UNIVERSITÄT BONN AIS

Lab CudaVision
Learning Vision Systems on Graphics Cards (MA-INF 4308)

Autoencoders

29.06.2021

PROF. SVEN BEHNKE, ANGEL VILLAR-CORRALES

Contact: villar@ais.uni-bonn.de

Motivation

Motivation

- Labeled data is scarce
- Labeling is time consuming and expensive

- Category labelling in COCO:
  - 330k images
  - 91 classes

- Instance segmentation in COCO:
  - 2.5 million instances

20 s/img

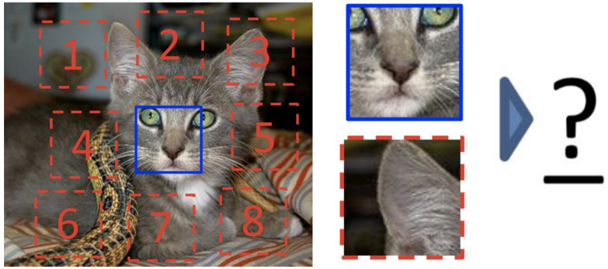
85 s/instance

amazon mechanical turk

Solution

- **Weakly supervised learning:**
  - Using labels from a related task
- **Semi-supervised learning:**
  - Using large datasets with only few labeled data
- **Unsupervised learning:**
  - Using no labeled data
- **Self-supervised learning:**
  - Learning representations on pretext tasks

Brushing teeth

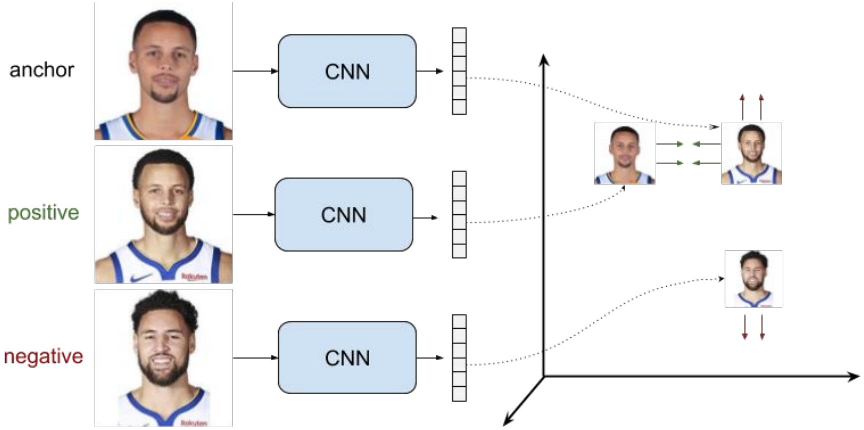


Applications

- Clustering and Similarity learning

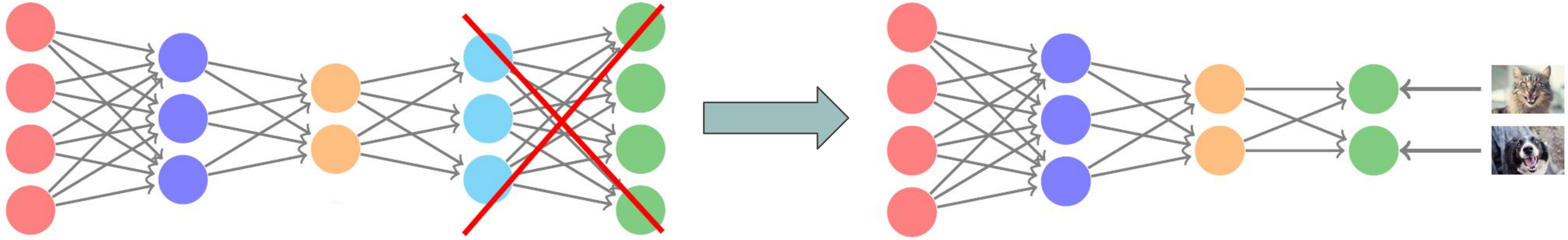
Image Clustering

Deep Metric Learning



Applications

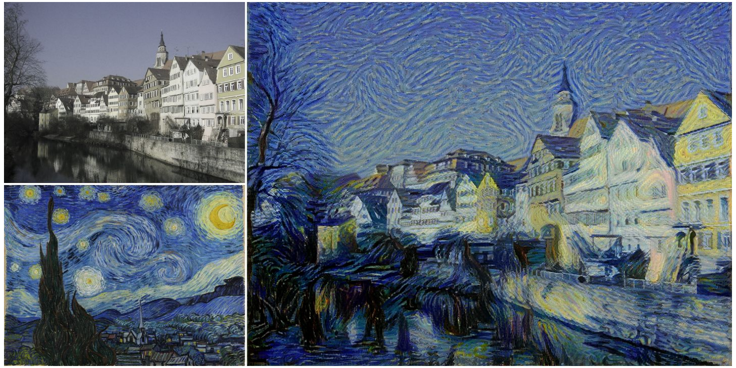
- Network initialization:
  - Model pretraining
  - Transfer learning



Applications

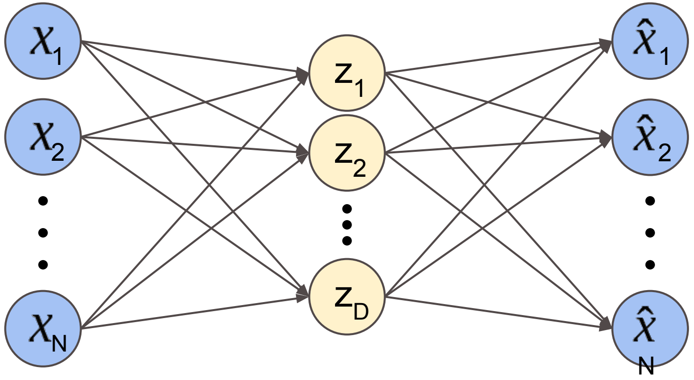
- Generative modelling
  - Generating new images
  - Image to image translation
  - Impainting and missing data

https://thispersondoesnotexist.com/



Autoencoders

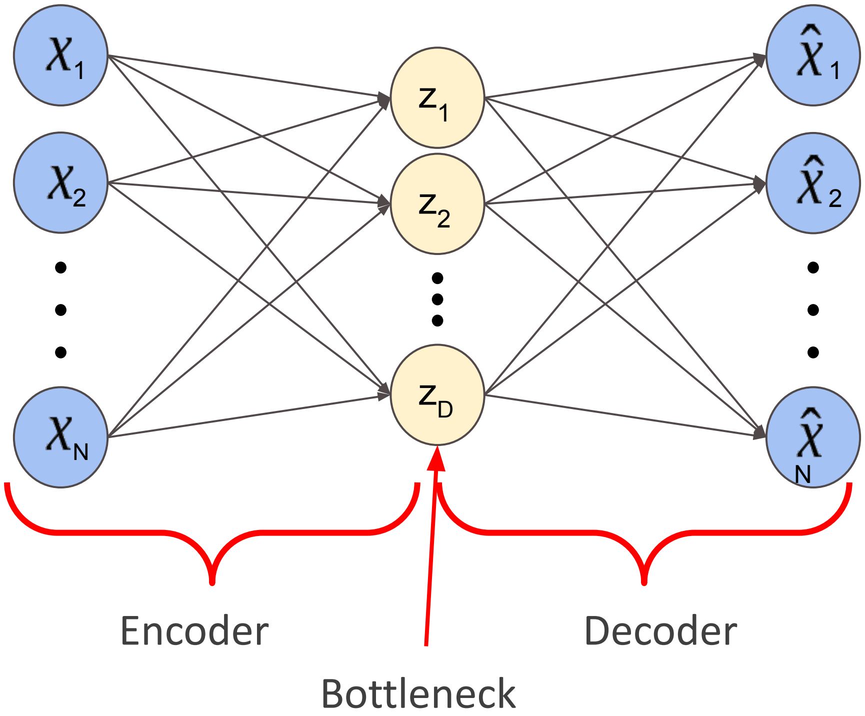
Autoencoder (AE)



- Models that are trained to predict their input
  - Dimensionality reduction
  - Representation learning

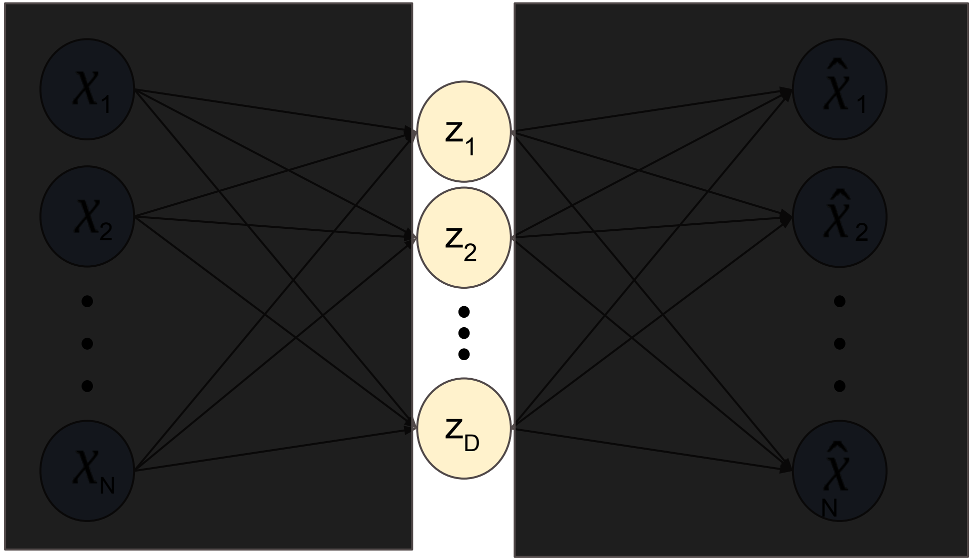
- Autoencoders learn an approximation of the identity

Autoencoder (AE)



Autoencoder (AE)

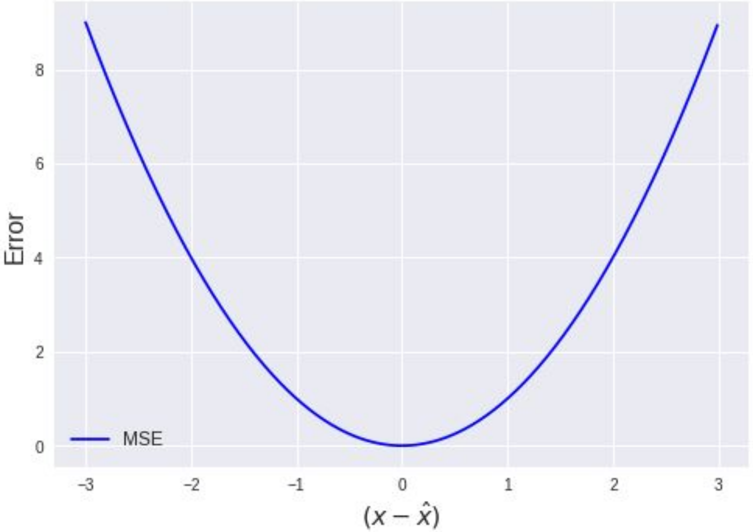
- $\mathbf{z} = E(\mathbf{X})$
- $\hat{\mathbf{X}} = D(\mathbf{z})$
- $\hat{\mathbf{X}} = D(E(\mathbf{X}))$



Training Autoencoders

- AEs are often trained with regression losses

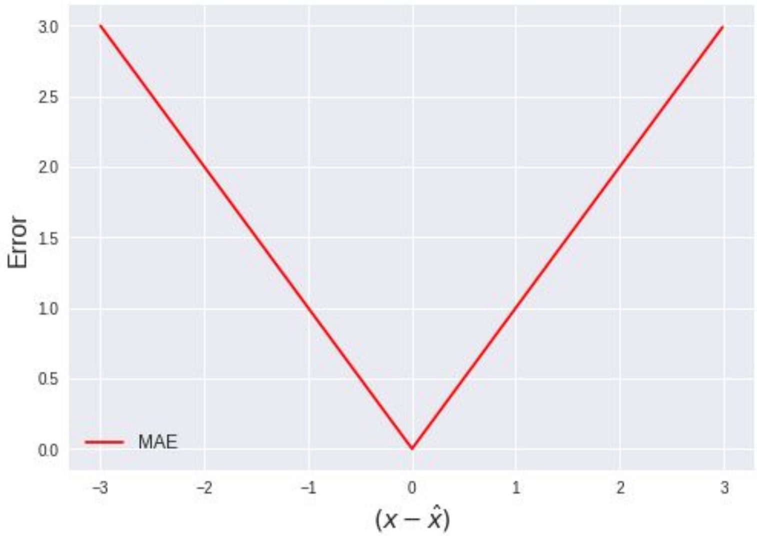
$$MSE = \frac{1}{N}\sum_{i}^{N}(\mathbf{X}_i - \hat{\mathbf{X}}_i)^2$$



Training Autoencoders

- AEs are often trained with regression losses

$$MAE = \frac{1}{N}\sum_{i}^{N} |\mathbf{X}_i - \hat{\mathbf{X}}_i|$$



Training Autoencoders

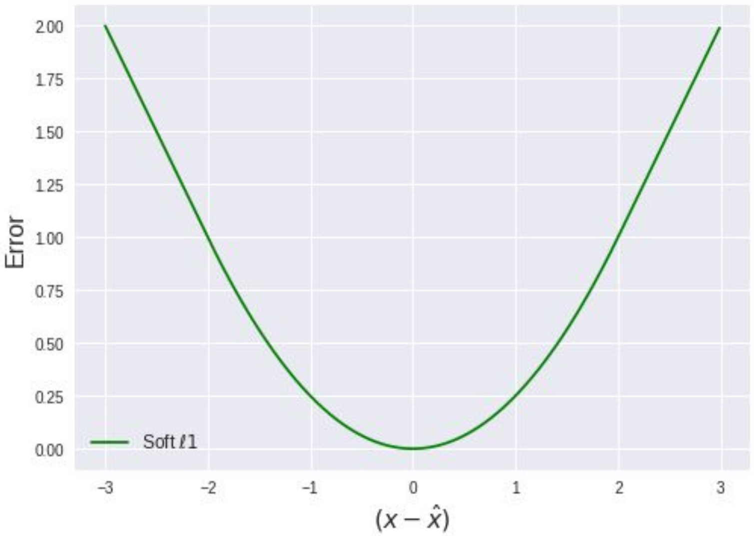
- AEs are often trained with regression losses

$$\text{Smooth}\ \ell 1 = \frac{1}{N}\sum_{i}^{N} l_i$$

MSE for small errors

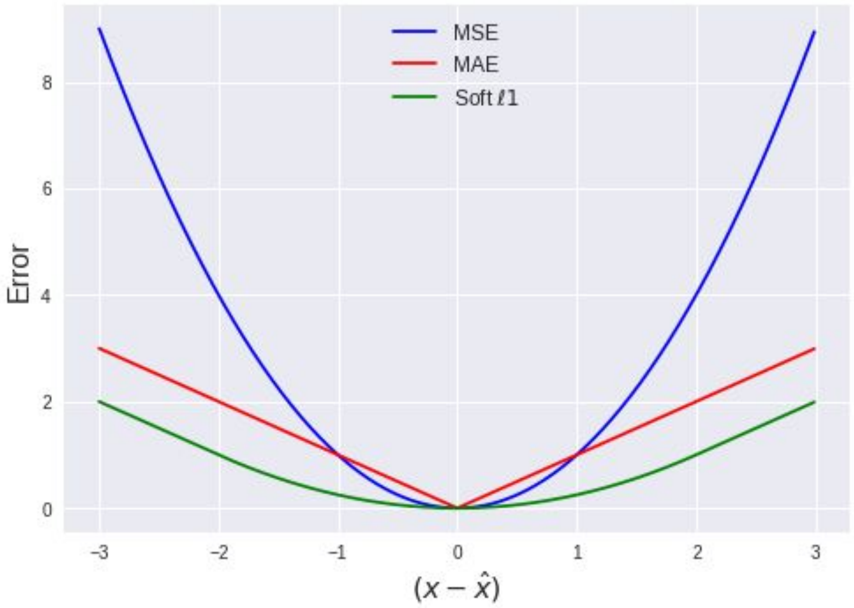
$$li = \begin{cases} \frac{1}{2\cdot\beta}(\mathbf{X}_i - \hat{\mathbf{X}}_i)^2 & |\mathbf{X}_i - \hat{\mathbf{X}}_i| \leq \beta \\ |\mathbf{X}_i - \hat{\mathbf{X}}_i| - 0.5\cdot\beta & \text{otherwise} \end{cases}$$

MAE for larger errors



Training Autoencoders

- $$MSE = \frac{1}{N}\sum_{i}^{N}(\mathbf{X}_i - \hat{\mathbf{X}}_i)^2$$
- $$MAE = \frac{1}{N}\sum_{i}^{N}|\mathbf{X}_i - \hat{\mathbf{X}}_i|$$
- $$\text{Smooth}\,\ell 1 = \frac{1}{N}\sum_{i}^{N} l_i$$

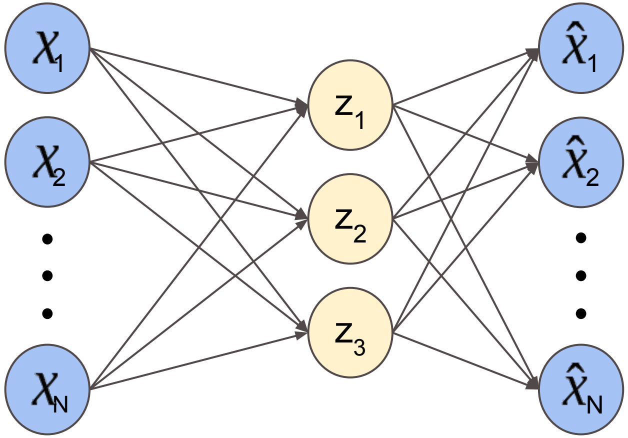


Regularizing AEs

- Without regularization, AEs learn an identity map
  - Enforce constraints on the architecture or loss

- **Undercomplete AE**
  - Low dimensional bottleneck
  - Prevents learning the identity
  - Enforces compression

- AE with one linear layer learns PCA
  - Encoder equivalent to projection matrix

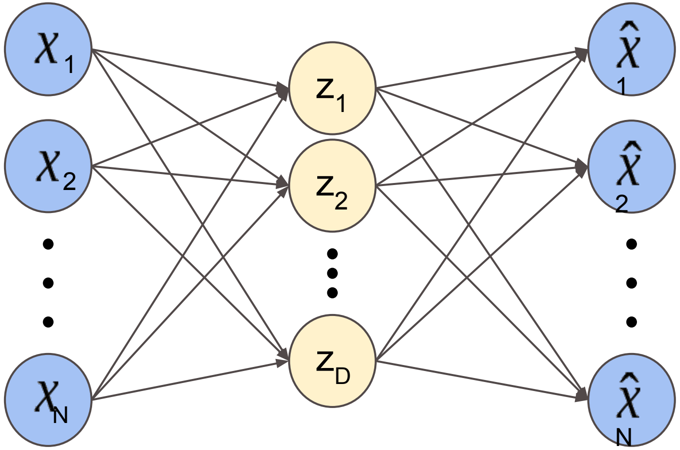


Regularizing AEs

- Without regularization, AEs learn an identity map
  - Enforce constraints on the architecture or loss

- **Sparse AE**
  - Enforce sparsity in bottleneck

$$\mathcal{L}_{\text{SAE}}(\mathbf{X}, \hat{\mathbf{X}}) = \mathcal{L}(\mathbf{X}, \hat{\mathbf{X}}) + \underbrace{\frac{1}{D}\sum_{i=1}^{D} |z_i|}_{\text{L1 reg.}}$$

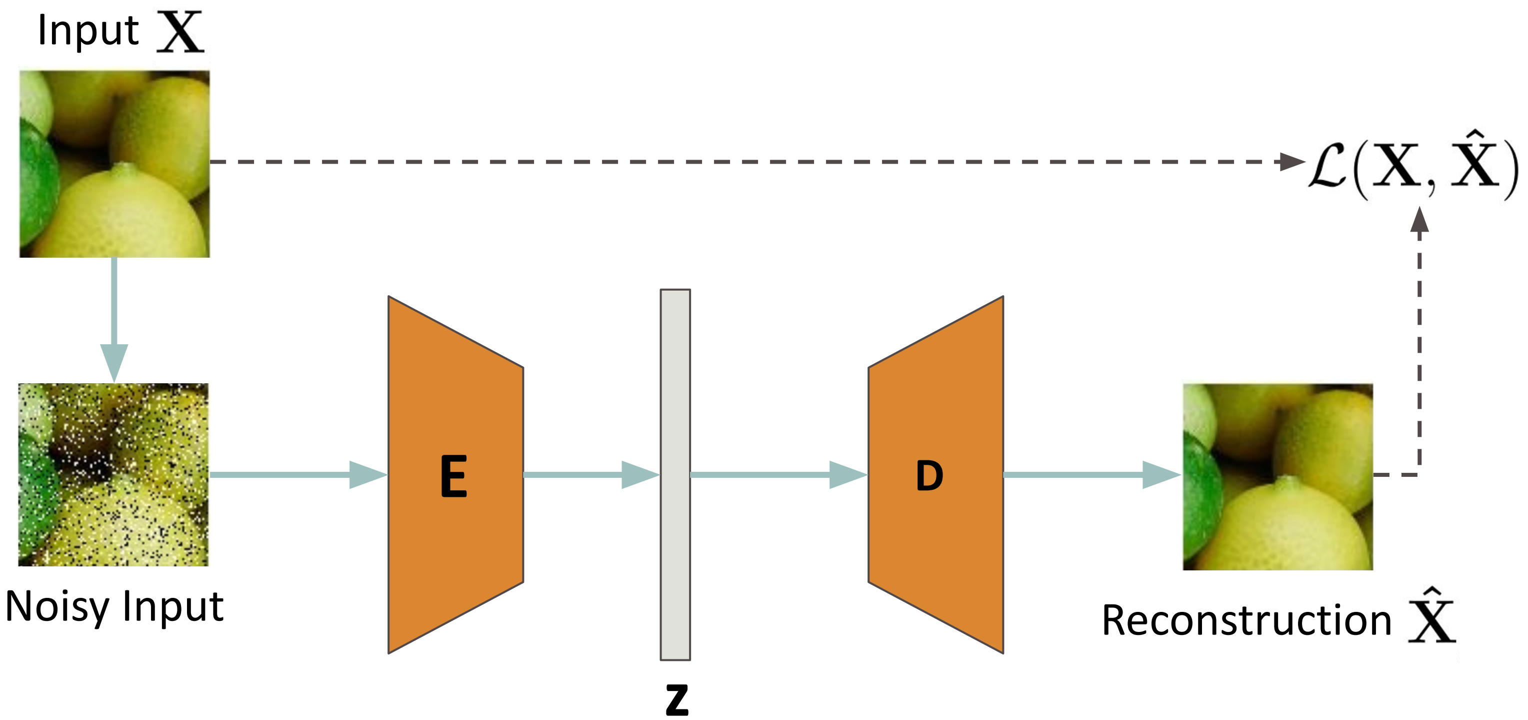


Denoising Autoencoders

Denoising Autoencoder (DAE)

- **Denoising**: *removing noise from a signal, removing as much information and features as possible*
- AEs excel at performing denoising in images
- Information bottleneck
  - Required features to reconstruct input
  - Noise does not contain information

DAE Pipeline



Let’s try it!

Variational Autoencoders

Variational Autoencoder (VAE)

- AEs compute a **deterministic** latent vector for the input

- Variational autoencoders (VAEs):
  - Describe latent space in a **probabilistic** manner

- VAEs map inputs into a probability distribution
  - Model uncertainty in the input data
  - Enforces smooth latent space

Auto-Encoding Variational Bayes

Diederik P. Kingma
Machine Learning Group
Universiteit van Amsterdam
dpkingma@gmail.com

Max Welling
Machine Learning Group
Universiteit van Amsterdam
welling.max@gmail.com

Abstract

How can we perform efficient inference and learning in directed probabilistic models, in the presence of continuous latent variables with intractable posterior distributions, and large datasets? We introduce a stochastic variational inference and learning algorithm that scales to large datasets and, under some mild differentiability conditions, even works in the intractable case. Our contributions is two-fold. First, we show that a reparameterization of the variational lower bound with an independent noise variable yields a lower bound estimator that can be jointly optimized w.r.t. variational and generative parameters using standard gradient-based stochastic optimization methods. Second, we show that posterior inference can be made especially efficient by optimizing a probabilistic encoder (also called a recognition model) to approximate the intractable posterior, using the proposed estimator. Theoretical advantages are reflected in experimental results.

1 Introduction

How can we efficiently learn the parameters of directed probabilistic models whose continuous latent variables have intractable posterior distributions? The variational approach to Bayesian inference involves the introduction of an approximation to the intractable posterior, used to maximize the variational lower bound on the marginal likelihood. Unfortunately, the common mean-field approach requires analytical solutions of expectations w.r.t. the approximate posterior, which are also intractable in the general case. We show how a reparameterization of the variational lower bound yields a practical differentiable estimator of the lower bound. This SGVB (Stochastic Gradient Variational Bayes) estimator can be straightforwardly used as a stochastic objective function, and that can be jointly optimized w.r.t. both the variational and generative parameters, using standard stochastic gradient ascent techniques.

VAE: Statistical Motivation

- **Assumption**: Sample **X** is generated by latent variable **z**
- Training VAE corresponds to determining $p(\mathbf{z}|\mathbf{X})$
  - Usually undefined and intractable
- Approximate $p(\mathbf{z}|\mathbf{X})$ by a tractable distribution $q(\mathbf{z}|\mathbf{X})$

$$\min \ \mathrm{KL}(p(\mathbf{z}|\mathbf{X})||q(\mathbf{z}|\mathbf{X}))$$

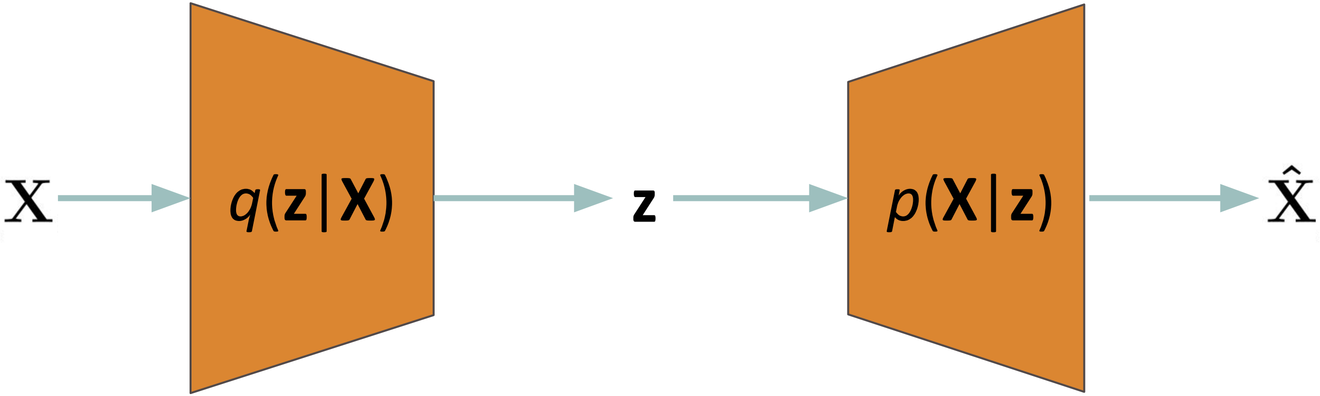
which is equivalent to

ELBO

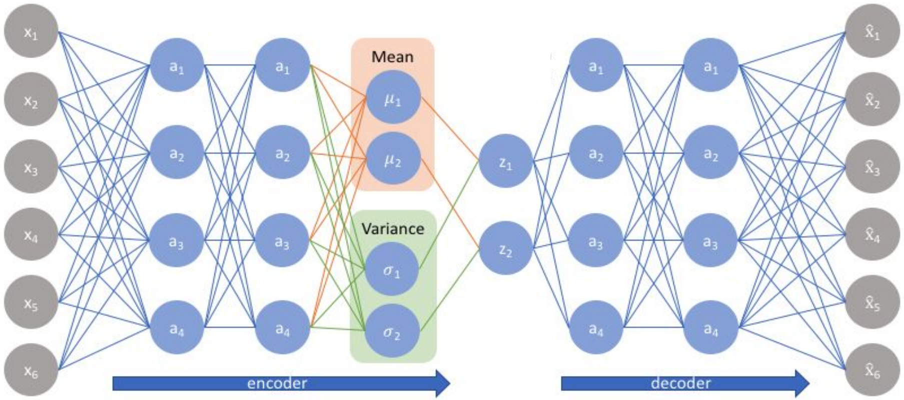
$$\max \ \underbrace{\mathrm{E}_{q(\mathbf{z}|\mathbf{X})} \log p(\mathbf{X}|\mathbf{z})}_{\text{Reconstruction likelihood}} - \underbrace{\mathrm{KL}(q(\mathbf{z}|\mathbf{X})||p(\mathbf{z}))}_{\text{Difference between } q(z|X) \text{ and true prior}}$$

VAE: Statistical Motivation

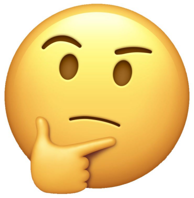
- $p(\mathbf{z})$ is often assumed to be an Isotropic Gaussian distribution
    - ➢ For determining $q(\mathbf{z}|\mathbf{X})$ we just need $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$
    - ➢ We use neural networks to estimate $q(\mathbf{z}|\mathbf{X})$ and $p(\mathbf{X}|\mathbf{z})$



VAE Training



Loss requires sampling
How do we solve this?

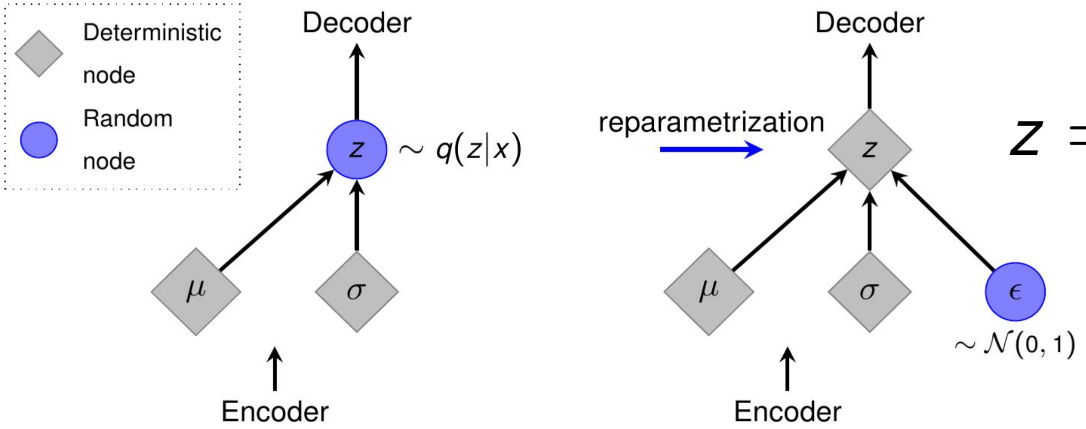


$$\mathcal{L} = \mathcal{L}_{\text{Recons}} - \mathcal{L}_{\text{VAE}}$$

$$\mathcal{L}_{\text{VAE}} = \boxed{\mathrm{E}_{q(\mathbf{z}|\mathbf{X})}} \log p(\mathbf{X}|\mathbf{z}) - \mathrm{KL}(q(\mathbf{z}|\mathbf{X})||p(\mathbf{z}))$$

Reparametrization Trick

- We cannot propagate through random sampling
- Move random sampling out of path by reparametrization
  - Backpropagation is deterministic



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

VAE as Generative Models

- New data can be generated by sampling from latent space distribution
  - Use learned mean and covariance
  - Sample from distribution
  - Reconstruct using decoder

- Diagonal covariance enforces independent latent variables

- Smooth latent space can be transversed

References

1. https://towardsdatascience.com/all-you-want-to-know-about-deep-learning-8d68dcffc258
2. Goodfellow, Ian, et al. Deep learning. Vol. 1. No. 2. Cambridge: MIT press, 2016.
3. Vincent, Pascal, et al. "Extracting and composing robust features with denoising autoencoders." Proceedings of the 25th international conference on Machine learning. 2008.
4. Alain, Guillaume, and Yoshua Bengio. "What regularized auto-encoders learn from the data-generating distribution." The Journal of Machine Learning Research 15.1 (2014): 3563-3593.
5. Doersch, Carl. "Tutorial on variational autoencoders." arXiv preprint arXiv:1606.05908 (2016).
6. Zhou, Bolei, et al. "Learning deep features for discriminative localization." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
7. https://lilianweng.github.io/lil-log/2018/08/12/from-autoencoder-to-beta-vae.html#vae-variational-autoencoder
8. https://www.jeremyjordan.me/variational-autoencoders/