Pattern Analysis & Machine Intelligence Praktikum: MLPR-WS19

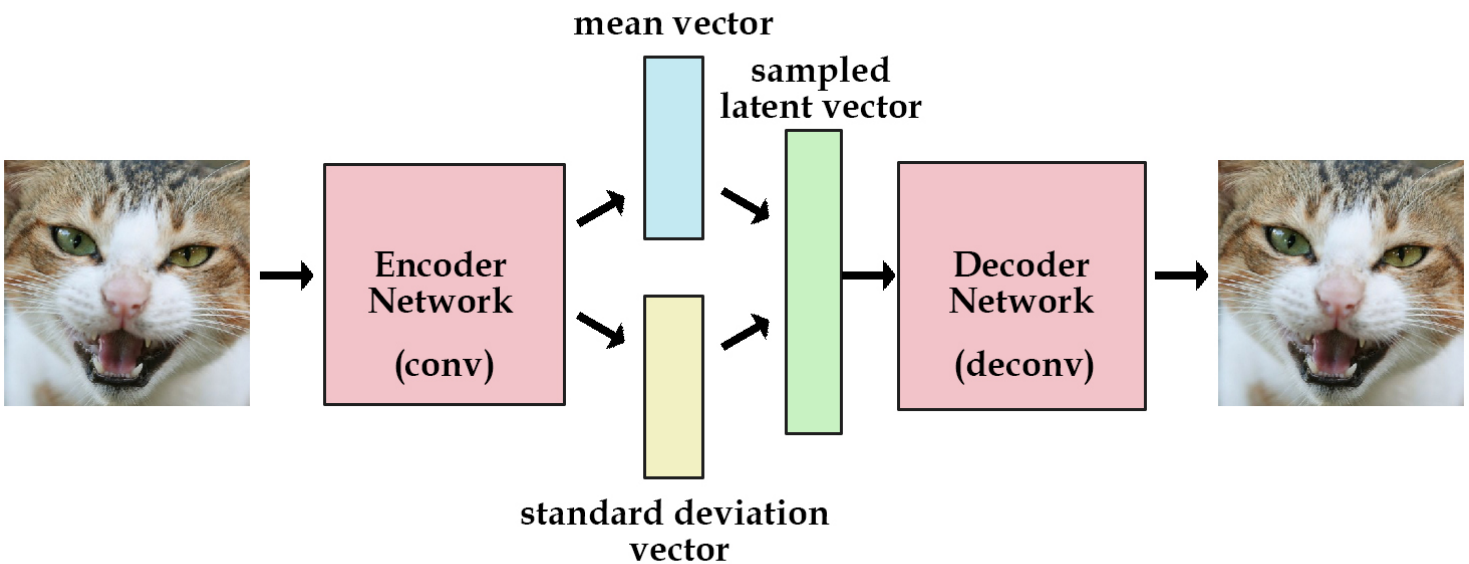
Martin Mundt, Dr. Iuliia Pliushch, Prof. Dr. Visvanathan Ramesh

Goethe Uni Frankfurt

Variational Autoencoders

Variational Autoencoders

- Our AE has a latent embedding/variables in the hidden layer connecting encoder to decoder. But difficult to grasp as it is unconstrained.
- We can add a constraint such as forcing the latent vector to follow a unit Gaussian (e.g. by optimizing KL divergence in addition to reconstruction, which measures how close we are to a unit Gaussian).



http://kvfrans.com/content/images/2016/08/vae.jpg

Variational Autoencoders

- Consider a dataset X with variable x
- Assume data is generated by some random process involving unobserved random variable z
- z is generated from some prior distribution $p_\theta(z)$
- a value x is generated from some conditional distribution $p_\theta(x|z)$

$\rightarrow$ The parameters and values of latent variables z are not known to us. $p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}, \mathbf{z}) dz$ is intractable.

$\rightarrow$ The true posterior density $p_\theta(z|x) = p_\theta(x|z)p_\theta(z)/p_\theta(x)$ is intractable.

Variational Autoencoders

"Auto-Encoding Variational Bayes", Kingma and Welling, ICLR 2014

- Why a Bayesian formulation? $\rightarrow$ Lets us learn about the distribution of seen data $p(\mathbf{x})$ by capturing it through latent variables $\mathbf{z}$. However, as $p(\mathbf{x}) = \int p(\mathbf{x}, \mathbf{z}) dz$ is intractable, we do this by optimizing a lower-bound to the marginal distribution $p(\mathbf{x})$.
- How to do approximate Bayesian inference with neural networks + learning with probabilistic models whose latent variables have intractable posterior distributions $\rightarrow$ variational inference.
- In a VAE we use a reparameterization trick to make the model backward differentiable.

Variational Autoencoders

The densities of the marginal and joint distribution are related through Bayes rule:

$$p_\theta(\mathbf{z}|\mathbf{x}) = \frac{p_\theta(\mathbf{x},\mathbf{z})}{p_\theta(\mathbf{x})} = \frac{p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})}{p_\theta(\mathbf{x})} \tag{1}$$

Using the logarithm on both sides we can write this as a sum:

$$\log p_\theta(\mathbf{x}) = \log p_\theta(\mathbf{x}|\mathbf{z}) + \log p_\theta(\mathbf{z}) - \log p_\theta(\mathbf{z}|\mathbf{x}) \tag{2}$$

Here, we do not know our real posterior $p_\theta(\mathbf{z}|\mathbf{x})$. We thus use variational inference and introduce an approximation $q_\phi(\mathbf{z}|\mathbf{x})$ to the posterior:

$$\log p_\theta(\mathbf{x}) = \int_z q_\phi(\mathbf{z}|\mathbf{x}) \left[\log p_\theta(\mathbf{x}|\mathbf{z}) + \log p_\theta(\mathbf{z}) - \log p_\theta(\mathbf{z}|\mathbf{x})\right] dz \tag{3}$$

Variational Autoencoders

$$\log p_\theta(\mathbf{x}) = \int_z q_\phi(\mathbf{z}|\mathbf{x})[\log p_\theta(\mathbf{x}|\mathbf{z}) + \log p_\theta(\mathbf{z}) - \log p_\theta(\mathbf{z}|\mathbf{x}) + \log q_\phi(\mathbf{z}|\mathbf{x}) - \log q_\phi(\mathbf{z}|\mathbf{x})]dz \tag{4}$$

Using the definition of reverse Kullback Leibler divergence $KL(Q \,||\, P) = \int_{-\infty}^{\infty} Q(x) \log \frac{Q(x)}{P(x)}$:

$$\log p_\theta(\mathbf{x}) = KL(q_\phi(\mathbf{z}|\mathbf{x}) \,||\, p_\theta(\mathbf{z}|\mathbf{x})) + \mathbb{E}_{z\sim q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - KL(q_\phi(\mathbf{z}|\mathbf{x}) \,||\, p_\theta(\mathbf{z})) \tag{5}$$

Here, the first term on the right hand side cannot be evaluated, but by definition is strictly positive. Therefore if we optimize the remaining terms on the right hand side, we optimize a lower-bound to $p(\mathbf{x})$.

Variational Autoencoder

Thus, the marginal likelihood can be rewritten as:

$$\log p_\theta(x) = D_{KL}\left(q_\phi(z|x)||p_\theta(z|x)\right) + \mathfrak{L}(\theta, \phi; x)$$

$\rightarrow$ First RHS term is the KL divergence between approximate and true posterior
$\rightarrow$ Second RHS term is called the variational lower bound on the marginal likelihood of a datapoint as the KLD is always positive. With our derivation from above we can write it as:

$$\log p_\theta(x) \geq \mathfrak{L}(\theta, \phi; x) = -D_{KL}\left(q_\phi(z|x)||p_\theta(z)\right) + \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right]$$

Variational Autoencoders

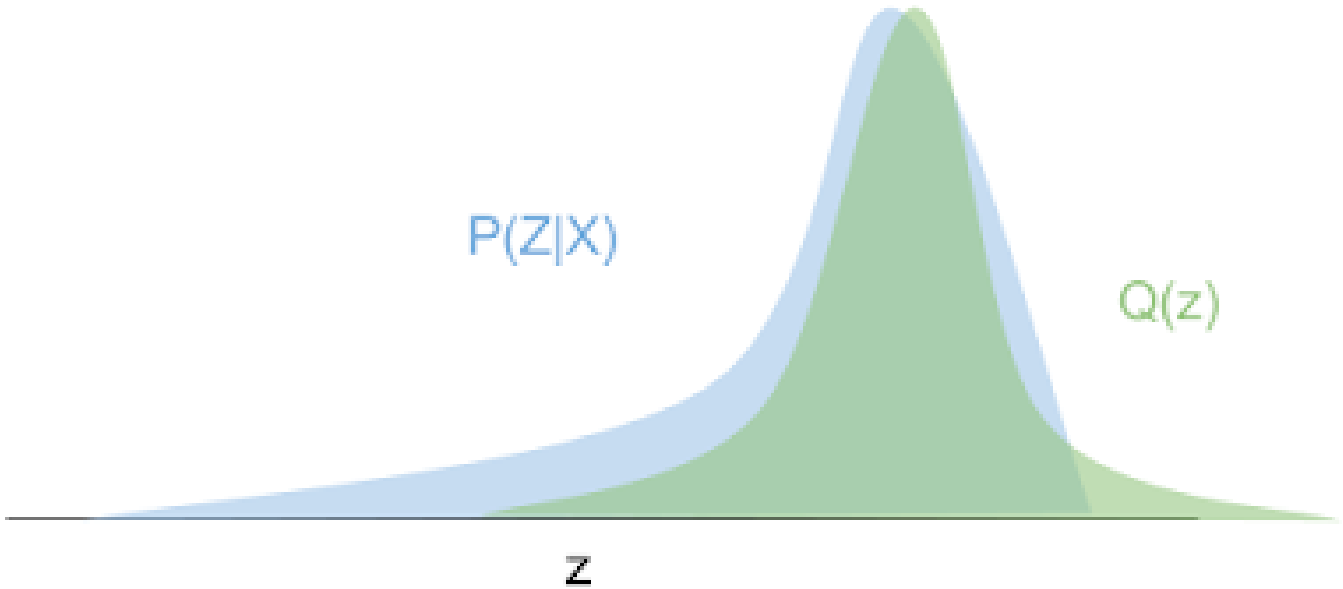
$$\mathfrak{L}(\theta, \phi; x) = -D_{KL}\left(q_\phi(z|x)||p_\theta(z)\right) + \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right]$$

- Recognition model/probabilistic encoder: $q_\phi(z|x)$: variational approximation to intractable posterior $p_\theta(z|x)$
- z has an interpretation as latent representation or code. $q_\phi(z|x)$: probabilistic encoder $\rightarrow$ given a datapoint x it produces a distribution over possible values of z from which it could have been generated
- $p_\theta(x|z)$: probabilistic decoder $\rightarrow$ given a z it produces a distribution over possible values of x

- The first RHS term is a KL divergence encouraging the approximate posterior to be close to the prior $p_\theta(z)$
- Second RHS term is expected reconstruction error as given by the log-likelihood (estimated by sampling)

Variational Autoencoders

https://blog.evjang.com/2016/08/variational-bayes.html - read more about variational inference and the nuances of Kullback-Leibler divergence at this blog post.



"Reverse KL divergence measures the amount of information (in nats, or units of $\frac{1}{\log 2}$ bits) required to "distort" $p_\theta(\mathbf{z})$ into $q_\phi(\mathbf{z})$"

Variational Autoencoders

Example: let approximate variational posterior be a multivariate Gaussian:

$$\log q_\phi(z|x) = \log \mathcal{N}(z; \mu, \sigma \, \mathtt{I})$$

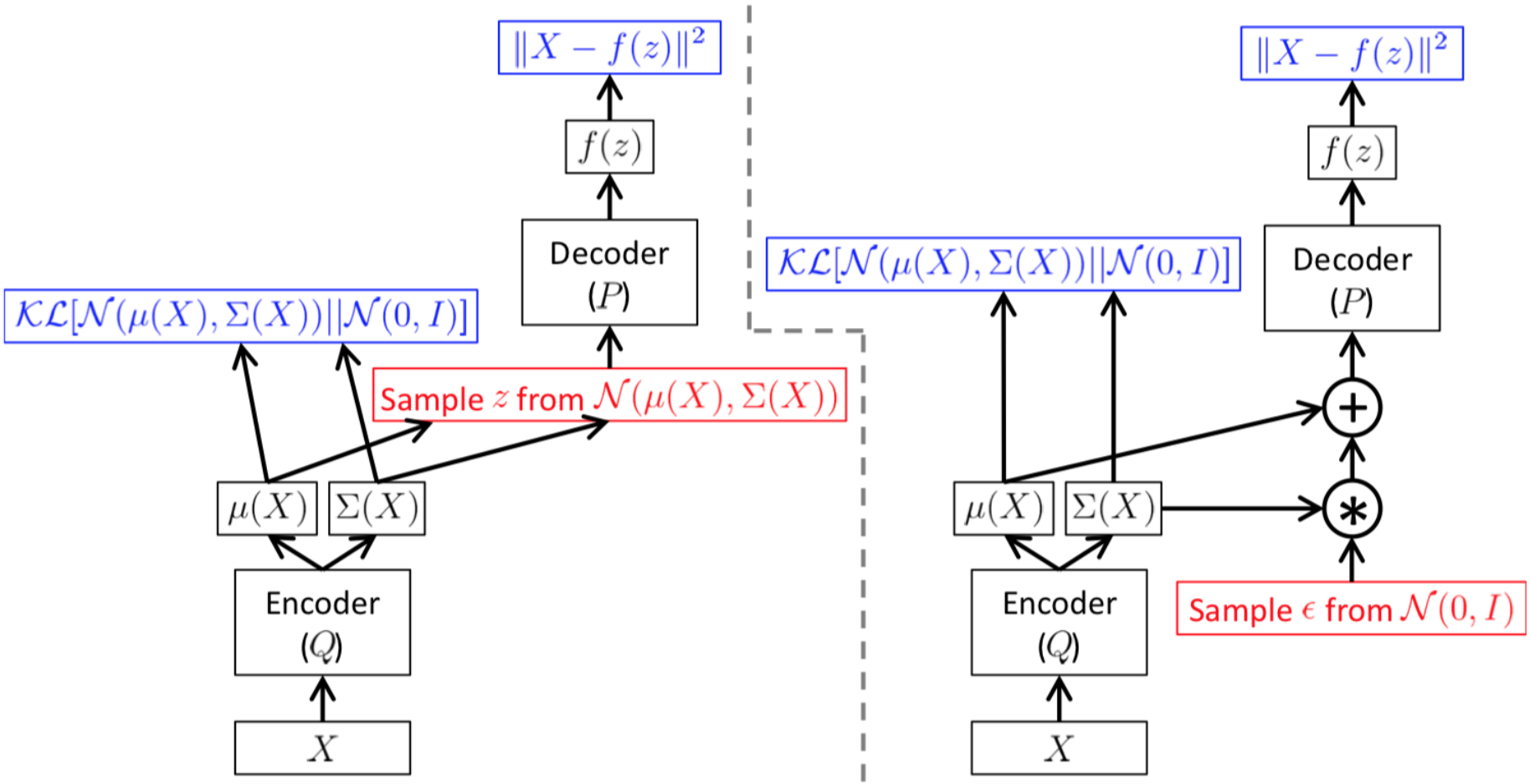
- Use a reparameterization trick to generate samples from $q_\phi(z|x) \rightarrow$ express random variable z as deterministic variable.
- Sample the posterior $z \sim q_\phi(z|x)$ using $z = \mu + \sigma \circ \epsilon$

$$\mathfrak{L}(\theta, \phi; x) \approx \tfrac{1}{2} \left(1 + \log \sigma^2 - \mu^2 - \sigma^2\right) + \tfrac{1}{L} \sum_{l=1}^{L} \log p_\theta(x|z^l)$$

where

$$z^l = \mu + \sigma \circ \epsilon^l \quad \text{and } \epsilon^l \sim \mathcal{N}(0, \mathtt{I})$$

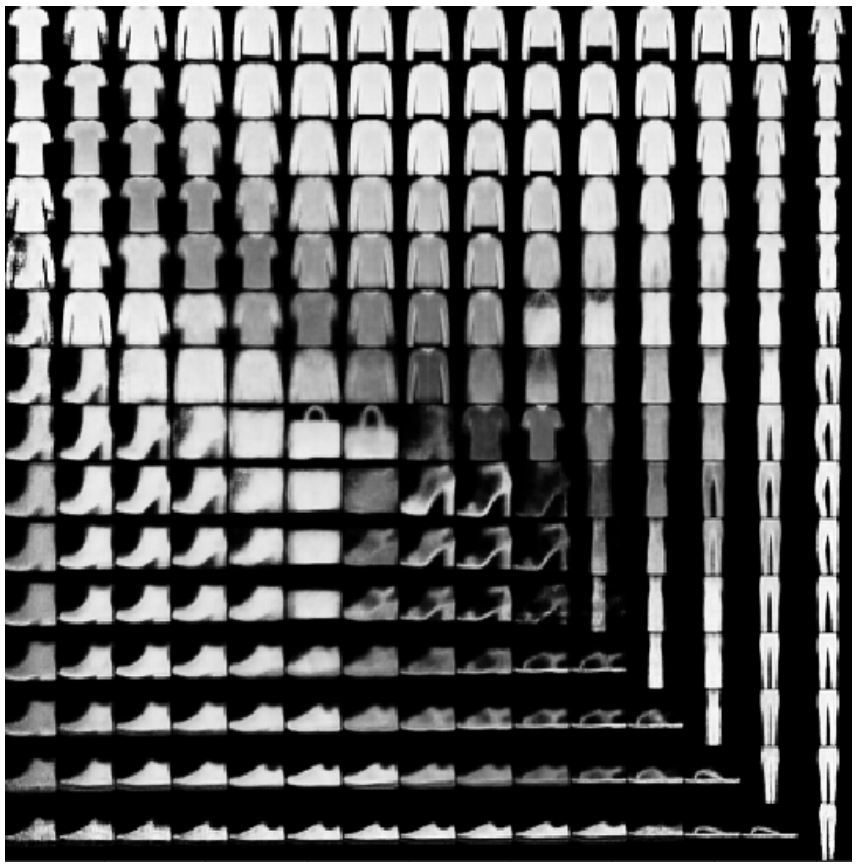
Variational Autoencoders



"Tutorial on Variational Autoencoders", Carl Doersch

Variational Autoencoders

- After the model is trained we can sample images by sampling from the prior $z \sim p_\theta(\mathbf{z})$, here $\mathcal{N}(0, 1)$, and calculating the decoder.
- Example for a 2-D latent space grid of z values for Fashion MNIST:



from: https://github.com/sadanand-singh/datasciencevision.com/blob/master/content/posts/2018-04-26-autoencoders.md