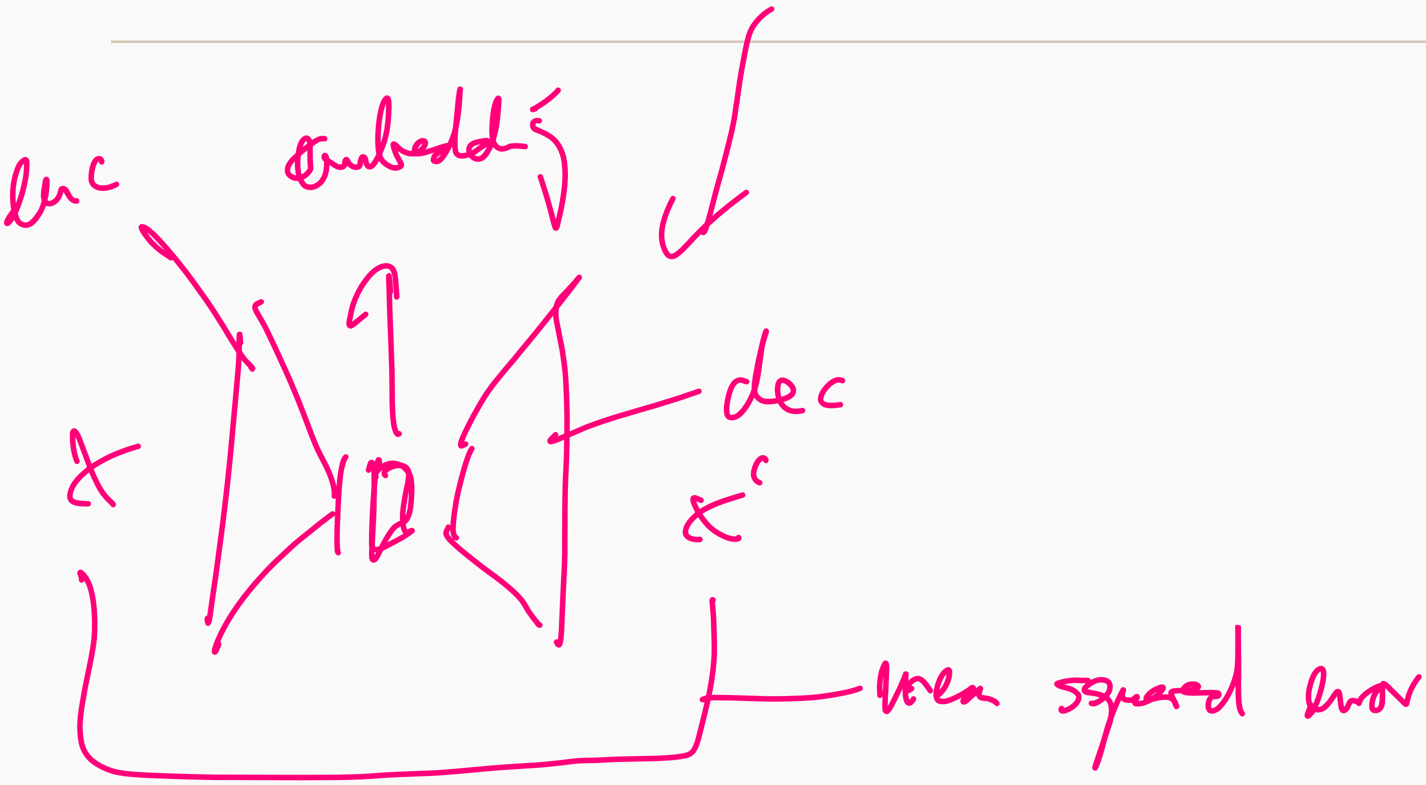
Deep Learning

Ulf Brefeld + SOHAM MAJUMDER !

build: June 17, 2024

Machine Learning Group
Leuphana University of Lüneburg

Variational Autoencoders



Autoencoders learn latent representations of data by a fairly straightforward procedure: encode data into the latent representation then attempt to reconstruct the original inputs from them.

A problem is that this *latent space* has no inherent structure: in principle, no assumption is made on which properties it should have beyond its dimensionality.

By reframing the problem from a probabilistic perspective, we can see how to induce a specific distribution for the latent space.

In generative modeling, we develop and use methods that learn the data generation process: a model of the data we observe.

As done previously, we start by considering a maximum likelihood problem: we model the true data distribution $p(x)$ with a parametrized distribution $p_\theta(x)$ and optimize θ such that $p_\theta(x)$ is maximimized under the data we observe.

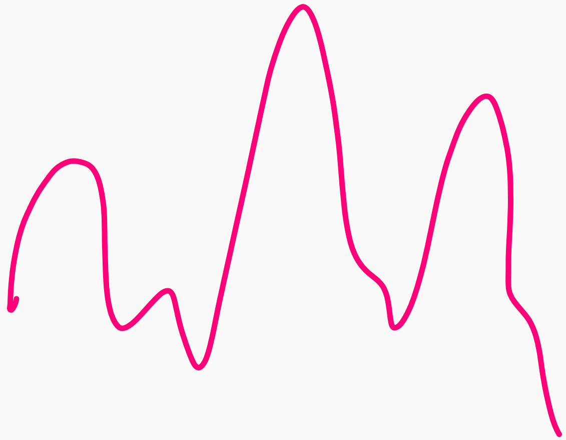
A *latent variable model* addresses this problem by introducing *unobserved* (latent) variables z.

Consider the distribution $p_\theta(x, z)$: the joint distribution of the input space $\mathcal{X}$ (where our data comes from) and the latent space $\mathcal{Z}$ of latent (unbserved) variables. This distribution is also parametrized by θ.

According to our joint distribution, we can calculate $p_\theta(x)$ as:

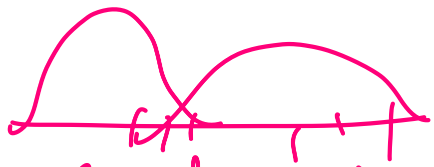
$$p_\theta(x) = \int p_\theta(x, z)dz,$$

which is also referred to as a *marginal* likelihood



→ Expectation Maximization

Model for $p(x)$!?



$p(z_{nk} \mid x \dots)$

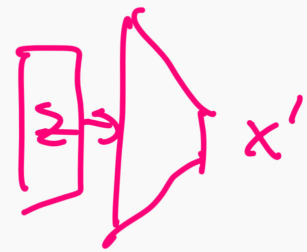
One advantage of deep neural networks is that they can model very complex functions.

As such, they seem a natural candidate to model $p_\theta(x, z)$. The most straightforward factorization (assumption) for this distribution is:

$$p_\theta(x, z) = p_\theta(z)p_\theta(x|z)$$

- $p_\theta(z)$: *prior* distribution over z

A model of $p(x, z)$ using a neural network for x representing independent binary variables following a Bernoulli distribution:



$$
\begin{aligned}
z &\sim \mathcal{N}(0, I) \\
(p_1, p_2, \ldots, p_D) &= \mathrm{NeuralNet}_\theta(z) \\
\log p(x|z) &= \sum_{j=1}^{D} \log p(x_j|z) = \sum_{j=1}^{D} \log \mathrm{Ber}(x_j; p_j) \\
&= \sum_{j=1}^{D} x_j \log p_j + (1 - x_j) \log(1 - p_j)
\end{aligned}
$$

where $0 \leq p_j \leq 1$ (e.g., output is a sigmoid layer of $\mathrm{NeuralNet}_\theta(\cdot)$) and $\mathrm{Ber}(\cdot; p_j)$ is a Bernoulli distribution with parameter p_j.

While those allow very flexible $p(x, z)$ to be modelled, the earlier marginal likelihood becomes intractable in this setting.

Note that this also affects the posterior $p_\theta(z|x)$, as per the relationship

$$p(z|x) = \frac{p(x, z)}{p(x)} \Rightarrow p(x) = \frac{p(x, z)}{p(z|x)}.$$

if $p(x)$ is intractable, then $p(z|x)$ is also intractable.

Variational Autoencoders solve this problem by introducing an approximation of $p_\theta(z|x)$, the *approximate posterior* $q_\phi(z|x)$, which is parametrized by *variational* parameters ϕ.

The distribution $q_\phi(z|x)$ is also parametrized by a neural network. For instance:

$$
\begin{aligned}
(\mu, \log \sigma) &= \mathrm{NeuralNet}_\phi(x) \\
q_\phi(z|x) &= \mathcal{N}(z; \mu, \mathrm{diag}(\sigma))
\end{aligned}
$$

Approximating the posterior $p_\theta(z|x)$ helps us optimize the marginal likelihood:

$$
\begin{aligned}
\log p_\theta(x) &= \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x)\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p_\theta(x,z)}{p_\theta(z|x)}\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p_\theta(x,z)q_\phi(z|x)}{q_\phi(z|x)p_\theta(z|x)}\right] \\
&= \underbrace{\mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p_\theta(x,z)}{q_\phi(z|x)}\right]}_{\text{ELBO } \mathcal{L}_{\theta,\phi}} + \underbrace{\mathbb{E}_{q_\phi(z|x)}\left[\log \frac{q_\phi(z|x)}{p_\theta(z|x)}\right]}_{D_{\mathrm{KL}}(q_\phi(z|x)\|p_\theta(z|x))}
\end{aligned}
$$

Importantly, one should note that:

$$\log p_\theta(x) = \underbrace{\mathbb{E}_{q(z|x)}\left[\log \frac{p_\theta(x,z)}{q_\phi(z|x)}\right]}_{\mathcal{L}_{\theta,\phi}(x)} + D_{\mathrm{KL}}(q_\phi(z|x)\|p_\theta(z|x))$$

>0

$$\begin{aligned}\mathcal{L}_{\theta,\phi}(x) &= \log p_\theta(x) - D_{\mathrm{KL}}(q_\phi(z|x)\|p_\theta(z|x)) \\ &\leq \log p_\theta(x) \quad \text{(since KL div. is strictly positive)}\end{aligned}$$

discard KL-D.

This means that $\mathcal{L}_{\theta,\phi}(x)$ represents a lower bound on the log-likelihood of the data, which is why it is referred to as *evidence lower bound* (ELBO). Furthermore, maximizing it brings the aproximate posterior $q_\phi(z|x)$ closer to the true posterior $p_\theta(z|x)$ as it implies the KL divergence becomes smaller.

One problem still remains: optimizing the ELBO through SGD requires sampling from $q_\phi(z|x)$:

$$\mathcal{L}_{\theta,\phi}(x) = \mathbb{E}_{q(z|x)}\left[\log p_\theta(x,z) - \log q_\phi(z|x)\right],$$

which is impractical.

To solve this, we can use the *reparametrization trick*: we rewrite $z \sim q_\phi(z|x)$ as a function h of another random variable ϵ which is independent from x and ϕ:

$$\begin{aligned}
\epsilon &\sim p(\epsilon) \\
z &= h(\phi, x, \epsilon) \\
\tilde{\mathcal{L}}_{\theta,\phi}(x) &= \log p_\theta(x,z) - \log q_\phi(z|x).
\end{aligned}$$

The gradients $\nabla_{\theta,\phi}\tilde{\mathcal{L}}_{\theta,\phi}$ are an unbiased estimator of the gradients of $\mathcal{L}_{\theta,\phi}$.

perceptron update:

$$w^t \leftarrow w^{t-1} + yx$$

here: / ADAM (similar side effects)

$$z \leftarrow z + yx$$

With these, a variational autoencoder can be built by modeling $p_\theta(x, z)$ and $q_\phi(z|x)$ and optimizing $\tilde{\mathcal{L}}_{\theta,\phi}$ with respect to θ and ϕ.

Usual choices for a VAE:

- Encoder $f(\phi, \cdot)$

$$\mu^f, \sigma^f = f(\phi, x)$$

turn x into parameters of distribution

$$q_\phi(z|x) = \mathcal{N}(\mu^f, \mathrm{diag}(\sigma^f))$$

- Decoder $g(\theta, \cdot, \cdot)$

draw z

$$p_\theta(x, z) = p_\theta(x|z)p_\theta(z)$$

$$p_\theta(z) = \mathcal{N}(0, I) \quad \text{(prior over latent space)}$$

$$\epsilon \sim p(\epsilon) = \mathcal{N}(0, I)$$

$$z = \epsilon \circledcirc \sigma^f + \mu^f \quad \text{(reparametrization trick)}$$

parameters another distribution

$$\mu^g, \sigma^g = g(\theta, z, \epsilon)$$

$$p_\theta(x|z) = \mathcal{N}(\mu^g, \mathrm{diag}(\sigma^g))$$

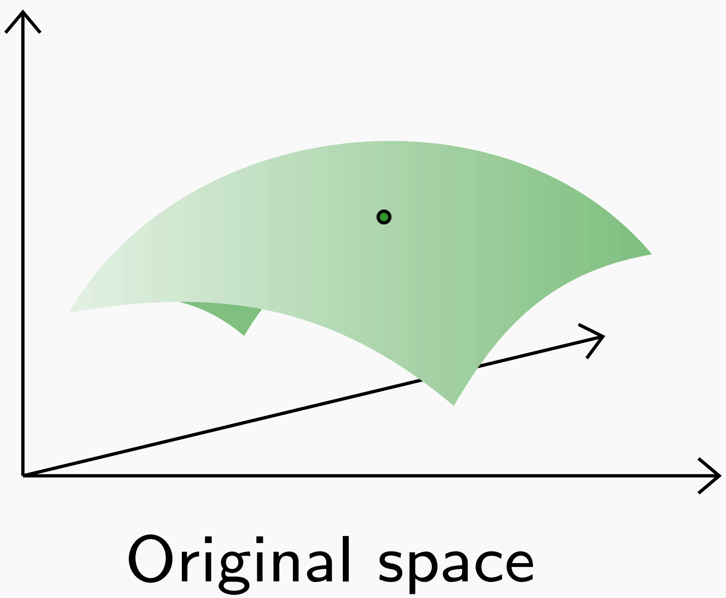
draw reconstruction x' from final distribution

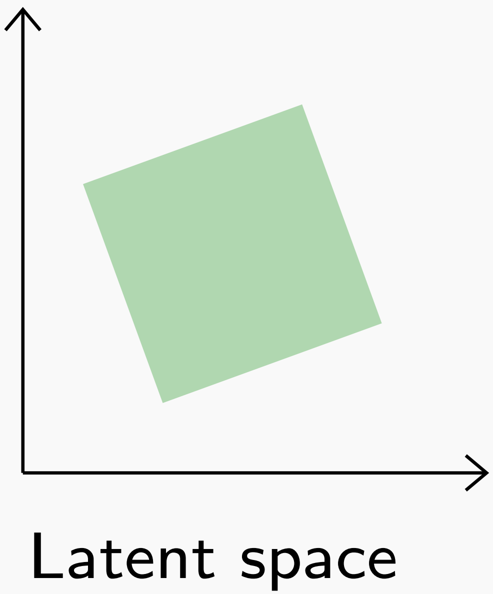
A usual interpretation for the ELBO $\mathcal{L}_{\theta,\phi}$ is:

$$
\begin{aligned}
\mathcal{L}_{\theta,\phi} &= \mathbb{E}_{q_\phi(z|x)}\left[\log \underbrace{p_\theta(x,z)}_{p_\theta(x|z)p_\theta(z)} - \log q_\phi(z|x)\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z) + \log p_\theta(z) - \log q_\phi(z|x)\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z) + \log \frac{p_\theta(z)}{q_\phi(z|x)}\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z) - \log \frac{q_\phi(z|x)}{p_\theta(z)}\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right] - \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{q_\phi(z|x)}{p_\theta(z)}\right] \\
&= \underbrace{\mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right]}_{\text{reconstruction error}} - \underbrace{D_{\mathsf{KL}}(q_\phi(z|x) \| p_\theta(z))}_{\text{latent space matches prior?}}
\end{aligned}
$$

KL term of the ELBO:

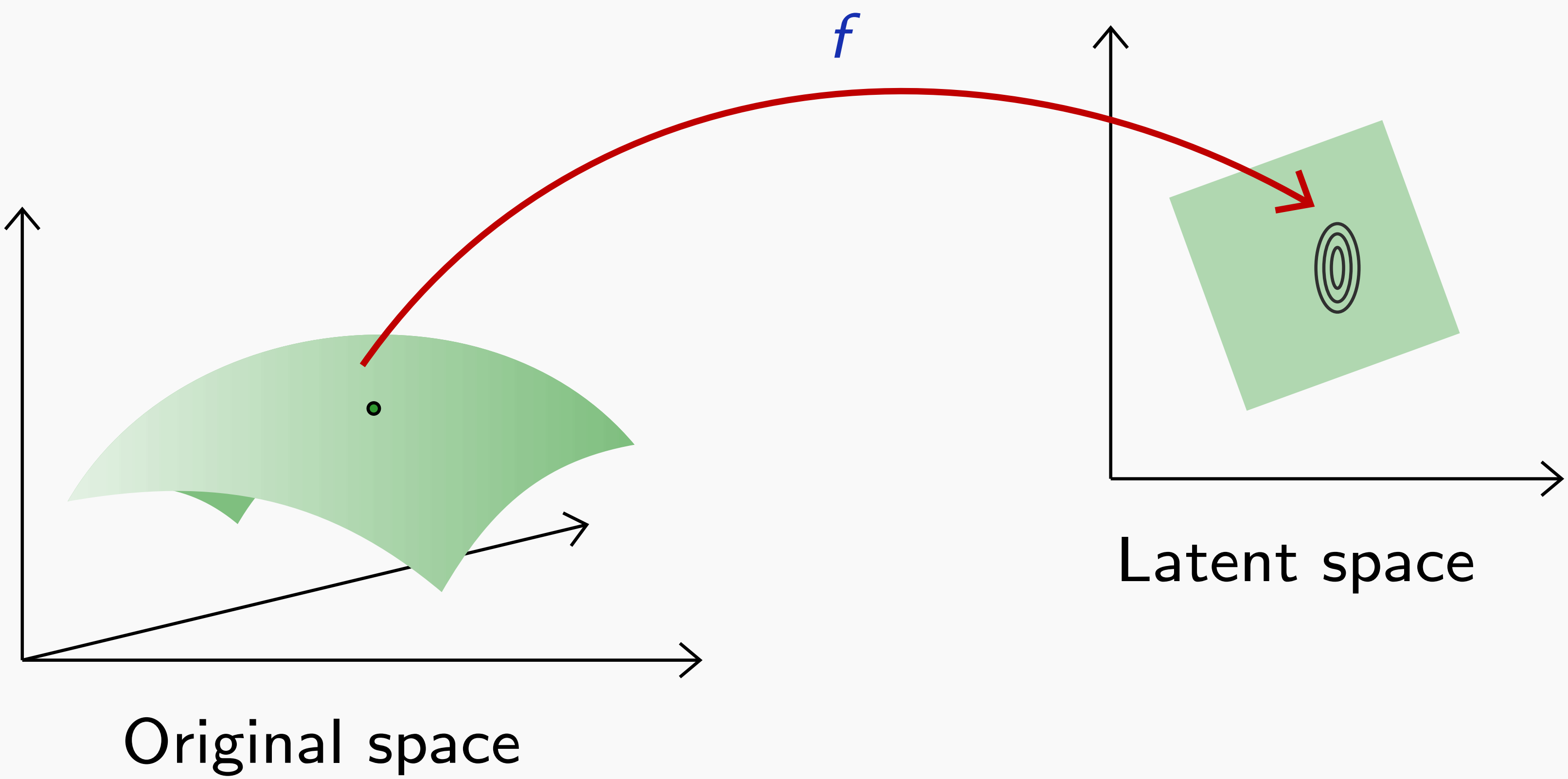
$$D_{\mathsf{KL}}\Big(\underbrace{q_\phi(z|x)}_{\mathcal{N}(\mu^f,\mathrm{diag}(\sigma^f))} \,\|\, \underbrace{p_\theta(z)}_{\mathcal{N}(0,I)}\Big) = -\frac{1}{2}\sum_{j=1}^{d_{\mathsf{latent}}}\left(1 + 2\log\sigma_j^f - (\mu_d^f)^2 - (\sigma_d^f)^2\right)$$





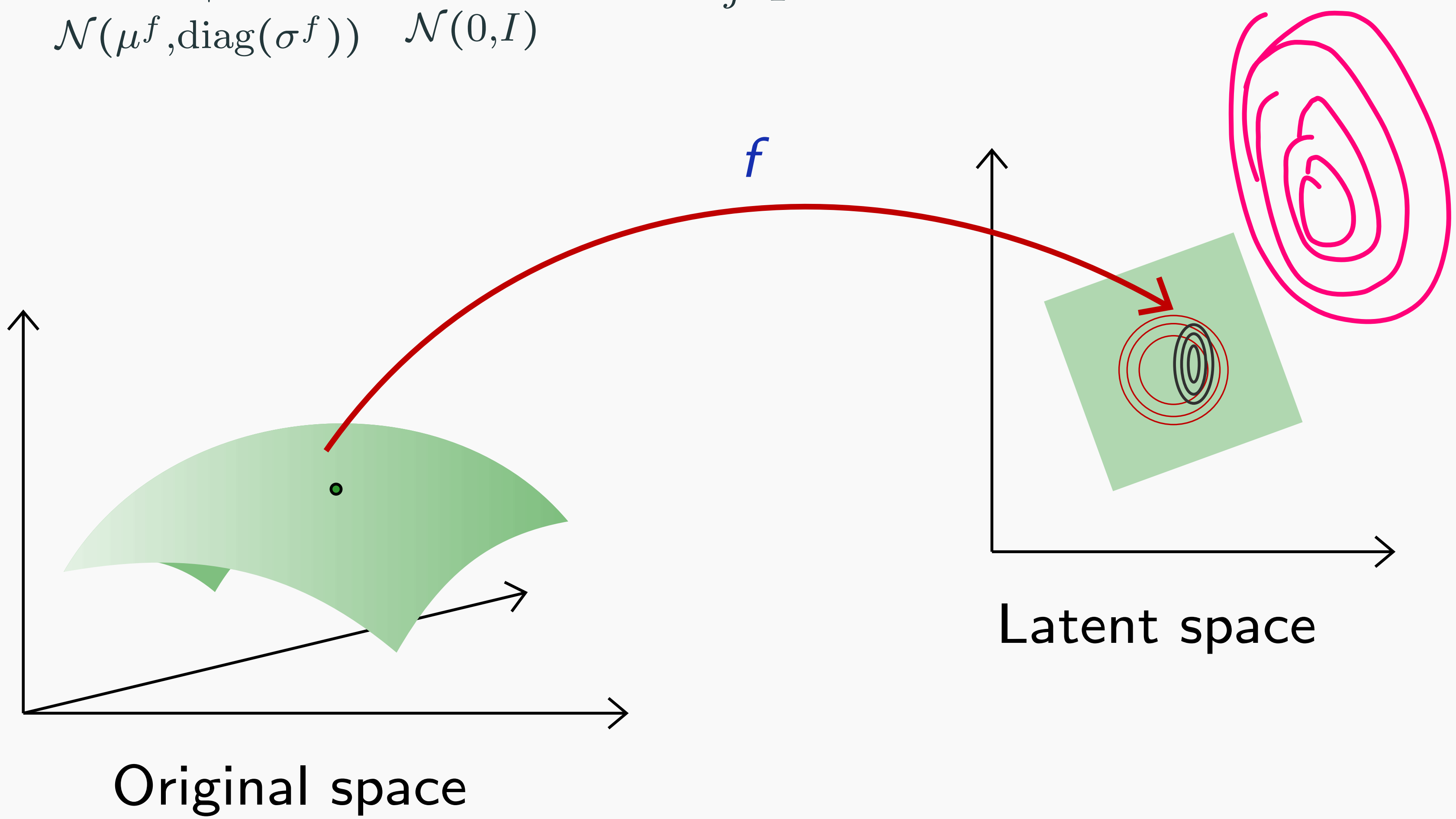
KL term of the ELBO:

$$D_{\mathsf{KL}}\Big(\underbrace{q_\phi(z|x)}_{\mathcal{N}(\mu^f,\mathrm{diag}(\sigma^f))} \,\Big\|\, \underbrace{p_\theta(z)}_{\mathcal{N}(0,I)}\Big) = -\frac{1}{2}\sum_{j=1}^{d_{\mathsf{latent}}}\left(1 + 2\log\sigma_j^f - (\mu_d^f)^2 - (\sigma_d^f)^2\right)$$



KL term of the ELBO:

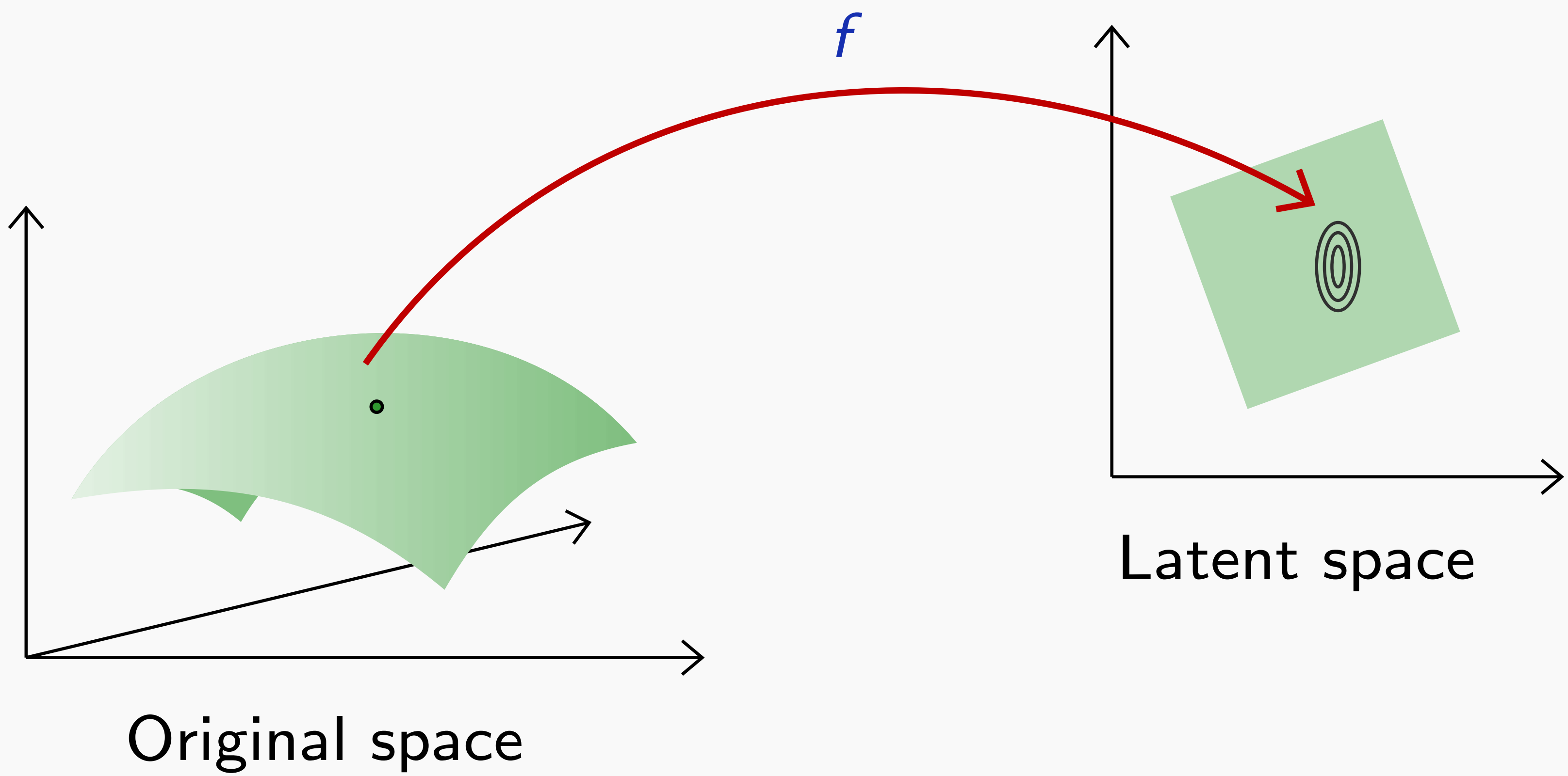
$$D_{\mathsf{KL}}\big(\underbrace{q_\phi(z|x)}_{\mathcal{N}(\mu^f,\mathrm{diag}(\sigma^f))} \,\|\, \underbrace{p_\theta(z)}_{\mathcal{N}(0,I)}\big) = -\frac{1}{2}\sum_{j=1}^{d_{\mathsf{latent}}}\Big(1 + 2\log\sigma_j^f - (\mu_d^f)^2 - (\sigma_d^f)^2\Big)$$



Assuming a Gaussian with identity covariance (meaning $\sigma^g = \mathbf{1}$), the other term becomes similar to a reconstruction error:

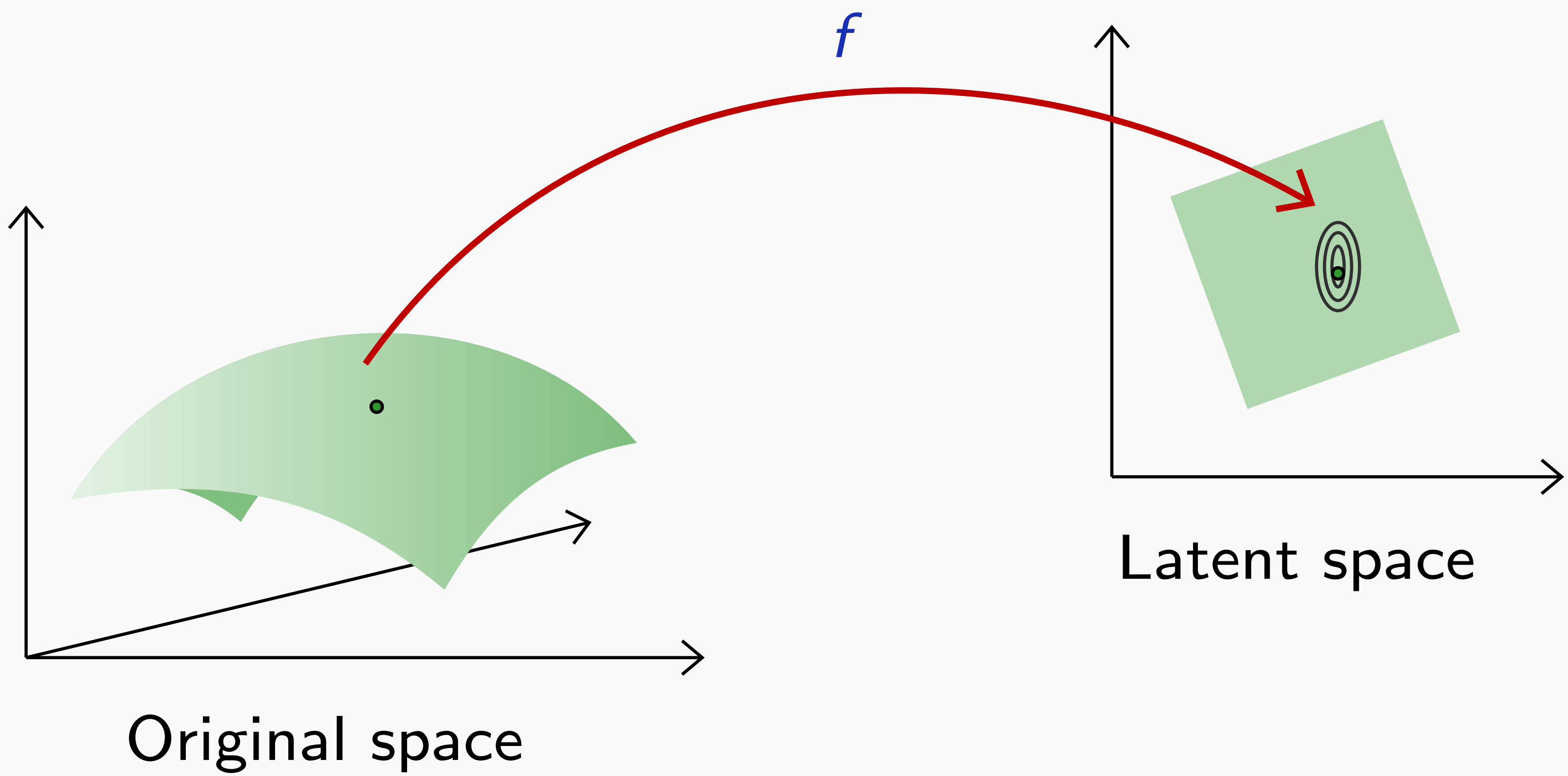
$p(x|z) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{(x-\mu)^2}{2\sigma^2} \right\}$

$$\log p_\theta(x|z) = \frac{1}{2} \sum_{j=1}^{d_{\text{input}}} (x_j - \mu_j^g)^2 + \text{constant}$$



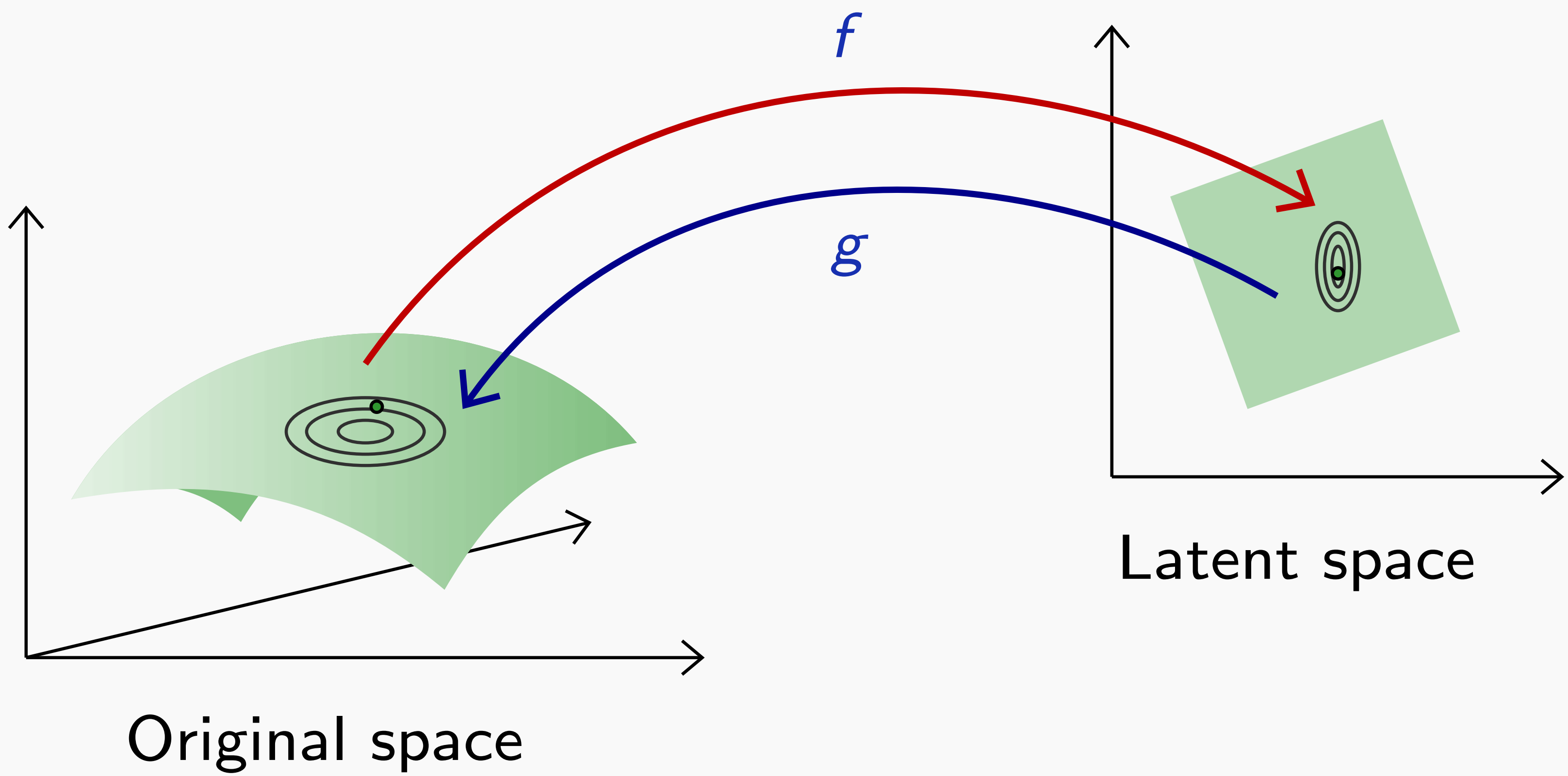
Assuming a Gaussian with identity covariance (meaning $\sigma^g = \mathbf{1}$), the other term becomes similar to a reconstruction error:

$$\log p_\theta(x|z) = \frac{1}{2} \sum_{j=1}^{d_{\text{input}}} (x_j - \mu_j^g)^2 + \text{constant}$$

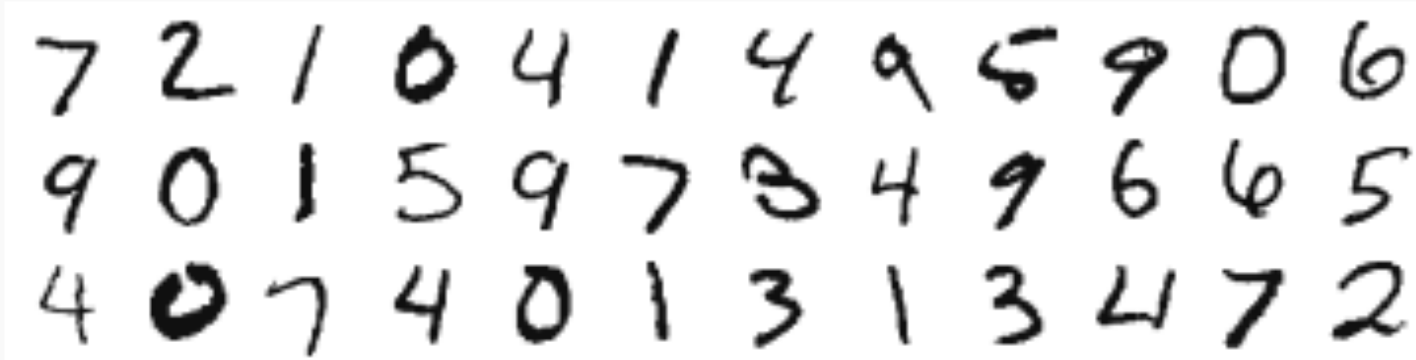


Assuming a Gaussian with identity covariance (meaning $\sigma^g = \mathbf{1}$), the other term becomes similar to a reconstruction error:

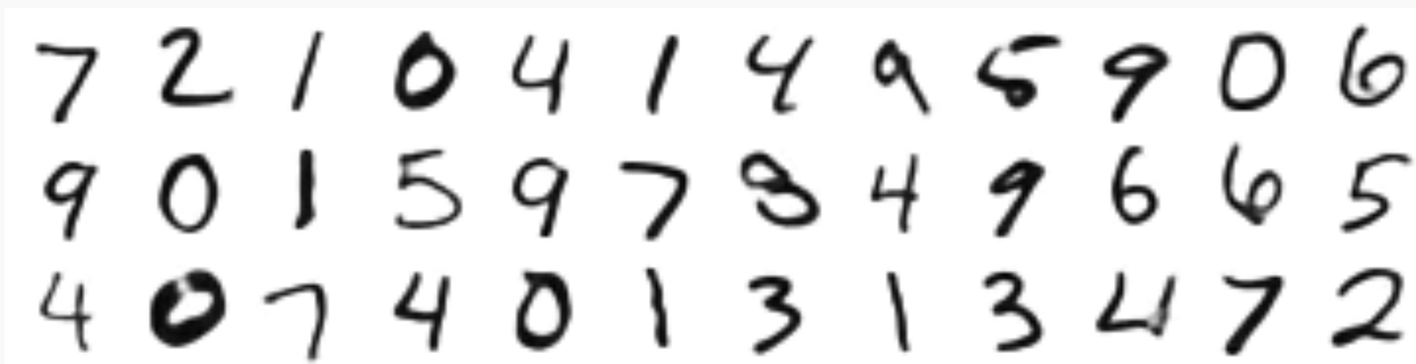
$$\log p_\theta(x|z) = \frac{1}{2}\sum_{j=1}^{d_{\text{input}}}(x_j - \mu_j^g)^2 + \text{constant}$$



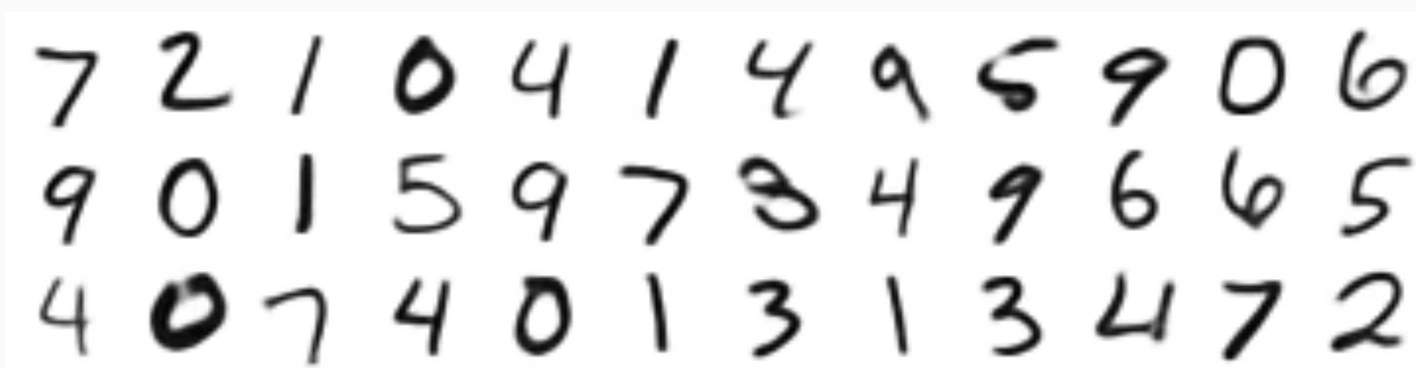
Original



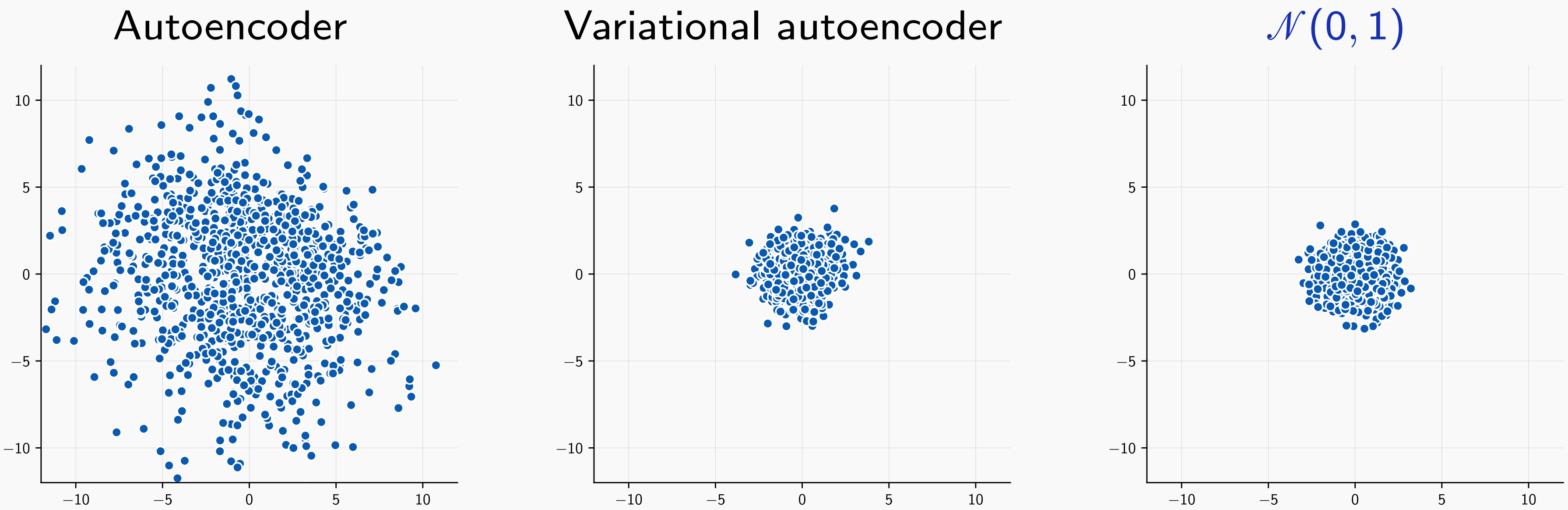
Autoencoder reconstruction ($d = 32$)



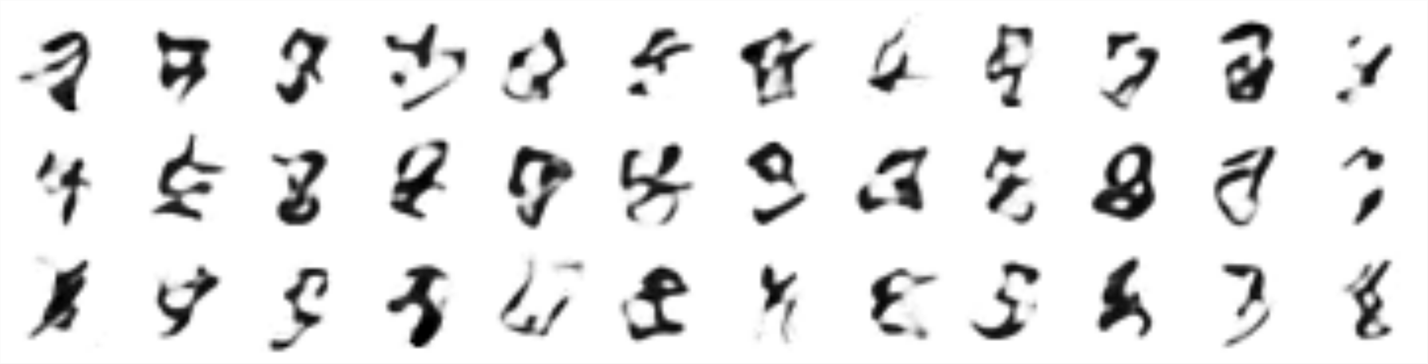
Variational Autoencoder reconstruction ($d = 32$)



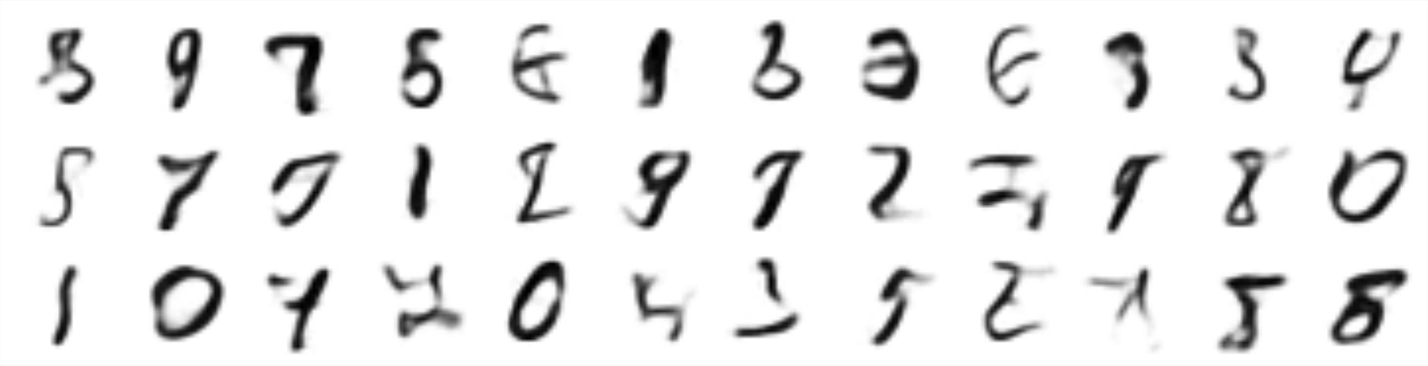
Inspecting 2 latent dimensions of autoencoders



Autoencoder sampling ($d = 32$)



Variational Autoencoder sampling ($d = 32$)

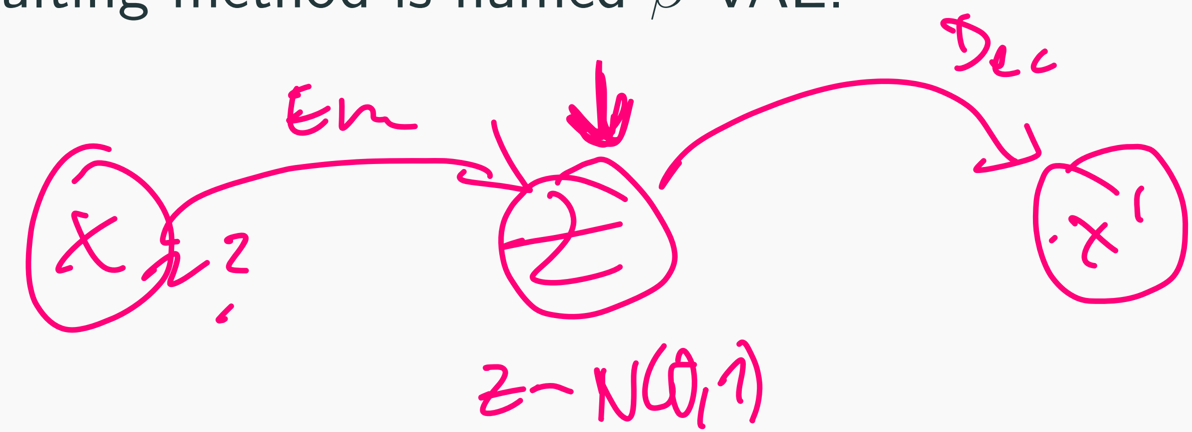


Encouraging the latent space to follow $\mathcal{N}(0, 1)$ often results in "disentangled" representations.

This effect can be controlled by introducting a hyperparameter β in the ELBO:

$$\mathcal{L}_{\theta,\phi} = \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right] - \beta D_{\mathsf{KL}}(q_\phi(z|x) \| p_\theta(z)).$$

The resulting method is named β-VAE.



References

$x \rightarrow z \rightarrow x$

Kingma, D. P., and Welling, M. An Introduction to Variational Autoencoders (2019). arXiv preprint arXiv:1906.02691.

$p(z|x)$