Machine Learning

A. G. Schwing & M. Telgarsky

University of Illinois at Urbana-Champaign, 2018

L22: Variational Auto-Encoders

Goals of this lecture

Goals of this lecture

- Getting to know Variational Auto-Encoders

Goals of this lecture

- Getting to know Variational Auto-Encoders
- Understanding generative methods

Goals of this lecture

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

Goals of this lecture

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

Reading material

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

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

Recap: Maximum likelihood so far?

Recap: Maximum likelihood so far?

Model:

$$p(\boldsymbol{y}|x) = \frac{\exp F(\boldsymbol{y}, x, \boldsymbol{w})/\epsilon}{\sum_{\hat{\boldsymbol{y}}} \exp F(\hat{\boldsymbol{y}}, x, \boldsymbol{w})/\epsilon}$$

Recap: Maximum likelihood so far?

Model:

$$p(\boldsymbol{y}|x) = \frac{\exp F(\boldsymbol{y}, x, \boldsymbol{w})/\epsilon}{\sum_{\hat{\boldsymbol{y}}} \exp F(\hat{\boldsymbol{y}}, x, \boldsymbol{w})/\epsilon}$$

- $\boldsymbol{y}$:

Recap: Maximum likelihood so far?

Model:

$$p(\boldsymbol{y}|x) = \frac{\exp F(\boldsymbol{y}, x, \boldsymbol{w})/\epsilon}{\sum_{\hat{\boldsymbol{y}}} \exp F(\hat{\boldsymbol{y}}, x, \boldsymbol{w})/\epsilon}$$

- $\boldsymbol{y}$: discrete output space

Recap: Maximum likelihood so far?

Model:

$$p(\boldsymbol{y}|x) = \frac{\exp F(\boldsymbol{y}, x, \boldsymbol{w})/\epsilon}{\sum_{\hat{\boldsymbol{y}}} \exp F(\hat{\boldsymbol{y}}, x, \boldsymbol{w})/\epsilon}$$

- $\boldsymbol{y}$: discrete output space
- x:

Recap: Maximum likelihood so far?

Model:

$$p(\boldsymbol{y}|x) = \frac{\exp F(\boldsymbol{y}, x, \boldsymbol{w})/\epsilon}{\sum_{\hat{\boldsymbol{y}}} \exp F(\hat{\boldsymbol{y}}, x, \boldsymbol{w})/\epsilon}$$

- $\boldsymbol{y}$: discrete output space
- x: input data

Recap: Maximum likelihood so far?

Model:

$$p(\boldsymbol{y}|x) = \frac{\exp F(\boldsymbol{y}, x, \boldsymbol{w})/\epsilon}{\sum_{\hat{\boldsymbol{y}}} \exp F(\hat{\boldsymbol{y}}, x, \boldsymbol{w})/\epsilon}$$

- $\boldsymbol{y}$: discrete output space
- x: input data

Now and the past few lectures:

Recap: Maximum likelihood so far?

Model:

$$p(\boldsymbol{y}|x) = \frac{\exp F(\boldsymbol{y}, x, \boldsymbol{w})/\epsilon}{\sum_{\hat{\boldsymbol{y}}} \exp F(\hat{\boldsymbol{y}}, x, \boldsymbol{w})/\epsilon}$$

- $\boldsymbol{y}$: discrete output space
- x: input data

Now and the past few lectures:

How about modeling a distribution $p(x)$ for the data?

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

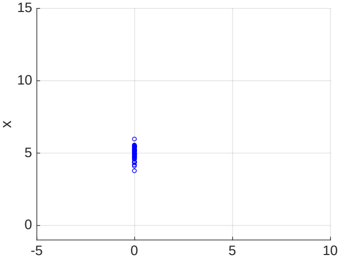
Given data points x, how can we model $p(x)$?

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

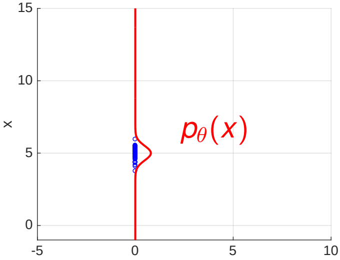
Given data points x, how can we model $p(x)$?

- 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)

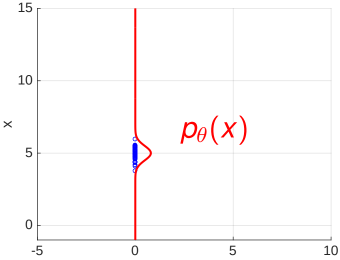
Example:



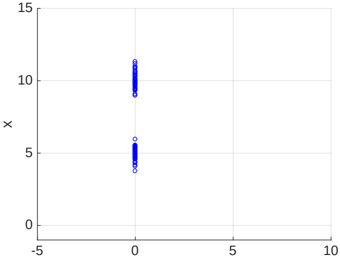
Example:



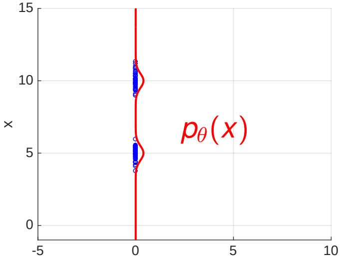
Example:



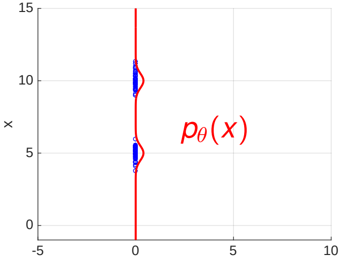
Example:



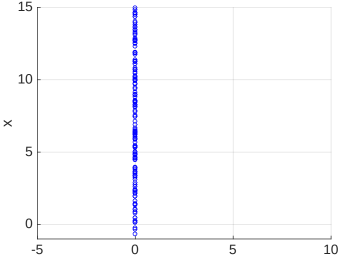
Example:



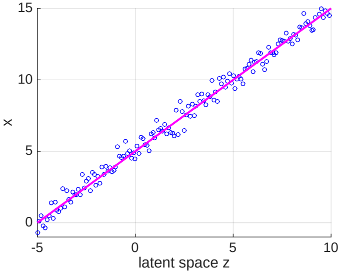
Example:



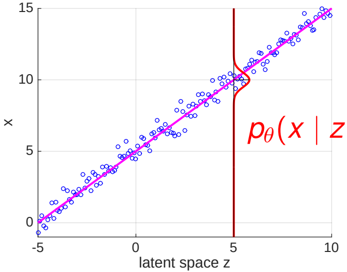
Example:



Example:

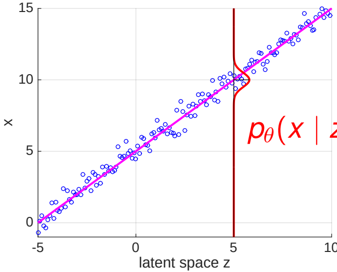


Example:



$$p_\theta(x \mid z) \propto \exp\left(-\frac{(x-\mu(z))^2}{2\sigma^2}\right)$$

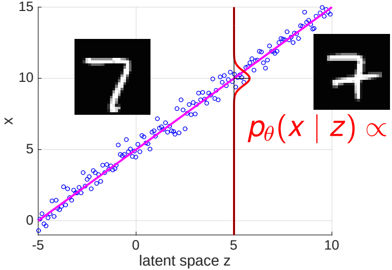
Example:



$$p_\theta(x \mid z) \propto \exp\left(-\frac{(x-\mu(z))^2}{2\sigma^2}\right)$$

Increasing dimensions:

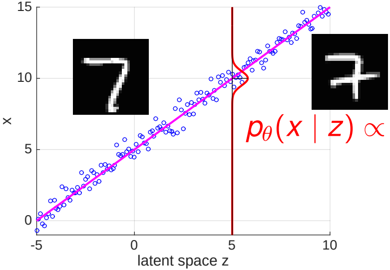
$x \in \mathbb{R}^{784}$



$$p_\theta(x \mid z) \propto \exp\left(-\frac{(x-\mu(z))^2}{2\sigma^2}\right)$$

Increasing dimensions:

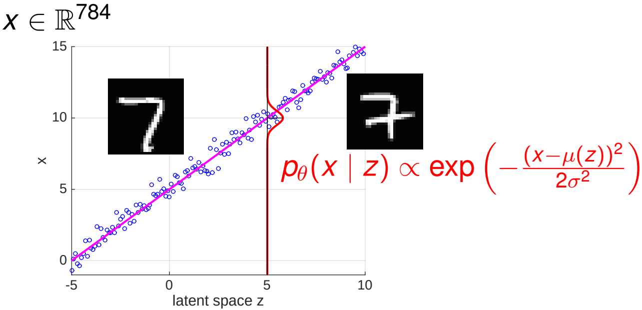
$x \in \mathbb{R}^{784}$



$p_\theta(x \mid z) \propto \exp\left(-\frac{(x-\mu(z))^2}{2\sigma^2}\right)$

Manifold hypothesis:

Increasing dimensions:

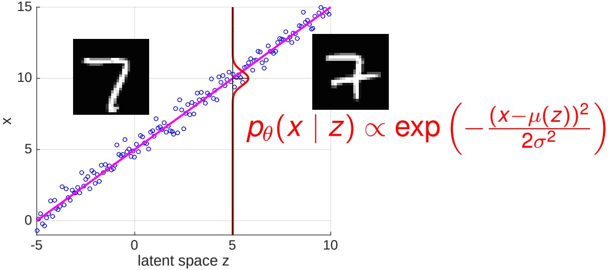


Manifold hypothesis:

- x is a high dimensional vector

Increasing dimensions:

$x \in \mathbb{R}^{784}$



Manifold hypothesis:

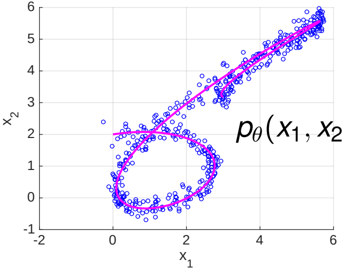
- x is a high dimensional vector
- data is concentrated around a low dimensional manifold

Examples of low-dimensional manifolds:

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

Examples of low-dimensional manifolds:

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

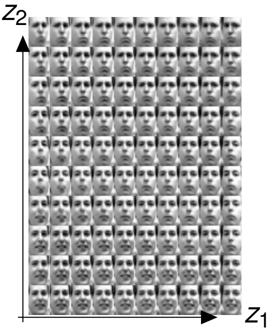


$p_\theta(x_1, x_2|z)$

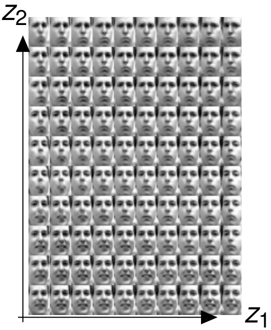
Examples of low-dimensional manifolds: ($z \in \mathbb{R}^2$)

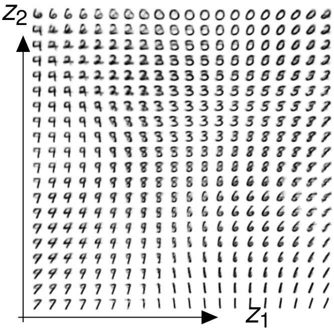
Examples of low-dimensional manifolds: ($z \in \mathbb{R}^2$)
Kingma and Welling: "Auto-Encoding Variational Bayes" (ICLR 2014)

Examples of low-dimensional manifolds: ($z \in \mathbb{R}^2$)
Kingma and Welling: “Auto-Encoding Variational Bayes” (ICLR 2014)



Examples of low-dimensional manifolds: ($z \in \mathbb{R}^2$)
Kingma and Welling: "Auto-Encoding Variational Bayes" (ICLR 2014)





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

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

- Given z, we can generate, compute or sample data x

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

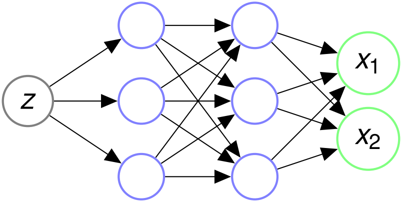
- Given z, we can generate, compute or sample data x
- Idea: z from simple Gaussian & transformation via deep net

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

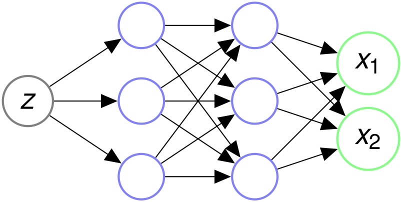
What is $p_\theta(x|z)$?

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



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

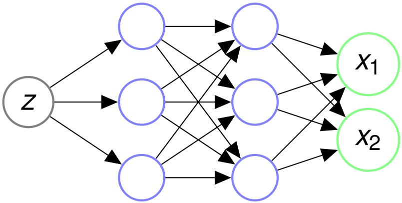
- Given z, we can generate, compute or sample data x
- Idea: z from simple Gaussian & transformation via deep net
- Decoder



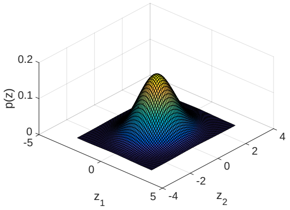
More generally:

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

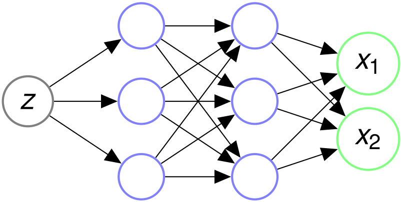


More generally:

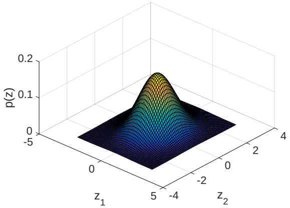


What is $p_\theta(x|z)$?

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



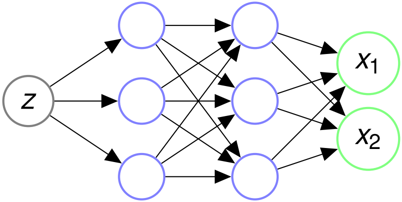
More generally:



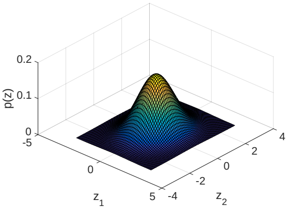
$\Longrightarrow$

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

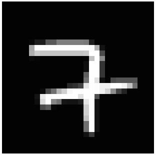
- Given z, we can generate, compute or sample data x
- Idea: z from simple Gaussian & transformation via deep net
- Decoder



More generally:

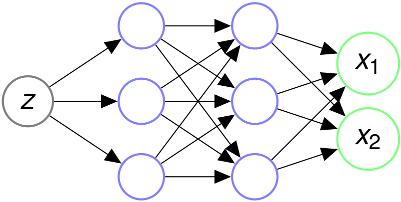


$\Longrightarrow$

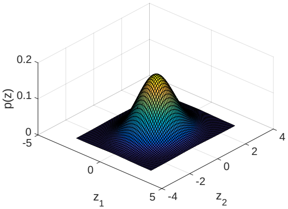


What is $p_\theta(x|z)$?

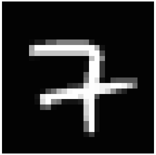
- Given z, we can generate, compute or sample data x
- Idea: z from simple Gaussian & transformation via deep net
- Decoder



More generally:

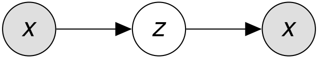


$\Longrightarrow$

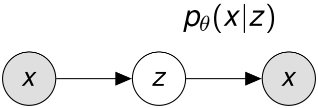


But how to optimize for θ? How do we know which z maps to which x?

Training as an auto-encoder: How about



Training as an auto-encoder: How about



Training as an auto-encoder: How about

$$x \xrightarrow{p_\theta(z|x)} z \xrightarrow{p_\theta(x|z)} x$$

Training as an auto-encoder: How about

$$\begin{tikzcd} x \arrow[r, "p_\theta(z|x)"] & z \arrow[r, "p_\theta(x|z)"] & x \end{tikzcd}$$

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

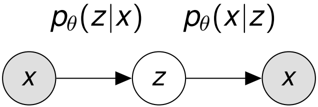
Training as an auto-encoder: How about

$$x \xrightarrow{p_\theta(z|x)} z \xrightarrow{p_\theta(x|z)} x$$

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

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

Training as an auto-encoder: How about



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

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

Training as an auto-encoder: How about

$$x \xrightarrow{p_\theta(z|x)} z \xrightarrow{p_\theta(x|z)} x$$

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?

Training as an auto-encoder: How about

$$x \xrightarrow{p_\theta(z|x)} z \xrightarrow{p_\theta(x|z)} x$$

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)

Training as an auto-encoder: How about

$$x \xrightarrow{p_\theta(z|x)} z \xrightarrow{p_\theta(x|z)} x$$

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)

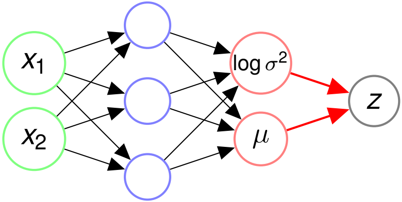
Encoder $q_{\phi}(z|x)$ as a deep net which computes mean and variance:

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

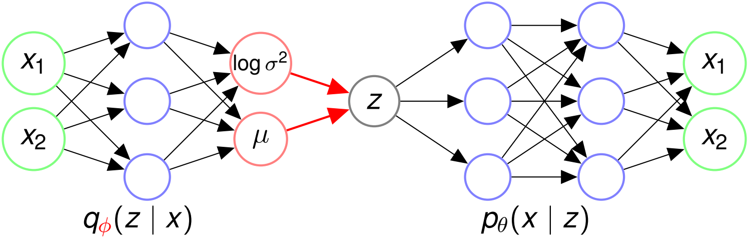
$$q_{\phi}(z|x) = \mathcal{N}(z; \mu_{\phi}(x), \sigma_{\phi}(x))$$

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?

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

$$\log p_\theta(x)$$

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

$$\log p_\theta(x) = \int_z q_\phi(z|x) \log p_\theta(x)$$

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

$$\begin{aligned}\log p_\theta(x) &= \int_z q_\phi(z|x)\log p_\theta(x)\\ &= \int_z q_\phi(z|x)\log\frac{p_\theta(x,z)}{p_\theta(z|x)}\end{aligned}$$

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

$$\begin{aligned}
\log p_\theta(x) &= \int_z q_\phi(z|x) \log p_\theta(x) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{p_\theta(z|x)} \\
&= \int_z q_\phi(z|x) \log \left(\frac{p_\theta(x,z)}{q_\phi(z|x)} \cdot \frac{q_\phi(z|x)}{p_\theta(z|x)} \right)
\end{aligned}$$

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

$$
\begin{aligned}
\log p_\theta(x) &= \int_z q_\phi(z|x) \log p_\theta(x) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{p_\theta(z|x)} \\
&= \int_z q_\phi(z|x) \log \left(\frac{p_\theta(x,z)}{q_\phi(z|x)} \cdot \frac{q_\phi(z|x)}{p_\theta(z|x)} \right) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} + \int_z q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z|x)}
\end{aligned}
$$

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

$$
\begin{aligned}
\log p_\theta(x) &= \int_z q_\phi(z|x) \log p_\theta(x) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{p_\theta(z|x)} \\
&= \int_z q_\phi(z|x) \log \left(\frac{p_\theta(x,z)}{q_\phi(z|x)} \cdot \frac{q_\phi(z|x)}{p_\theta(z|x)} \right) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} + \int_z q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z|x)} \\
&= \mathcal{L}(p_\theta, q_\phi) + D_{KL}(q_\phi, p_\theta)
\end{aligned}
$$

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

$$\begin{aligned}
\log p_\theta(x) &= \int_z q_\phi(z|x) \log p_\theta(x) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{p_\theta(z|x)} \\
&= \int_z q_\phi(z|x) \log \left(\frac{p_\theta(x,z)}{q_\phi(z|x)} \cdot \frac{q_\phi(z|x)}{p_\theta(z|x)} \right) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} + \int_z q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z|x)} \\
&= \mathcal{L}(p_\theta, q_\phi) + D_{KL}(q_\phi, p_\theta) \\
&\geq \mathcal{L}(p_\theta, q_\phi)
\end{aligned}$$

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

$$
\begin{aligned}
\log p_\theta(x) &= \int_z q_\phi(z|x) \log p_\theta(x) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{p_\theta(z|x)} \\
&= \int_z q_\phi(z|x) \log \left(\frac{p_\theta(x,z)}{q_\phi(z|x)} \cdot \frac{q_\phi(z|x)}{p_\theta(z|x)} \right) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} + \int_z q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z|x)} \\
&= \mathcal{L}(p_\theta, q_\phi) + D_{KL}(q_\phi, p_\theta) \\
&\geq \mathcal{L}(p_\theta, q_\phi)
\end{aligned}
$$

- Assume $q_\phi(z|x)$ can approximate $p_\theta(z|x)$ well

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

$$
\begin{aligned}
\log p_\theta(x) &= \int_z q_\phi(z|x) \log p_\theta(x) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{p_\theta(z|x)} \\
&= \int_z q_\phi(z|x) \log \left(\frac{p_\theta(x,z)}{q_\phi(z|x)} \cdot \frac{q_\phi(z|x)}{p_\theta(z|x)} \right) \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} + \int_z q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z|x)} \\
&= \mathcal{L}(p_\theta, q_\phi) + D_{KL}(q_\phi, p_\theta) \\
&\geq \mathcal{L}(p_\theta, q_\phi)
\end{aligned}
$$

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

Approximate Inference:

$$\mathcal{L}(p_\theta, q_\phi) \quad = \quad \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)}$$

Approximate Inference:

$$\begin{aligned}
\mathcal{L}(p_\theta, q_\phi) &= \int_z q_\phi(z|x) \log \frac{p_\theta(x, z)}{q_\phi(z|x)} \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x|z)p(z)}{q_\phi(z|x)}
\end{aligned}$$

Approximate Inference:

$$\begin{aligned}
\mathcal{L}(p_\theta, q_\phi) &= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x|z)p(z)}{q_\phi(z|x)} \\
&= \int_z q_\phi(z|x) \log \frac{p(z)}{q_\phi(z|x)} + \int_z q_\phi(z|x) \log p_\theta(x|z)
\end{aligned}$$

Approximate Inference:

$$
\begin{aligned}
\mathcal{L}(p_\theta, q_\phi) &= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x|z)p(z)}{q_\phi(z|x)} \\
&= \int_z q_\phi(z|x) \log \frac{p(z)}{q_\phi(z|x)} + \int_z q_\phi(z|x) \log p_\theta(x|z) \\
&= -D_{KL}(q_\phi, p) + \mathbb{E}_{q_\phi}[\log p_\theta(x|z)]
\end{aligned}
$$

Approximate Inference:

$$\begin{aligned}
\mathcal{L}(p_\theta, q_\phi) &= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \\
&= \int_z q_\phi(z|x) \log \frac{p_\theta(x|z)p(z)}{q_\phi(z|x)} \\
&= \int_z q_\phi(z|x) \log \frac{p(z)}{q_\phi(z|x)} + \int_z q_\phi(z|x) \log p_\theta(x|z) \\
&= -D_{KL}(q_\phi, p) + \mathbb{E}_{q_\phi}[\log p_\theta(x|z)]
\end{aligned}$$

- Regularization $-D_{KL}(q_\phi, p)$ with prior $p(z)$ (Gaussian)

Approximate Inference:

$$\begin{aligned}\mathcal{L}(p_\theta, q_\phi) &= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \\ &= \int_z q_\phi(z|x) \log \frac{p_\theta(x|z)p(z)}{q_\phi(z|x)} \\ &= \int_z q_\phi(z|x) \log \frac{p(z)}{q_\phi(z|x)} + \int_z q_\phi(z|x) \log p_\theta(x|z) \\ &= -D_{KL}(q_\phi, p) + \mathbb{E}_{q_\phi}[\log p_\theta(x|z)]\end{aligned}$$

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

Regularization:

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

Regularization:

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

- $p(z) = \mathcal{N}(z; 0, 1)$

Regularization:

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

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

Regularization:

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

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

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

Regularization:

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

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

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

Reconstruction:

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

Reconstruction:

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

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

Reconstruction:

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

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:

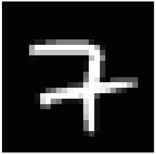
$$\mathcal{L}(p_\theta, q_\phi) \approx -D_{KL}(q_\phi, p) + \frac{1}{N}\sum_{i=1}^{N} \log p_\theta(x|z^i)$$

Intuitively:

Variational auto-encoder loss function:

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

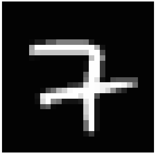
Intuitively:

 $\rightarrow$

Variational auto-encoder loss function:

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

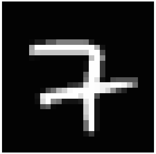
Intuitively:

 $\rightarrow$ z $\rightarrow$

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:



$\rightarrow$ z $\rightarrow$

$\log p_\theta(x|z^1)$

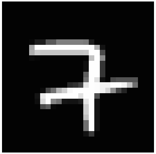
$\vdots$

$\log p_\theta(x|z^N)$

Variational auto-encoder loss function:

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

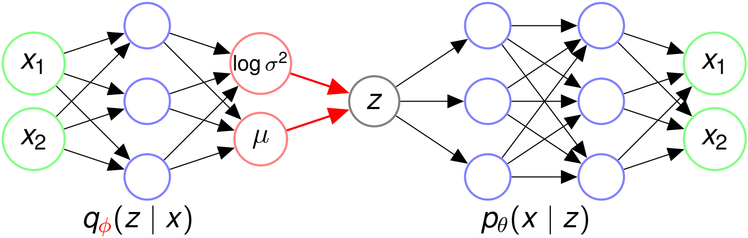
Intuitively:



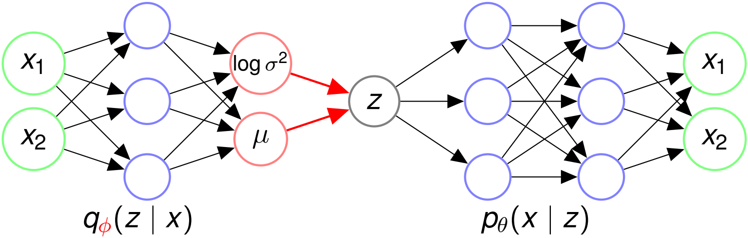
$$\rightarrow \quad z \quad \rightarrow \quad \begin{array}{l} \log p_\theta(x|z^1) \\ \vdots \\ \log p_\theta(x|z^N) \end{array}$$

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

Where is an issue?



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:

Reparameterization trick:

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

is equivalent to:

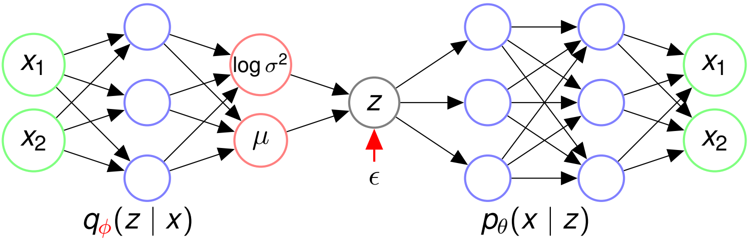
$$z = \mu_\phi(x) + \sigma_\phi(x) \cdot \epsilon \quad \text{where} \quad \epsilon \sim \mathcal{N}(0, 1)$$

Reparameterization trick:

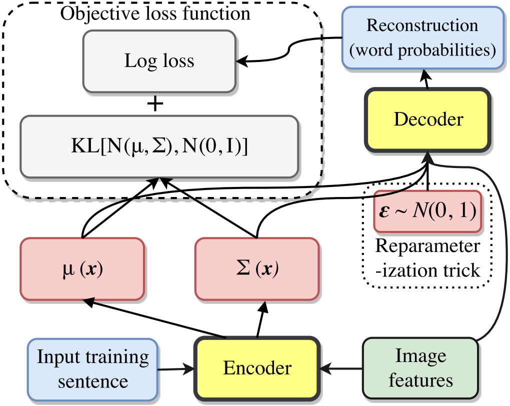
$$z \sim q_{\phi}(z|x) = \mathcal{N}(z; \mu_{\phi}(x), \sigma_{\phi}(x))$$

is equivalent to:

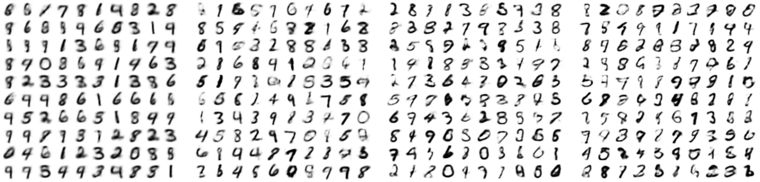
$$z = \mu_{\phi}(x) + \sigma_{\phi}(x) \cdot \epsilon \quad \text{where} \quad \epsilon \sim \mathcal{N}(0, 1)$$



Variational auto-encoder flowchart:



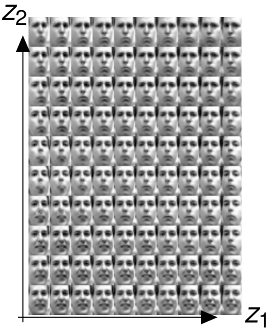
Variational auto-encoder for MNIST digits:



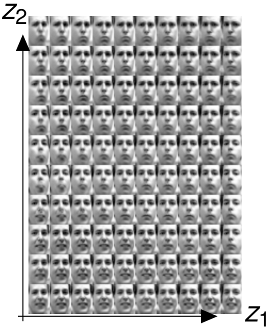
(a) 2-D latent space (b) 5-D latent space (c) 10-D latent space (d) 20-D latent space

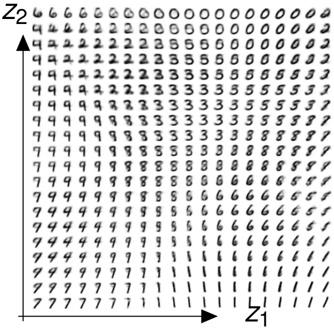
Examples of low-dimensional manifolds: ($z \in \mathbb{R}^2$)
Kingma and Welling: "Auto-Encoding Variational Bayes" (ICLR 2014)

Examples of low-dimensional manifolds: ($z \in \mathbb{R}^2$)
Kingma and Welling: "Auto-Encoding Variational Bayes" (ICLR 2014)

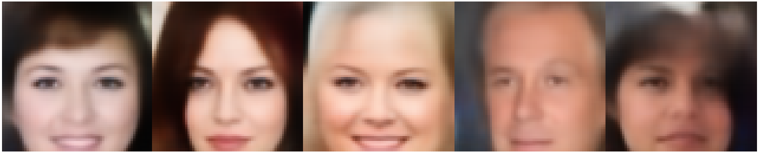


Examples of low-dimensional manifolds: ($z \in \mathbb{R}^2$)
Kingma and Welling: “Auto-Encoding Variational Bayes” (ICLR 2014)





Variational auto-encoder for face generation:



Variational auto-encoder for semi-supervised learning:

Maaloe et al. "Improving Semi-Supervised Learning with Auxiliary Deep Generative Models" (2015)

	100 labels
AtlasRBF (Pitelis et al., 2014)	8.10% ($\pm$0.95)
Deep Generative Model (M1+M2) (Kingma et al., 2014)	3.33% ($\pm$0.14)
Virtual Adversarial (Miyato et al., 2015)	2.12%
Ladder (Rasmus et al., 2015)	1.06% ($\pm$0.37)
Auxiliary Deep Generative Model (1 MC)	2.25% ($\pm$ 0.08)
Auxiliary Deep Generative Model (10 MC)	**0.96% ($\pm$ 0.02)**

Quiz:

Quiz:

- What is the difference between generative and discriminative modeling?

Quiz:

- What is the difference between generative and discriminative modeling?
- What generative modeling techniques do you know about?

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?

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

Important topics of this lecture

- Getting to know variational auto-encoders, a generative modeling technique

Important topics of this lecture

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

Important topics of this lecture

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

Next up:

Autoregressive Methods