Tutorial on Variational Autoencoders

David Nagy @dvgnagy
David Szepesvari

EEML 2021, Budapest

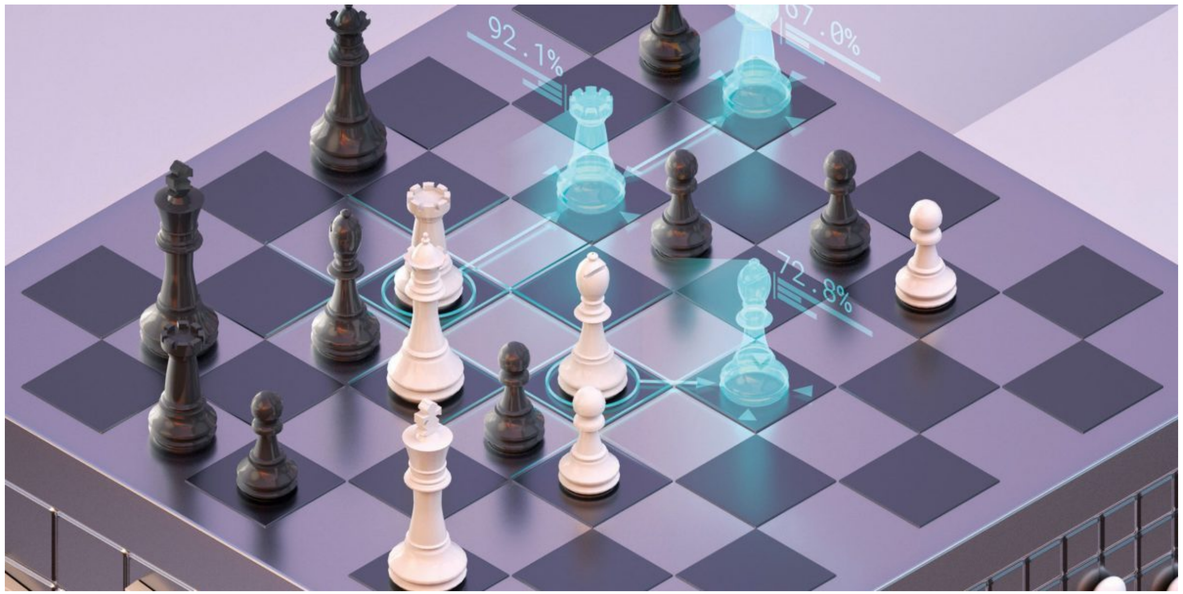
Why are VAEs interesting?

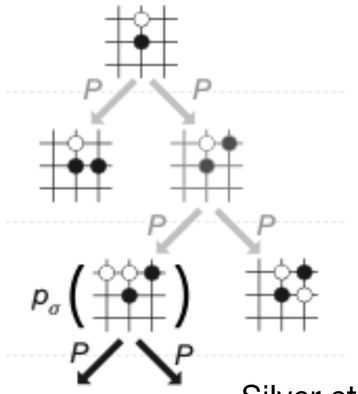
- a single model that can both
    - work as a generative model
    - infer latent variables
- both
    - theoretical interpretation in probabilistic modeling framework
    - but also works on naturalistic data
- a building block in many larger models

generative models

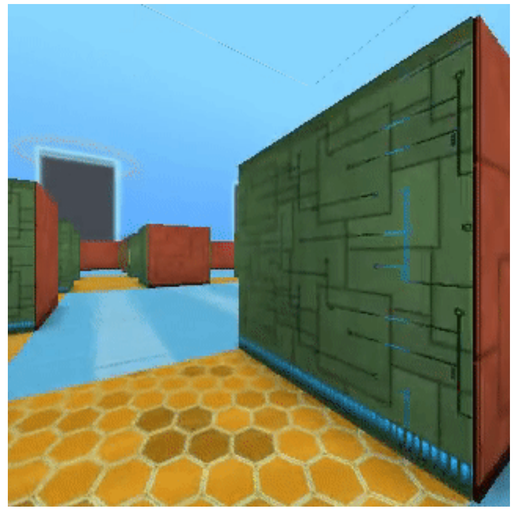
Karras et al. 2017

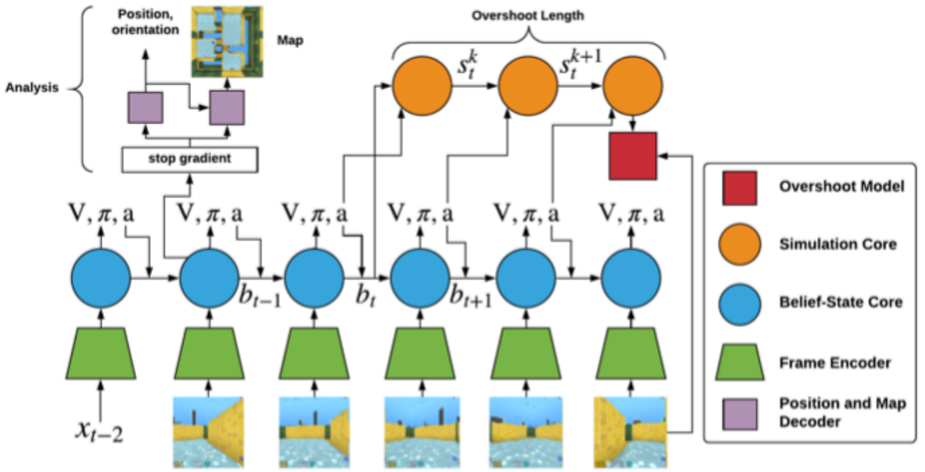
generative models





Silver et al 2016

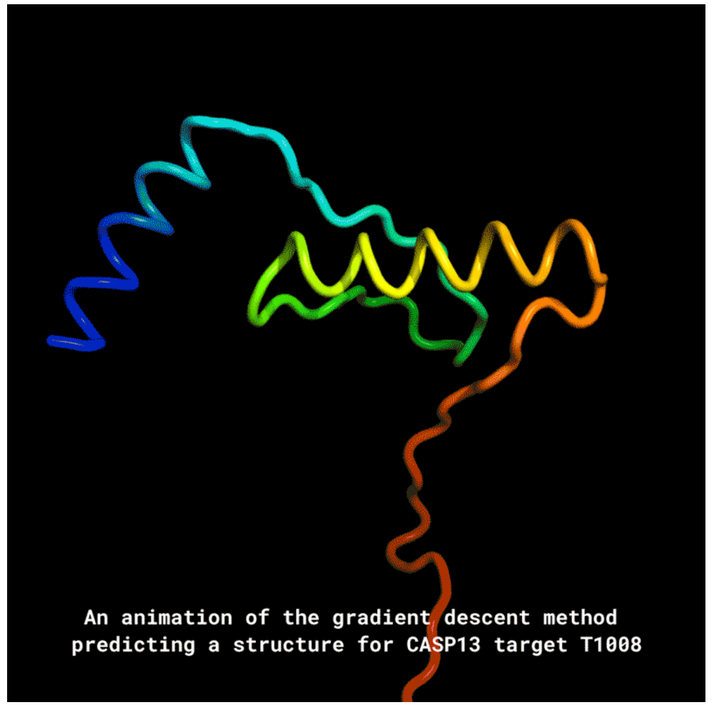


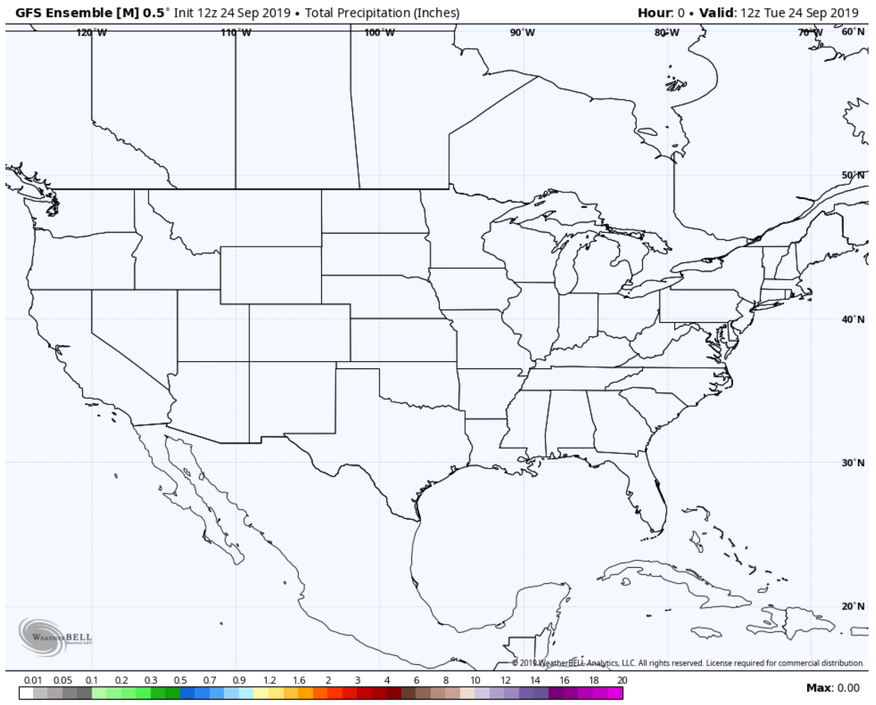


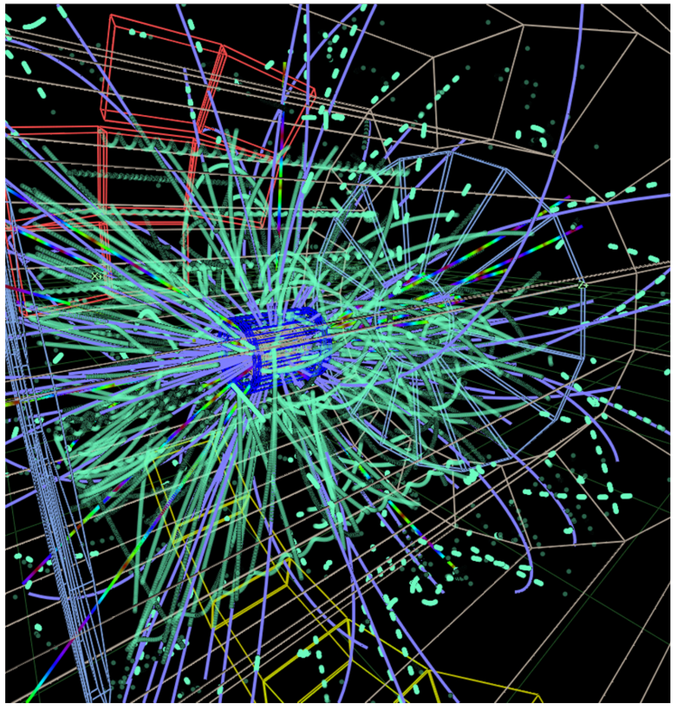
Besse et al 2019

generative models

generative models







generative models

latent state

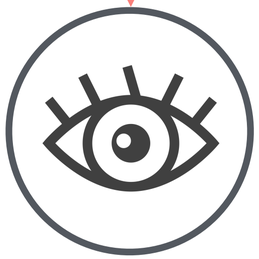
z

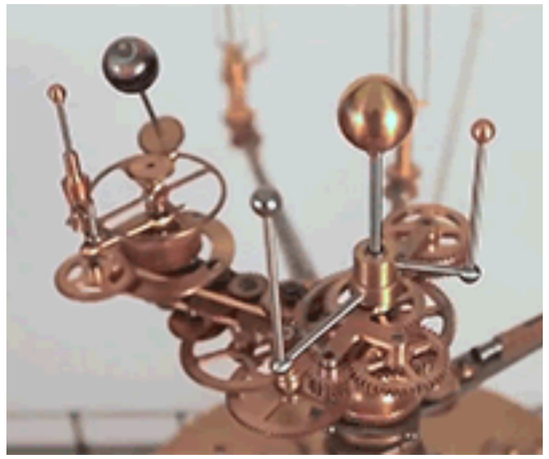
generative model

$p(x \mid z)$

observation

x

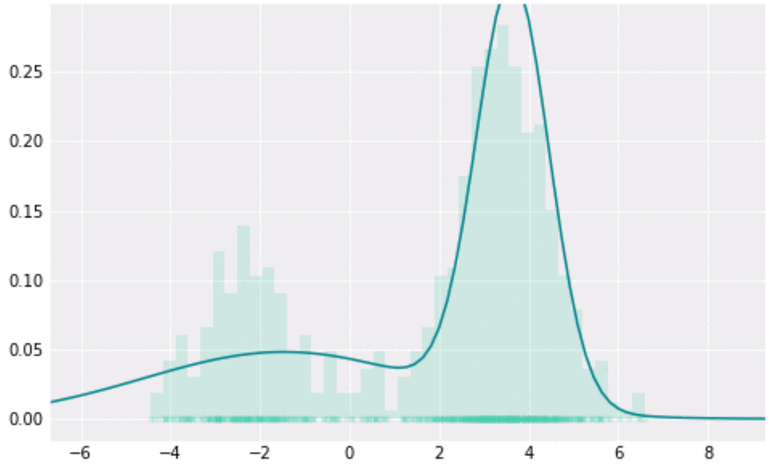


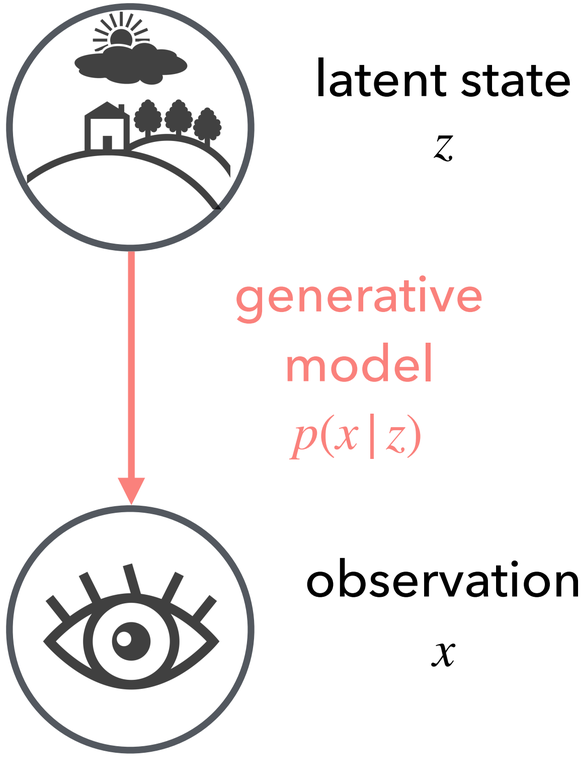


how do we learn a generative model?

- match empirical distribution to predictive distribution

$$\arg\min_{model} \; KL\left[\, p_{data}(x) \,||\, p_{model}(x) \,\right]$$

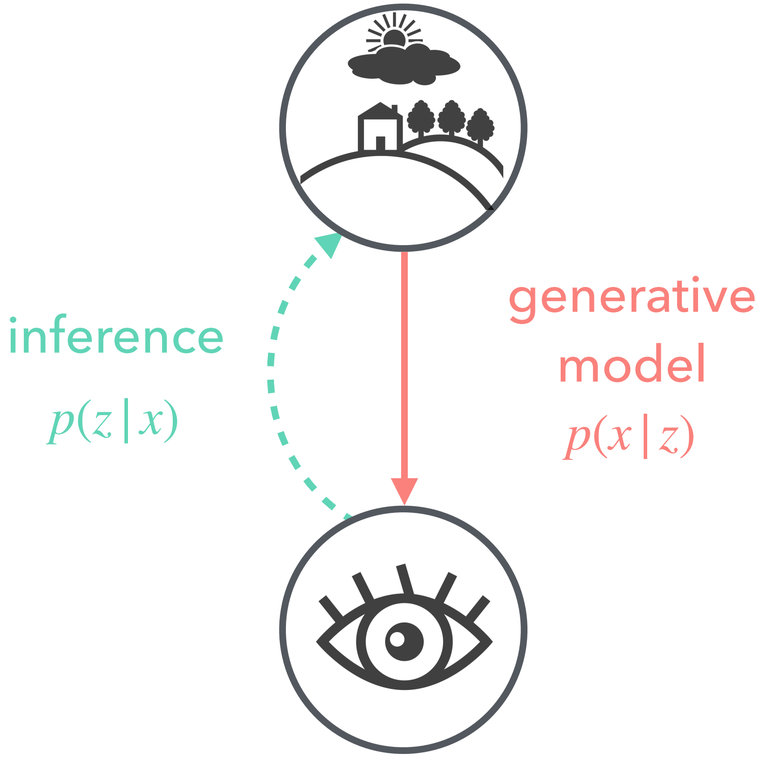




- equivalent to maximum likelihood

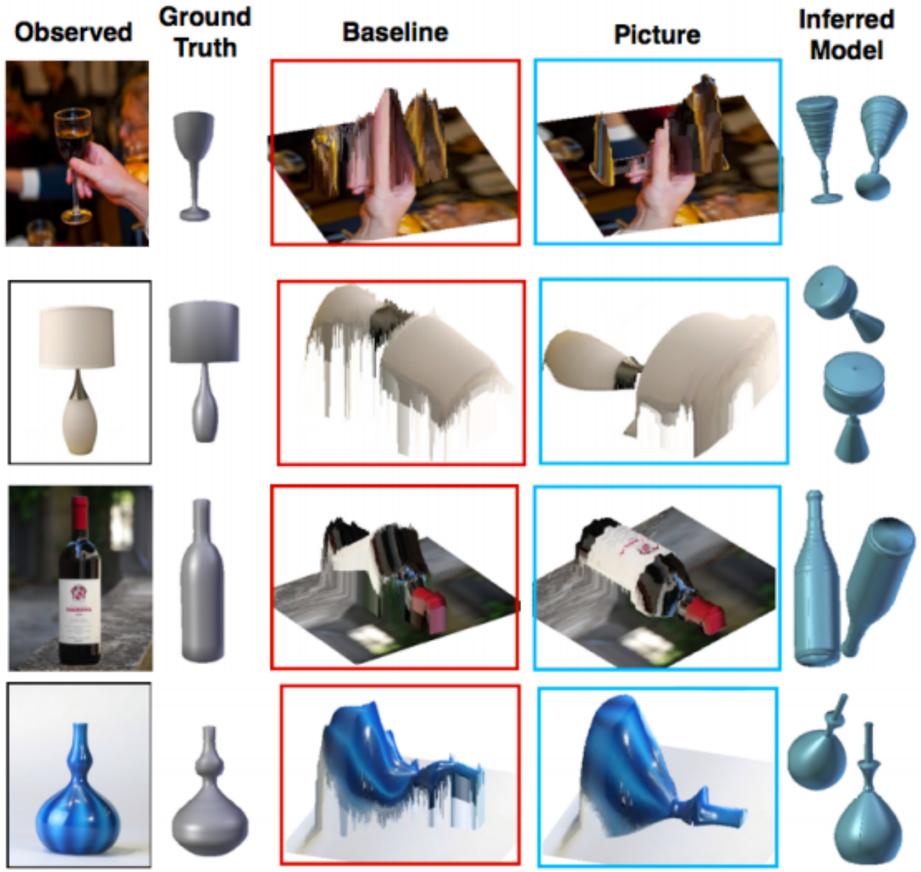
$$\arg\max_{\theta} \log p_{\theta}(x)$$

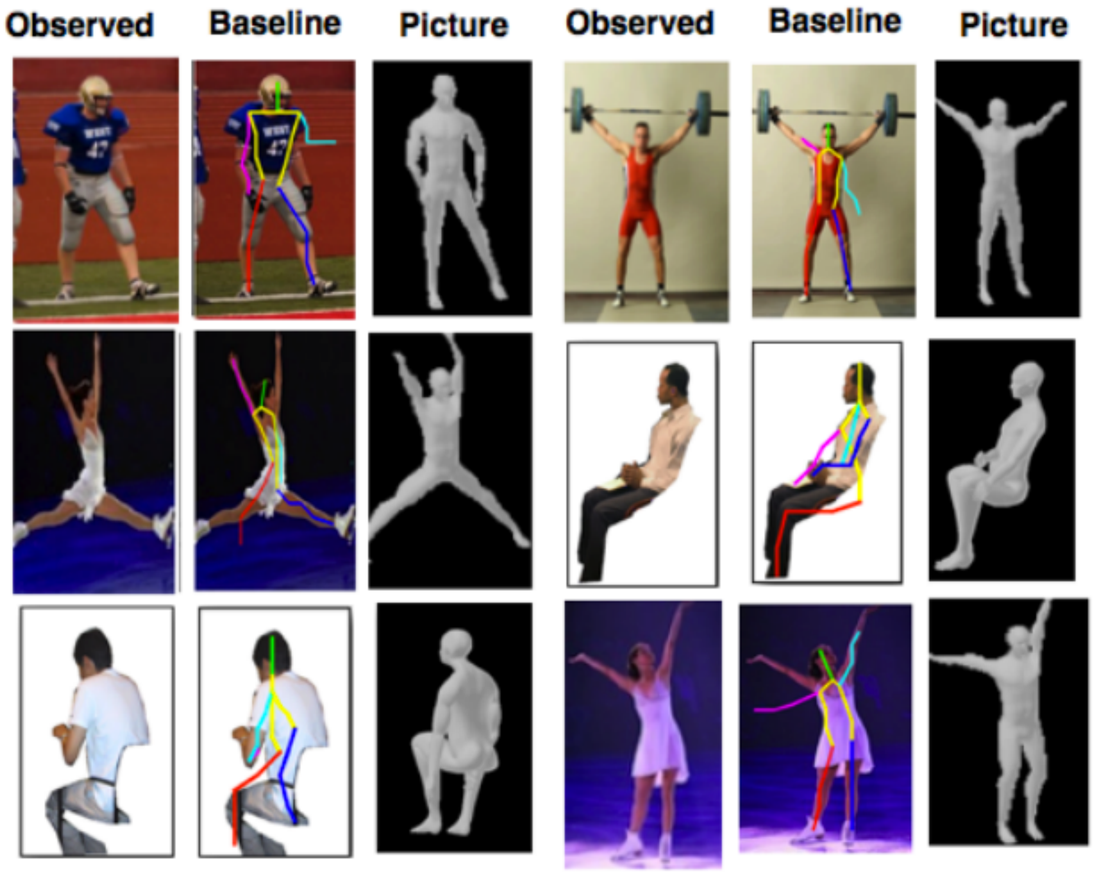
inference



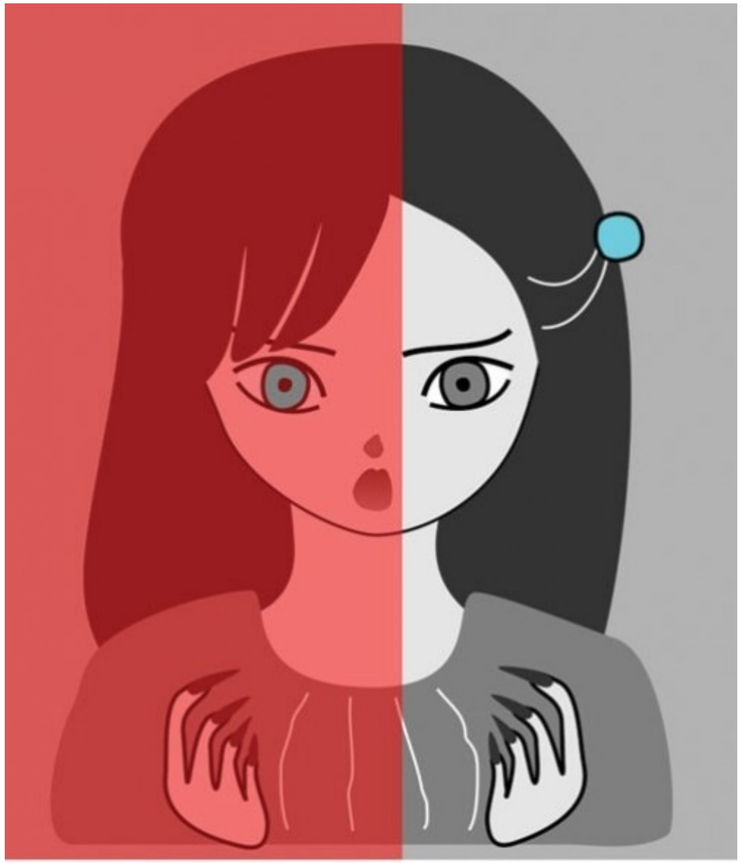
inference

inference

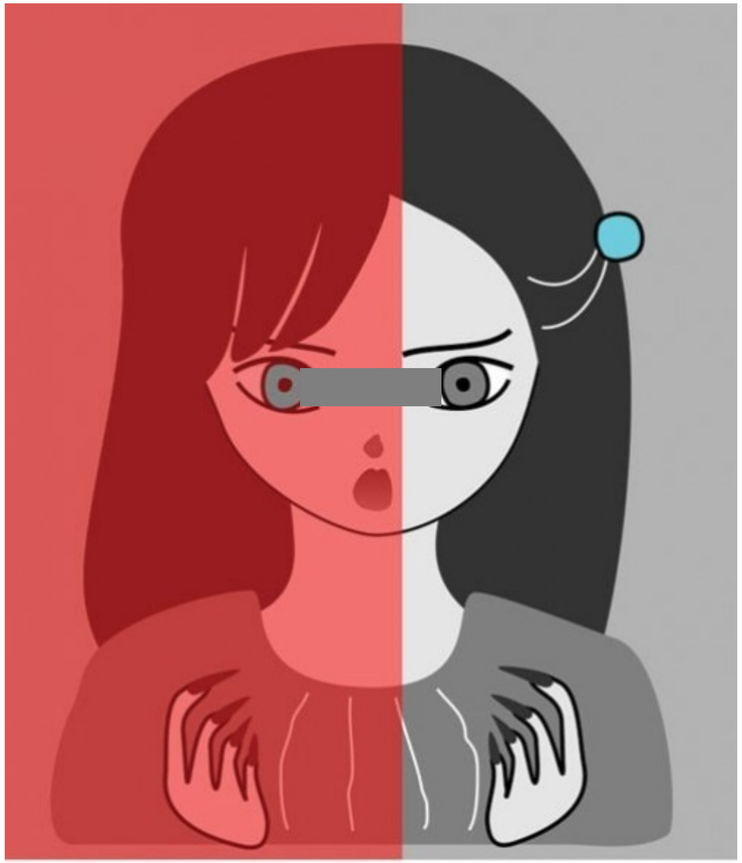

Observed
Ground Truth
Baseline
Picture
Inferred Model


Observed
Baseline
Picture
Observed
Baseline
Picture

inference

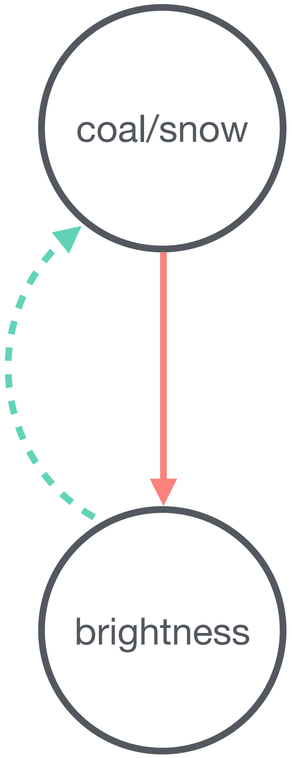


inference



- what we see is not the data but an interpretation of the data
- ‘unconscious inference’ over latent variables

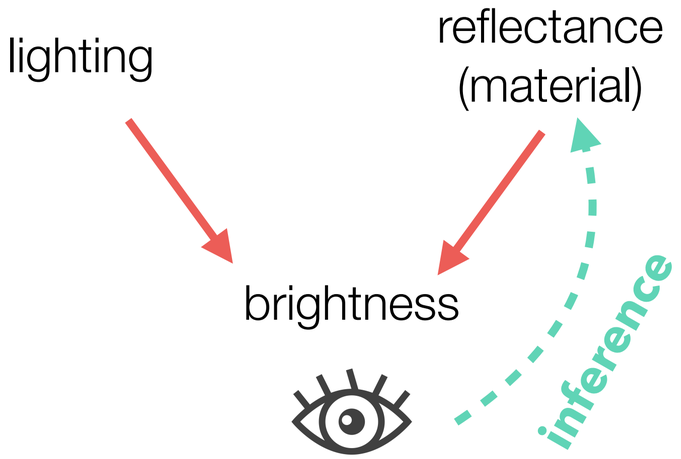
how do we do inference?



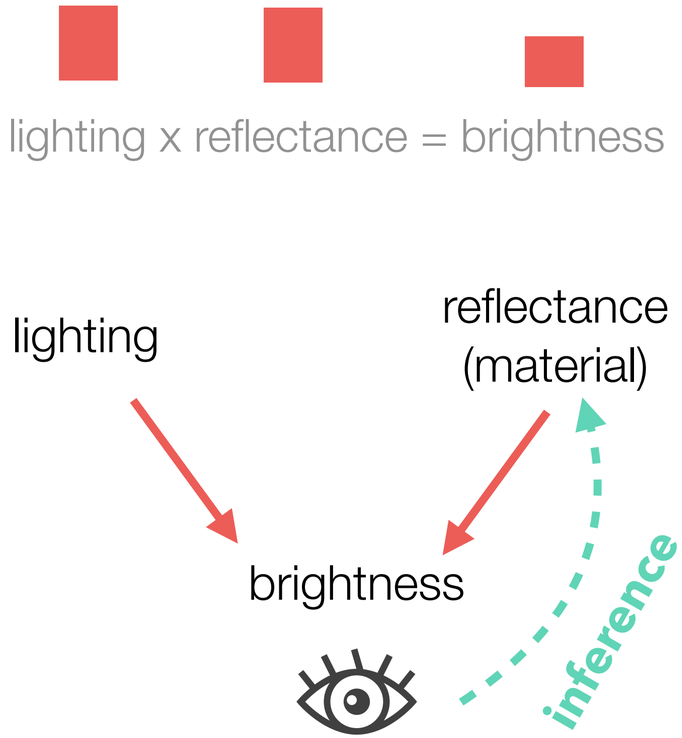
how do we do inference?

how do we do inference?

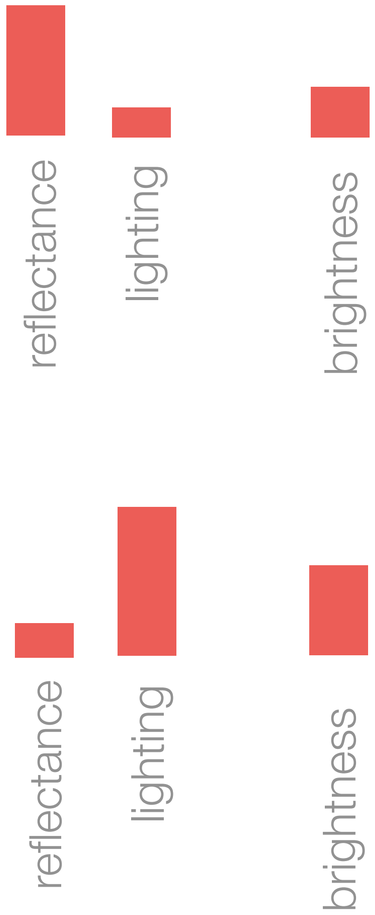
lighting x reflectance = brightness

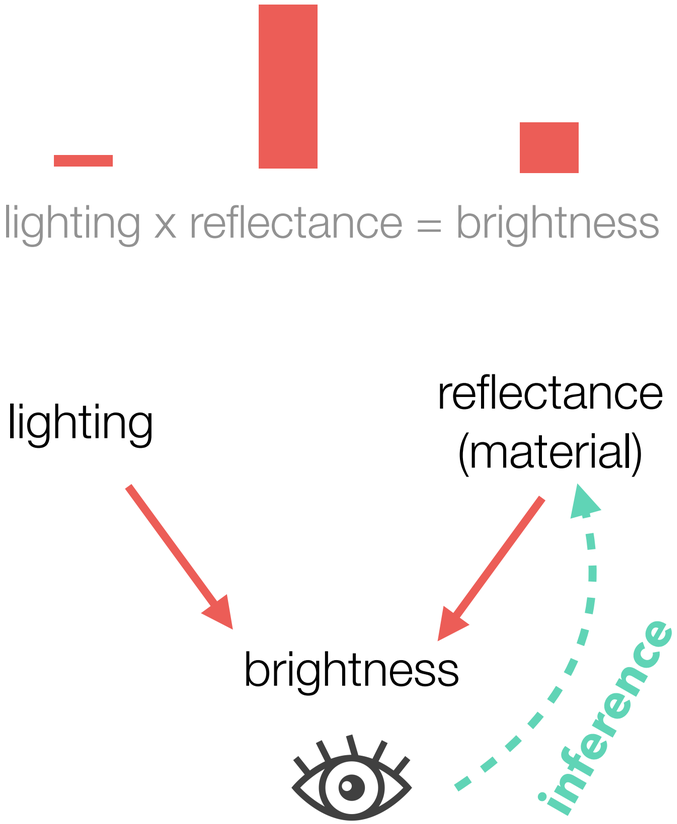


how do we do inference?

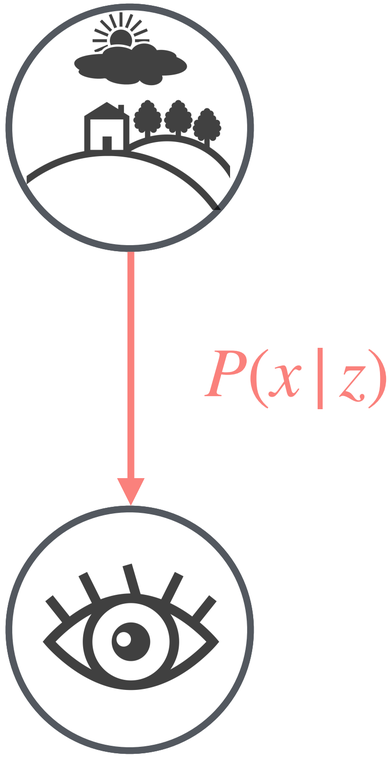


how do we do inference?

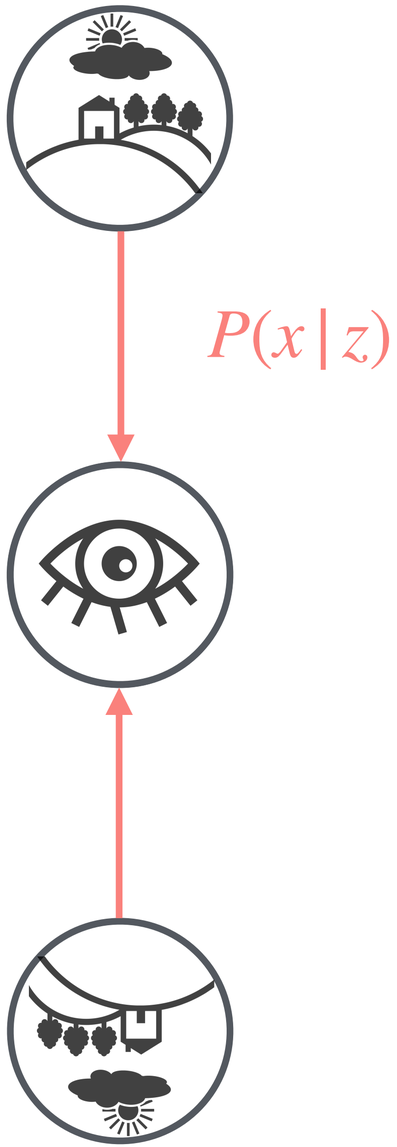




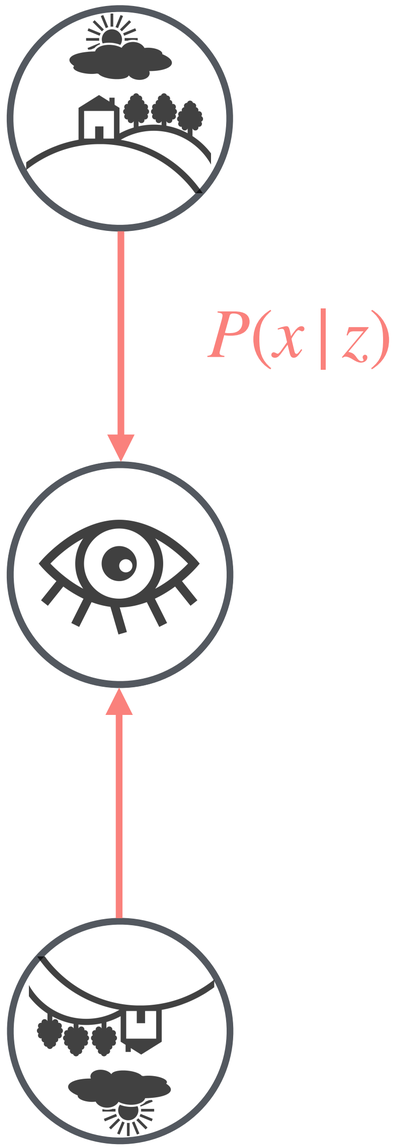
how do we do inference?



how do we do inference?

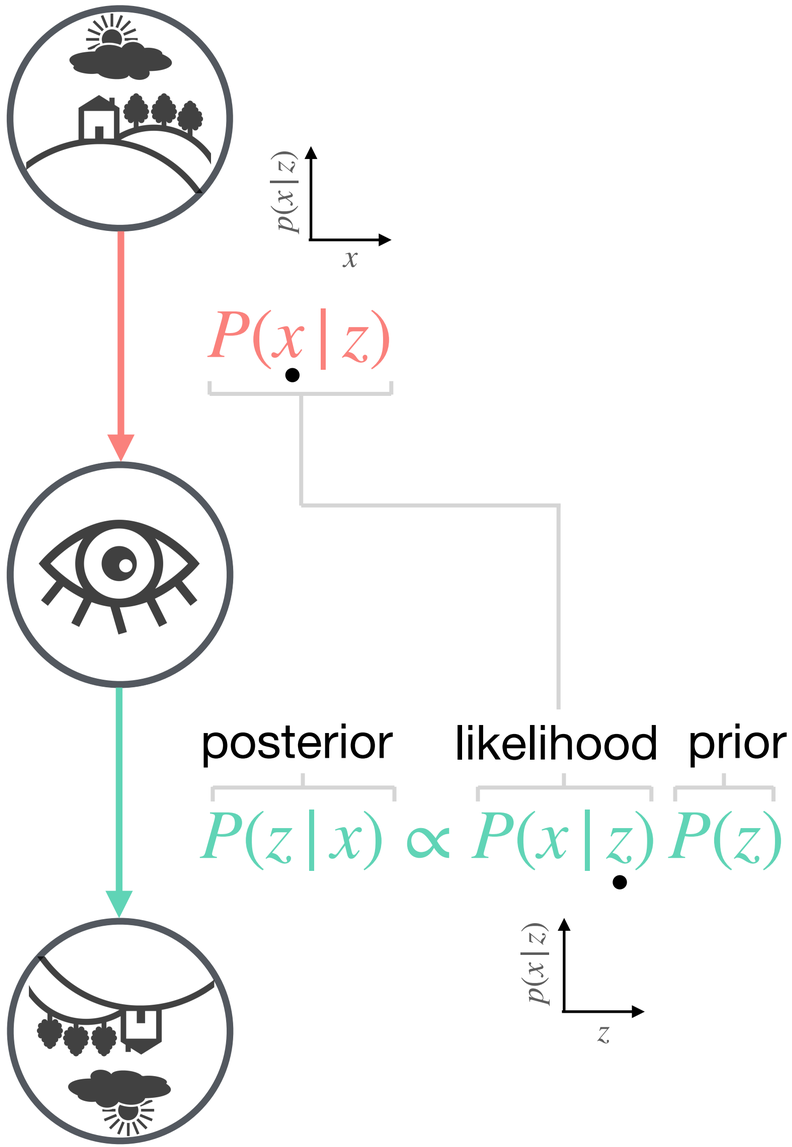


how do we do inference?



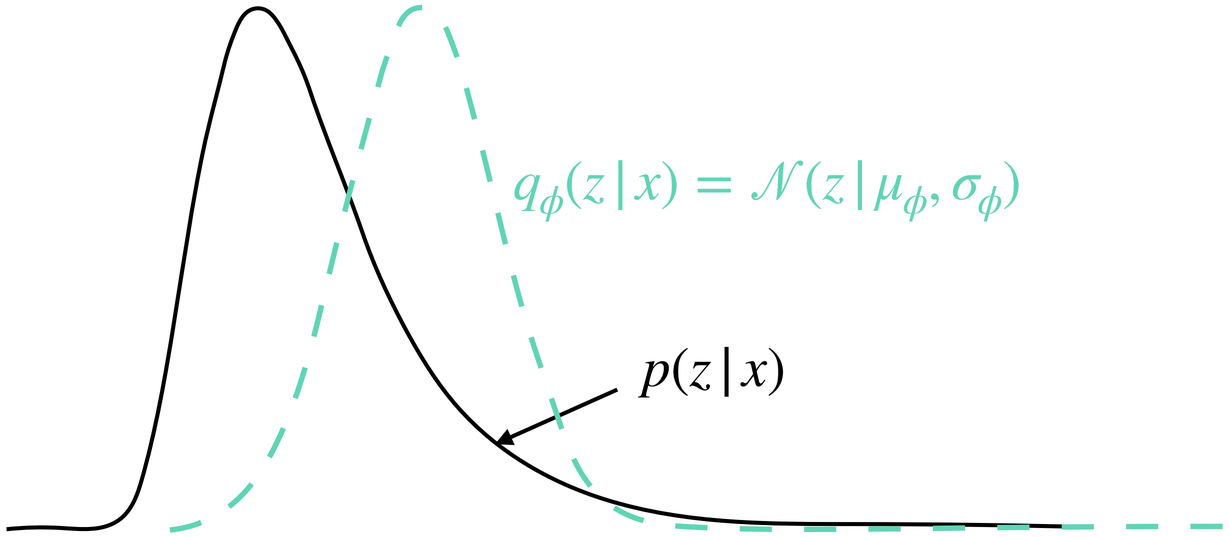
how do we do inference?

- inference can be performed by inverting the generative model
- “vision is inverse graphics”
- Bayesian inference



how do we do inference?

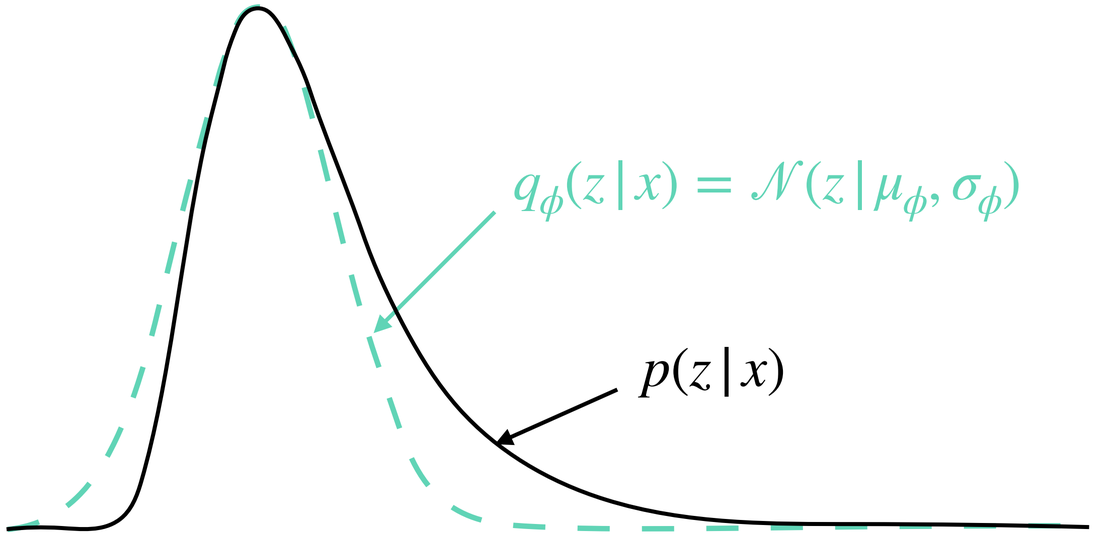
approximation 1: variational bayes



$$\arg\min_{\phi} KL[q_\phi(z\,|\,x)\,||\,p(z\,|\,x)]$$

how do we do inference?

approximation 1: variational bayes



Ferenc notation (inference.vc)

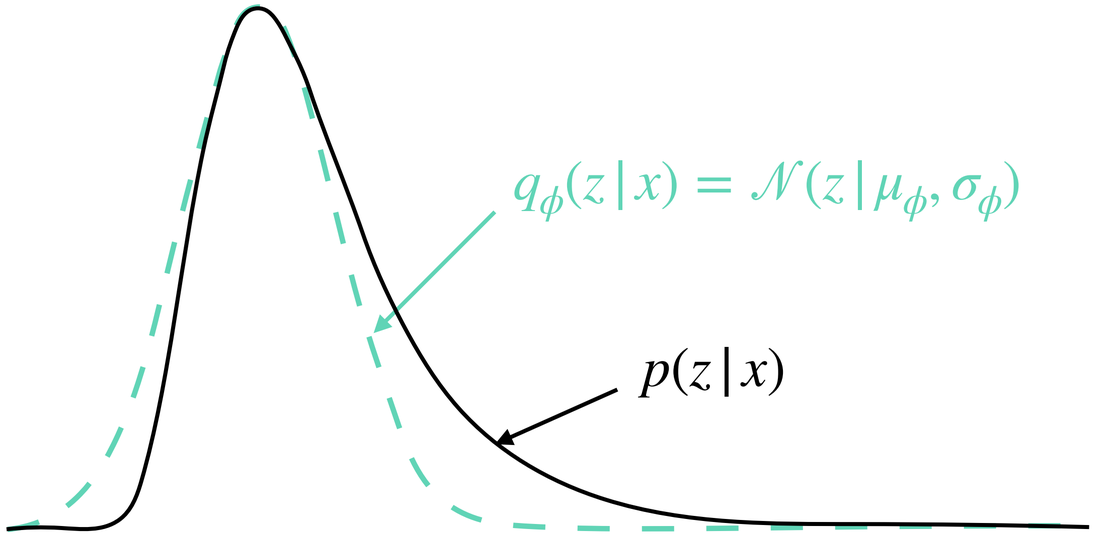
- $p_\theta(z)$ is very easy 🐣,
- $p_\theta(x|z)$ is easy 🐹,
- $p_\theta(x, z)$ is easy 🐨,
- $p_\theta(x)$ is super-hard 🐍,
- $p_\theta(z|x)$ is mega-hard 🐲

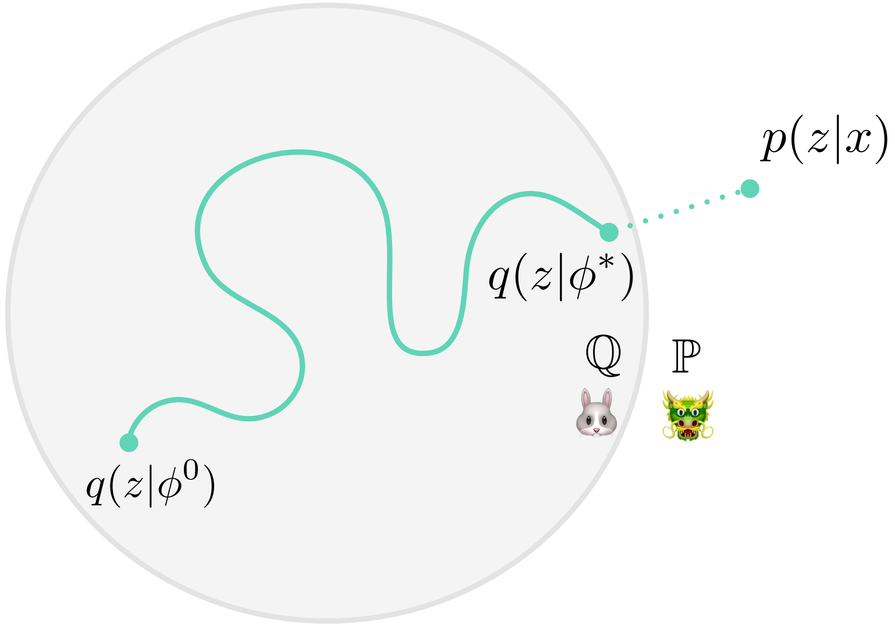
$$\arg\min_\phi KL[q_\phi(z|x)||p(z|x)]$$

$$\mathrm{KL}[🐰|🐲]$$

how do we do inference?

approximation 1: variational bayes





$$\arg\min_{\phi} KL[q_\phi(z\,|\,x)\,||\,p(z\,|\,x)]$$

how do we do inference?

approximation 2: amortised inference

$$
\begin{array}{ccc}
 & f & \\
x_1 & \mapsto & q(z \mid x_1) \\
x_2 & \mapsto & q(z \mid x_2) \\
 & \vdots & \\
x_N & \mapsto & q(z \mid x_N)
\end{array}
$$

- f mapping between datapoint and posterior
- if we have x,q pairs for a set of datapoints,
- we can treat learning f as a supervised regression problem

how do we do inference?

approximation 2: amortised inference

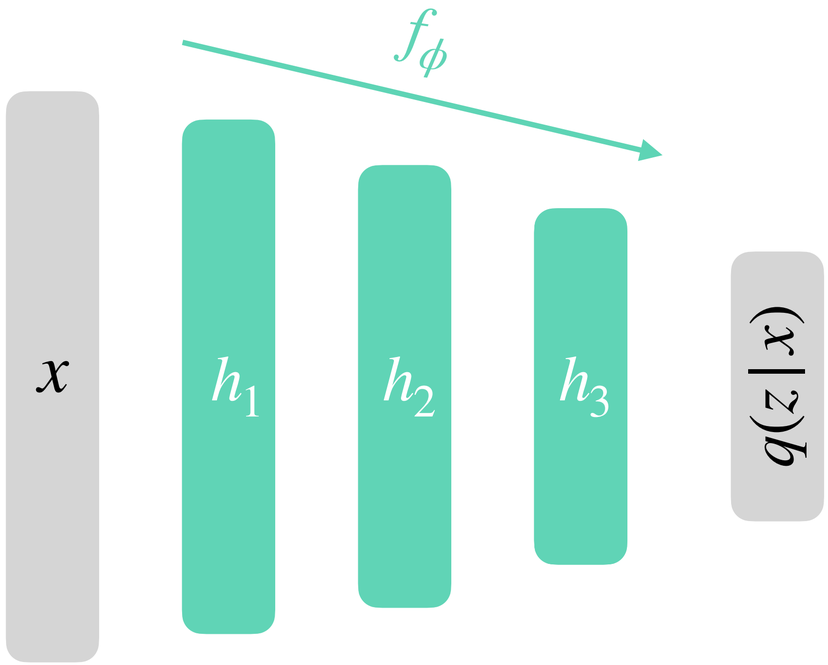
$$f \approx f_\phi$$

$$x_1 \mapsto q(z \mid x_1)$$

$$x_2 \mapsto q(z \mid x_2)$$

$$\vdots$$

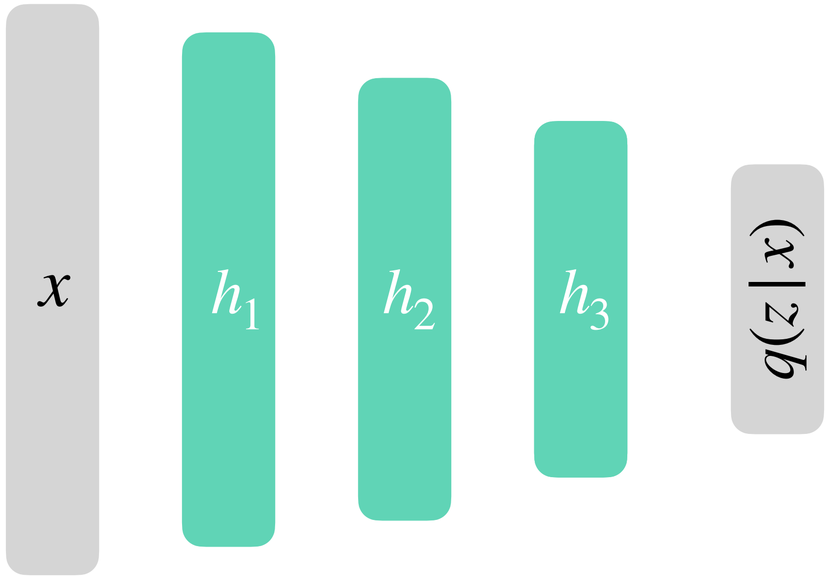
$$x_N \mapsto q(z \mid x_N)$$



- treat learning f as a supervised regression problem
- approximate the mapping with a simpler function (NN)
  - inference network

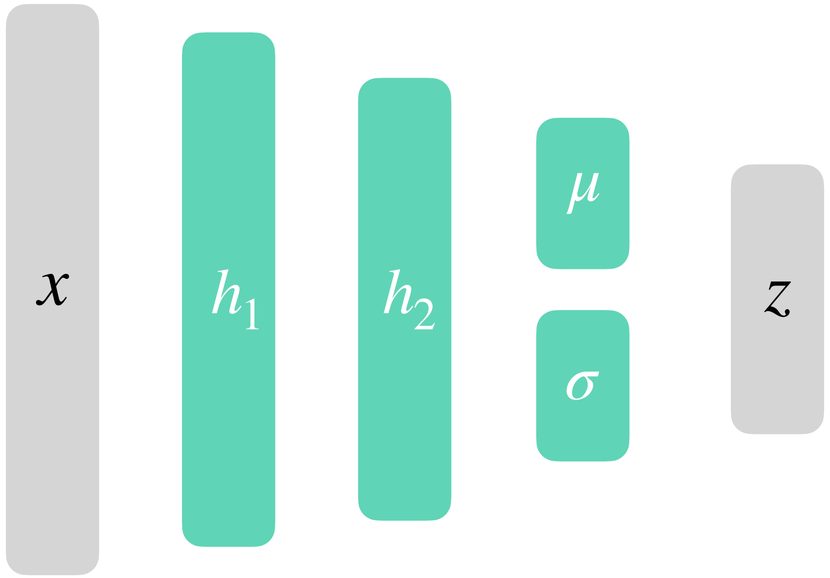
how do we do inference?

reparameterisation trick



how do we do inference?

reparameterisation trick

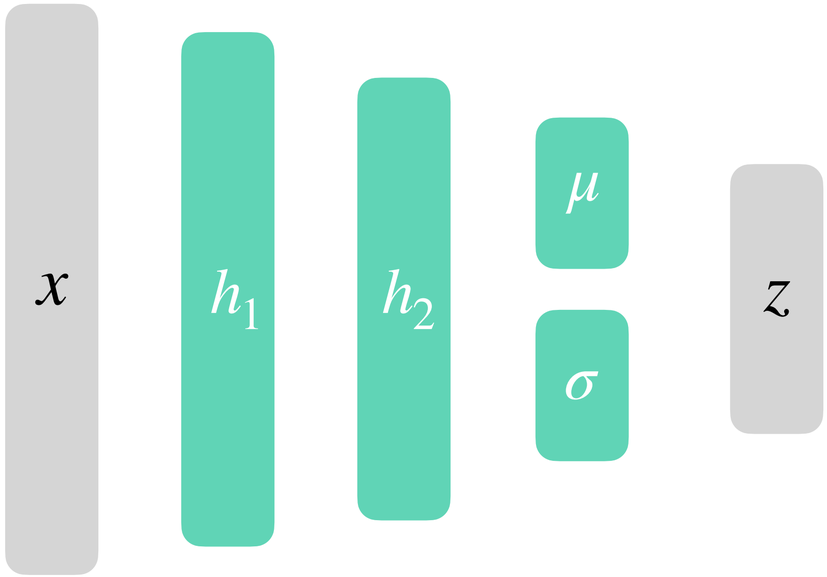


$$q_\phi(z \mid x) = \mathcal{N}(z \mid f_\phi^\mu(x), f_\phi^\sigma(x))$$

$$z \sim q_\phi(z \mid x)$$

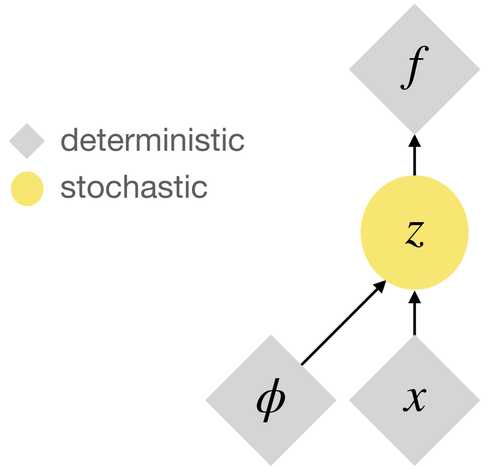
how do we do inference?

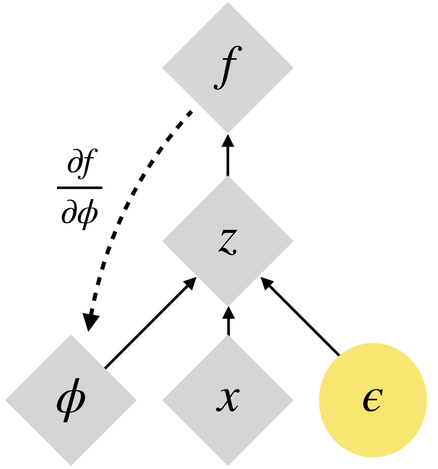
reparameterisation trick



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

$$z \sim q_\phi(z\,|\,x)$$

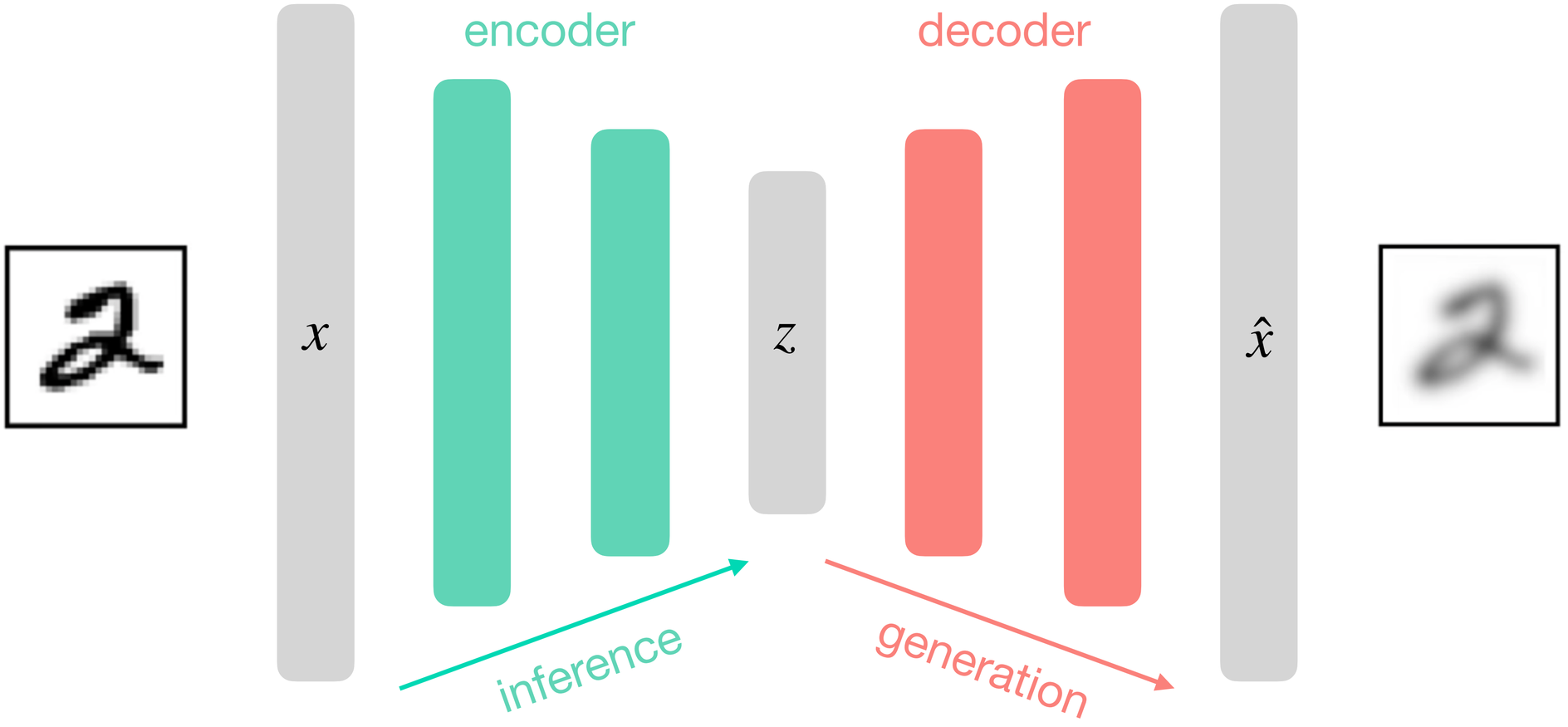




$$z = \mu + \sigma \odot \epsilon$$

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

Variational Autoencoder



Combined objective

evidence lower bound (ELBO)

log 🐍 KL[🐰|🐲]

$$\log p_\theta(x) - KL[q_\phi(z|x)||p(z|x)] = \mathscr{L}_{ELBO}(\theta, \phi, x)$$

Combined objective

evidence lower bound (ELBO)

log 🐍 KL[🐰|🐲]

$$\log p_\theta(x) - KL[q_\phi(z|x)||p(z|x)] = \mathscr{L}_{ELBO}(\theta, \phi, x)$$

$$\mathscr{L}_{ELBO}(\theta, \phi, x) = \mathbb{E}_{🐰} \log 🐹 - \mathbb{E}_{🐰} \mathrm{KL}[🐰|🐣]$$

Combined objective

evidence lower bound (ELBO)

$$\log 🐍 \qquad\qquad \mathrm{KL}[🐰 \,|\, 🐲]$$

$$\log p_\theta(x) - KL[q_\phi(z\,|\,x)\,||\,p(z\,|\,x)] = \mathscr{L}_{ELBO}(\theta, \phi, x)$$

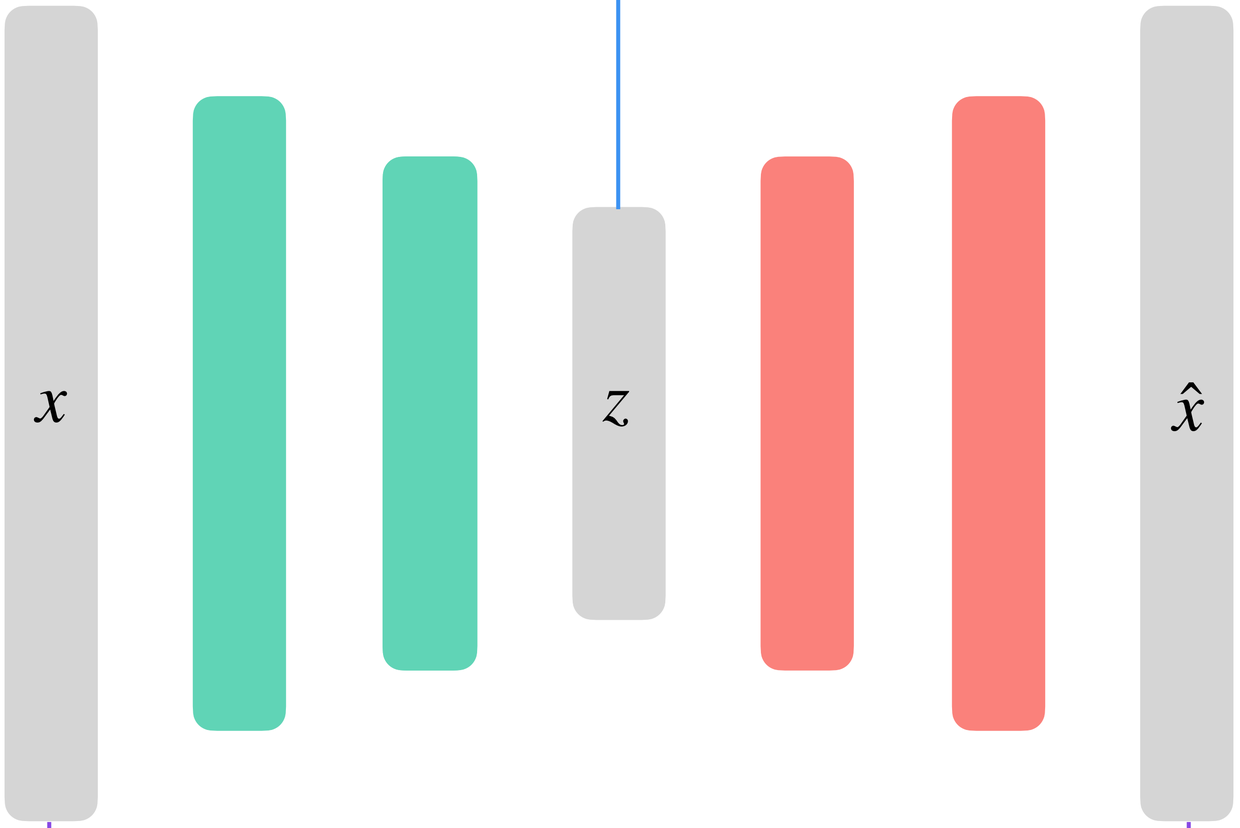
$$\mathbb{E}_{🐰}\ \log 🐹 \qquad\qquad \mathbb{E}_{🐰}\ \mathrm{KL}[🐰 \,|\, 🐣]$$

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

reconstruction — regularisation

$-KL[q_\phi(z\,|\,x)\,||\,p(z)]$

regularisation



reconstruction

$\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x\,|\,z)] \quad = |\,x-\hat{x}\,|^2$

for gaussian noise model

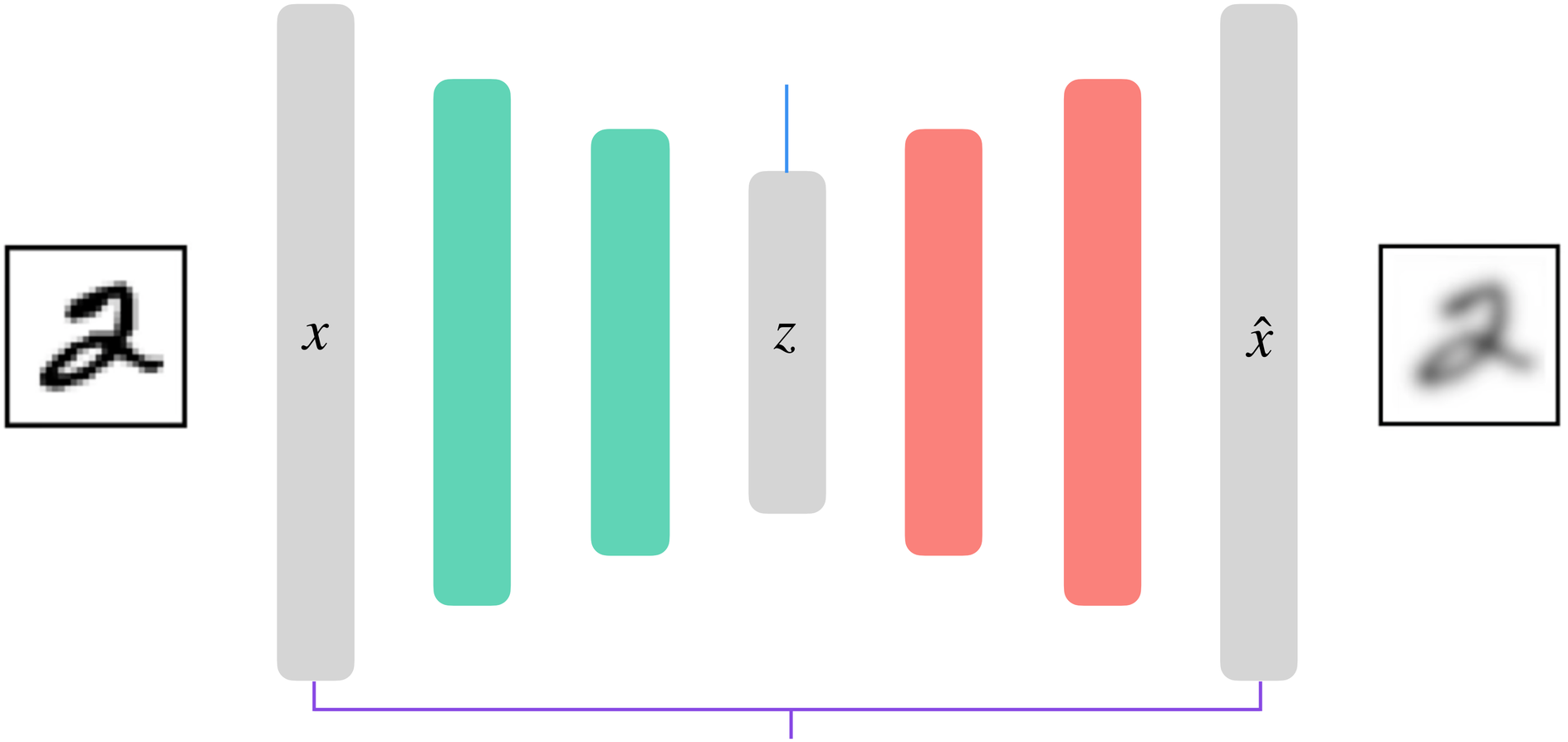
$-KL[q_\phi(z\,|\,x)\,||\,p(z)]$

regularisation

$$q_\phi(z\,|\,x) = \mathcal{N}(z\,|\,f^\mu_\phi(x), f^\sigma_\phi(x))$$
$$p(z) = \mathcal{N}(z\,|\,0, I)$$

analytical result for KL if Gaussian prior and posterior:

$$\frac{1}{2}(1 + \log\sigma^2 - \mu^2 - \sigma^2)$$

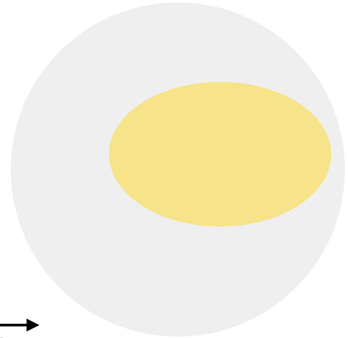


reconstruction

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

$-KL[q_\phi(z\,|\,x)\,||\,p(z)]$

regularisation



$p(z)$

$q_\phi(z\,|\,x)$

low KL

x

z

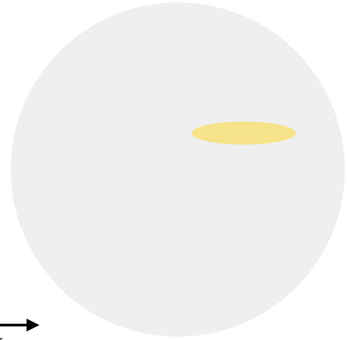
$\hat{x}$

reconstruction

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

$-KL[q_\phi(z\,|\,x)\,||\,p(z)]$

regularisation



$p(z)$

$q_\phi(z\,|\,x)$

high KL

x

z

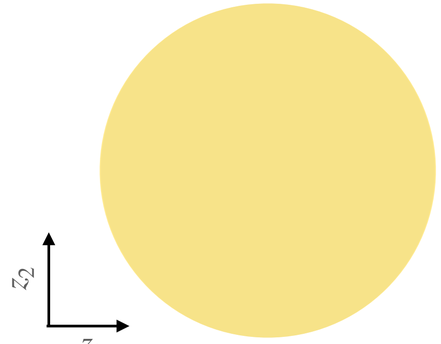
$\hat{x}$

reconstruction

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

$-KL[q_\phi(z\,|\,x)\,||\,p(z)]$

regularisation



$p(z)$

$q_\phi(z\,|\,x)$

zero KL

reconstruction

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

Demo

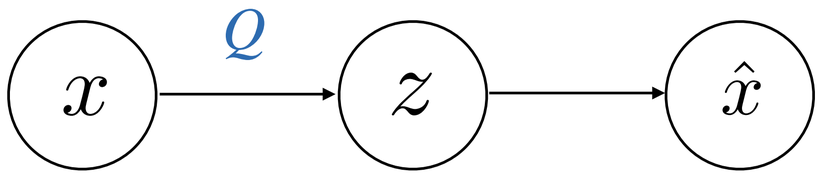
https://github.com/eemlcommunity/PracticalSessions2021/tree/main/generative

Part II

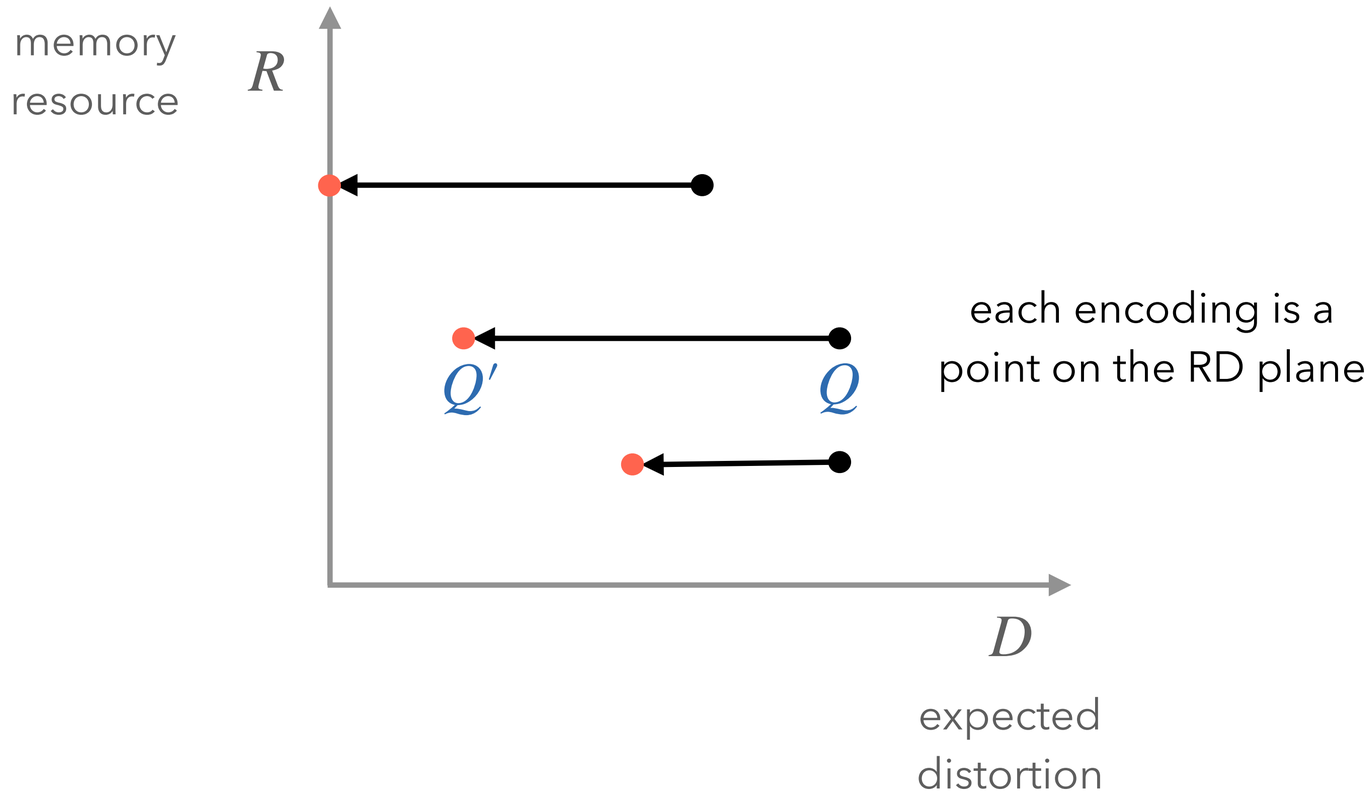
β-VAE

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

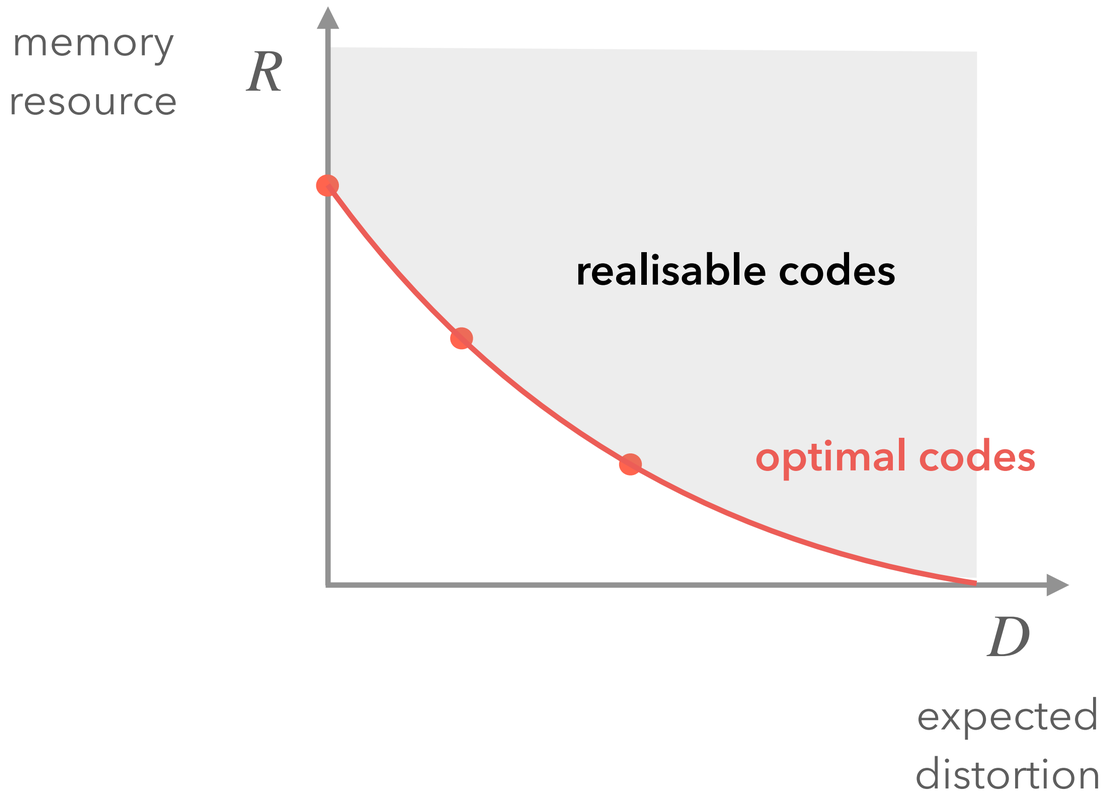
Compression view



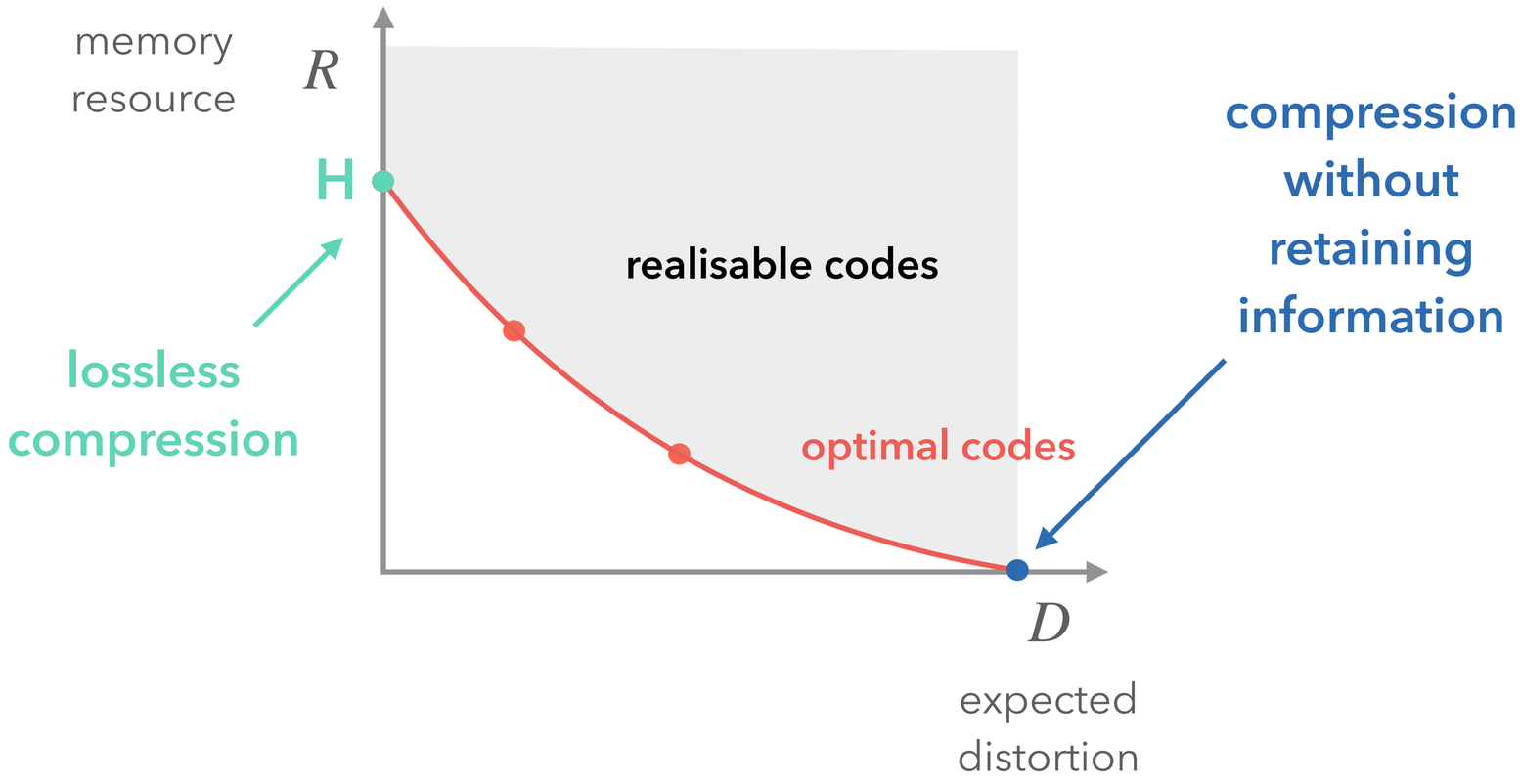
Optimal compression



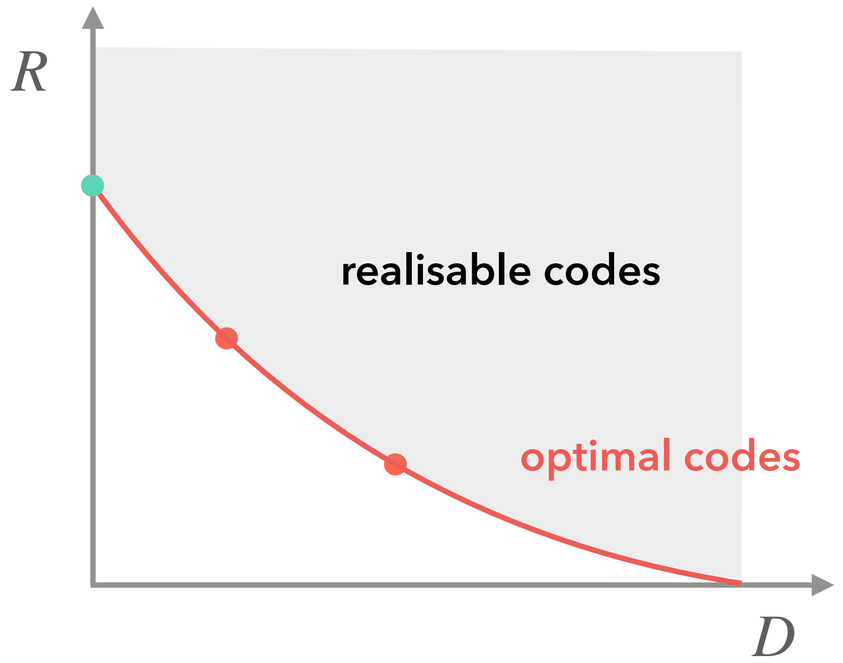
Optimal compression



Optimal compression



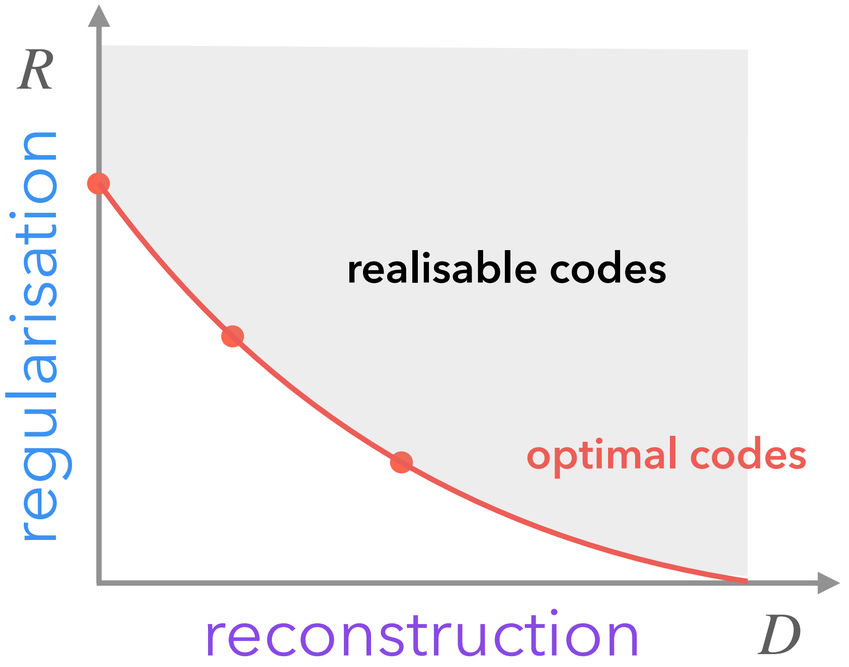
Compression view of ELBO



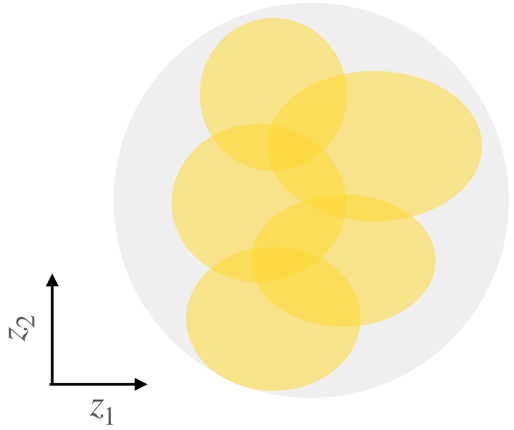
$$\mathscr{L}(\theta, \phi, x) = \mathbb{E}_{z \sim q_\phi(z|x)}[\log p_\theta(x \mid z)] - KL[q_\phi(z \mid x) \mid\mid p(z)]$$

reconstruction regularisation

Compression view of ELBO



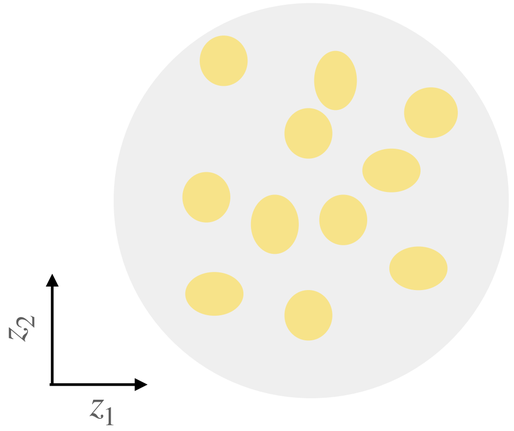
Compression view of ELBO



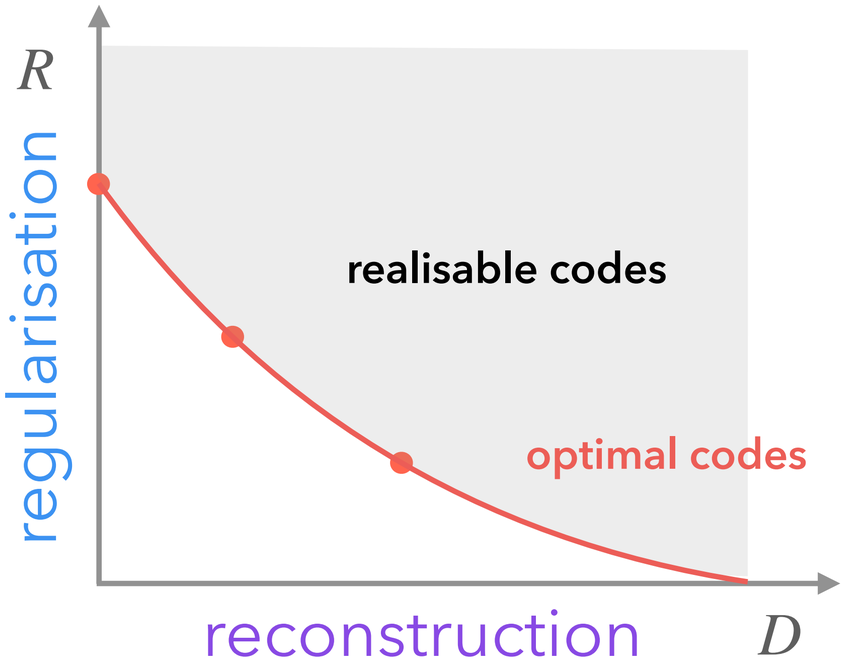
$p(z)$

$q_\phi(z \mid x)$

- low expected KL
- low mutual information



- high expected KL
- high mutual information

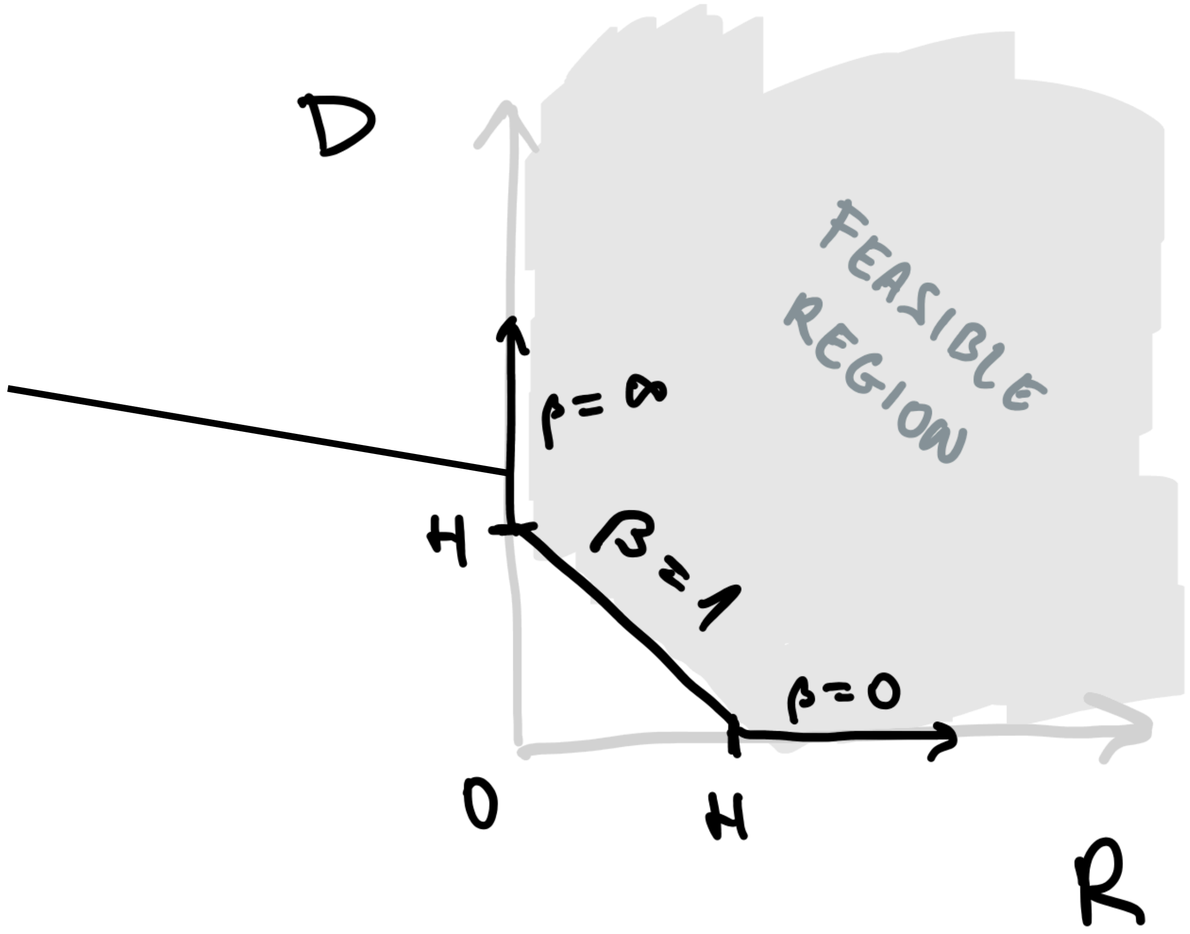


$|x - \hat{x}|^2$

RD plane

no constraints on model family

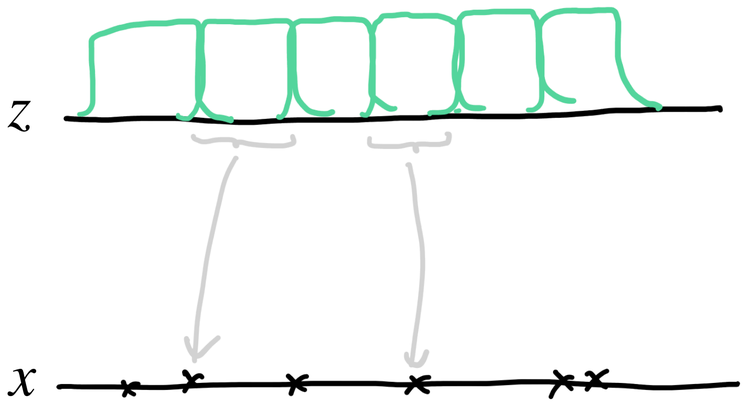
- encoder
- decoder
- prior

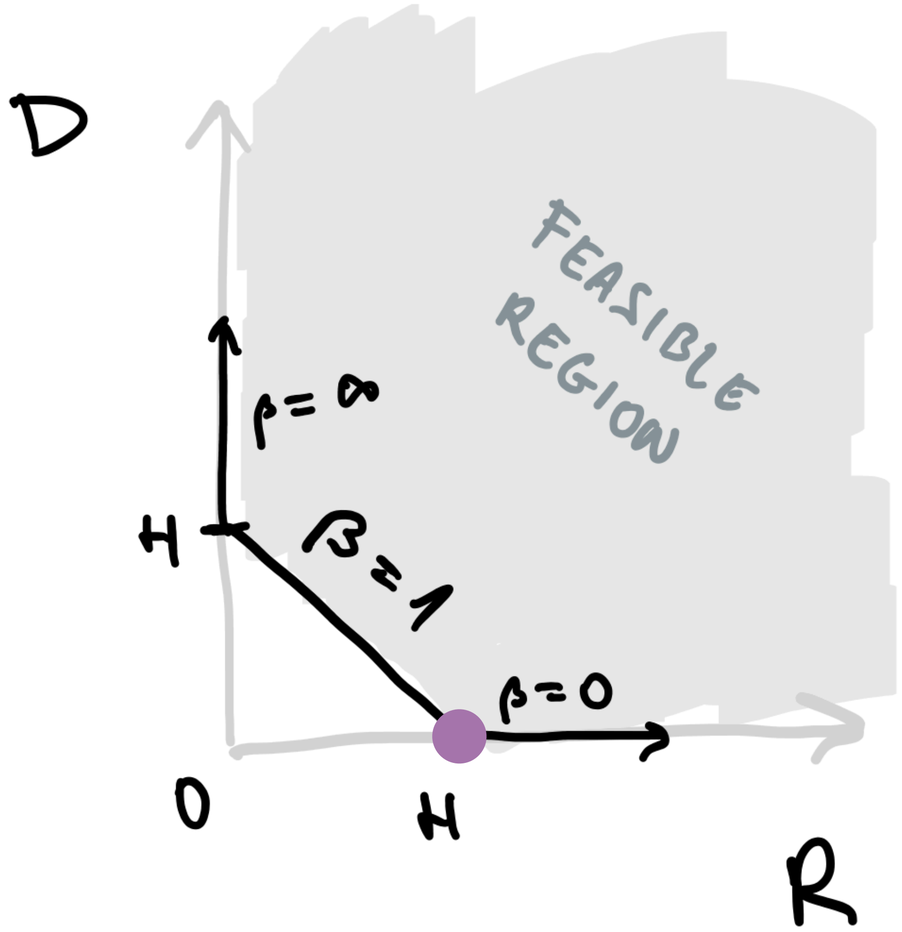


(Alemi et al. 2017)

Auto-encoding limit

- lossless compression
- store and return data points
- each region of Z corresponds to a datapoint

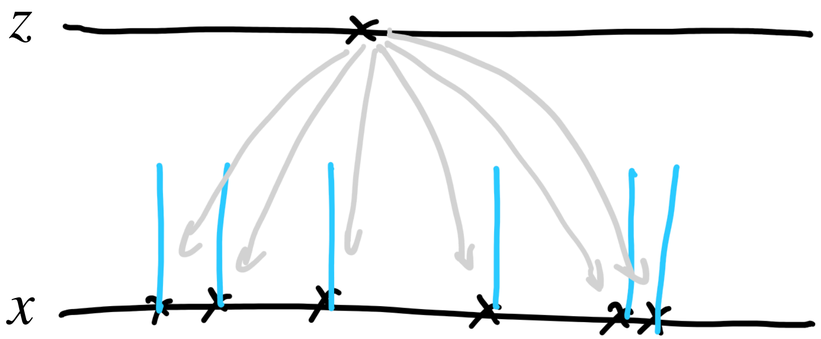


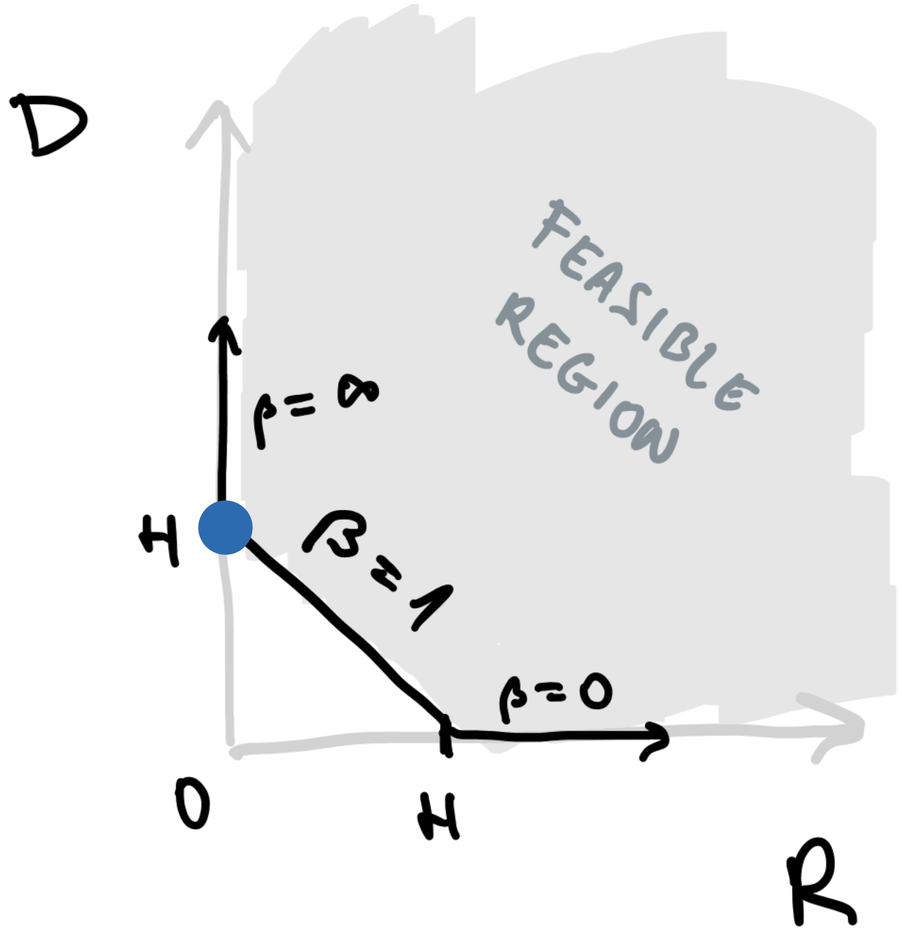


(Alemi et al. 2017)

Auto-decoding limit

- compression without retaining information
- z independent of x
- density estimation without representation learning

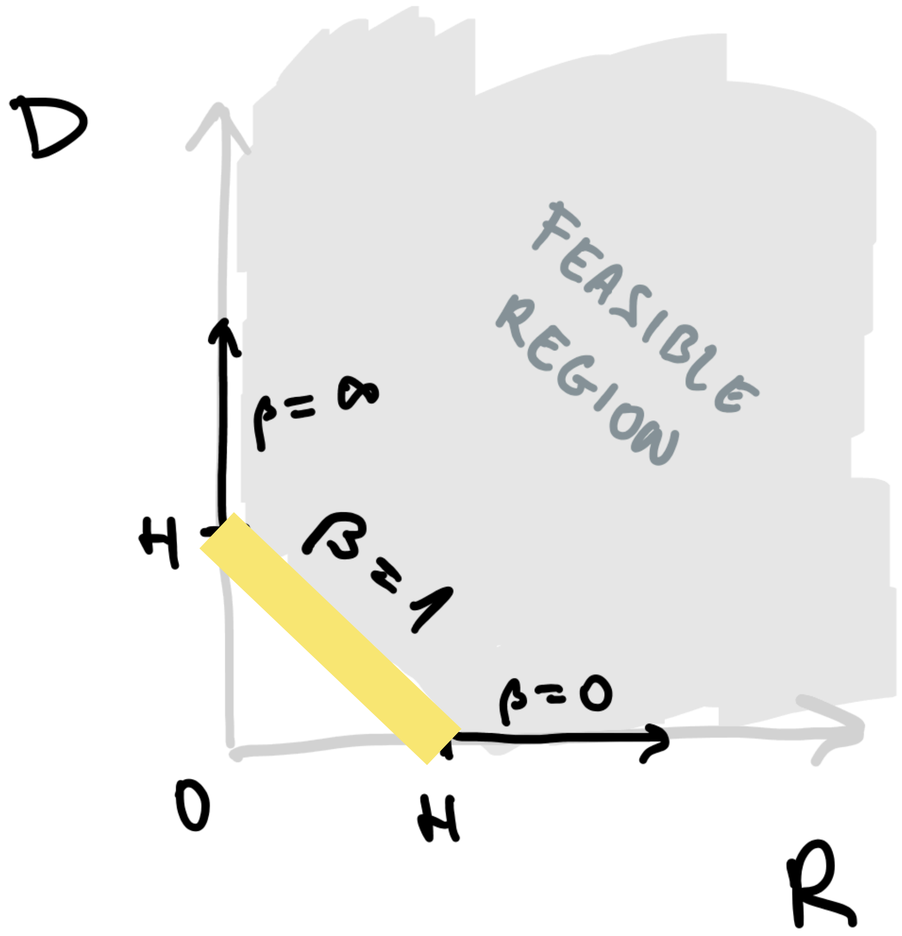




(Alemi et al. 2017)

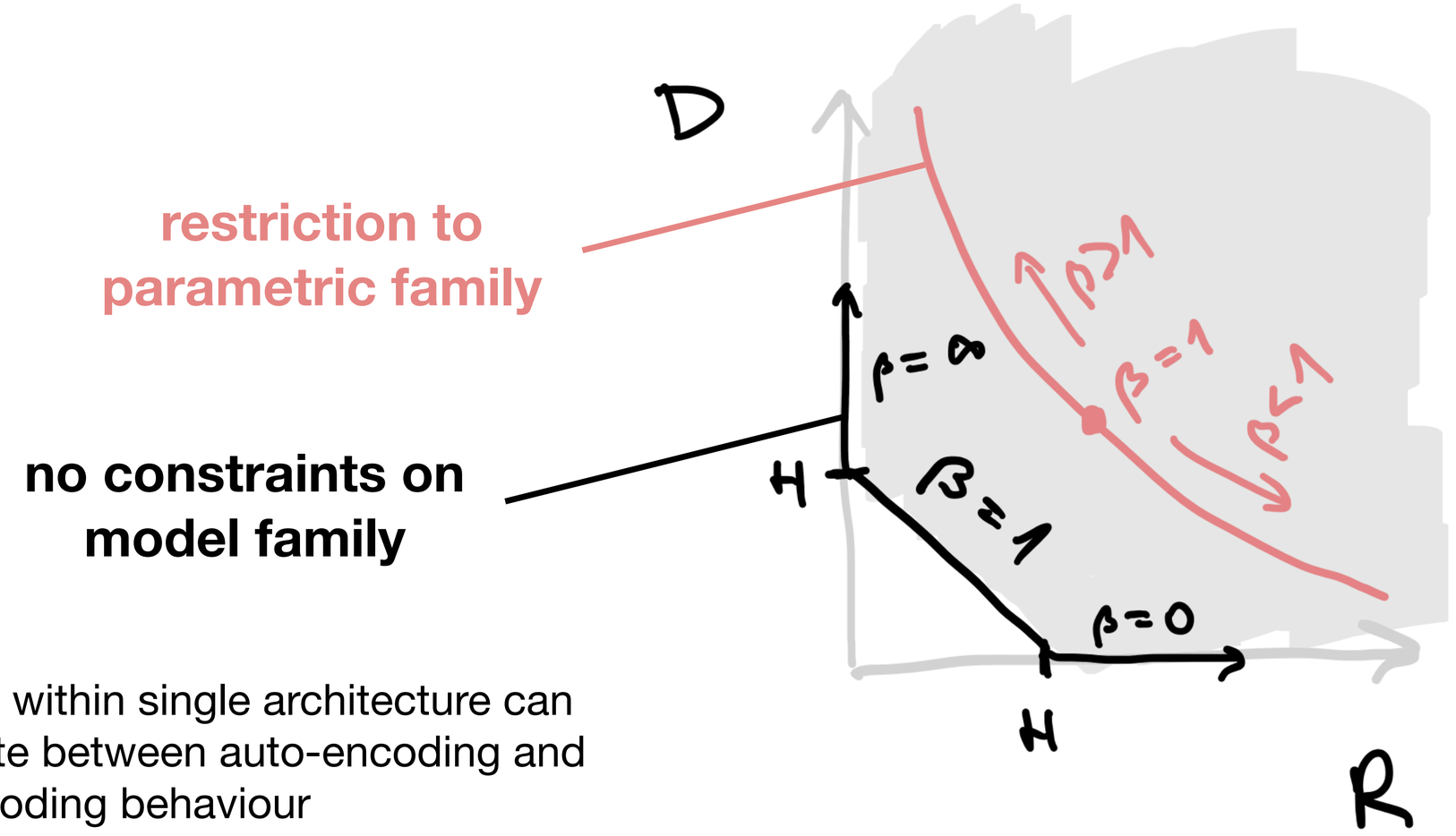
Intermediate points

- intermediate trade-offs between rate and distortion
- these points can’t be targeted through β-VAE objective since they all belong to β=1
- point is selected implicitly through model architecture and initial conditions



(Alemi et al. 2017)

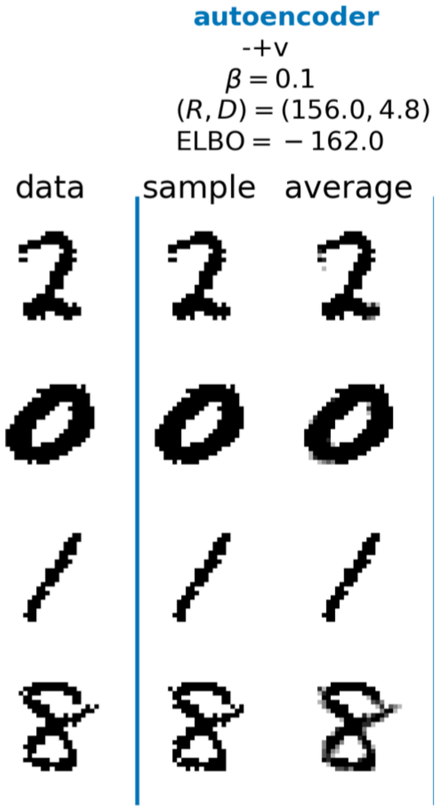
Constrained model family

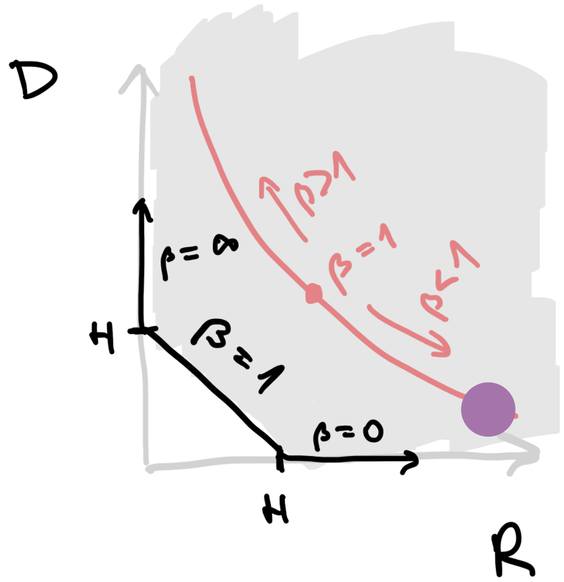


- varying β within single architecture can interpolate between auto-encoding and auto-decoding behaviour

(Alemi et al. 2017)

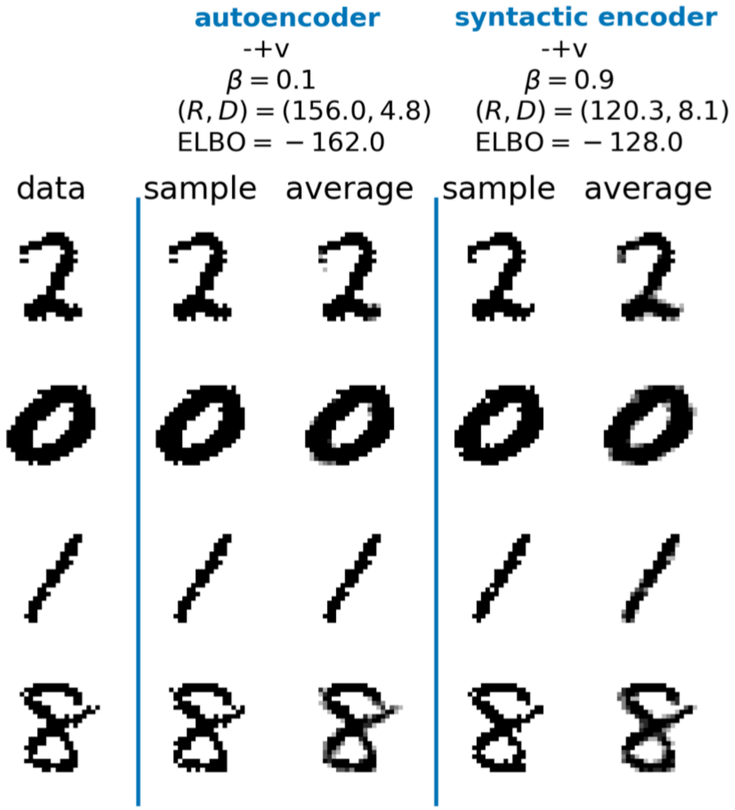
Reconstruction

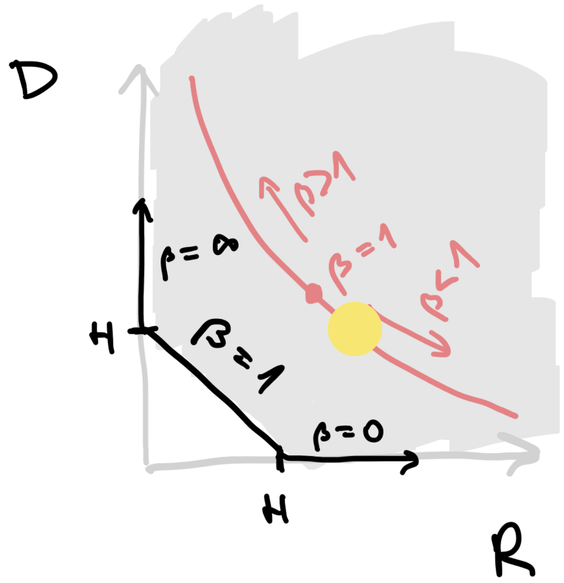




(Alemi et al. 2017)

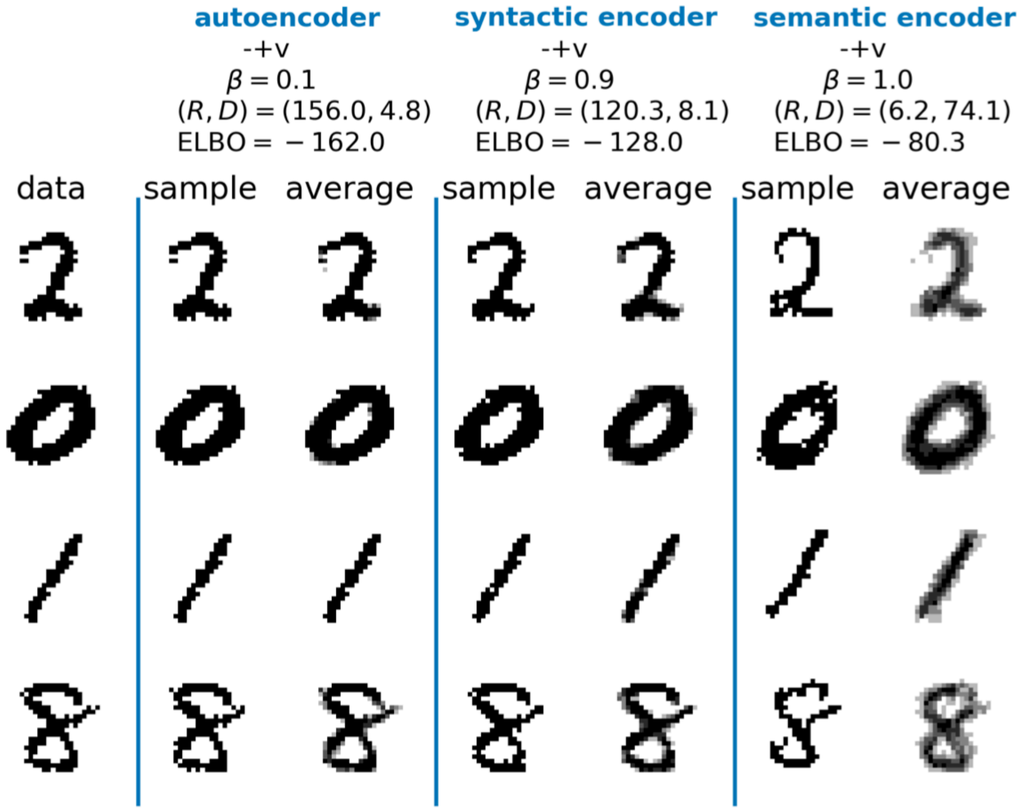
Reconstruction

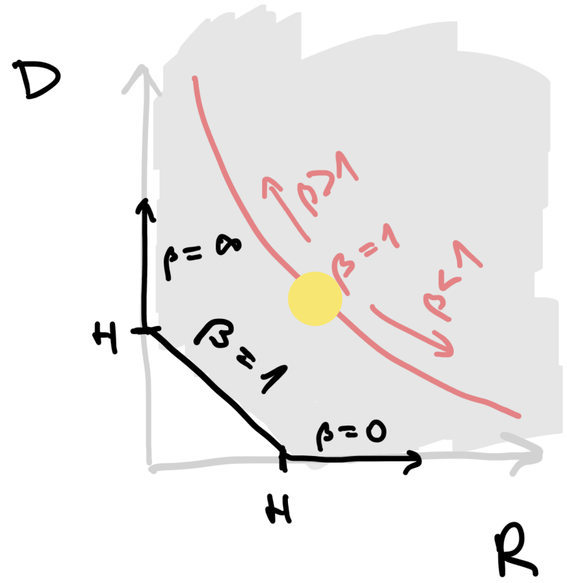




(Alemi et al. 2017)

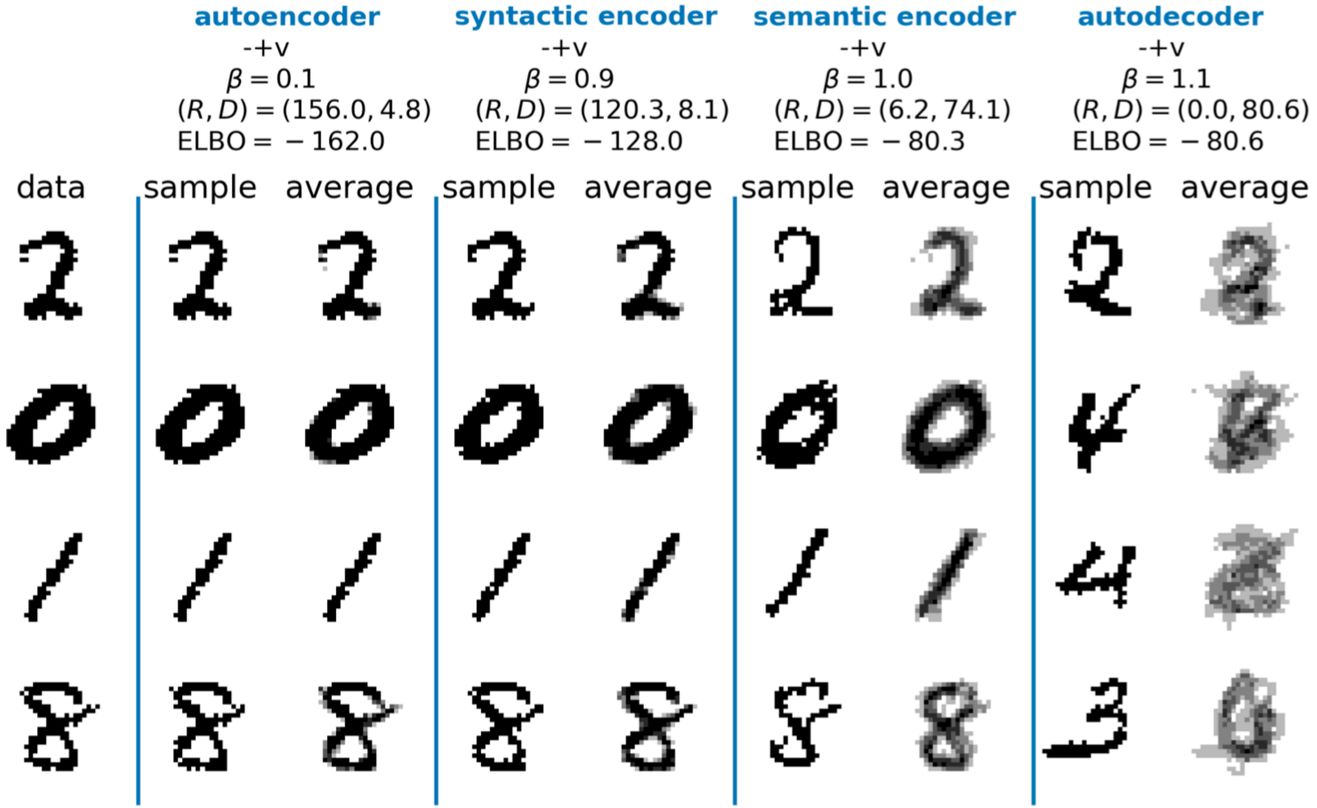
Reconstruction

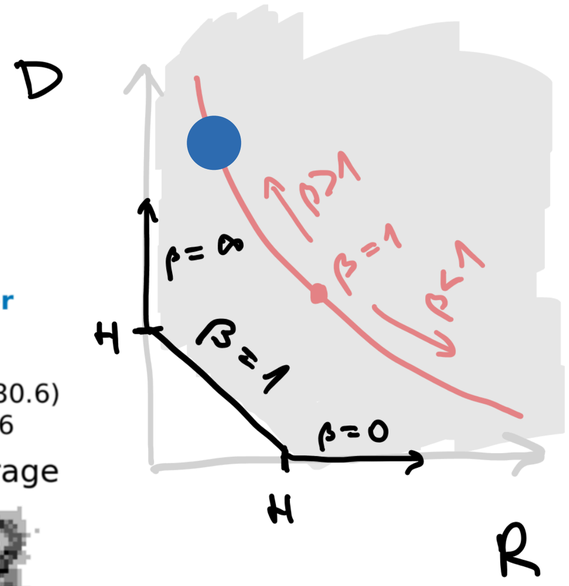




(Alemi et al. 2017)

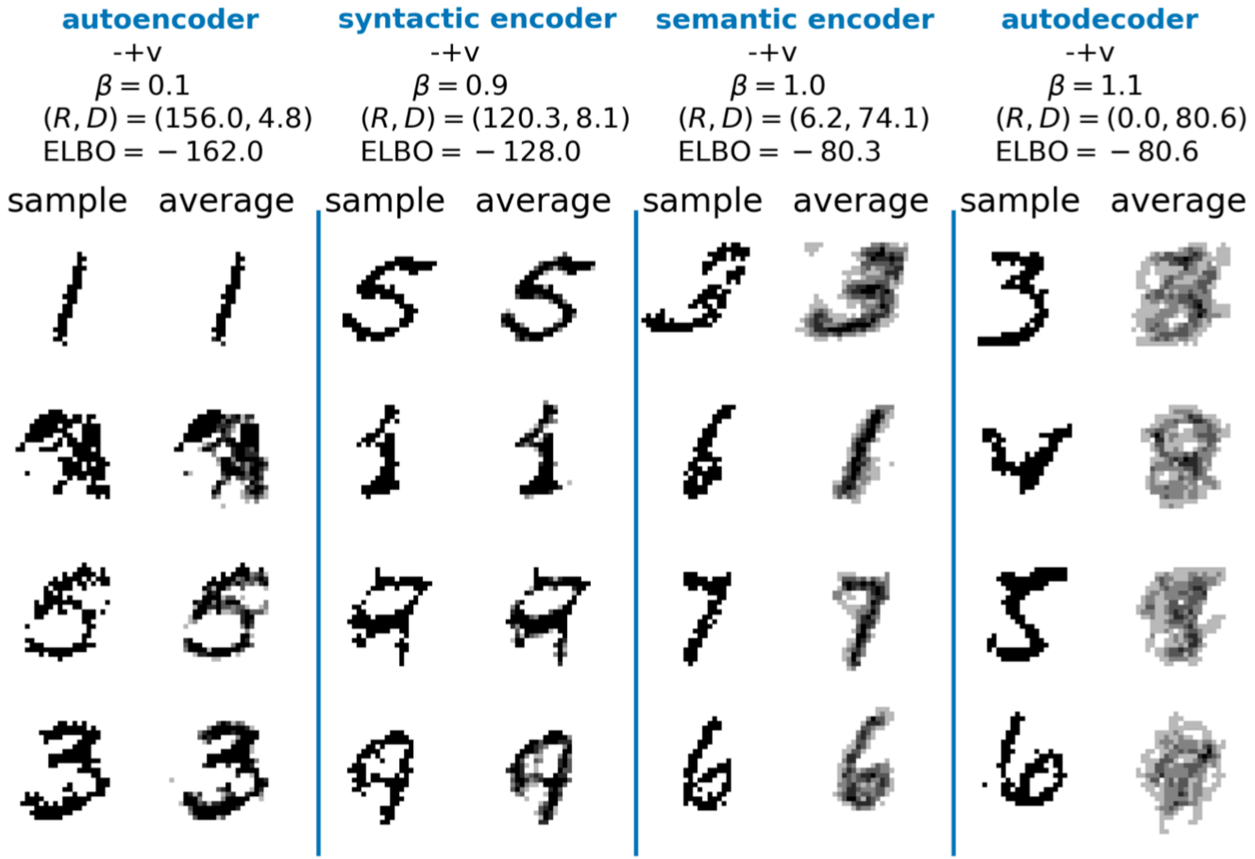
Reconstruction





(Alemi et al. 2017)

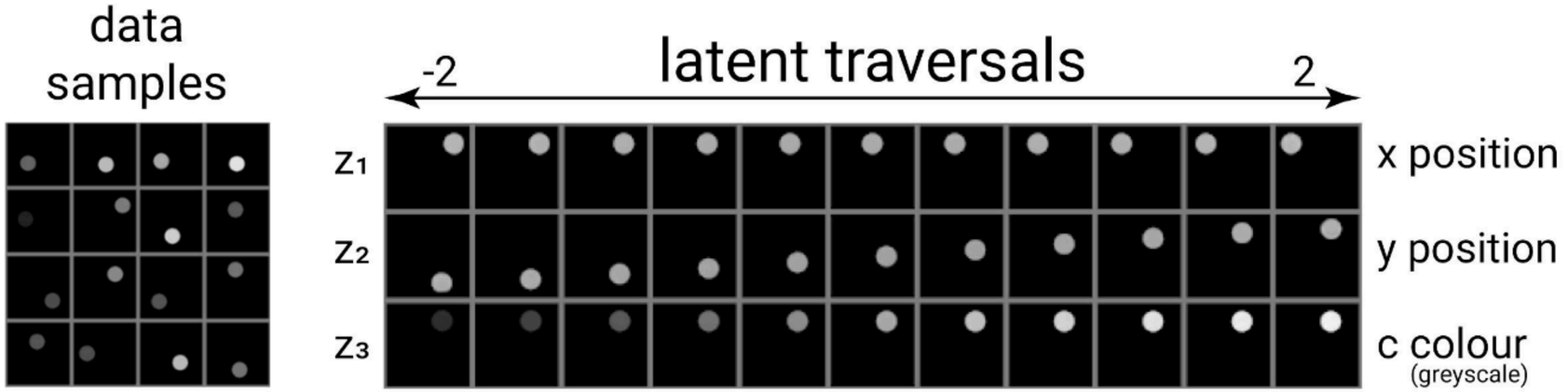
Generation

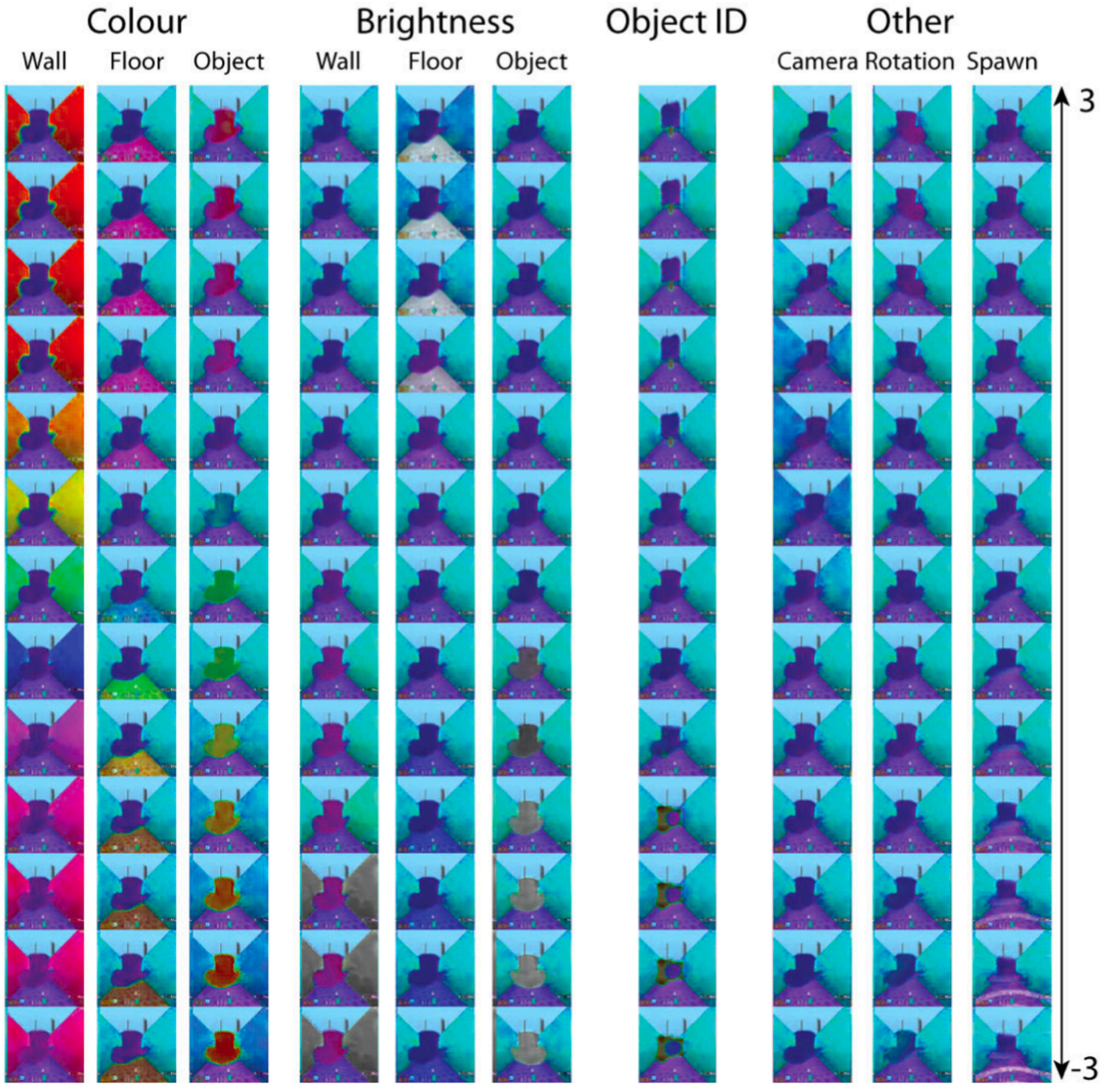


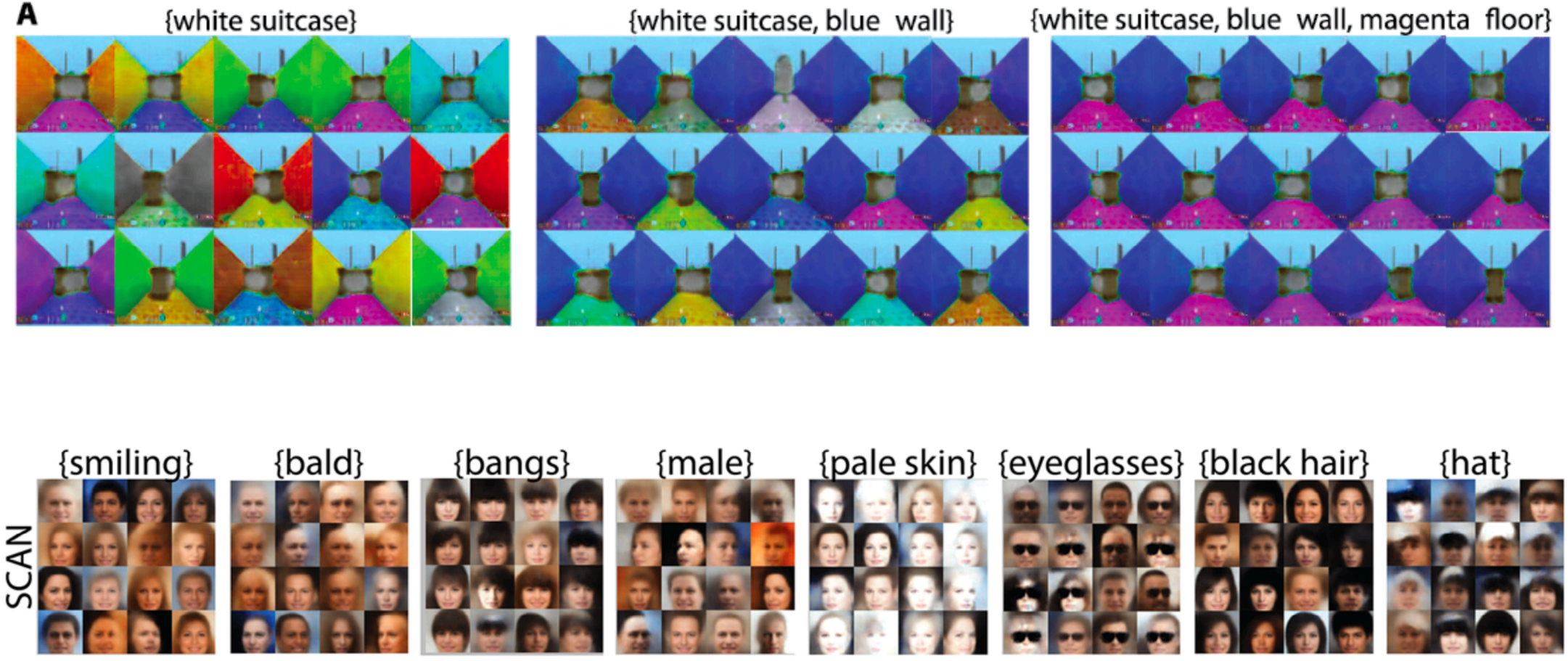
(Alemi et al. 2017)

Disentangled representations

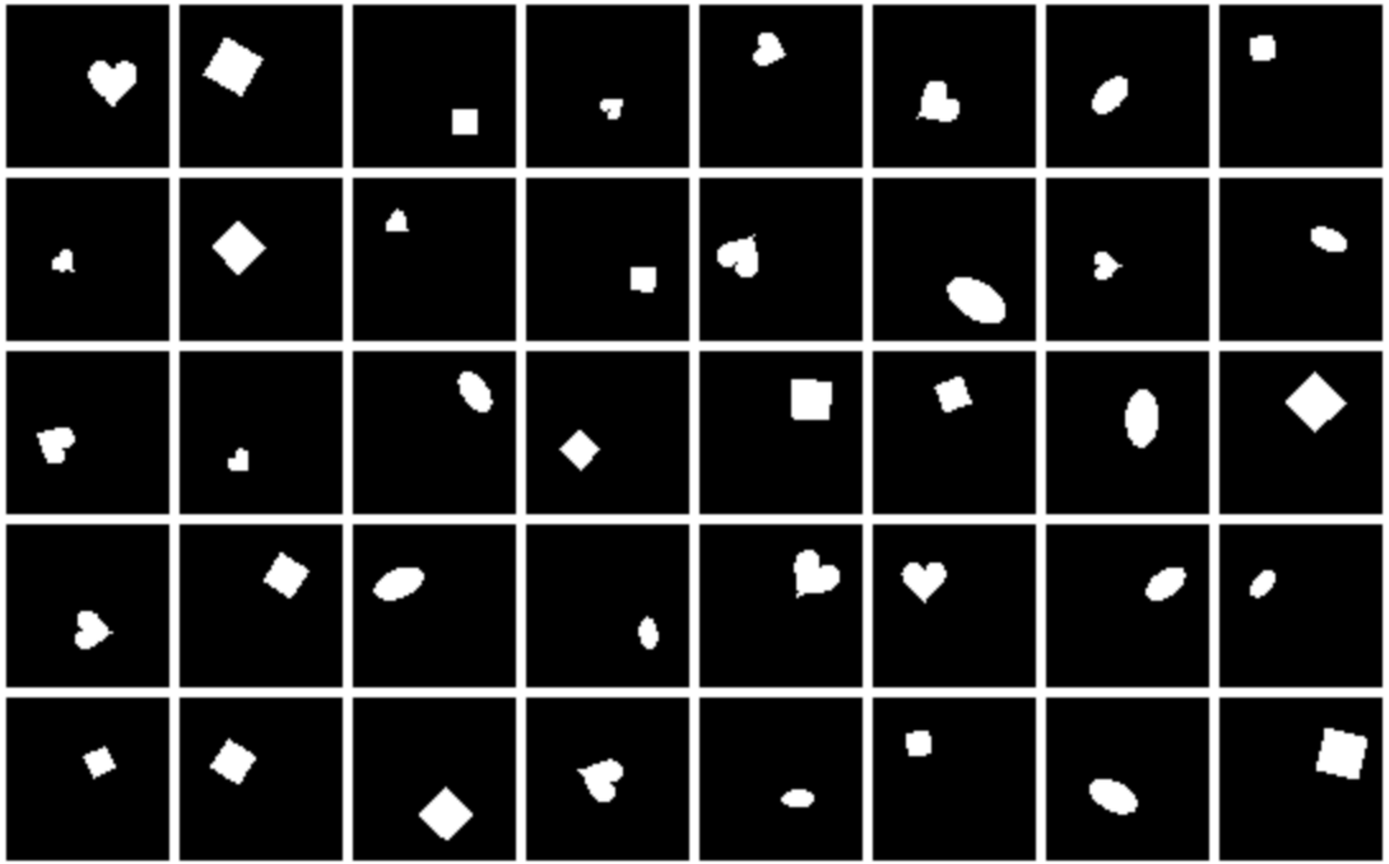
- we were able to control the information content of the latent representation with β
- we would also prefer representations that are **disentangled**
  - disentangled = factorised + interpretable (Bengio, 2013)
  - one generative factor per latent dimension
  - capture symmetry transformations




Colour
Wall
Floor
Object
Brightness
Wall
Floor
Object
Object ID
Other
Camera Rotation
Spawn
3
-3

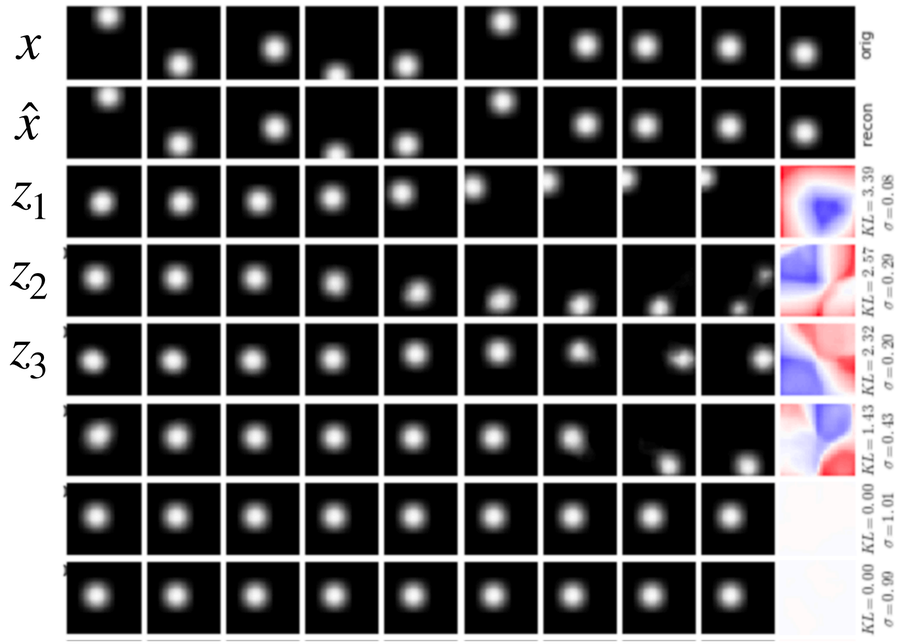

A
{white suitcase}
{white suitcase, blue wall}
{white suitcase, blue wall, magenta floor}
SCAN
{smiling}
{bald}
{bangs}
{male}
{pale skin}
{eyeglasses}
{black hair}
{hat}

dsprites dataset

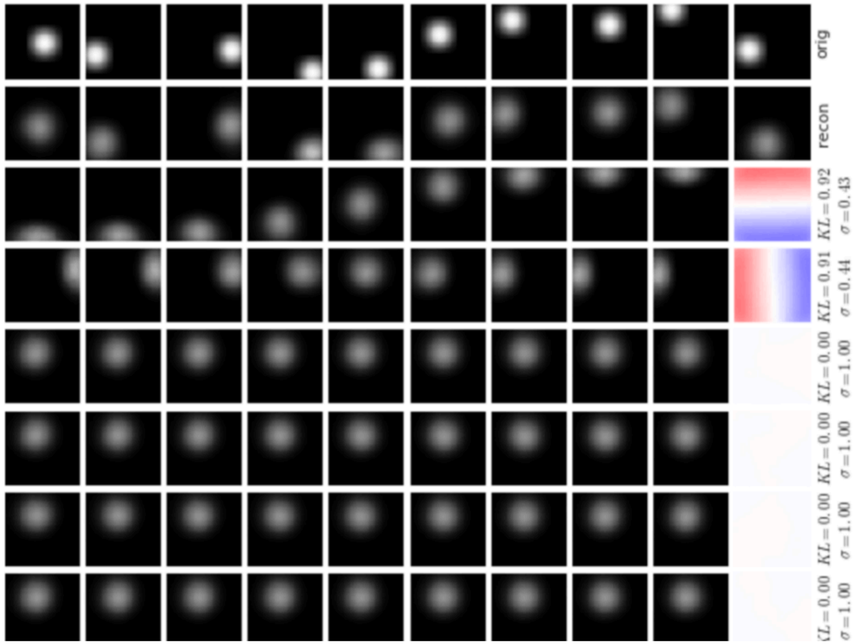


β-VAE

β = 1

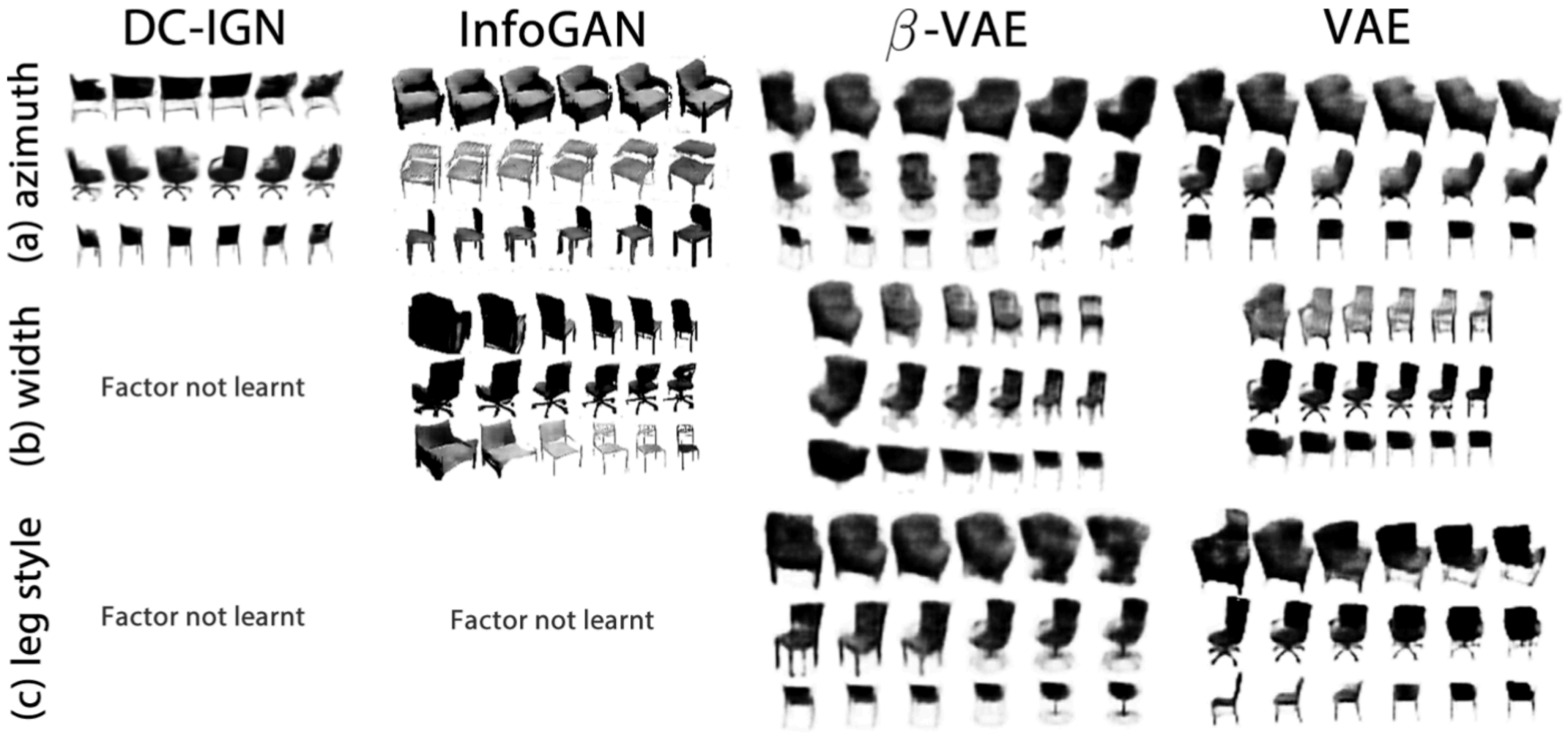


β = 150



(Burgess et al. 2017)

β-VAE


DC-IGN
InfoGAN
β-VAE
VAE
(a) azimuth
(b) width
(c) leg style
Factor not learnt
Factor not learnt
Factor not learnt

Why does β-VAE disentangle?

- mainly because of the diagonal normal approximate posterior
  - different generative factors give different average contributions to reconstruction
  - these factors should have different capacities allocated to them
  - with a diagonal posterior, this necessitates using different dimensions for each factor

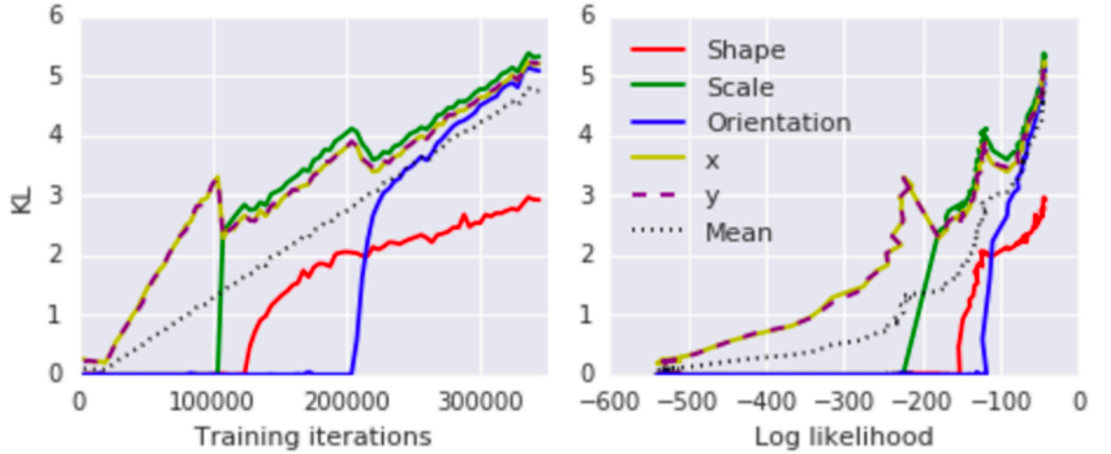
$p(z)$

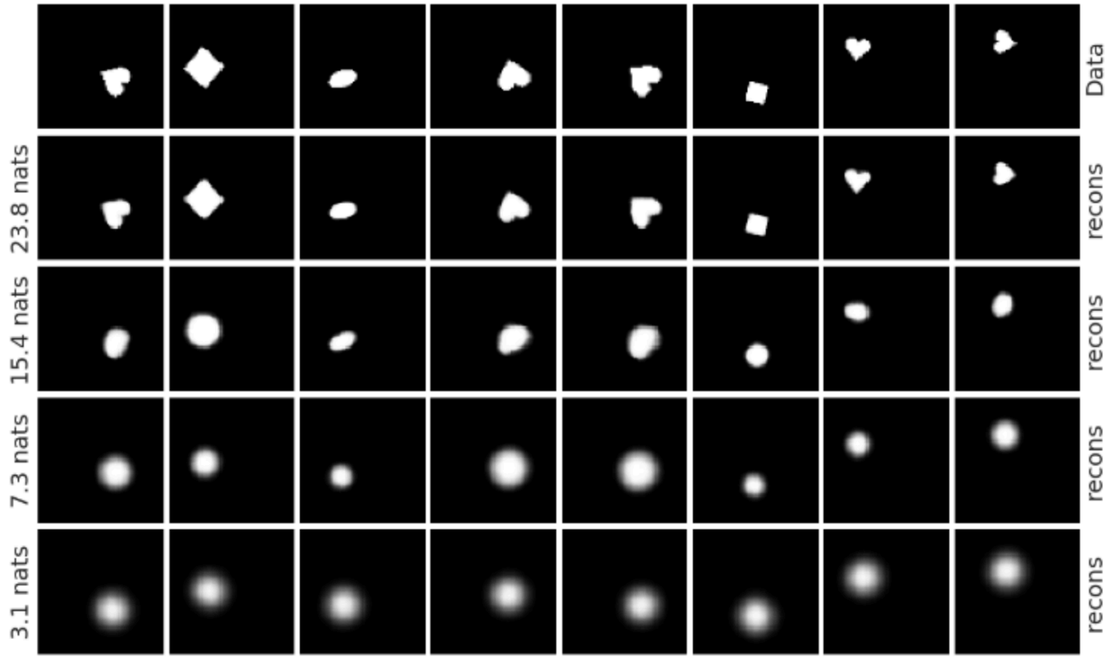
$q_\phi(z \mid x)$

Why does β-VAE disentangle?

- mainly because of the diagonal normal approximate posterior
  - different generative factors give different average contributions to reconstruction
  - these factors should have different capacities allocated to them
  - with a diagonal posterior, this necessitates using different dimensions for each factor

- Burgess et al., 2017. “**Understanding Disentangling in Beta-VAE**.” http://arxiv.org/abs/1804.03599
- Rolinek et al., 2019. “**Variational Autoencoders Pursue Pca Directions (by Accident)**.” https://doi.org/10.1109/CVPR.2019.01269
- Zietlow, Dominik, Michal Rolínek, and Georg Martius. 2020. “**Demystifying Inductive Biases for β-VAE Based Architectures**.”





$$\mathcal{L}(\theta, \phi; \mathbf{x}(\mathbf{f}), \mathbf{z}, C) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{f})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \gamma \left| D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{f}) \parallel p(\mathbf{z})\right) - C \right|$$

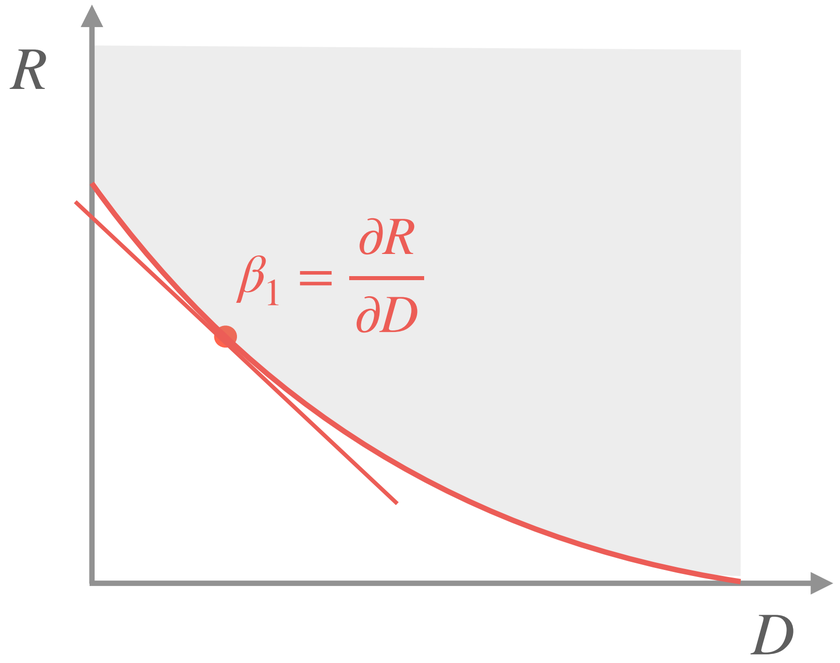
(Burgess et al. 2017)

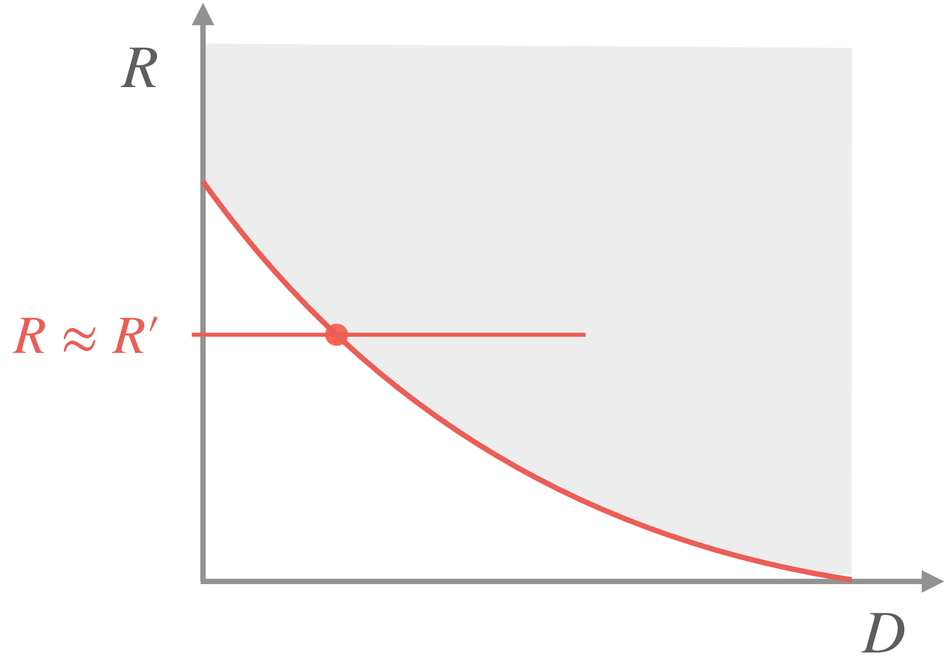
Modified objective

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

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

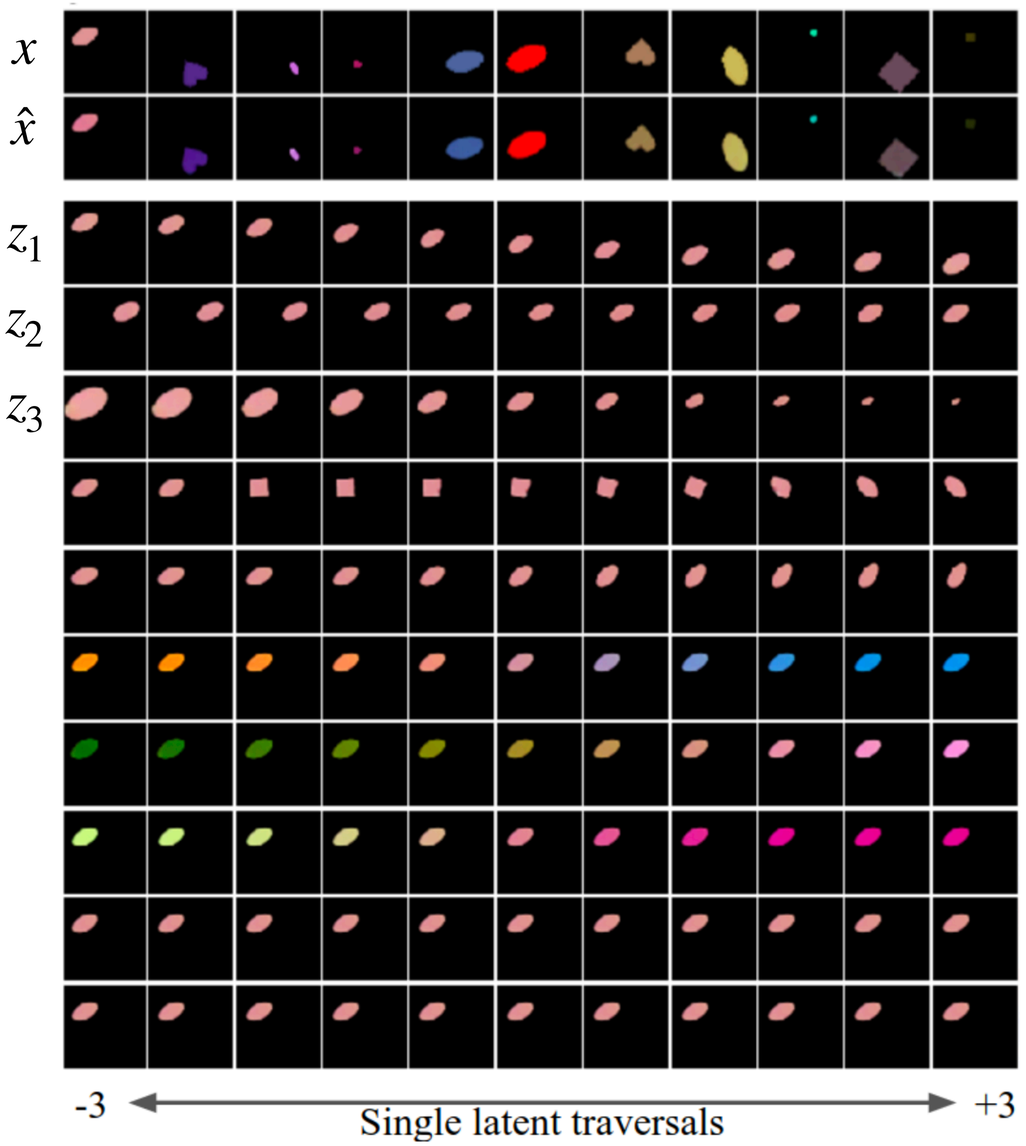
Modified objective





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

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



(Burgess et al. 2017)

Demo

https://github.com/eemlcommunity/PracticalSessions2021/tree/main/generative