

CS 446/ECE 449: Machine Learning

Shenlong Wang

University of Illinois at Urbana-Champaign, 2024

Variational Auto-Encoders

Goals of this lecture

- Getting to know Variational Auto-Encoders
- Understanding generative methods
- Differentiating between discriminative and generative methods

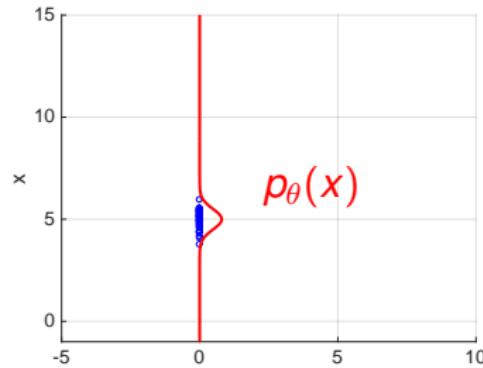
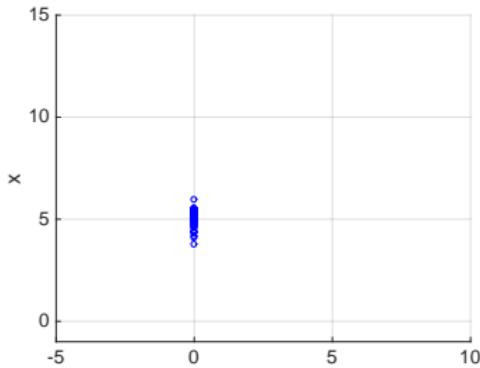
Reading material:

- D.P. Kingma and M. Welling; Auto-Encoding Variational Bayes;
arxiv.org/abs/1312.6114
- C. Doersch; Tutorial on Variational Autoencoders;
arxiv.org/abs/1606.05908

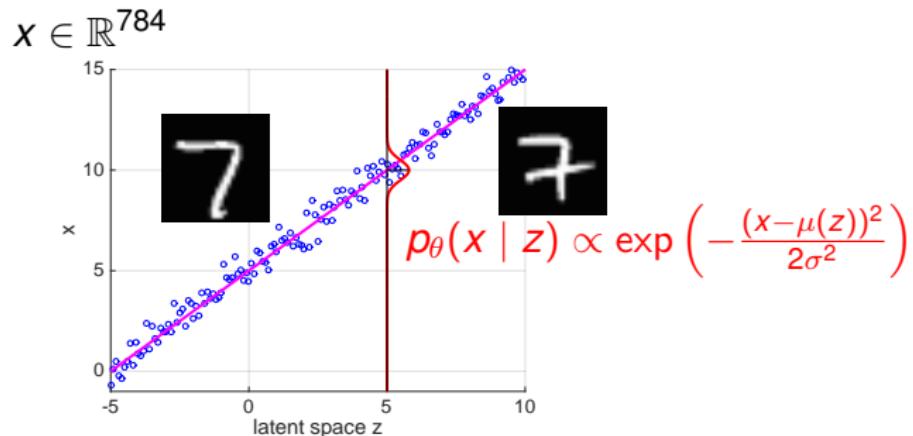
Given data points x , how can we model $p(x)$?

- Fit mean and variance (= parameters θ) of a distribution (e.g., Gaussian)
- Fit parameters θ of a mixture distribution (e.g., mixture of Gaussian, k-means)

Example:



Increasing dimensions:

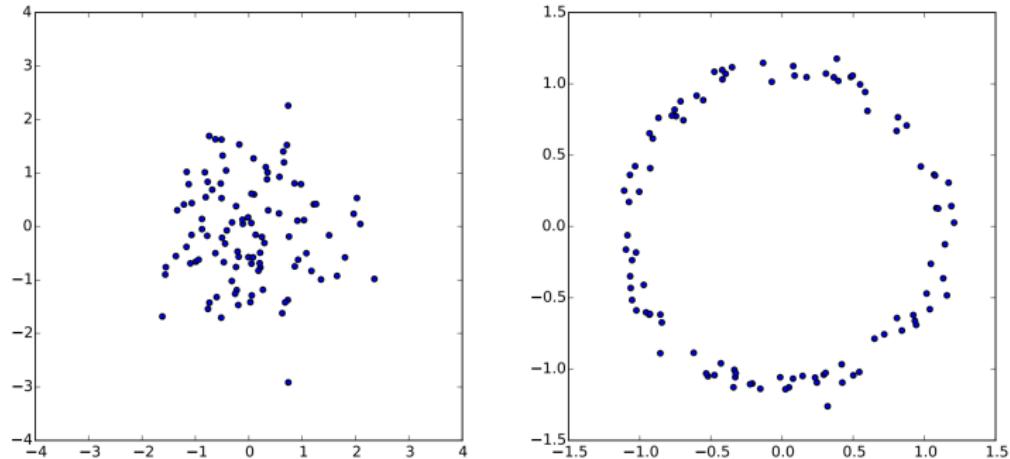


Manifold hypothesis:

- x is a high dimensional vector
- data is concentrated around a low dimensional manifold

How about non-linear manifold?

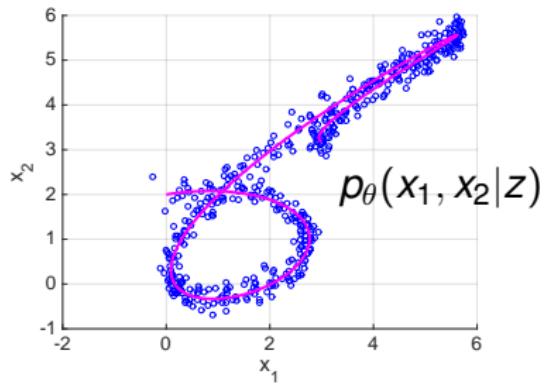
Reparameterization trick:



$$z \sim \mathcal{N}(0, I); g(z) = z/10 + z/\|z\|$$

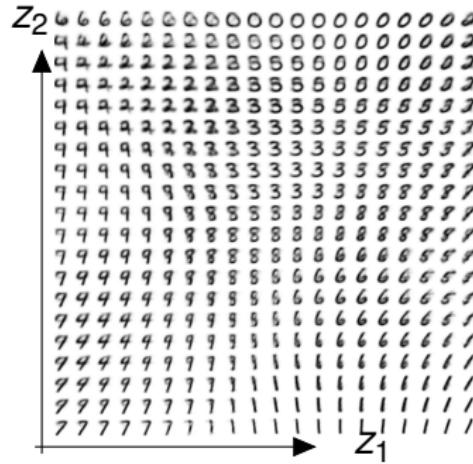
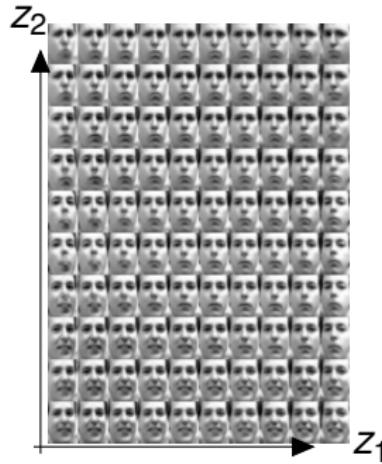
Examples of low-dimensional manifolds:

$$z \in [0, 1] \rightarrow$$



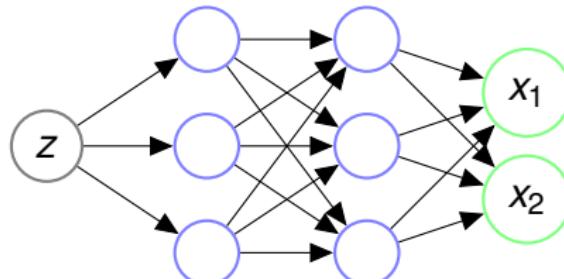
Examples of low-dimensional manifolds: ($z \in \mathbb{R}^2$)

Kingma and Welling: “Auto-Encoding Variational Bayes” (ICLR 2014)

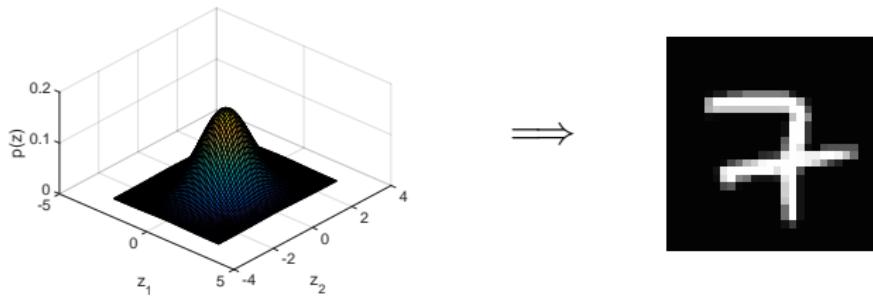


What is $p_\theta(x|z)$?

- Given z , we can generate, compute or sample data x
- Idea: z from simple Gaussian & transformation via deep net
- Decoder

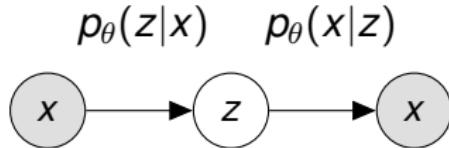


More generally:



But how to optimize for θ ? How do we know which z maps to which x ?

Training as an auto-encoder: How about



What is $p_{\theta}(z|x)$?

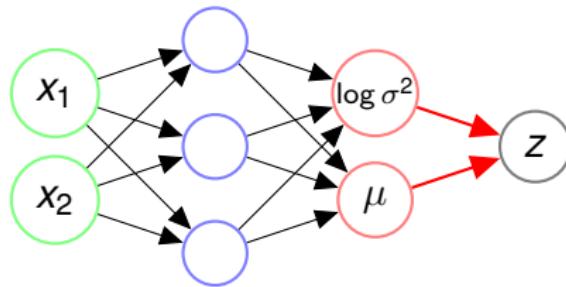
$$p_{\theta}(z|x) = \frac{p_{\theta}(x|z)p(z)}{p_{\theta}(x)} = \frac{p_{\theta}(x|z)p(z)}{\int_{\hat{z}} p_{\theta}(x|\hat{z})p(\hat{z})d\hat{z}}$$

How to compute normalization constant?

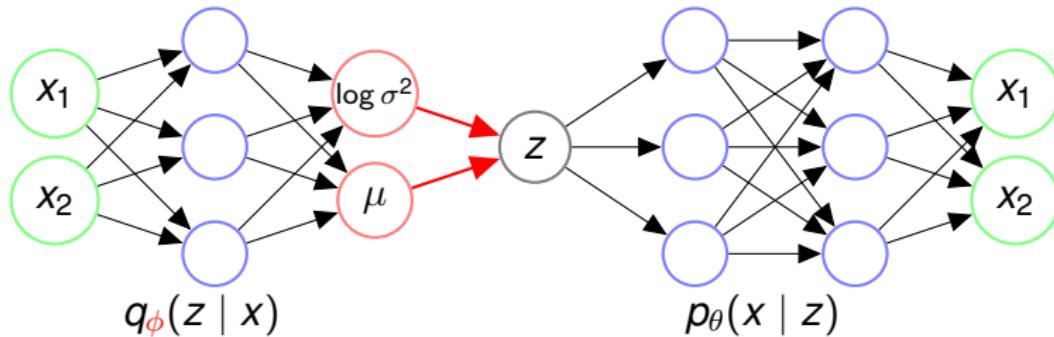
- Sampling techniques (generally very costly)
- Approximate $p_{\theta}(z|x)$ with $q_{\phi}(z|x)$ (encoder)

Encoder $q_\phi(z|x)$ as a deep net which computes mean and variance:

$$q_\phi(z|x) = \mathcal{N}(z; \mu_\phi(x), \sigma_\phi(x))$$



Variational auto-encoder architecture:



Learning parameters θ, ϕ via back-propagation

What do we want to optimize? What is the loss function?

$$\begin{aligned}\log p_\theta(x) &= \log \int p_\theta(x, z) dz \quad (\text{marginalization}) \\ &= \log \int q_\phi(z|x) \frac{p_\theta(x, z)}{q_\phi(z|x)} dz \quad (\text{multiple and divide by } q_\phi(z|x)) \\ &\geq \int q_\phi(z|x) \log \frac{p_\theta(x, z)}{q_\phi(z|x)} dz \quad (\text{Jensen-inequality}) \\ &= \mathcal{L}(p_\theta, q_\phi) \quad (\text{ELBO})\end{aligned}$$

- Assume $q_\phi(z|x)$ can approximate $p_\theta(z|x)$ well
- \mathcal{L} is often referred to as evidence lower bound (ELBO)

ELBO derived by KL-divergence:

$$\begin{aligned}\log p_\theta(x) &= \int q_\phi(z|x) \log p_\theta(x) dz \quad (\text{constant wrt } z) \\ &= \int q_\phi(z|x) \log \frac{p_\theta(x,z)}{p_\theta(z|x)} dz \quad (\text{bayes}) \\ &= \int q_\phi(z|x) \log \left(\frac{p_\theta(x,z)}{q_\phi(z|x)} \cdot \frac{q_\phi(z|x)}{p_\theta(z|x)} \right) dz \quad (\text{div and multiply}) \\ &= \int q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} dz + \int q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z|x)} dz \quad \text{log(abc)} \\ &= \mathcal{L}(p_\theta, q_\phi) + D_{KL}(q_\phi, p_\theta) \\ &\geq \mathcal{L}(p_\theta, q_\phi) \quad (\text{KLdiv is nonnegative})\end{aligned}$$

- Assume $q_\phi(z|x)$ can approximate $p_\theta(z|x)$ well
- \mathcal{L} is often referred to as evidence lower bound (ELBO)

Approximate Inference (maximize ELBO):

$$\begin{aligned}\mathcal{L}(p_\theta, q_\phi) &= \int q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} dz \quad (\text{div and multiply}) \\ &= \int q_\phi(z|x) \log \frac{p_\theta(x|z)p(z)}{q_\phi(z|x)} dz \quad (\text{bayes}) \\ &= \int q_\phi(z|x) \log \frac{p(z)}{q_\phi(z|x)} dz + \int q_\phi(z|x) \log p_\theta(x|z) dz \quad \log(\text{ab}) \\ &= -D_{KL}(q_\phi, p) + \mathbb{E}_{q_\phi} [\log p_\theta(x|z)]\end{aligned}$$

- Regularization $-D_{KL}(q_\phi, p)$ with prior $p(z)$ (Gaussian)
- Reconstruction $\mathbb{E}_{q_\phi} [\log p_\theta(x|z)]$

Regularization:

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

- $p(z) = \mathcal{N}(z; 0, 1)$
- $q_\phi(z|x)$ is Gaussian with mean $\mu_\phi(x)$, variance $\sigma_\phi^2(x)$

$$-D_{KL}(q_\phi, p) = \textbf{Homework}$$

Reconstruction:

$$\mathbb{E}_{q_\phi} [\log p_\theta(x|z)] = \int q_\phi(z|x) \log p_\theta(x|z) dz$$

Approximate $\mathbb{E}_{q_\phi} [\log p_\theta(x|z)]$ by sampling from $q_\phi(z|x)$

$$\mathbb{E}_{q_\phi} [\log p_\theta(x|z)] \approx \frac{1}{N} \sum_{i=1}^N \log p_\theta(x|z^i) \quad \text{where} \quad z^i \sim \mathcal{N}(z; \mu_\phi(x), \sigma_\phi(x))$$

Variational auto-encoder loss function:

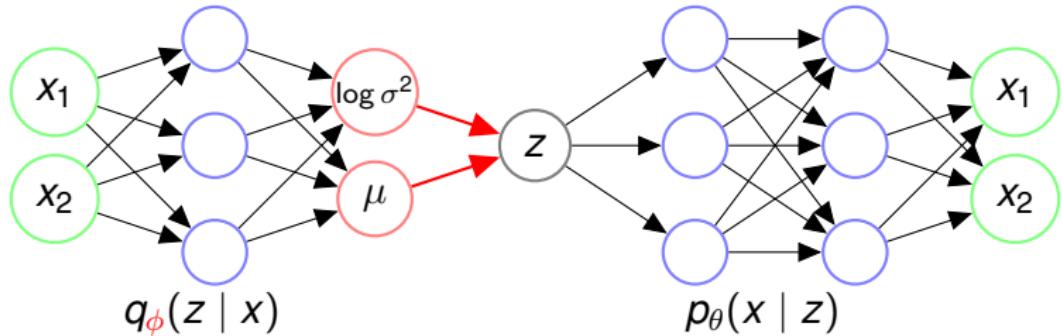
$$\mathcal{L}(p_\theta, q_\phi) \approx -D_{KL}(q_\phi, p) + \frac{1}{N} \sum_{i=1}^N \log p_\theta(x|z^i)$$

Intuitively:



Typically N small, e.g., N = 1

Where is an issue?



How to backpropagate through the sampling step?

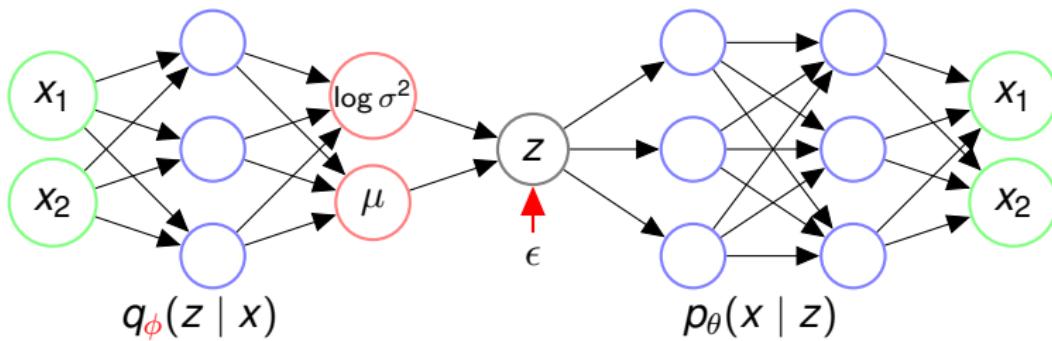
$$z \sim q_\phi(z|x) = \mathcal{N}(z; \mu_\phi(x), \sigma_\phi(x))$$

Reparameterization trick:

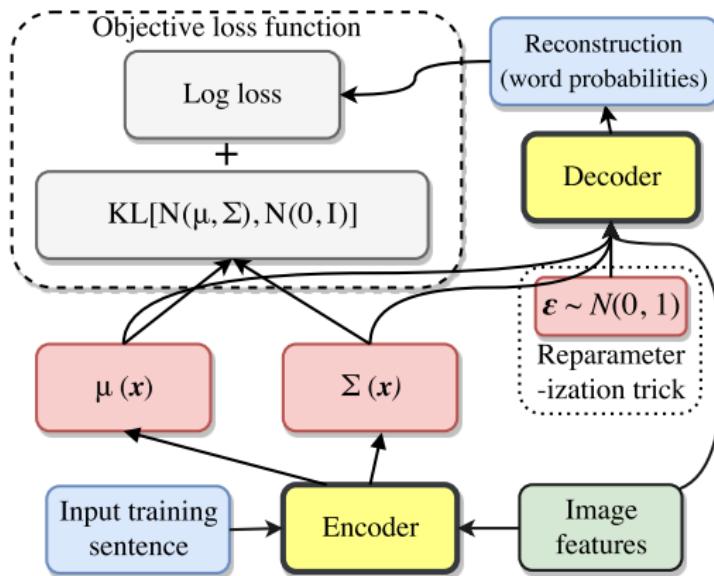
$$z \sim q_{\phi}(z|x) = \mathcal{N}(z; \mu_{\phi}(x), \sigma_{\phi}(x))$$

is equivalent to:

$$z = \mu_{\phi}(x) + \sigma_{\phi}(x) \cdot \epsilon \quad \text{where} \quad \epsilon \sim \mathcal{N}(0, 1)$$



Variational auto-encoder flowchart:



Variational auto-encoder for MNIST digits:

6 6 1 7 8 1 4 8 2 8	3 1 6 5 1 0 1 6 7 2	2 8 3 1 3 8 5 9 3 8	2 2 0 8 9 2 2 3 9 8 0
9 6 8 3 9 6 0 3 1 9	8 5 9 4 6 9 2 1 6 2	8 3 8 2 7 9 2 3 3 8	7 5 1 9 1 1 7 1 4 4
3 3 7 1 3 6 8 1 7 9	6 1 5 3 2 8 8 1 3 3	3 5 9 9 2 3 8 5 1 1	8 9 6 2 0 8 2 8 2 9
8 9 0 8 6 9 1 9 6 3	2 1 6 8 4 1 0 0 4 1	1 9 8 8 9 8 3 4 9 2	2 4 8 6 3 8 7 0 6 1
9 2 3 3 3 1 3 8 6	5 1 9 2 0 1 5 3 5 9	2 7 3 6 4 3 0 2 0 3	5 7 4 1 8 9 9 9 1 0
6 9 9 8 6 1 6 6 6 8	6 5 6 1 4 9 1 7 5 8	5 9 7 0 5 8 3 3 4 5	6 8 8 4 9 8 8 2 8 1
9 5 2 6 6 5 1 8 9 9	1 3 4 3 9 1 3 4 7 0	6 9 4 3 6 2 8 5 5 7	7 5 8 2 3 6 1 3 8 3
7 9 8 7 3 1 2 8 2 3	4 5 8 2 9 7 0 4 5 9	8 4 9 0 5 0 7 0 5 5	7 9 8 2 2 9 3 5 0
0 4 6 1 2 3 2 0 8 5	6 9 4 4 8 7 2 3 9 3	7 4 3 6 2 0 3 6 0 1	4 5 2 4 3 9 0 1 8 4
9 7 5 4 9 3 4 8 5 1	2 6 4 5 6 0 9 7 9 8	2 1 2 0 4 7 1 8 5 0	2 8 7 2 3 8 1 6 2 3 1

(a) 2-D latent space

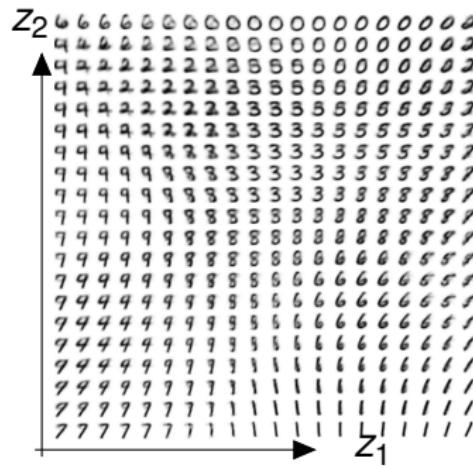
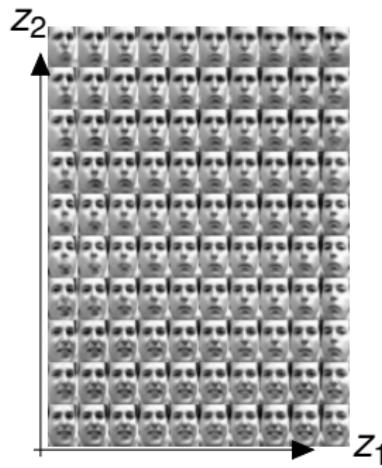
(b) 5-D latent space

(c) 10-D latent space

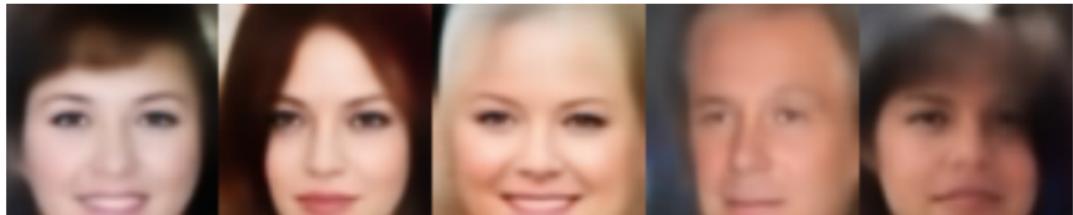
(d) 20-D latent space

Examples of low-dimensional manifolds: ($z \in \mathbb{R}^2$)

Kingma and Welling: “Auto-Encoding Variational Bayes” (ICLR 2014)



Variational auto-encoder for face generation:



Quiz:

- What is the difference between generative and discriminative modeling?
- What generative modeling techniques do you know about?
- What are the approximations used in variational auto-encoders?
- Why are variational auto-encoder results smooth?

Important topics of this lecture

- Getting to know variational auto-encoders, a generative modeling technique
- Understanding the reasons for approximations

Next up:

Adversarial Nets, another generative modeling technique