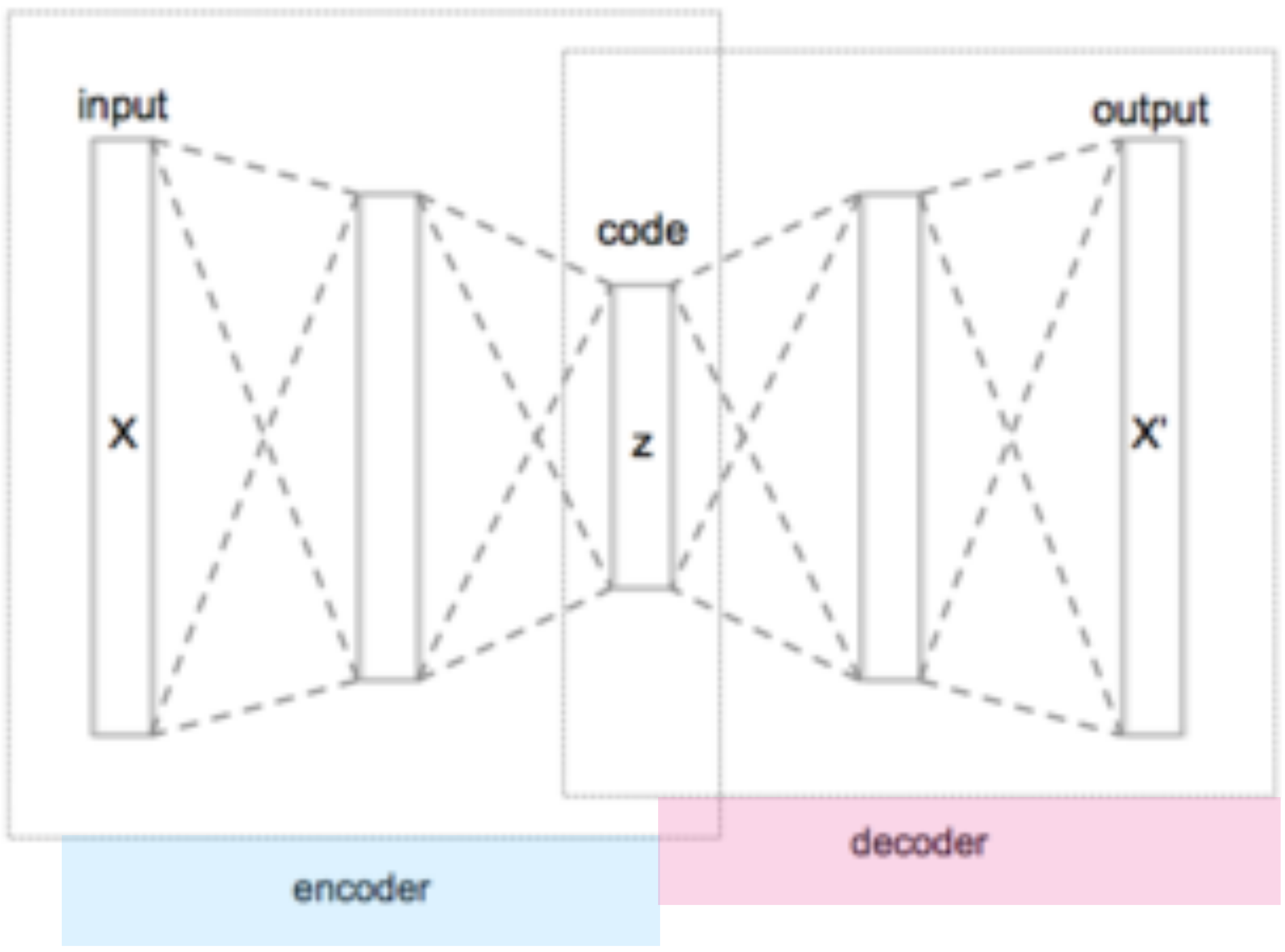
How to Generate Random Variables

- Generate U(0, 1)
  - Generate uniform over integers: 1, 2, …, N, where N is a large number
  - **pseudo-random number generator:** Algorithms that produce sequences of numbers that appear random but are entirely deterministic, based on an initial value called a **seed.**

- CDF method
- How to generate normal random variables

- Latent variable method

AutoEncoder



$$\mathbf{x}_{d\times 1}, \quad \mathbf{z}_{k\times 1}$$

$$\mathbf{z}_{k\times 1} = U_{k\times d}\mathbf{x}_{d\times 1}$$

$$\mathbf{x}'_{d\times 1} = W_{d\times k}\mathbf{z}_{k\times 1}$$

$$\min_{U,W} \sum_{i=1}^{n} \|\mathbf{x}_i - WU\mathbf{x}_i\|^2$$

Essentially PCA, if we restrict encoder and decoder to be linear functions

Probabilistic PCA

Reference: Tipping & Bishop (1997, 1999); Roweis (1998); Bishop PRML book (chap 12)

I. Model

$$\mathbf{z}_i \sim N(\mathbf{0}, I_k)$$

$$\mathbf{x}_i|\mathbf{z}_i, \theta, \sim p_\theta(\mathbf{x}_i \mid \mathbf{z}_i)$$

$$\mathbf{x}_i = W z_i + N(\mathbf{0}, \sigma^2 I_d)$$

We could solve MLE of theta = (W, sigma-square) directly: the integrated likelihood of X is still normal.

$$\begin{aligned} \mathbf{z}_i &\sim N_k(\mathbf{0}, I_k) \\ \mathbf{x}_i|\mathbf{z}_i, \theta &\sim N_d(W_{d\times k} z_i, \sigma^2 I_d) \\ \Longrightarrow \mathbf{x}_i &\sim N_d(\mathbf{0}, WW^t + \sigma^2 I_d) \end{aligned}$$

Probabilistic PCA

Reference: Tipping & Bishop (1997, 1999); Roweis (1998); Bishop PRML book (chap 12)

I. Model

$$\mathbf{z}_i \sim N(\mathbf{0}, I_k)$$

$$\mathbf{x}_i|\mathbf{z}_i, \theta, \sim p_\theta(\mathbf{x}_i \mid \mathbf{z}_i)$$

$$\mathbf{x}_i = W z_i + N(\mathbf{0}, \sigma^2 I_d)$$

II. EM Algorithm

$$J(\theta, Q) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{Q(\mathbf{z}_{1:n})}$$

Probabilistic PCA => Variational AutoEncoder

Reference: Tipping & Bishop (1997, 1999); Roweis (1998); Bishop PRML book (chap 12)

I. Model

$$\mathbf{z}_i \sim N(\mathbf{0}, I_k)$$

$$\mathbf{x}_i | \mathbf{z}_i, \theta, \sim p_\theta(\mathbf{x}_i \mid \mathbf{z}_i)$$

$$\mathbf{x}_i = W z_i + N(\mathbf{0}, \sigma^2 I_d)$$

II. EM Algorithm

$$J(\theta, Q) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{Q(\mathbf{z}_{1:n})}$$

1. **Variational EM**
2. **Amortized Inference**
3. **Replace all models by DNN**
4. **The reparameterization trick**

Variational AutoEncoder

I. Model

$$\mathbf{z}_i \sim N(\mathbf{0}, I_k)$$

$$\mathbf{x}_i | \mathbf{z}_i, \theta, \sim p_\theta(\mathbf{x}_i \mid \mathbf{z}_i)$$

$$\mathbf{x}_i = \boxed{W} z_i + N(\mathbf{0}, \boxed{\sigma^2} I_d)$$

II. EM Algorithm

$$\cancel{J(\theta, Q) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{Q(\mathbf{z}_{1:n})}}$$

$$J(\theta, Q) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{\prod_i q_i(\mathbf{z}_i)}$$

Variational AutoEncoder

I. Model

$$\mathbf{z}_i \sim N(\mathbf{0}, I_k)$$

$$\mathbf{x}_i | \mathbf{z}_i, \theta, \sim p_\theta(\mathbf{x}_i \mid \mathbf{z}_i)$$

$$\mathbf{x}_i = W z_i + N(\mathbf{0}, \sigma^2 I_d)$$

II. VEM Algorithm

$$\cancel{J(\theta, Q) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{Q(\mathbf{z}_{1:n})}}$$

$$J(\theta, \Phi) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{\prod_i q(\mathbf{z}_i \mid \phi_i)}$$

$$\cancel{J(\theta, Q) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{\prod_i q_i(\mathbf{z}_i)}}$$

1. Variational EM

Variational AutoEncoder

I. Model

$$\mathbf{z}_i \sim N(\mathbf{0}, I_k)$$

$$\mathbf{x}_i | \mathbf{z}_i, \theta, \sim p_\theta(\mathbf{x}_i \mid \mathbf{z}_i)$$

$$\mathbf{x}_i = \boxed{W} z_i + N(\mathbf{0}, \boxed{\sigma^2} I_d)$$

II. VEM Algorithm

2. Amortized Inference

$$\cancel{J(\theta, Q) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{Q(\mathbf{z}_{1:n})}}$$

$$\cancel{J(\theta, \Phi) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{\prod_i q(\mathbf{z}_i \mid \phi_i)}}$$

$$\cancel{J(\theta, \Phi) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{\prod_i q_i(\mathbf{z}_i)}}$$

$$J(\theta, \phi) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{\prod_i q(\mathbf{z}_i \mid \phi(\mathbf{x}_i))}$$

Variational AutoEncoder

I. Model

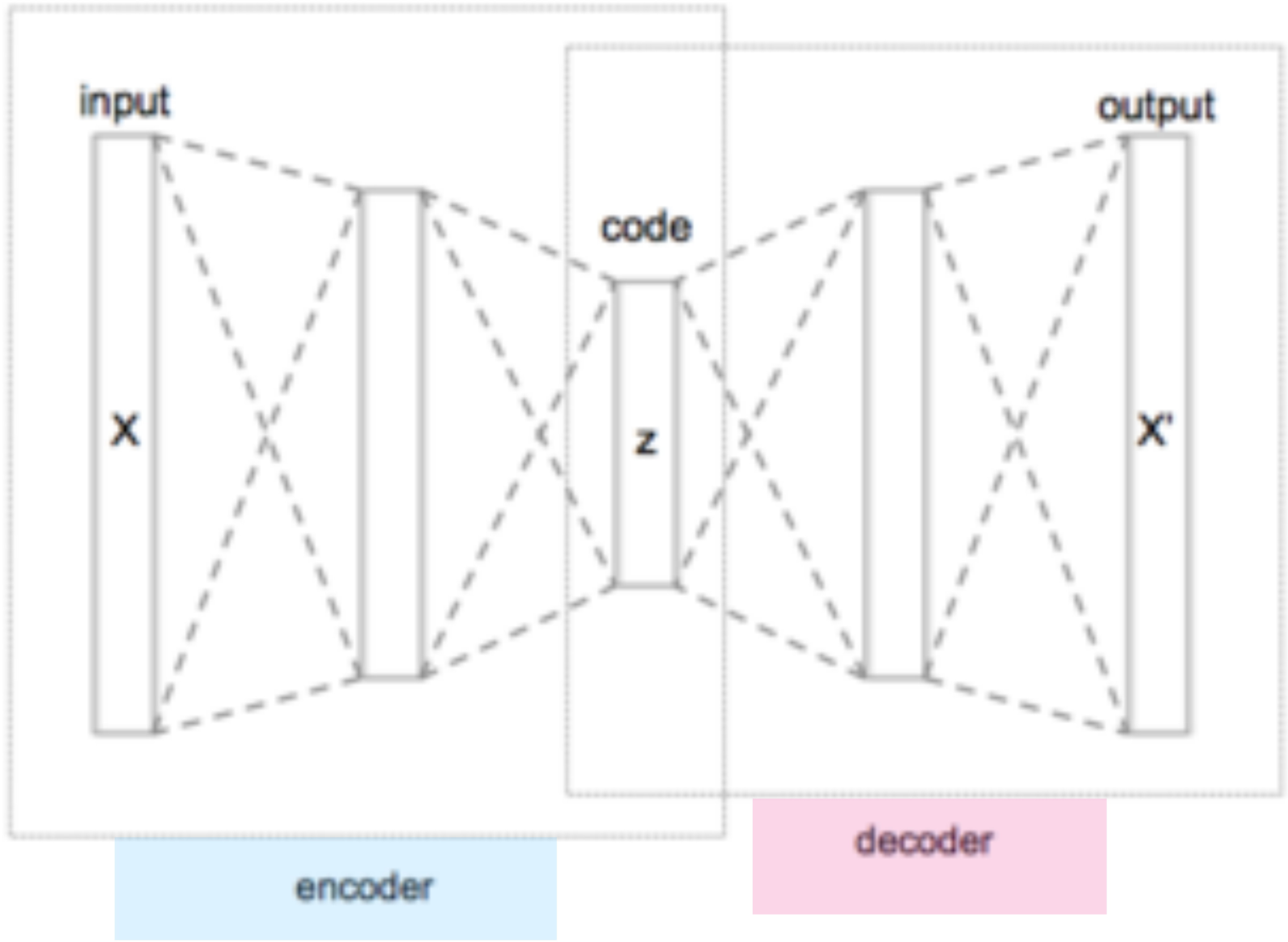
$$\mathbf{z}_i \sim N(\mathbf{0}, I_k)$$

$$\mathbf{x}_i | \mathbf{z}_i, \theta, \sim p_\theta(\mathbf{x}_i \mid \mathbf{z}_i)$$

II. VEM Algorithm

$$J(\theta, \phi) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{\prod_i q(\mathbf{z}_i \mid \phi(\mathbf{x}_i))}$$

3. Replace all models by DNN



Variational AutoEncoder

$$J(\theta, \phi) = \mathbb{E}_{\mathbf{z}_1 \cdots \mathbf{z}_n} \sum_i \log p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) + \sum_i \mathbb{E}_{\mathbf{z}_i} \log \frac{\pi(\mathbf{z}_i)}{q(\mathbf{z}_i \mid \phi(\mathbf{x}_i))}$$

I. Model

$$\mathbf{z}_i \sim N(\mathbf{0}, I_k)$$

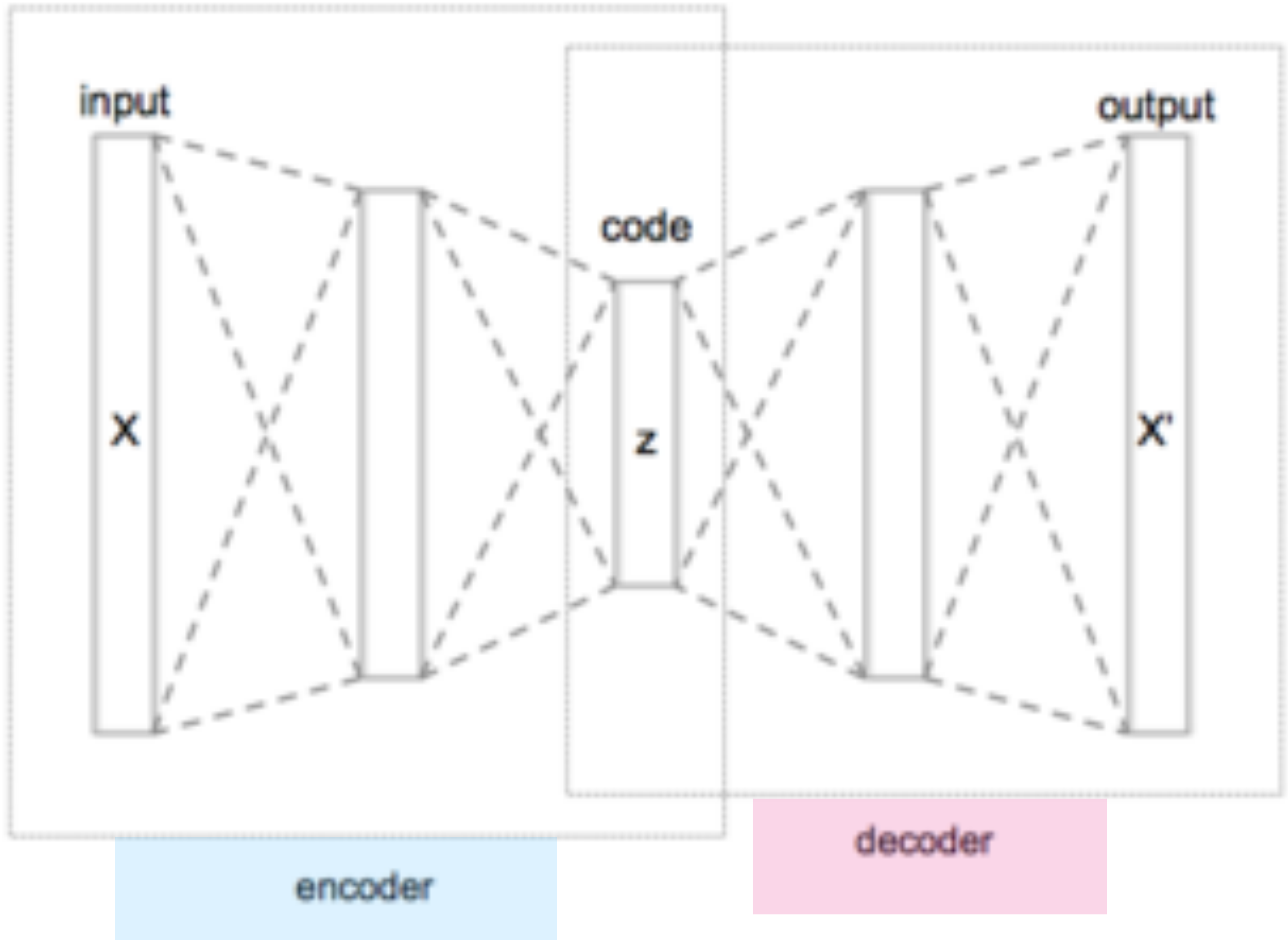
$$\mathbf{x}_i | \mathbf{z}_i, \theta, \sim p_\theta(\mathbf{x}_i \mid \mathbf{z}_i)$$

$$\mathbb{E}_{\epsilon_1 \cdots \epsilon_n} \sum_i \log p(\mathbf{x}_i \mid \sigma_i^2 \epsilon_i + \mu_i, \theta)$$

4. The reparameterization trick

II. VEM Algorithm

$$J(\theta, \phi) = \mathbb{E}_Q \log \frac{\prod_i p(\mathbf{x}_i \mid \mathbf{z}_i, \theta) \prod_i \pi(\mathbf{z}_i)}{\prod_i q(\mathbf{z}_i \mid \phi(\mathbf{x}_i))}$$



$$
\begin{aligned}
&\mathbb{E}_{\mathbf{z}_i} \log \frac{\pi(\mathbf{z}_i)}{q(\mathbf{z}_i \mid \phi(\mathbf{x}_i))} \\
&= \mathbb{E}_{\mathbf{z}_i \sim N(\mu_i, \sigma_i^2 I_k)} \log \frac{\frac{1}{(\sqrt{2\pi})^k} \exp\{-\frac{\|\mathbf{z}_i\|^2}{2}\}}{\frac{1}{(\sqrt{2\pi\sigma^2})^k} \exp\{-\frac{\|\mathbf{z}_i - \mu_i\|^2}{2}\}} \\
&= \frac{k}{2} \log \sigma^2 - \frac{1}{2}(\|\mu_i\|^2 + k\sigma_i^2) + C
\end{aligned}
$$