Kirill Struminsky

Variational Auto-Encoders

2021

MLHEP 2021

NATIONAL RESEARCH UNIVERSITY HIGHER SCHOOL OF ECONOMICS

LAMBDA • HSE

Yandex

SCHOOL OF DATA ANALYSIS

EPFL

SIT Schaffhausen Institute of Technology

A Quick Overview

Unsupervised learning: dataset $D = \{x_i\}_{i=1}^{N}$ with N i.i.d. samples

Goal: approximate the data distribution

Solution: Variational Auto-Encoder ***(VAE)****

based on the Bayesian Deep Learning toolkit

Useful for down-stream tasks

- semi-supervised learning
- anomaly detection
- data imputation, etc.

Kingma D. P., Welling M. Auto-encoding variational bayes // ICLR 2014.

Roadmap

- How to approximate data distribution?
  A. Design approximating distributions
  B. Fit the data
- Why is it called “auto-encoder”?
- Use cases and further insights

Designing Distributions

Example: Modeling Human Height

- Gaussian approximation

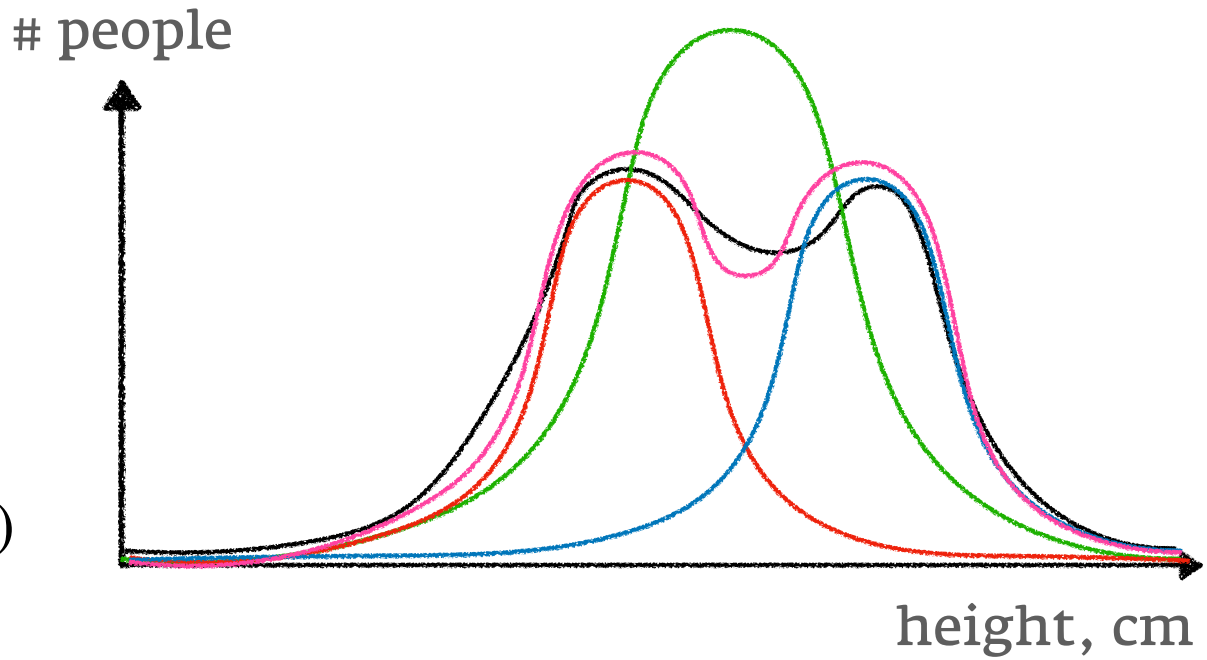
$$p(x) = \mathcal{N}(1.65,\ 0.15^2)$$

- Take gender into account

$$p(x, z) = p(x \mid z)p(z)$$

- Take age into account?

$$p(x, z, t) = p(x \mid z, t)p(z)p(t)$$



Latent Variable Models

- Introduce **a latent variable** $z \sim p(z)$
- Define $p(x \mid z)$ to connect **the observed variable** x and z

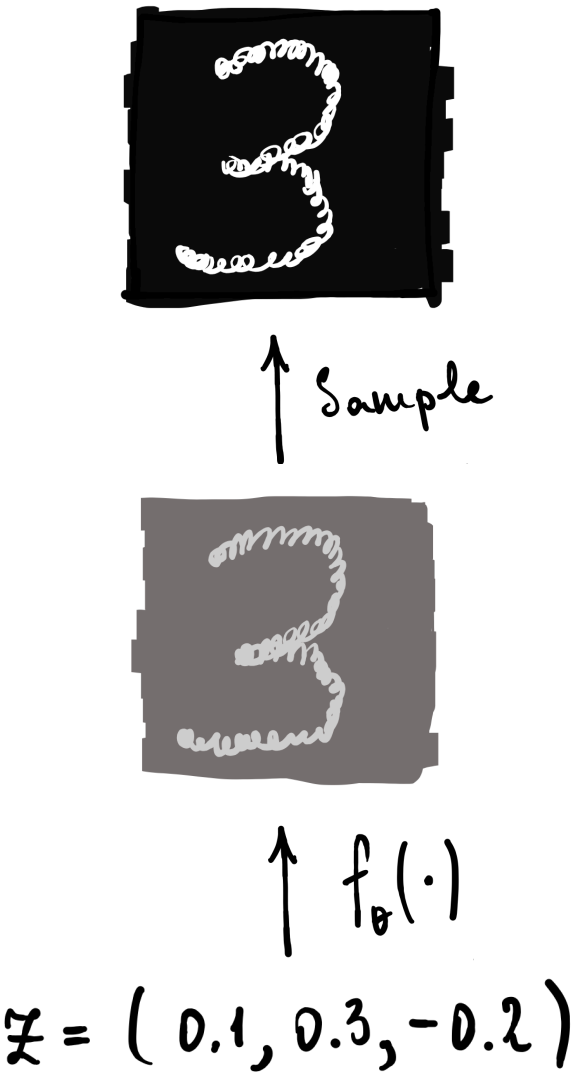
  i.e. $p(x \mid z) = \mathcal{N}(\mu(z),\, \sigma^2(z))$
- Model x with **the marginal distribution** $p(x) = \mathbb{E}_{p(z)} p(x \mid z)$

Vanilla VAE Distribution

- Model binary images $x \in \{0,1\}^{28\times28}$
- Latent variable is a real vector $z \in \mathbb{R}^d$

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

- Define $p_\theta(x \mid z)$ with **a decoder**
  $f_\theta(\cdot) : \mathbb{R}^d \to \mathbb{R}^{28x28}$
  - Pixels are independent
  - Decoder $f_\theta(z)$ defines mean values



Fitting The Model

Maximum (Marginal) Likelihood

- At this point, we have
  - Input data $D = \{x_i\}_{i=1}^N$
  - Latent variable model $p_\theta(x) = \mathbb{E}_{p(z)} p_\theta(x \mid z)$
- To choose $\hat{\theta}$ solve $\max_\theta \sum_i \log p_\theta(x_i)$
- We cannot compute the expectation $p_\theta(x) = \mathbb{E}_{p(z)} p_\theta(x \mid z)$

Estimating $\log p_\theta(x) = \log \mathbb{E}_{p(z)} p_\theta(x \mid z)$

- MC-estimate $\log p_\theta(x) \approx \log p_\theta(x \mid z)$ for $z \sim p(z)$ is biased & noisy
- Solution:
  1. use ELBO: $\log p_\theta(x) \geq \mathbb{E}_{q(z)} \log \frac{p_\theta(x \mid z) p(z)}{q(z)}$
  2. find $q(z) \approx p_\theta(z \mid x)$ by maximising ELBO w.r.t. $q(z)$
- Estimate $\frac{\log p_\theta(x \mid z) p(z)}{q(z)}$ with $z \sim q(z) \approx p_\theta(z \mid x)$ works great!

Encoder Network for Amortised Inference

Bayesian networks:

Need to infer $q(w) \approx p(w \mid D)$ only once

Variational auto-encoder:

Need to infer $q_i(z_i) \approx p(z_i \mid x_i)$ for each x_i in D

- Amortised inference: **an encoder** $g_\phi(x)$ parametrises $q(z_i)$ for each x_i
  - Pros: fast inference
  - Cons: $g_\phi(x_i)$ may give suboptimal $q_i(z_i)$
- Example: $q_\phi(z \mid x) = \mathcal{N}(\mu_\phi(x), \sigma^2_\phi(x))$ for $g_\phi(x) = [\mu_\phi(x), \sigma_\phi(x)]$

The Objective

$$\max_{\theta,\phi} \mathscr{L}(D, \phi, \theta) = \sum_{x_i \in D} \mathbb{E}_{q_\phi(z_i|x_i)} \log \frac{p_\theta(x_i \mid z_i) p(z_i)}{q_\phi(z_i \mid x_i)}$$

- Optimise ϕ (encoder) to improve the likelihood approximation
- Optimise θ (decoder) to fit the model

Estimating the Objective

$$\frac{1}{N}\sum_{x_i \in D} \mathbb{E}_{q_\phi(z_i|x_i)} \log \frac{p_\theta(x_i \mid z_i)p(z_i)}{q_\phi(z_i \mid x_i)} \approx \log \frac{p_\theta(x_k \mid z_k)p(z_k)}{q_\phi(z_k \mid x_k)}$$

- Re-sample data to estimate $\frac{1}{N}\sum_{x_i \in D}$
- Use **the reparametrization trick** to sample z_k and estimate $\mathbb{E}_{q_\phi(z_k|x_k)}$
- Use gradient ascent for optimization

Quick Summary

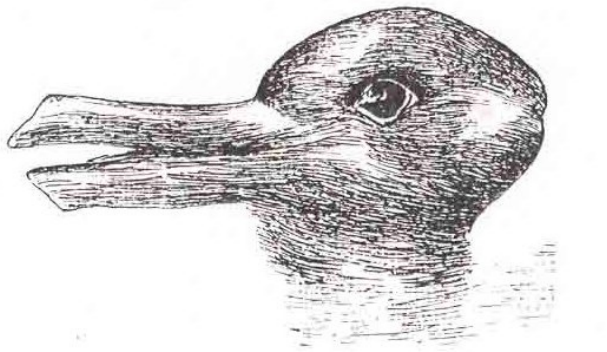
What is a VAE?

1. Probabilistic model $p(x, z) = p(x \mid z)p(z)$ for data
   - Example: $p_\theta(x \mid z) = \mathcal{N}(\mu_\theta(z), \Sigma_\theta(z)); p(z) = \mathcal{N}(0; I)$
2. Additional model $q_\phi(z \mid x)$ for **amortised inference**
   - Example: $q_\phi(z \mid x) = \mathcal{N}(\mu_\phi(x), \sigma^2_\phi(x))$
3. ELBO objective (+ reparametrisation trick for training)

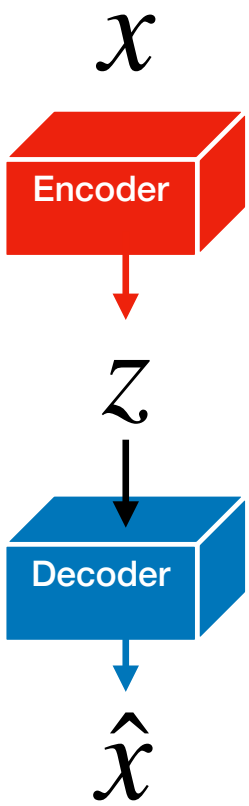
A Different Perspective on VAE

The Two Faces
of Variational Auto-Encoder

- VAE is a probabilistic framework
- VAE is auto-encoder



Auto-Encoders



- Classic deep learning concept
- Backpropagation without supervision
- Reconstruction error objective

$$\min_{\theta,\phi} \|x - f_\theta(g_\phi(x)\|^2$$

VAE as an Auto-Encoder

Objective

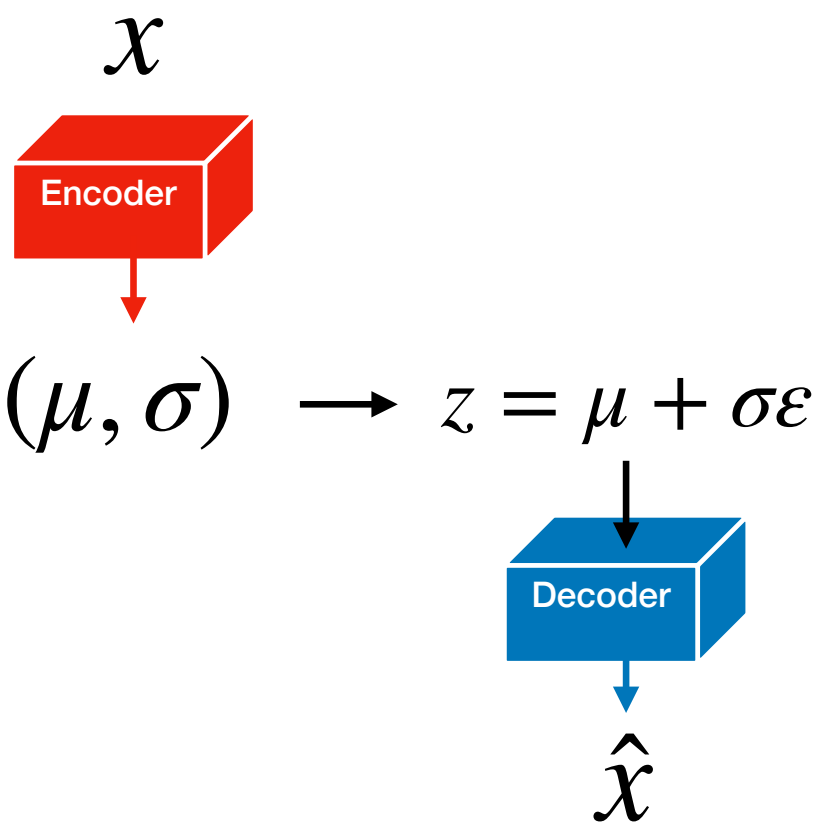
$$\mathbb{E}_{q(z|x)}\left[\log p(x\,|\,z) - \log \frac{p(z)}{q(z\,|\,x)}\right]$$

Example:

for $p(x\,|\,z) = \mathcal{N}(f(z), \sigma^2 I)$

and $q(z\,|\,x) = \mathcal{N}(g(x), \sigma^2 I)$

$$\log p(x\,|\,z) = C_1 + C_2 \|x - f(g(x) + \sigma\varepsilon)\|^2$$

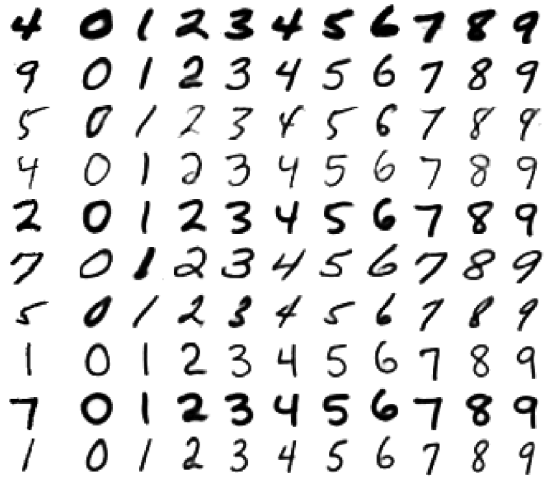


AE vs. VAE

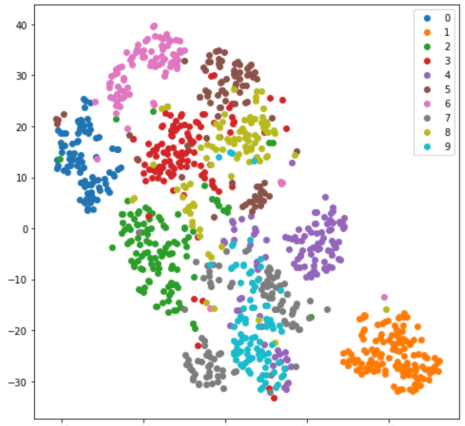
- Additional noise makes difference
  - AE only learns to reconstruct $g_\phi(x)$
  - VAE learns to reconstruct vicinities of $g_\phi(x)$
- VAE approximates data distribution
- Easy to sample from $p_\theta(x)$
  - Generate $z \sim p(z)$
  - Compute $p_\theta(x \mid z)$ and generate $x \sim p_\theta(x \mid z)$

Use Cases & Further Insights

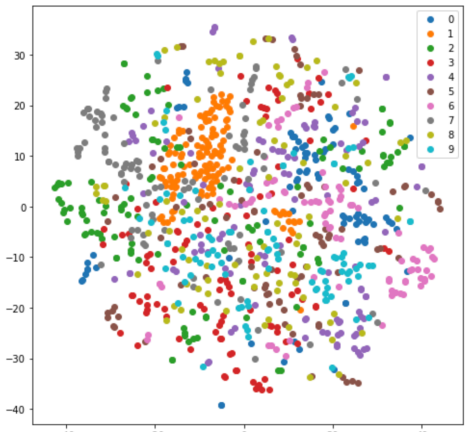
Perks of VAE



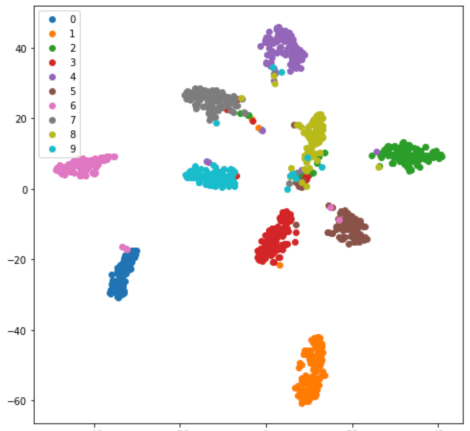
- Prior $p(z)$ allows to induce representation structure
  - Latent variable is a parse tree *(arxiv.org/abs/1807.09875)*
  - Decouple digit classes & handwriting *(arxiv.org/abs/1406.5298)*



z



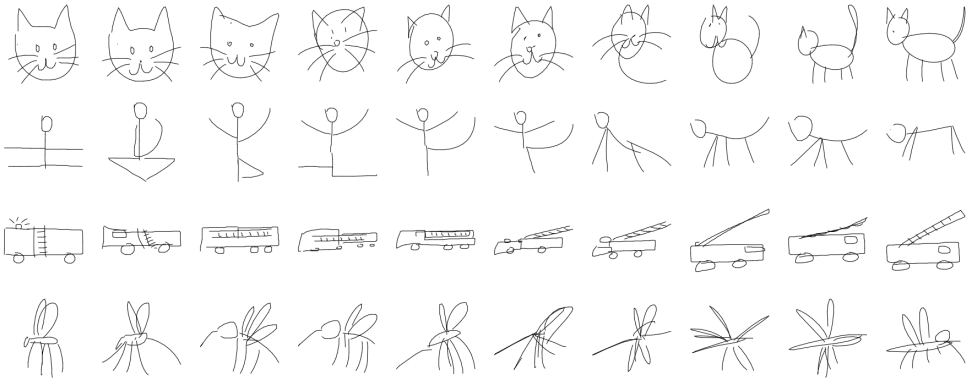
z

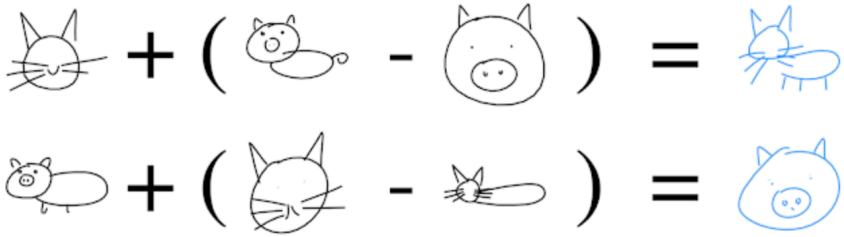


y

Perks of VAE

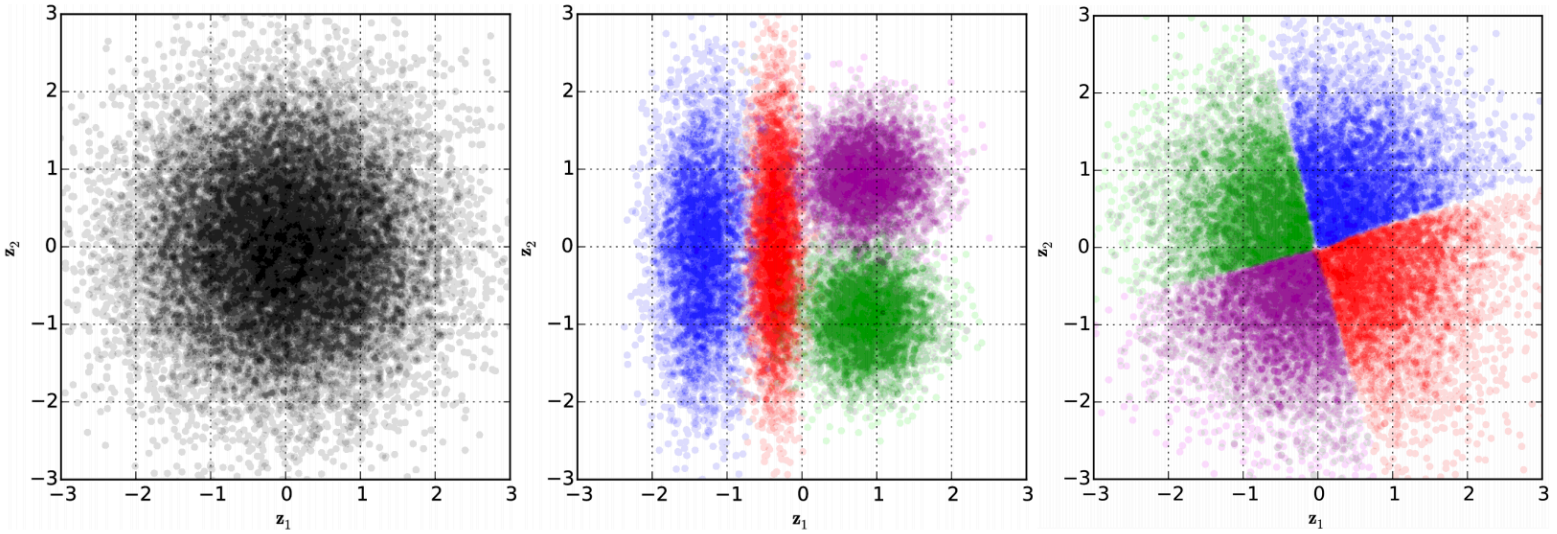
- New domains with likelihood $p(x \mid z)$ instead of reconstruction error
  - VAEs for automated molecule design *(arxiv.org/abs/1802.04364)*
  - Vector graphics with RNNs *(arxiv.org/abs/1704.03477)*





Pitfalls of VAE

- Gap in posterior approximation $q(z \mid x) \neq p(z \mid x)$ *(arxiv.org/abs/1606.04934)*



Pitfalls of VAE

- Optimisation tends to find local optima

$$L(x, \theta, \phi) = \mathbb{E}_{q(z|x)} \log p(x \mid z) - D_{KL}(q(z \mid x) \| p(z))$$

KL-term wins → Posterior collapse *(arxiv.org/abs/1911.02469)*

$$p(z) = q(z \mid x)$$

Decoder $f_\theta(z)$ does not depend on z

Reconstruction wins → “holes” in posterior *(arxiv.org/abs/1810.00597)*

Pitfalls of VAE

- Good performance may require years of architecture design

GANs:

VAE July 2020 (*arxiv.org/abs/2007.03898*):

Key Takeaways

- VAE approximates data distributions
- Decoder constitues a latent variable model
- Encoder approximates posterior for training