Lecture 8: Autoencoders

Learning representation – no supervision

How do we know what is relevant and what is not?

Main idea of unsupervised representation learning: a new representation should be "smaller" but allow to recover the original one.

Why smaller? (because we need to eliminate irrelevant, because we need something small)

Recap: PCA

We now want to pick the *d* optimal components $\{v_1, \ldots, v_D\} \subset \mathbb{R}^N$

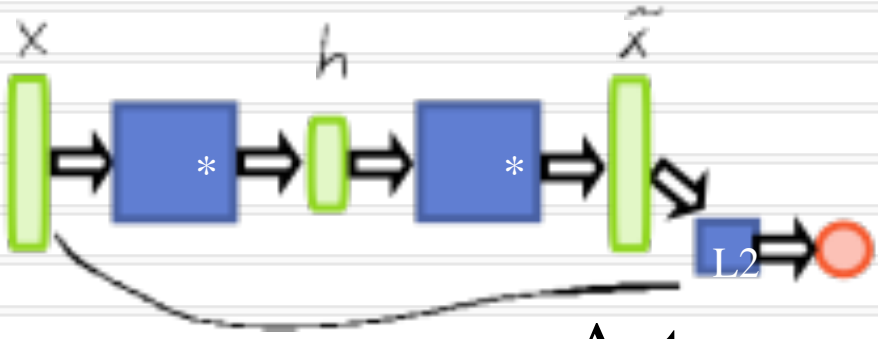
Let $\tilde{X} = [\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_M]$
be the reconstruction of the original vectors.

We want to minimize the reconstruction error:

$$\sum_{i}^{M} \| x_i - \tilde{x}_i \|_2^2 \rightarrow \min$$

$$\| X - \tilde{X} \|_F^2 \rightarrow \min$$

Neural network version (autoencoder)



PCA

$h = U x$

$\tilde{x} = U h$

$\min_U \| x - \tilde{x} \|_2^2$

s.t. $\langle u u' \rangle = \delta_{ij}$

Autoencoder:

$h = W x$

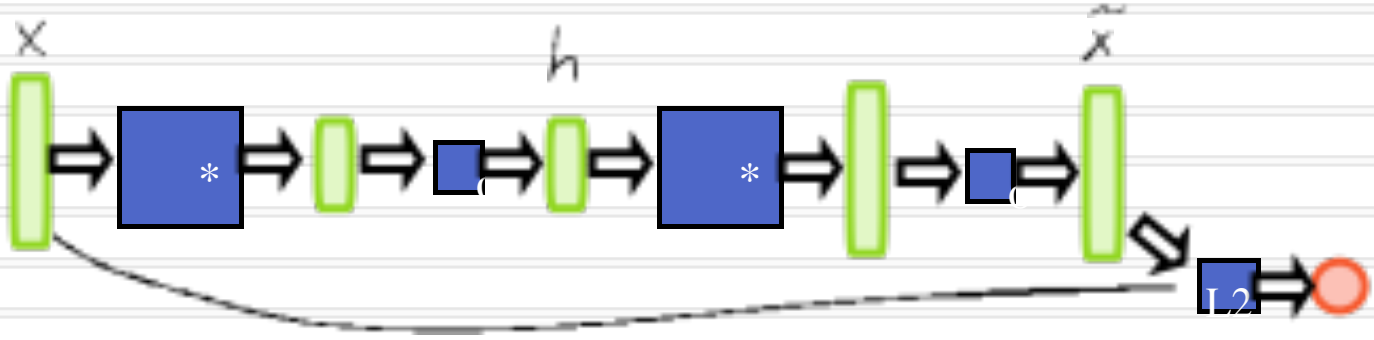
$\tilde{x} = W' h$

$\min_W \| x - \tilde{x} \|_2^2$

[Bengio et al. 07]

Autoencoder: adding non-linearity

encoder *decoder*

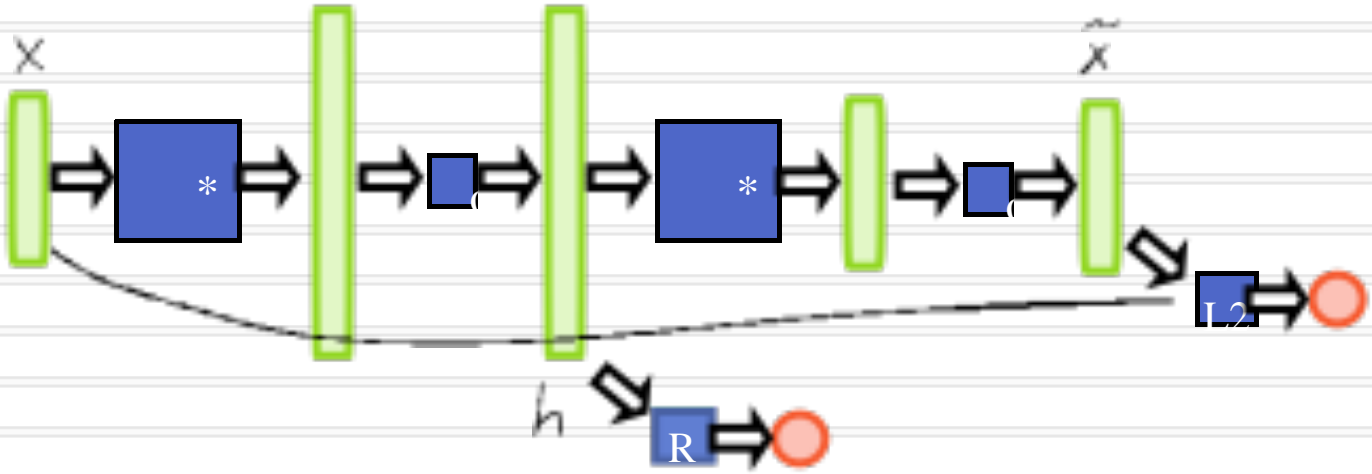


$$h = \sigma(Wx)$$

$$\tilde{x} = \sigma(W'h)$$

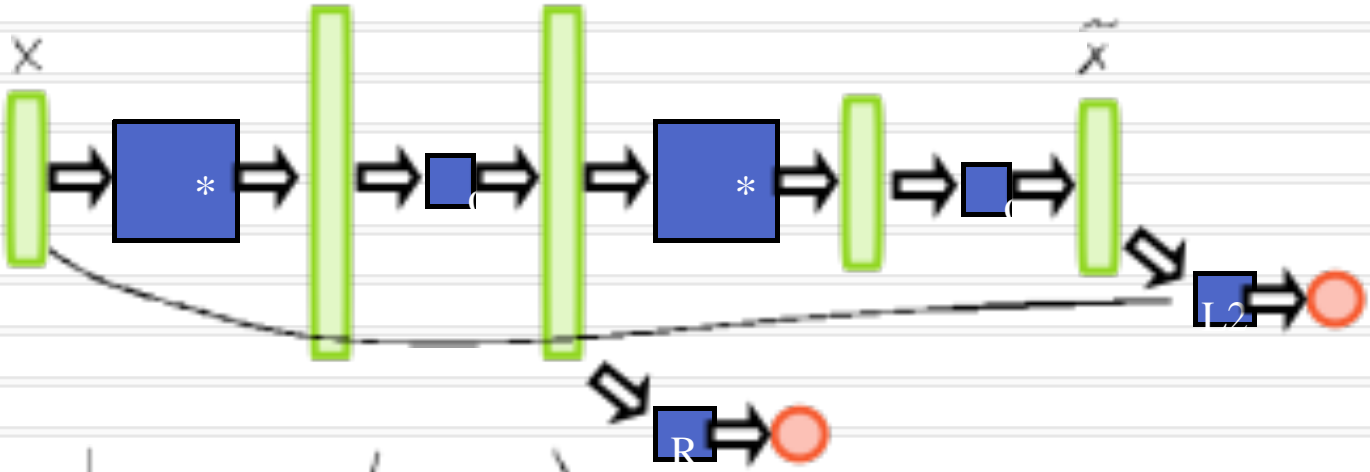
$$\min_{W,W'} \| x - \tilde{x} \|_2^2$$

Learning bigger/richer representation



- Naïve approach will learn an identity function
- Solution 1: regularize hidden representation

Sparse autoencoders



$$h = \sigma(Wx)$$

$$\tilde{x} = \sigma(W'h)$$

$$\min_{W, W'} \| x - \tilde{x} \|_2^2 + \lambda R(h)$$

$$\sum_{ij} |h_{ij}|$$

Sparse autoencoders

$$\min_{W,\,W'} \; \| x - \tilde{x} \|_2^2 + \lambda R(h)$$

estimating from e.g. minibatch

$$R(h) = \sum KL(\text{Bernoulli}(\rho) \,\|\, \text{Bernoulli}(\hat{\rho}_j))$$

$$= \sum \rho \log \frac{\rho}{\hat{\rho}_j} + (1-\rho) \log \frac{1-\rho}{1-\hat{\rho}_j}$$

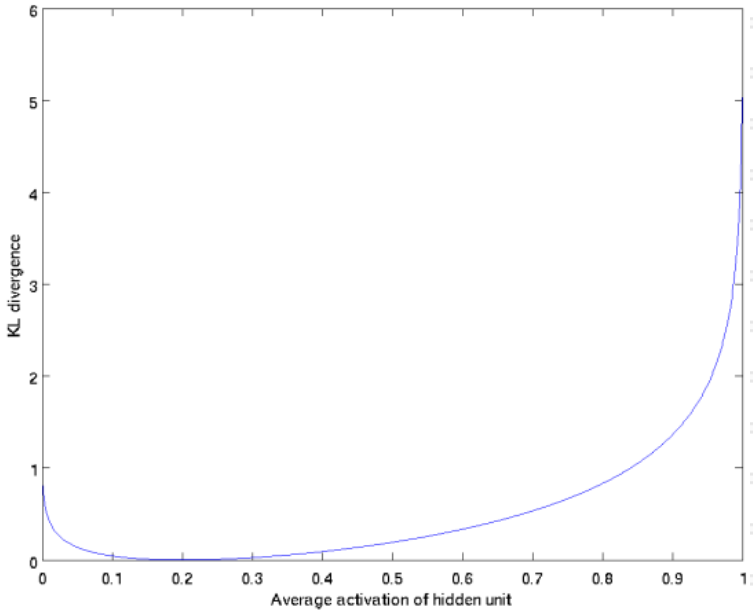
$$\frac{\partial R}{\partial \hat{\rho}_j} = \frac{\rho}{\hat{\rho}_j} - \frac{1-\rho}{1-\hat{\rho}_j}$$

[Ng 2011]

Sparse autoencoders

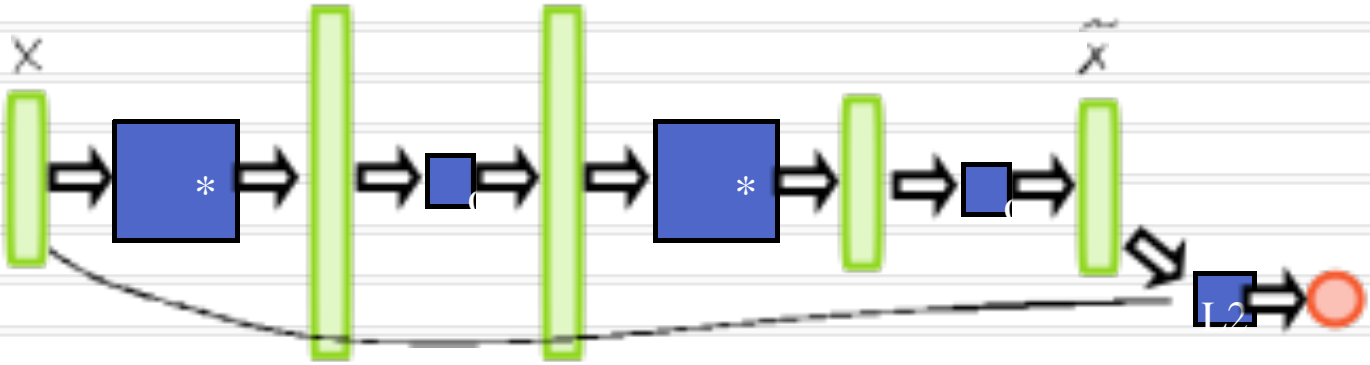
$$R(l) = \sum KL(\text{Bernoulli}(\rho) \,\|\, \text{Bernoulli}(\hat{\rho}_l))$$

$$\sum \rho \log \frac{\rho}{\hat{\rho}_l} + (1-\rho) \log \frac{1-\rho}{1-\hat{\rho}_l}$$



[Ng 2011]

Learning bigger representation

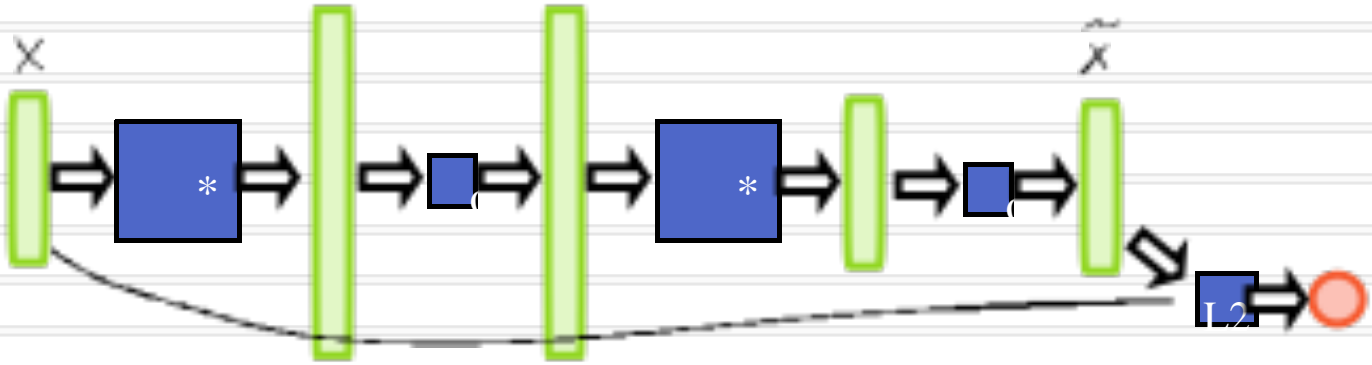


Naïve approach will learn an identity function

Solution 2: corrupt the input and reconstruct uncorrupted

[Vincent et al. 2010]

Denoising autoencoder



$$h = \sigma(W(x + n))$$

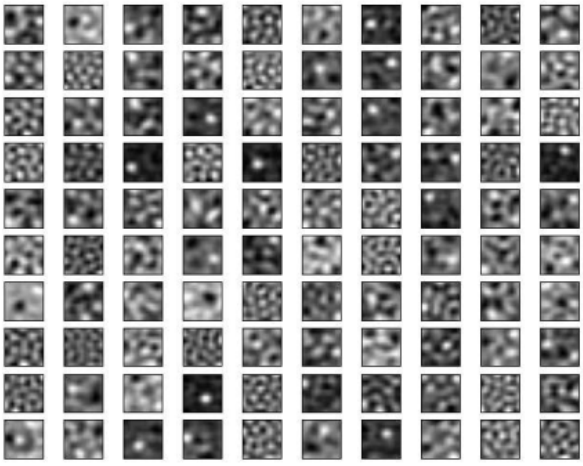
$$\tilde{x} = \sigma(W' h)$$

$$\min_{W, W'} \sum_n \| x_n - \tilde{x}_n \|_2^2$$

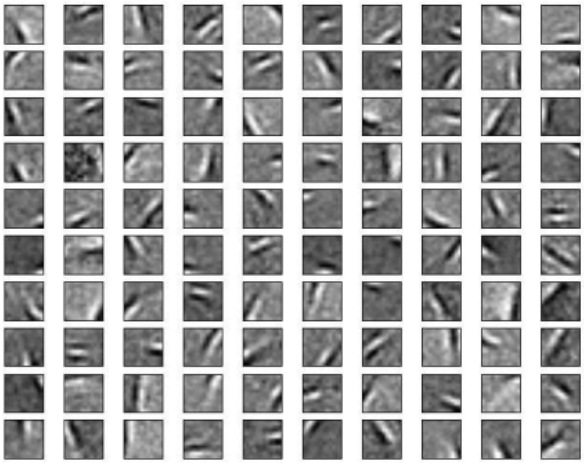
[Vincent et al. 2010]

Denoising autoencoder

12x12 patches, 200 hidden units

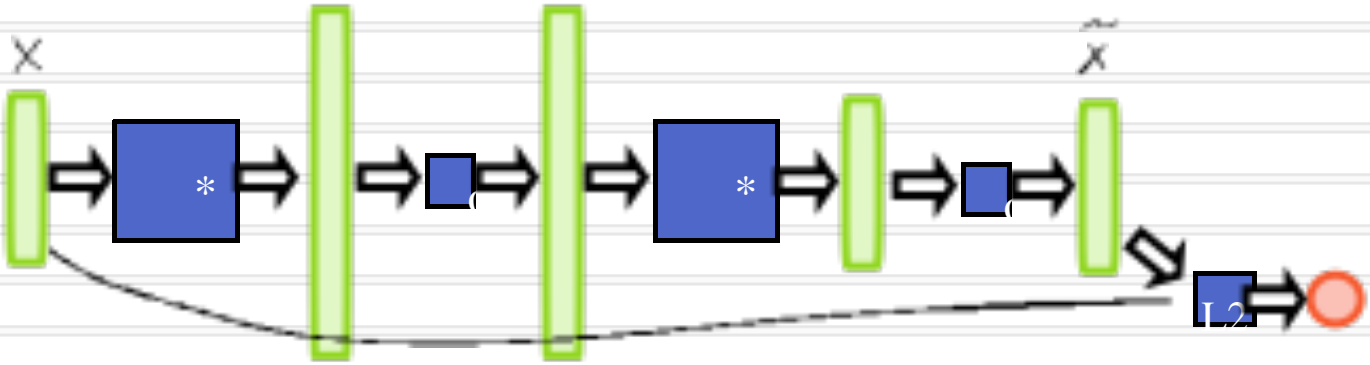


Autoencoder with weight-decay



Denoising autoencoder (Gaussian noise) [Vincent et al. 2010]

Contractive autoencoders



$$h = \sigma(Wx)$$

$$\tilde{x} = \sigma(W'h)$$

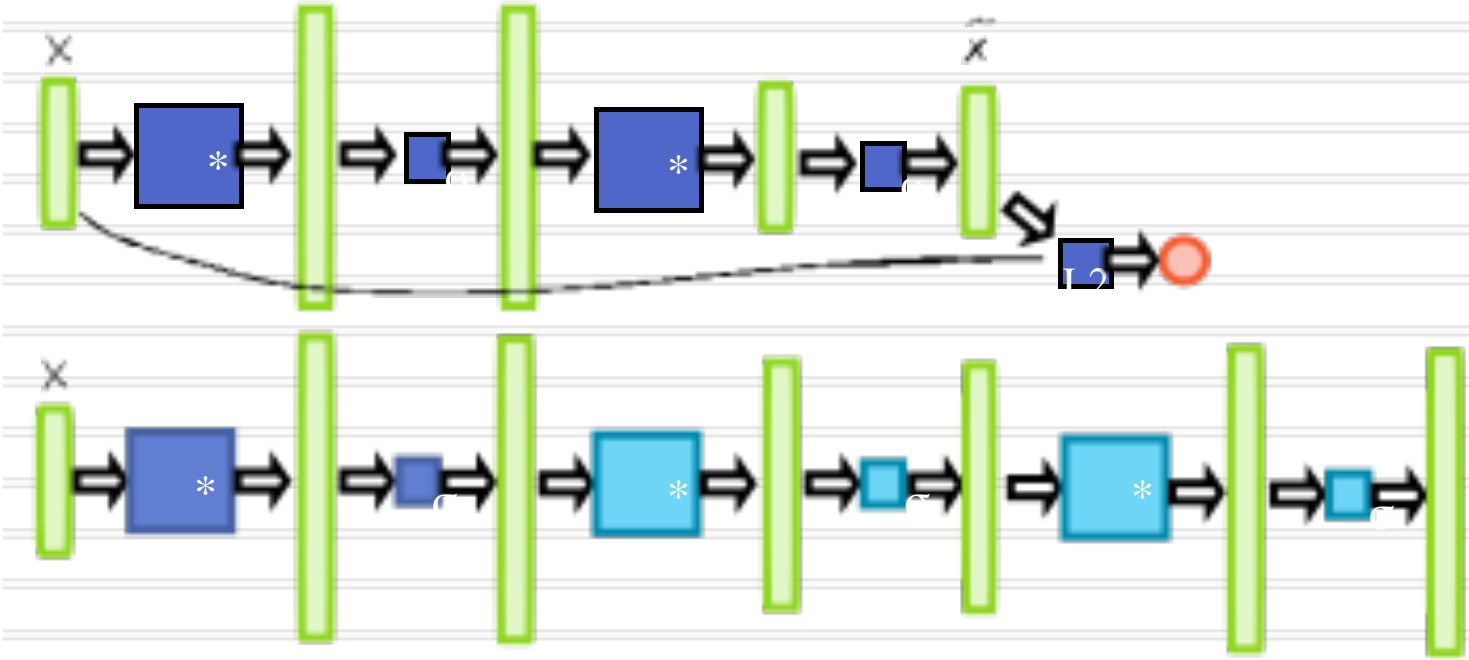
encoder

$$\min_{W, W'} \; \| x - \tilde{x} \|_2^2 + \lambda \left\| \frac{\partial h(x)}{\partial x_j} \right\|_1^2$$

$$\frac{\partial h_j}{\partial x_j} \quad (h_j(1 - h_j)) \; W_j$$

[Rifai et al. 2011]

Training deep autoencoders



Historically: layerwise pretraining

Most modern implementations: end-to-end

Commonly including convolutional and up-convolutional layers

Maximum likelihood

$p_\theta(x \mid z)$

Very natural idea: train θ by maximum likelihood:

$$\log p_\theta(x) - \log \int_z p_\theta(x, z)\, dz \to \max$$

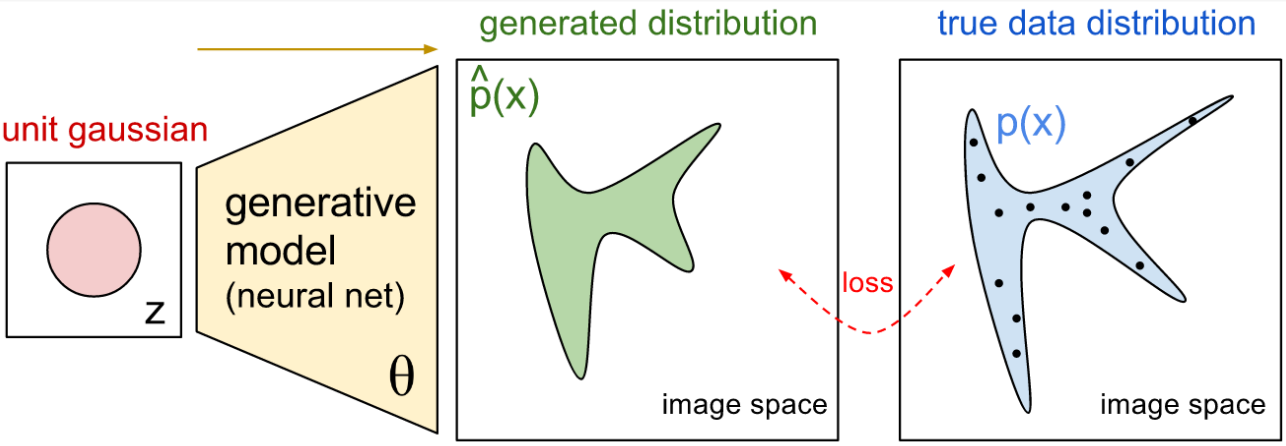


Image credit: OpenAI blog

Maximum likelihood via VAE

$p_\theta(x \mid z)$

Natural idea: train θ by maximum likelihood.

$$\log p_\theta(x) = \log \int_z p_\theta(x, z)\, dz \to \max$$

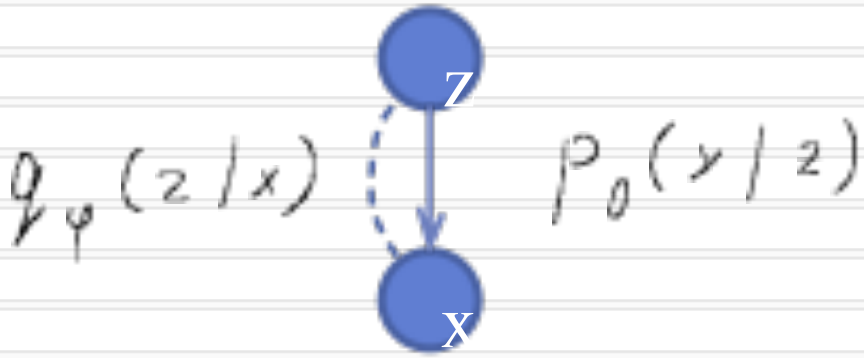
The integral is intractable.

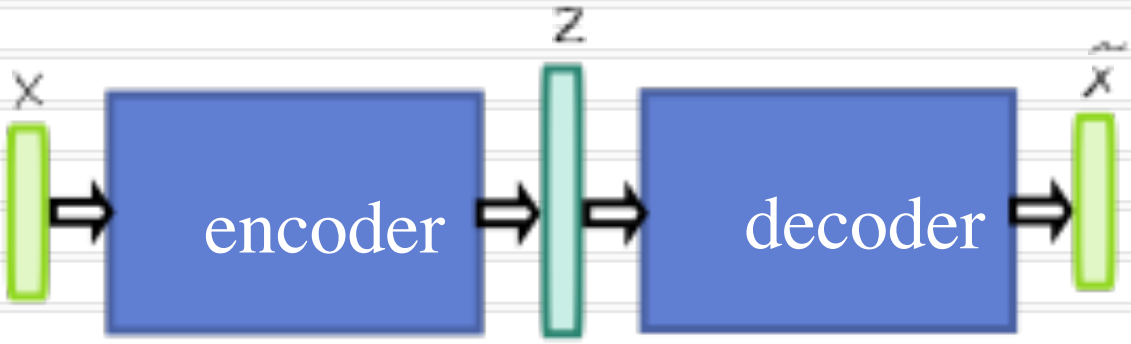
VAE solution: introduce an *encoder*

$$q_\varphi(z \mid x) \approx p(z \mid x)$$

Derive an approximation to ML objective

Variational autoencoder as a graphical model





[Kingma & Welling 14]

Variational lower bound

$$\log p(x) - \log \int_z p(x,z)\,dz =$$

$$-\log \int_z p(x,z)\frac{q(z|x)}{q(z|x)}\,dz -$$

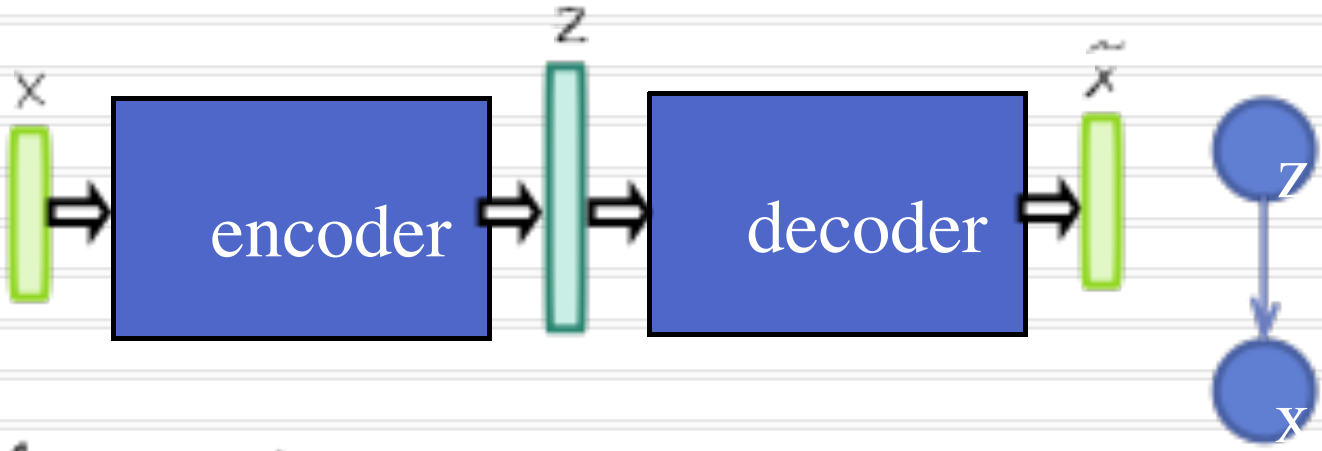
$$= -\log E_{q(z|x)} \frac{p(x,z)}{q(z|x)} -$$

$$\log E_{q(z|x)} \frac{p(x|z)\,p(z)}{q(z|x)} \geq$$

$$\geq E_{q(z|x)} \log p(x|z) \quad E_{q(z|x)} \log \frac{p(z)}{q(z|x)} =$$

$$- E_{q(z|x)} \log p(x|z) - KL(q(z|x)\,||\,p(z))$$

Variational lower-bound



$$\log p(x) \geq$$

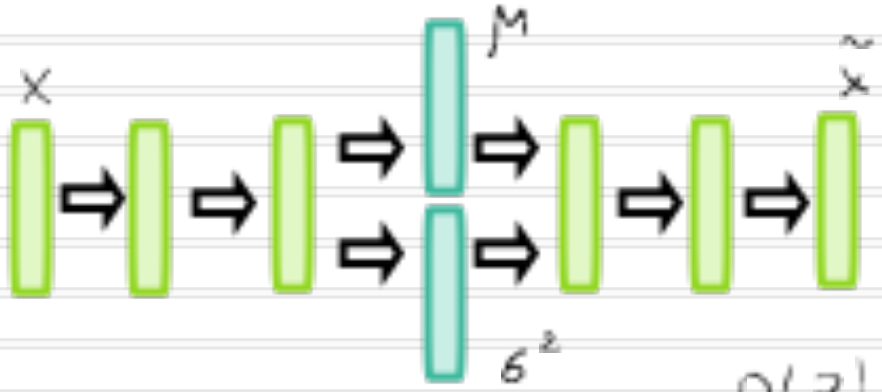
$$KL(q_\phi(z|x) \| p(z)) + E_{q_\phi(z|x)}[\log p_\theta(x|z)]$$

regularization

~"denoising auto-encoder"

[Kingma & Welling 14]

Minimizing regularization term



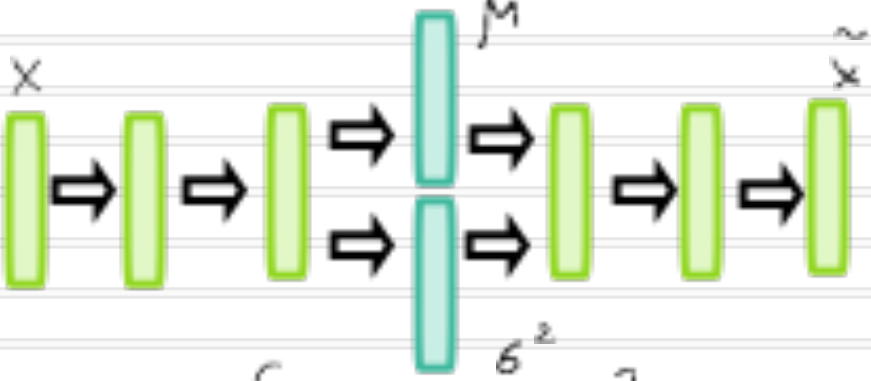
$$p(z) = \mathcal{N}(0, 1)$$

$$KL(q_\phi(z|x) \| p(z)) =$$

$$= \frac{1}{2} \sum (\mu_j^2 - \sigma_j^2 - 1 - \log \sigma_j^2)$$

[Kingma & Welling 14]

Optimizing reconstruction term



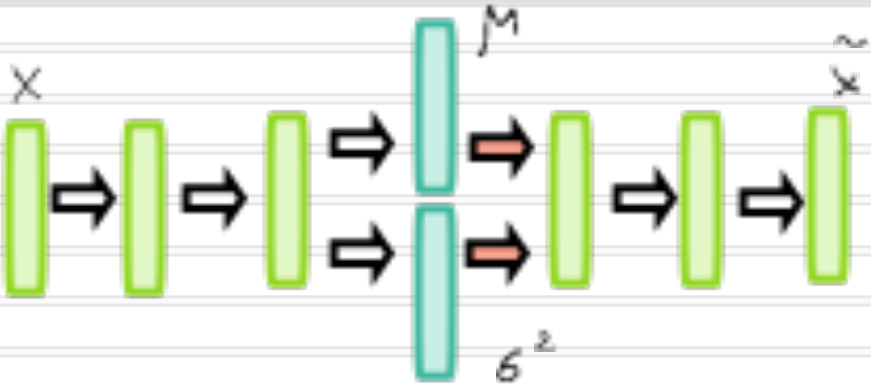
$$E_{q_\phi(z|x)}[\log p_\theta(x|z)] \to \max$$

$$p_\theta(x|z) \propto \exp(-\|x - \tilde{x}\|^2)$$

$$E_{q_\phi(z|x)}\|d_\theta(z) - x\|_2^2 \to \min$$

[Kingma & Welling 14]

Reparameterization trick



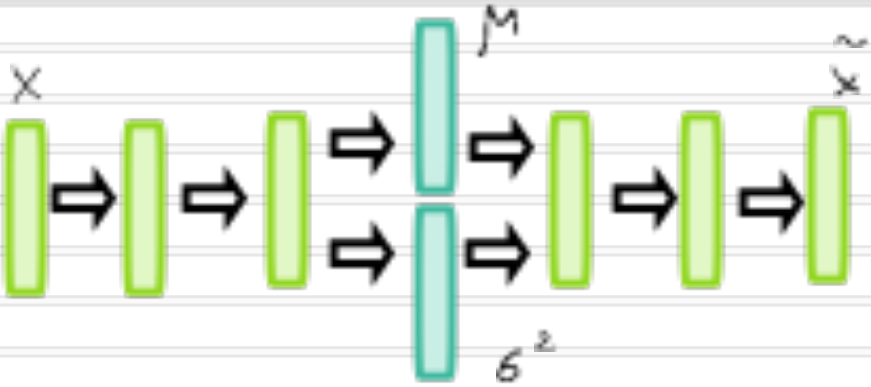
$$E_{q_\phi(z|x)} \| d_\theta(z) - x \|_2^2 \to \min$$

$$\tilde{z} = \mu + \sigma^2 \odot \varepsilon \sim \mathcal{N}(0, I)^N$$

$$L = \| d_\theta(\tilde{z}) - x \|_2^2 \to \min$$

[Kingma & Welling 14]

Reparameterization trick



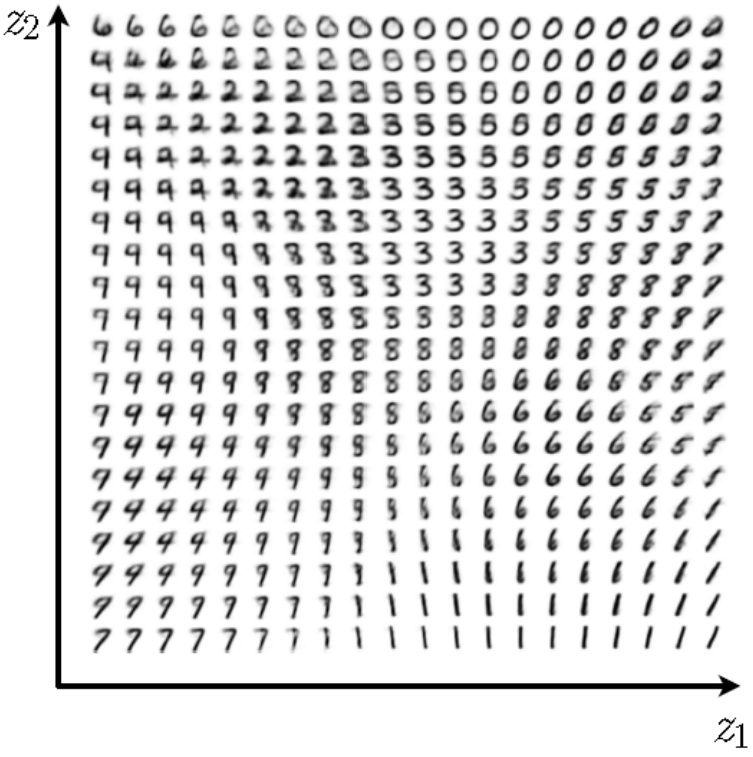
$$\tilde{z} = \mu + \sigma^2 \odot \varepsilon \sim \mathcal{N}(0, .)^N$$

$$\frac{dL}{d\mu} = \frac{dL}{d\tilde{z}} \qquad \frac{dL}{d\sigma^2} = \frac{dL}{d\tilde{z}} \odot \varepsilon$$

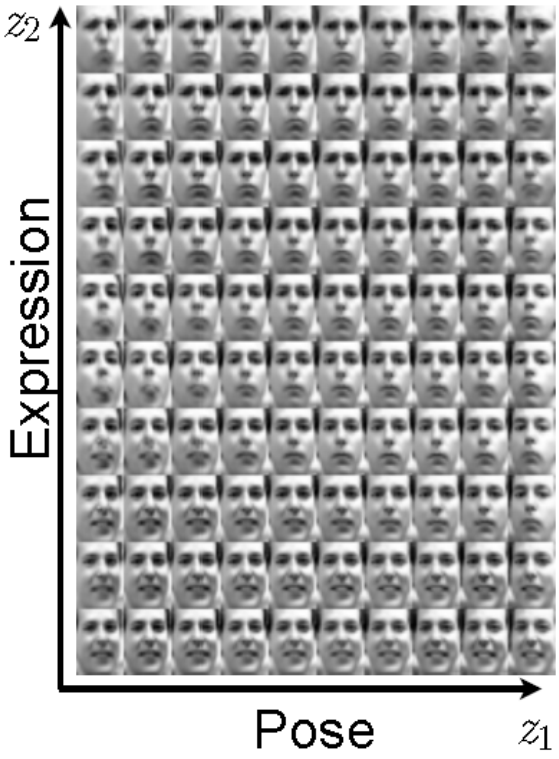
[Kingma & Welling 14]

VAE-learned manifolds

MNIST:



Frey Face dataset:



[Kingma & Welling 14]

VAE-learned manifolds

(a) 2-D latent space

(b) 5-D latent space

(c) 10-D latent space

(d) 20-D latent space

[Kingma & Welling 14]

Collapsing components

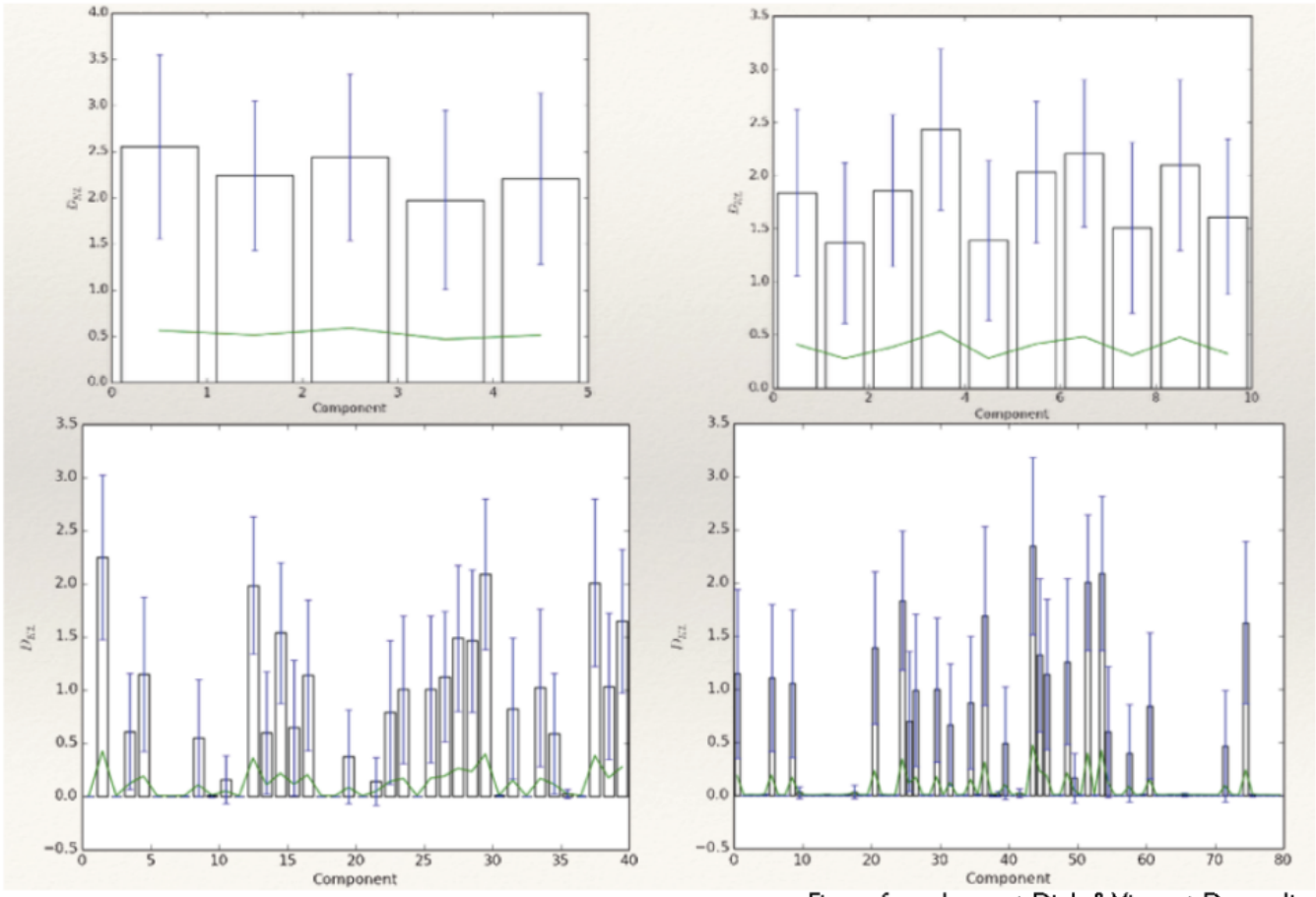


Figure from Laurent Dinh & Vincent Dumoulin

Collapsing components

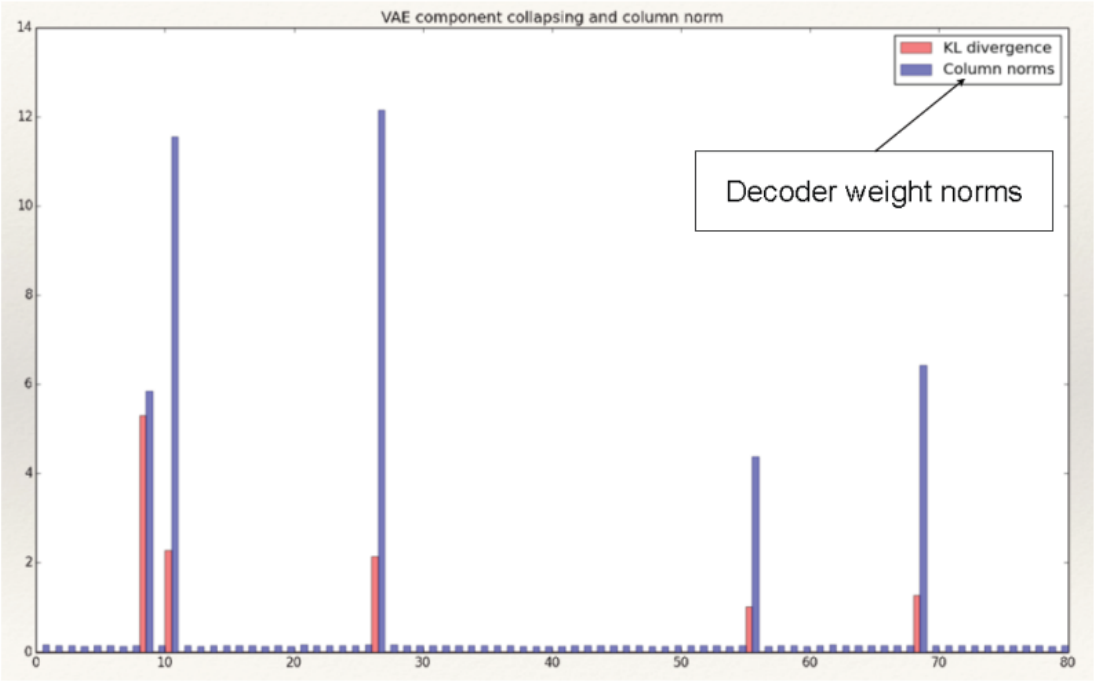


Figure from Laurent Dinh & Vincent Dumoulin

- Components that are shrunk to *N(0,1)* are not used by decoder

Sample quality comparison

VAE

GAN

Source: OpenAI blog