Variational Autoencoders

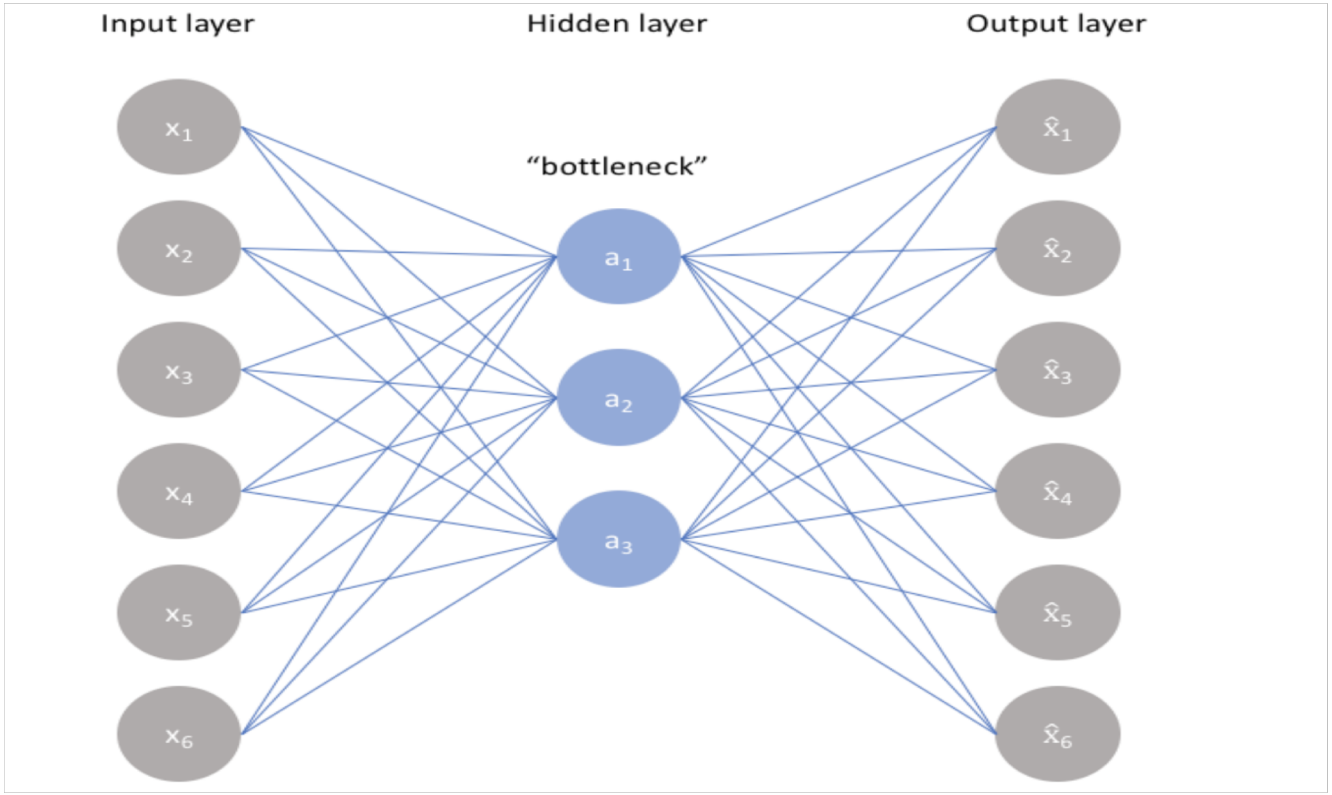
Agenda

- Motivation
- Intuition
  - Why autoencoder doesn’t work?
  - Why it works?
- Variational Inference
- Training

Generative models

- Unsupervised learning
- Model should be able to capture the underlying distribution well
- Able to draw samples repeatedly which are different but still fits the distribution
- Explicit generative models learn the structured-latent space in the process
- Desired
  - Interpolation
  - **Disentanglement**

Autoencoder



Why not Autoencoder ?

- Manipulation in z space not possible
- Can we interpolate?
- How can we fix this?

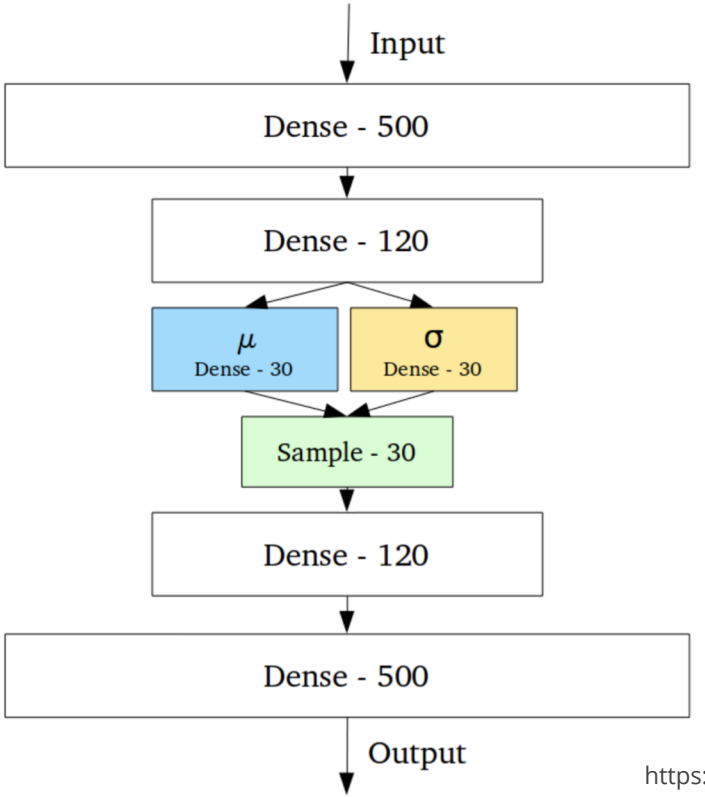
https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf

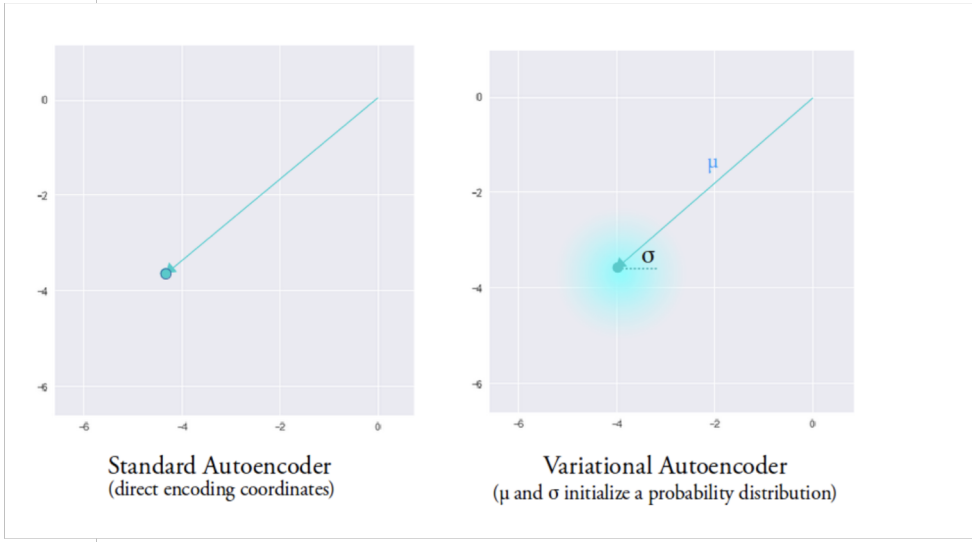
Why not Autoencoder ?

- Manipulation in z space not possible
- Can we interpolate?
- How can we fix this?
  - **Stochasticity**
  - **Regularization**

https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf

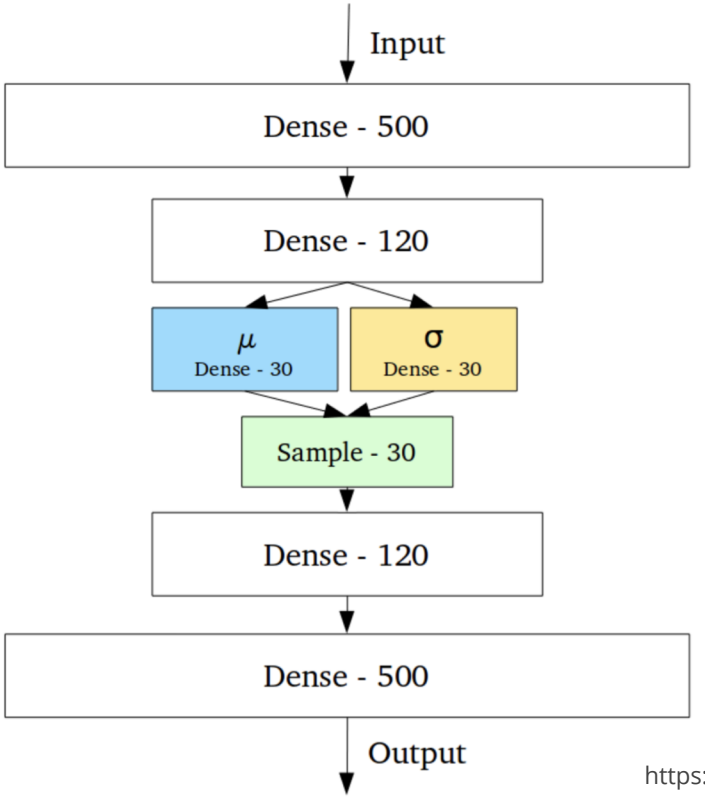
Stochasticity

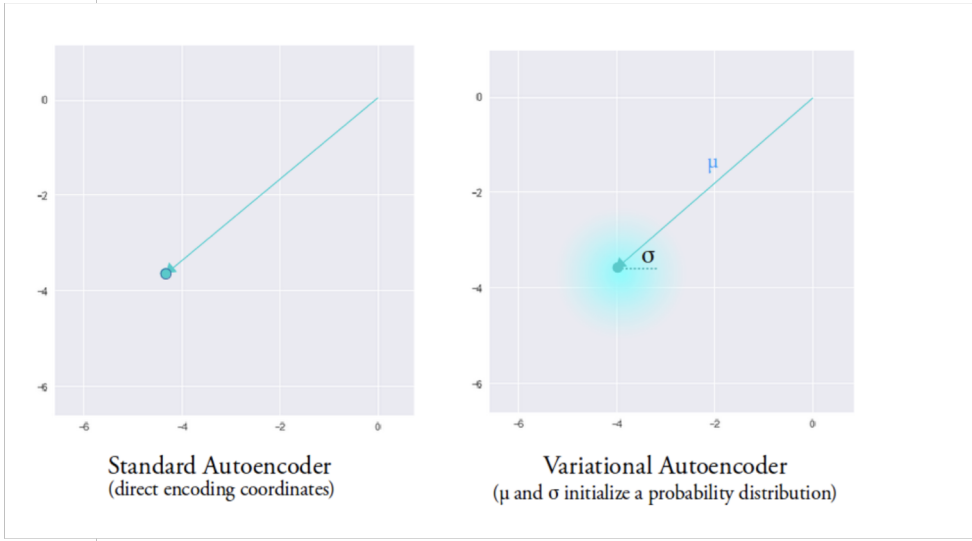




https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf

Stochasticity

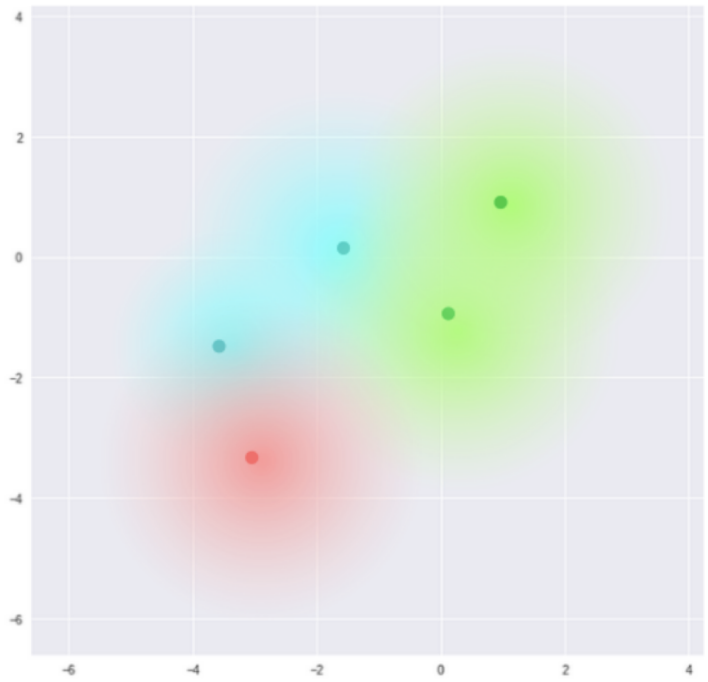




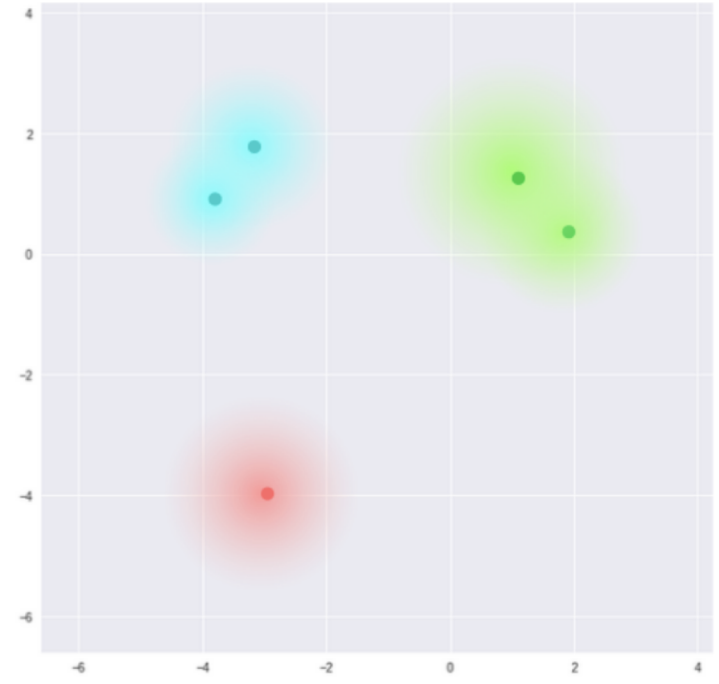
- Is it enough?

https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf

Stochasticity



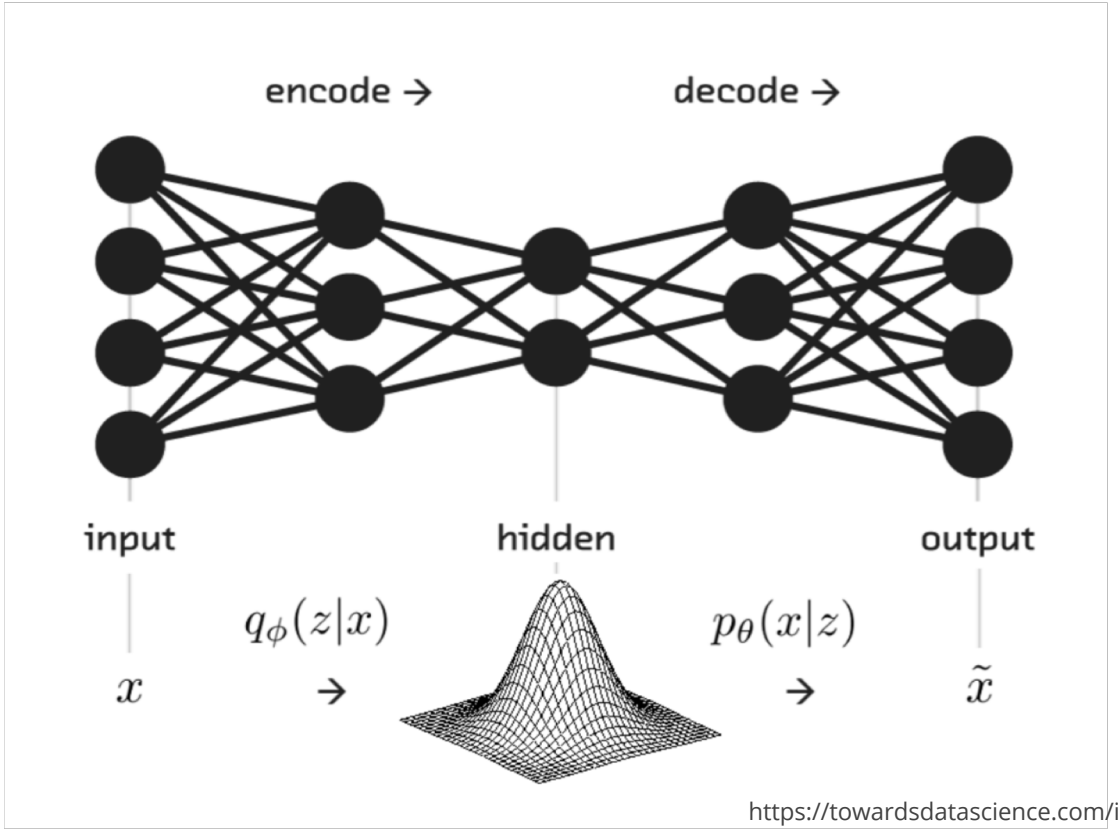
What we require



What we may inadvertently end up with

https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf

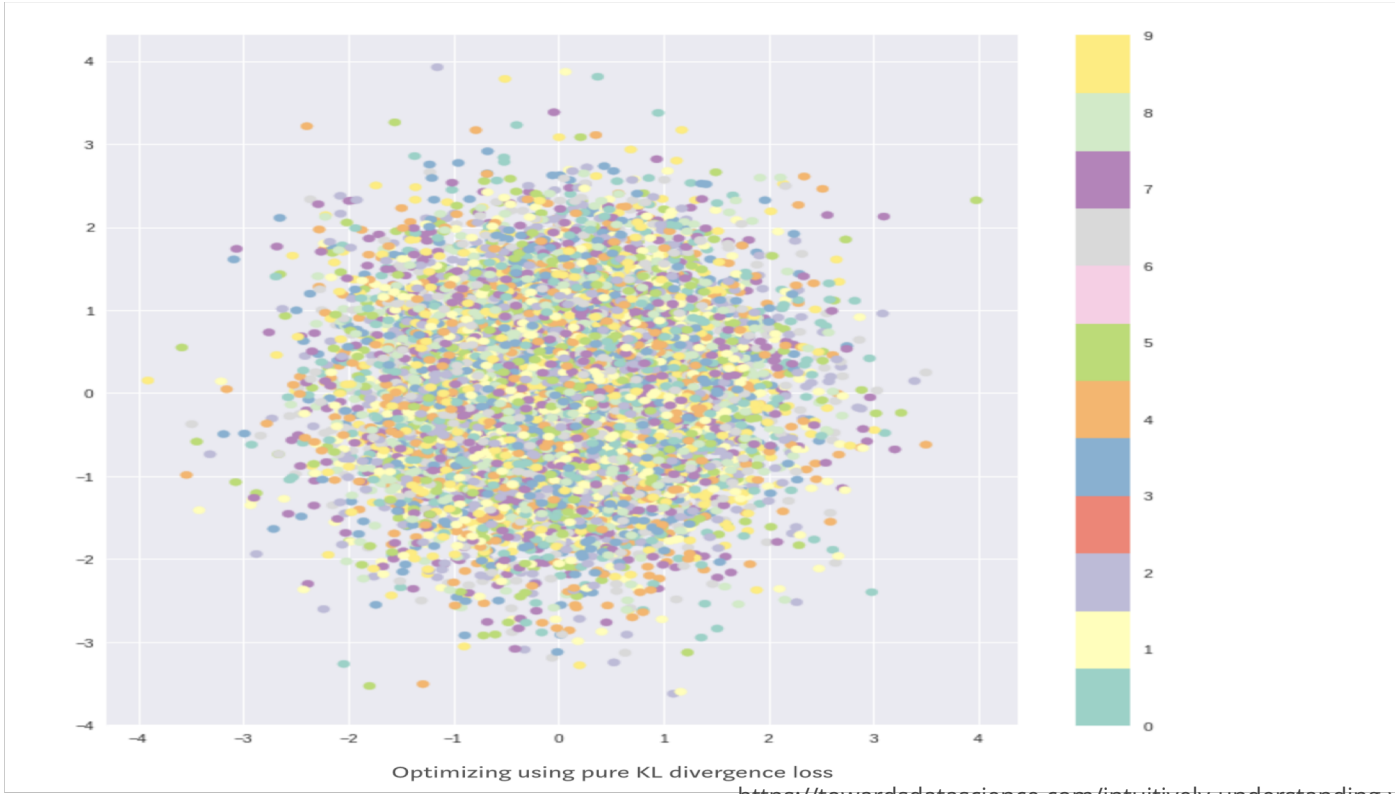
Regularization



- Enforce structure in z-space

https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf

Regularization



https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf

Regularization + Stochasticity

https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf

Variational Inference view

Probabilistic Machine Learning

- A probabilistic model is a joint distribution of hidden variables $\mathbf{z}$ and observed variables $\mathbf{x}$,

$$p(\mathbf{z}, \mathbf{x}).$$

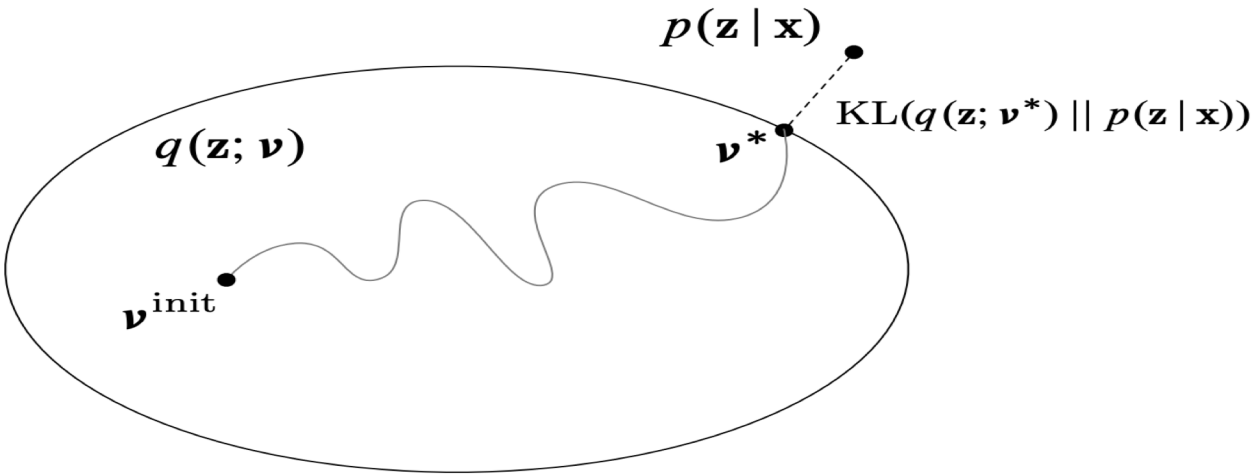
- Inference about the unknowns is through the **posterior**, the conditional distribution of the hidden variables given the observations

$$p(\mathbf{z} \mid \mathbf{x}) = \frac{p(\mathbf{z}, \mathbf{x})}{p(\mathbf{x})}.$$

- For most interesting models, the denominator is not tractable. We appeal to **approximate posterior inference**.

https://www.youtube.com/watch?v=ogdv_6dbvVQ&t=309s

Variational Inference



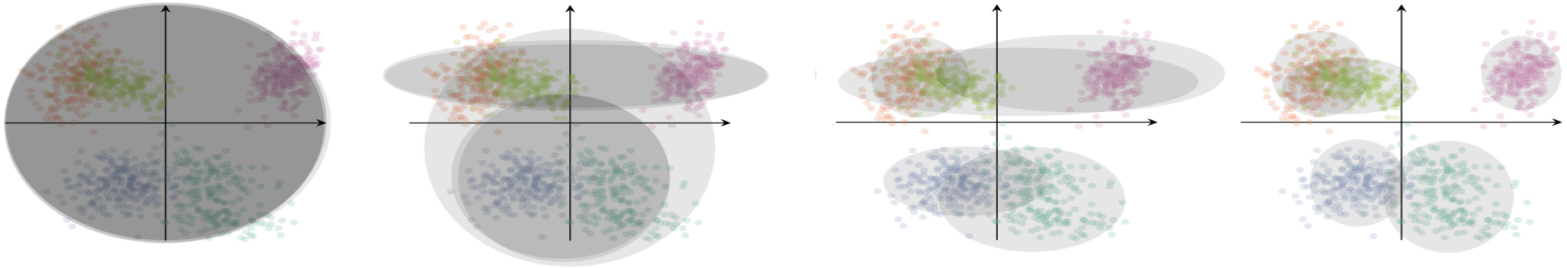
- VI turns **inference into optimization.**
- Posit a **variational family** of distributions over the latent variables,

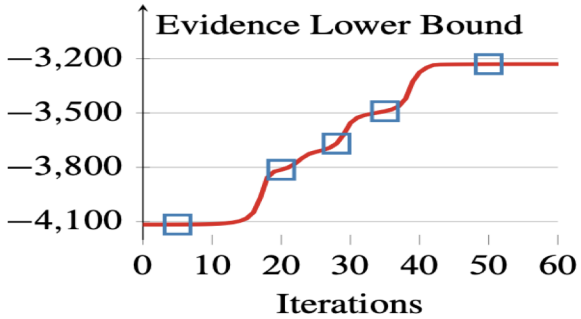
$$q(\mathbf{z}; \boldsymbol{\nu})$$

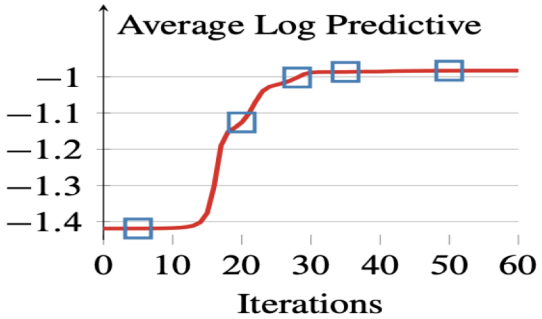
- Fit the **variational parameters** $\boldsymbol{\nu}$ to be close (in KL) to the exact posterior.
  (There are alternative divergences, which connect to algorithms like EP, BP, and others.)

https://www.youtube.com/watch?v=ogdv_6dbvVQ&t=309s

Example: Mixture of Gaussians







[images by Alp Kucukelbir]

https://www.youtube.com/watch?v=ogdv_6dbvVQ&t=309s

Variational Autoencoders - ELBO

- $q_\phi(z|x)$ is a good approximation to $p(z|x)$:

$$
\begin{aligned}
& KL(q_\phi(z|x) \;||\; p(z|x)) \\
&= -\int q_\phi(z|x) \log \frac{p(z|x)}{q_\phi(z|x)} dz \\
&= -\int q_\phi(z|x) \log \frac{p_\theta(x|z)p(z)}{q_\phi(z|x)p(x)} dz \\
&= -\int q_\phi(z|x) \log p_\theta(x|z) dz - \int q_\phi(z|x) \log \frac{p(z)}{q_\phi(z|x)} dz + \int q_\phi(z|x) \log p(x) dz \\
&= -\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] + KL(q_\phi(z|x) \;||\; p(z)) + \log p(x)
\end{aligned}
$$

$$
\log p(x) - KL(q_\phi(z|x) \;||\; p_\theta(z|x)) = \mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - KL(q_\phi(z|x) \;||\; p(z))
$$

Slide borrowed from Siddhesh

Variational Autoencoders - ELBO

- $q_\phi(z|x)$ is a good approximation to $p(z|x)$:

$$\log p(x) - KL(q_\phi(z|x) \,||\, p_\theta(z|x)) = \mathbb{E}_{z\sim q_\phi(z|x)}[\log p(x|z)] - KL(q_\phi(z|x) \,||\, p(z))$$

$$\log p(x) \geq \underbrace{\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - KL(q_\phi(z|x) \,||\, p(z))}_{\text{ELBO}}$$

Slide borrowed from Siddhesh

Variational Autoencoders - ELBO

- $q_\phi(z|x)$ is a good approximation to $p(z|x)$:

$$\log p(x) - KL(q_\phi(z|x) \,||\, p_\theta(z|x)) = \mathbb{E}_{z\sim q_\phi(z|x)}[\log p(x|z)] - KL(q_\phi(z|x) \,||\, p(z))$$

$$\log p(x) \geq \underbrace{\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)]}_{\text{Reconstruction Error}} - KL(q_\phi(z|x) \,||\, p(z))$$

Slide borrowed from Siddhesh

Variational Autoencoders - ELBO

- $q_\phi(z|x)$ is a good approximation to $p(z|x)$:

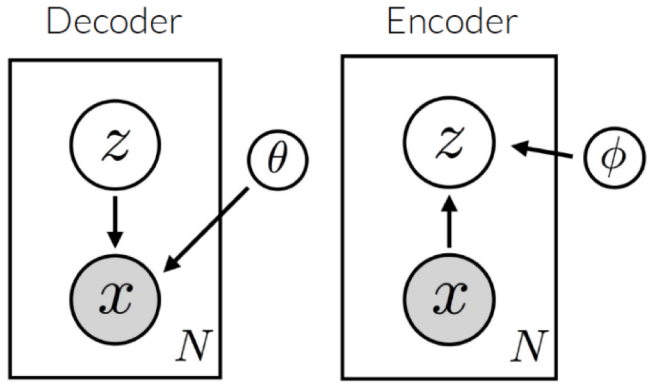
$$\log p(x) - KL(q_\phi(z|x) \,||\, p_\theta(z|x)) = \mathbb{E}_{z\sim q_\phi(z|x)}[\log p(x|z)] - KL(q_\phi(z|x) \,||\, p(z))$$

$$\log p(x) \geq \mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - \underbrace{KL(q_\phi(z|x) \,||\, p(z))}_{\text{Proposed distribution is close to prior}}$$

Slide borrowed from Siddhesh

Variational Autoencoders - Overview

- Decoder : $p_\theta(x|z)$
- Inference: $p(z|x) = \dfrac{p_\theta(x|z)p(z)}{p(x)}$
- **Key Idea:** Approximate posterior with a family of distributions $q_\phi(z|x)$
- Encoder : $q_\phi(z|x)$

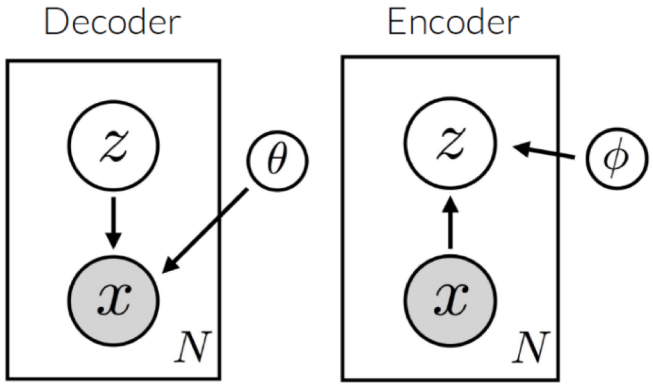


Figures taken from http://pyro.ai/examples/vae.html

Slide borrowed from Siddhesh

Variational Autoencoders - Overview

- Decoder : $p_\theta(x|z)$
- Encoder : $q_\phi(z|x)$
- **Goal:** Estimate parameters $\theta$and ϕ
  - $q_\phi(z|x)$ is a good approximation to $p(z|x)$:



Figures taken from http://pyro.ai/examples/vae.html

Slide borrowed from Siddhesh

Stochasticity in training

- How to train this model end-to-end?

Stochasticity in training

- How to train this model end-to-end?
  - **Reparameterization trick**
  - Possible due to gaussian prior

Variational Autoencoders - Training

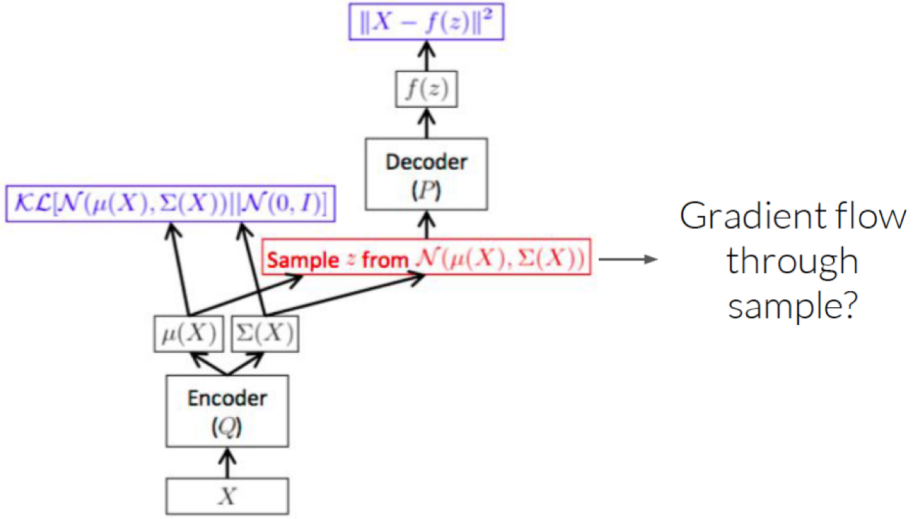


Figure taken from http://blog.qure.ai/notes/using-variational-autoencoders

Slide borrowed from Siddhesh

Variational Autoencoders - Training

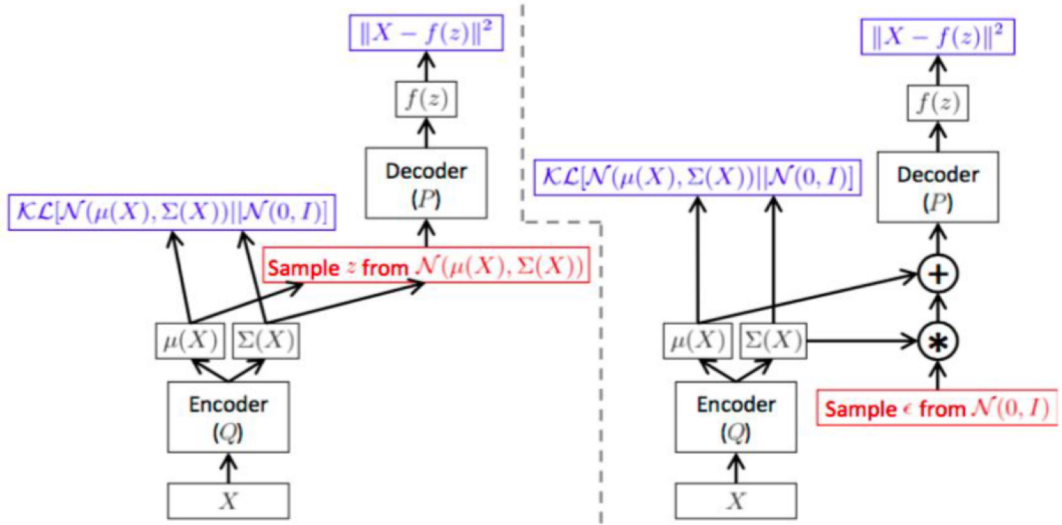


Figure taken from http://blog.qure.ai/notes/using-variational-autoencoders

Slide borrowed from Siddhesh

Variational Autoencoders - Training

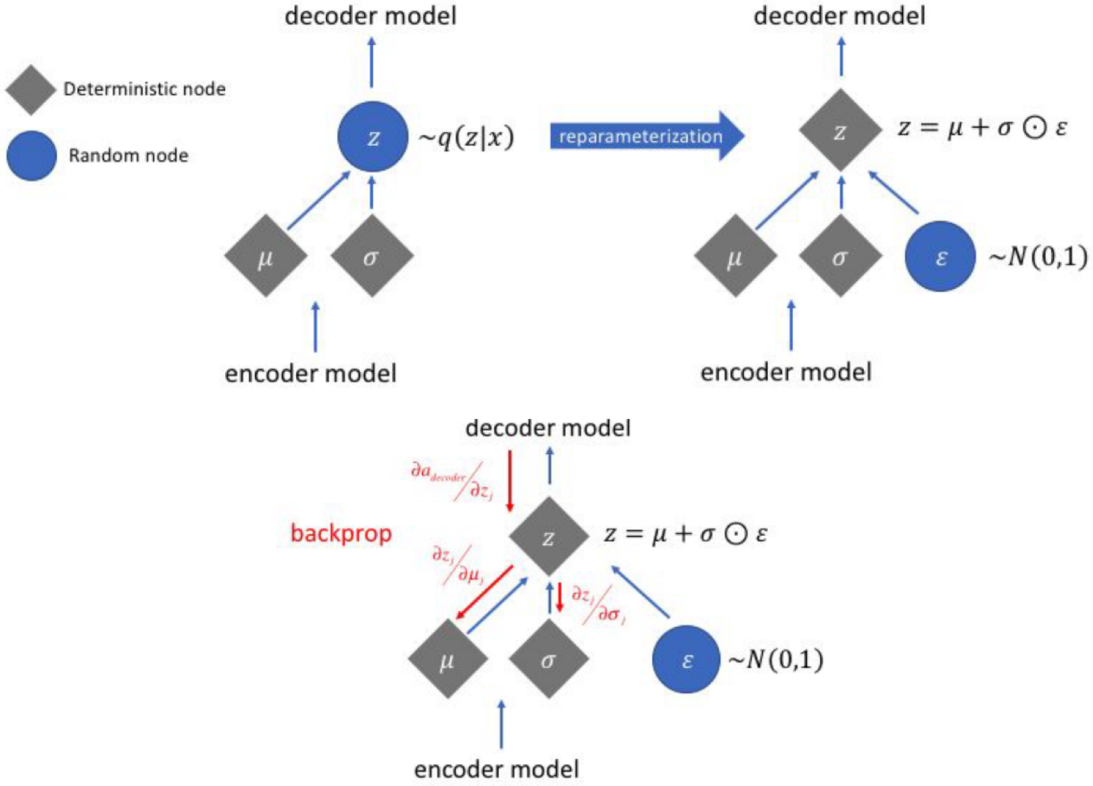


Figure taken from https://www.jeremyjordan.me/variational-autoencoders/

Slide borrowed from Siddhesh

Disentanglement*

- Imagine if you are generating images of person
  - You might want to tell the model : “I want to generate someone who looks like this person, but is taller”
  - Easy if height was disentangled from other features in z-space encoding
- Idea of beta-VAEs
  - Increase the regularization

$$\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \beta\, D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z}))$$

Conclusion

- VAEs try to learn meaningful representation of data
- 2 parts
  - Reconstruction
  - Regularization | KL Divergence term
- Training possible due to reparameterization trick
- Useful to have disentanglement
  - Future works like beta-VAE focus on this