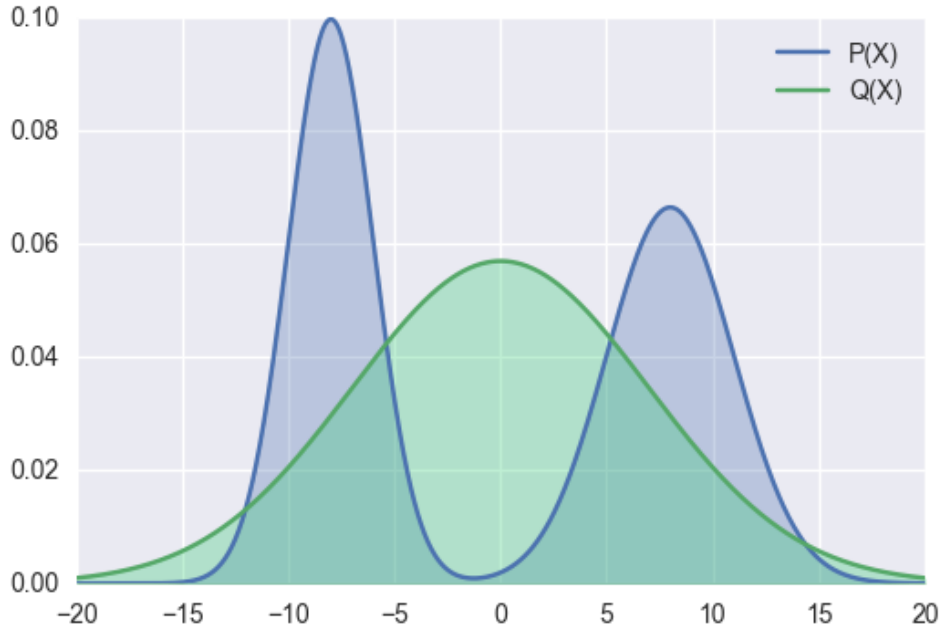
EM for GMM

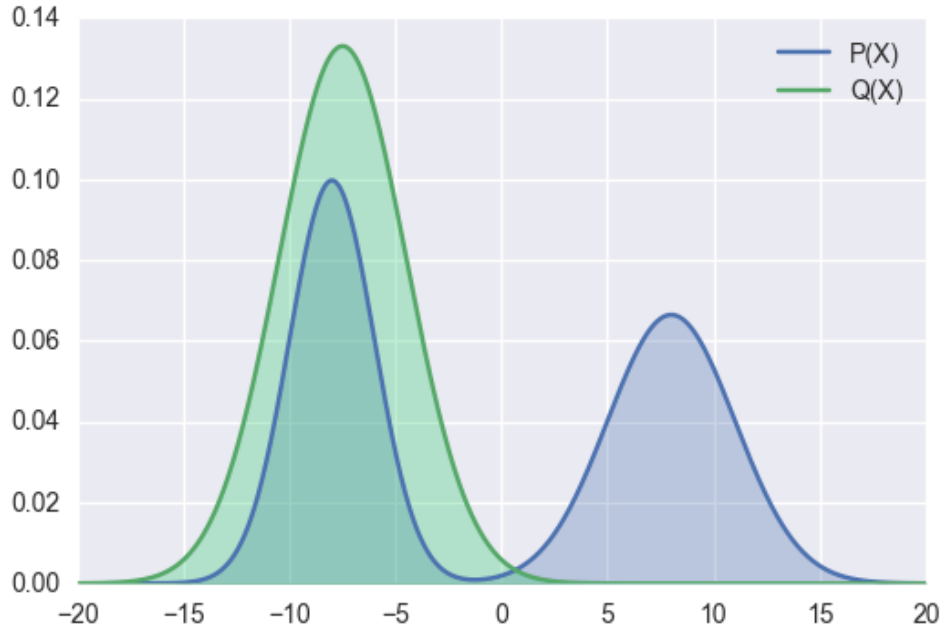


Christopher Bishop, Pattern Recognition & Machine Learning

KL Divergence

- Forward KL vs. Reverse KL
- $\int p(x) \log \frac{p(x)}{q(x)} dx$ vs. $\int q(x) \log \frac{q(x)}{p(x)} dx$





Agustinus Kristiadi, KL Divergence: Forward vs Reverse?, https://wiseodd.github.io/techblog/2016/12/21/forward-reverse-kl/