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

Autoencoders

11.12.2024

PROF. SVEN BEHNKE, ANGEL VILLAR-CORRALES

Contact: villar@ais.uni-bonn.de

Learning without Supervision

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
- Semantic segmentation in Cityscapes:
  - > 5000 fully annotated frames

20 s/img

85 s/instance

1.5 hours/img

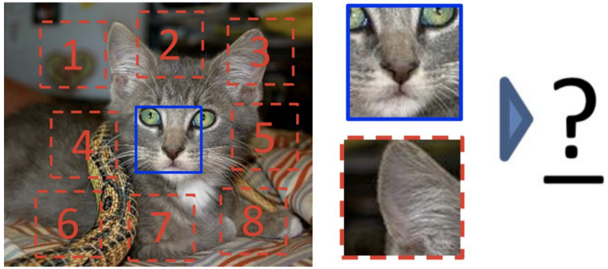
amazon mechanical turk

Solution

- Weak supervision or no supervision

- **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

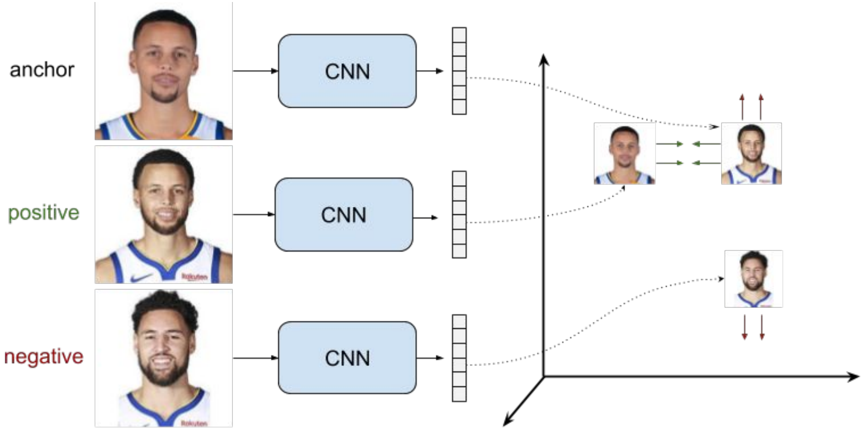


Applications

- Clustering and Similarity learning

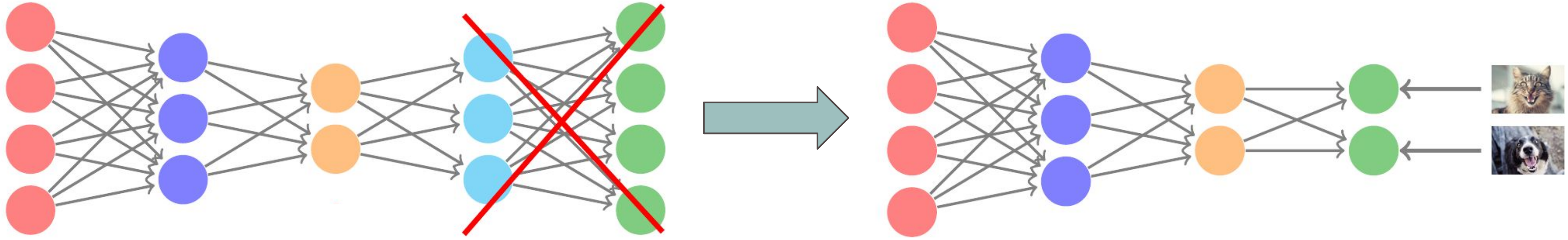
Image Clustering

Deep Metric Learning



Applications

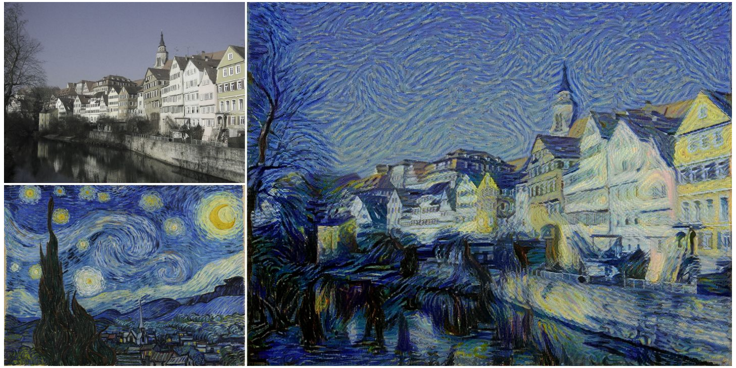
- Network initialization:
  - Model pretraining
  - Transfer learning



Applications

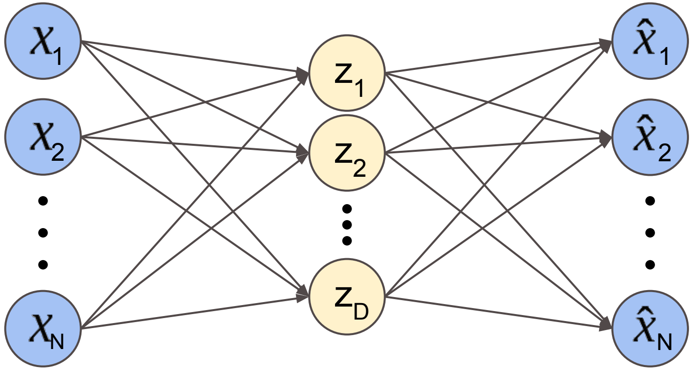
- Generative modelling
  - Generating new images
  - Image to image translation
  - Impainting and missing data

https://thispersondoesnotexist.com/



Autoencoders

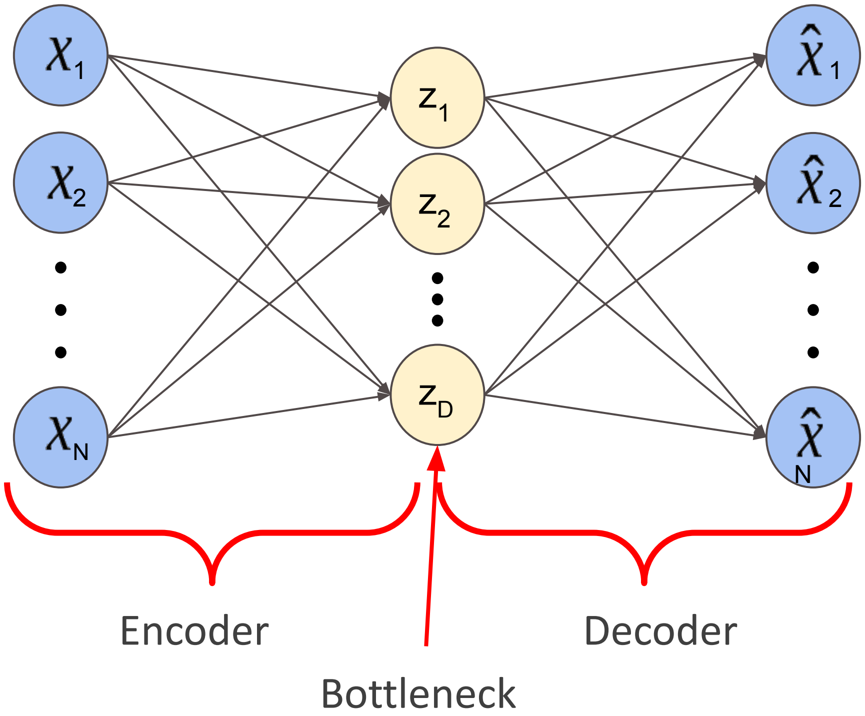
Autoencoder (AE)



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

- Autoencoders learn an approximation of the identity

Autoencoder (AE)

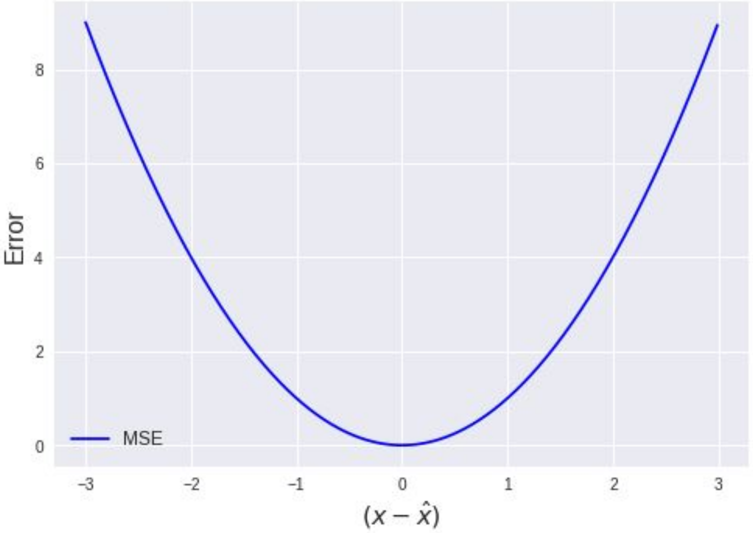


- $\mathbf{z} = E(\mathbf{X})$
- $\hat{\mathbf{X}} = D(\mathbf{z})$
- $\hat{\mathbf{X}} = D(E(\mathbf{X}))$

Training Autoencoders

- AEs are often trained with regression losses

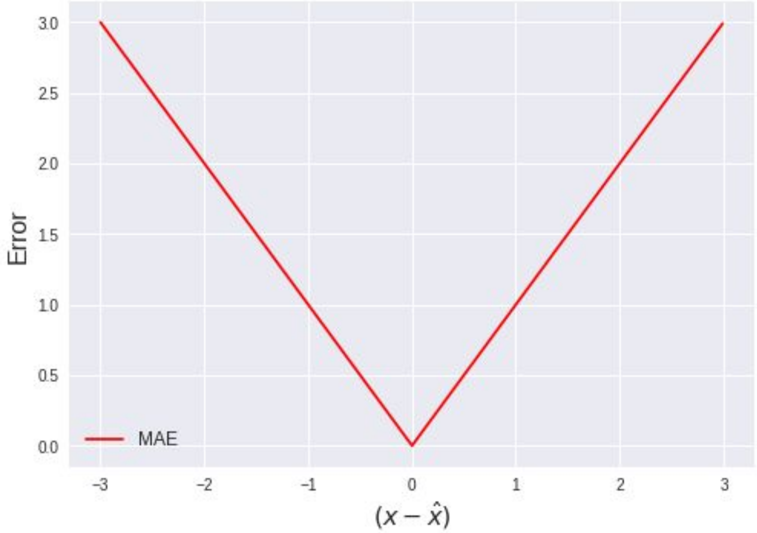
$$MSE = \frac{1}{N}\sum_{i}^{N}(\mathbf{X}_i - \hat{\mathbf{X}}_i)^2$$



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

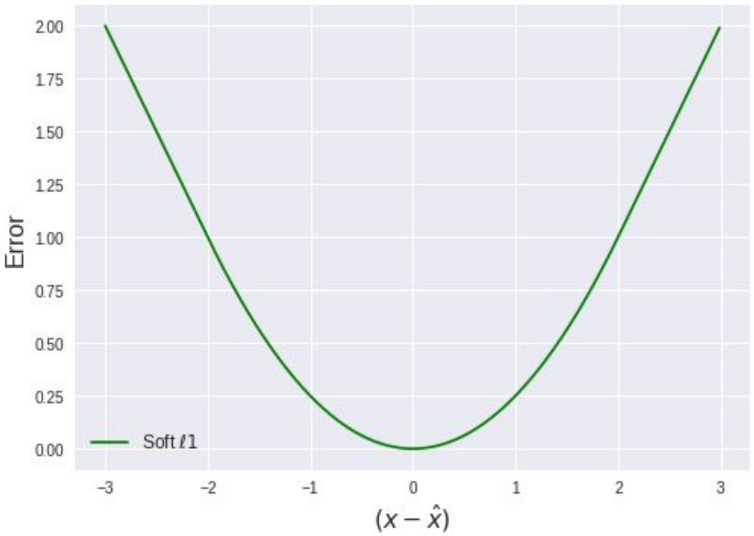
- AEs are often trained with regression losses

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

MSE for small errors

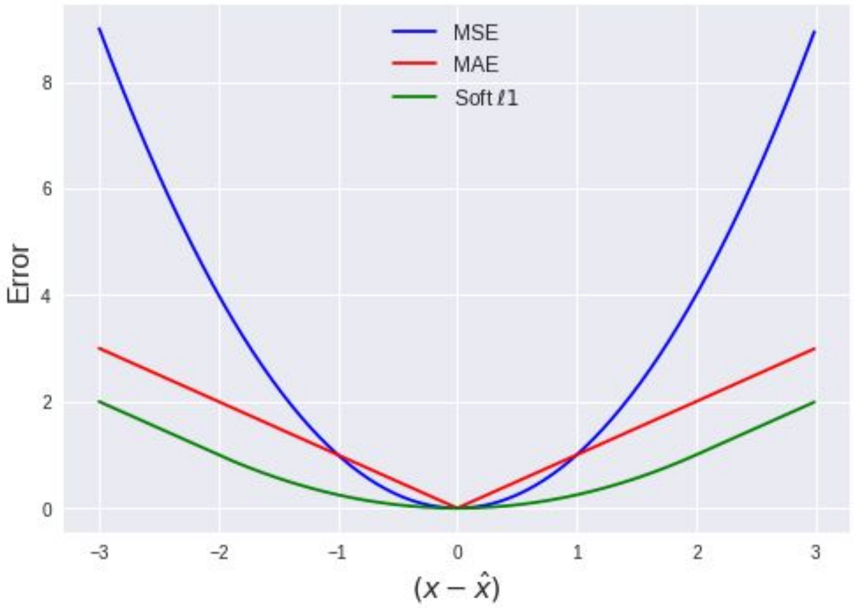
$$li = \begin{cases} \frac{1}{2 \cdot \beta} (\mathbf{X}_i - \hat{\mathbf{X}}_i)^2 & |\mathbf{X}_i - \hat{\mathbf{X}}_i| \leq \beta \\ |\mathbf{X}_i - \hat{\mathbf{X}}_i| - 0.5 \cdot \beta & \text{otherwise} \end{cases}$$

MAE for larger errors



Training Autoencoders

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

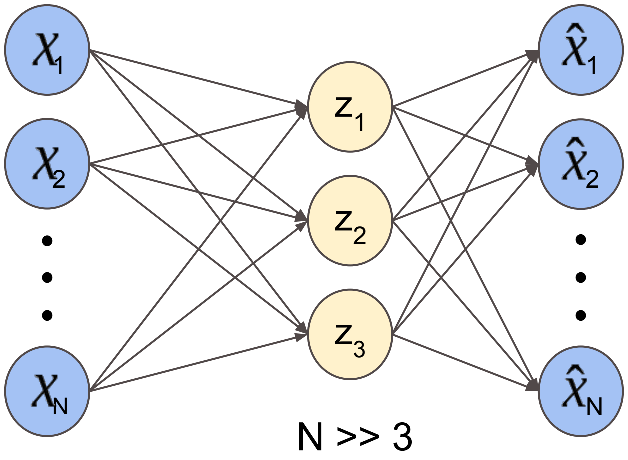


Regularizing AEs

- 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

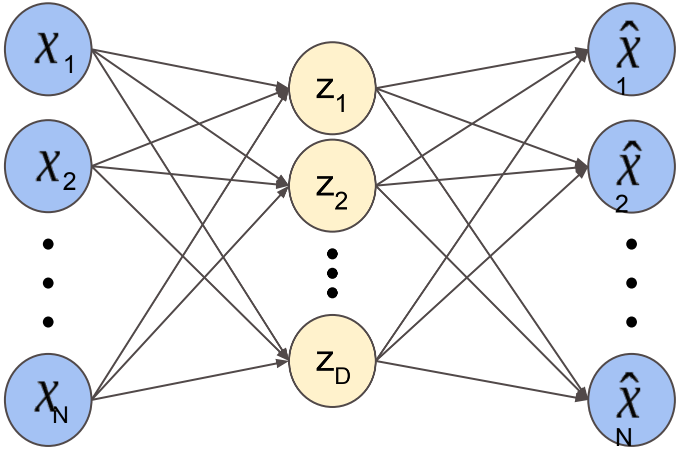


Regularizing AEs

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

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

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

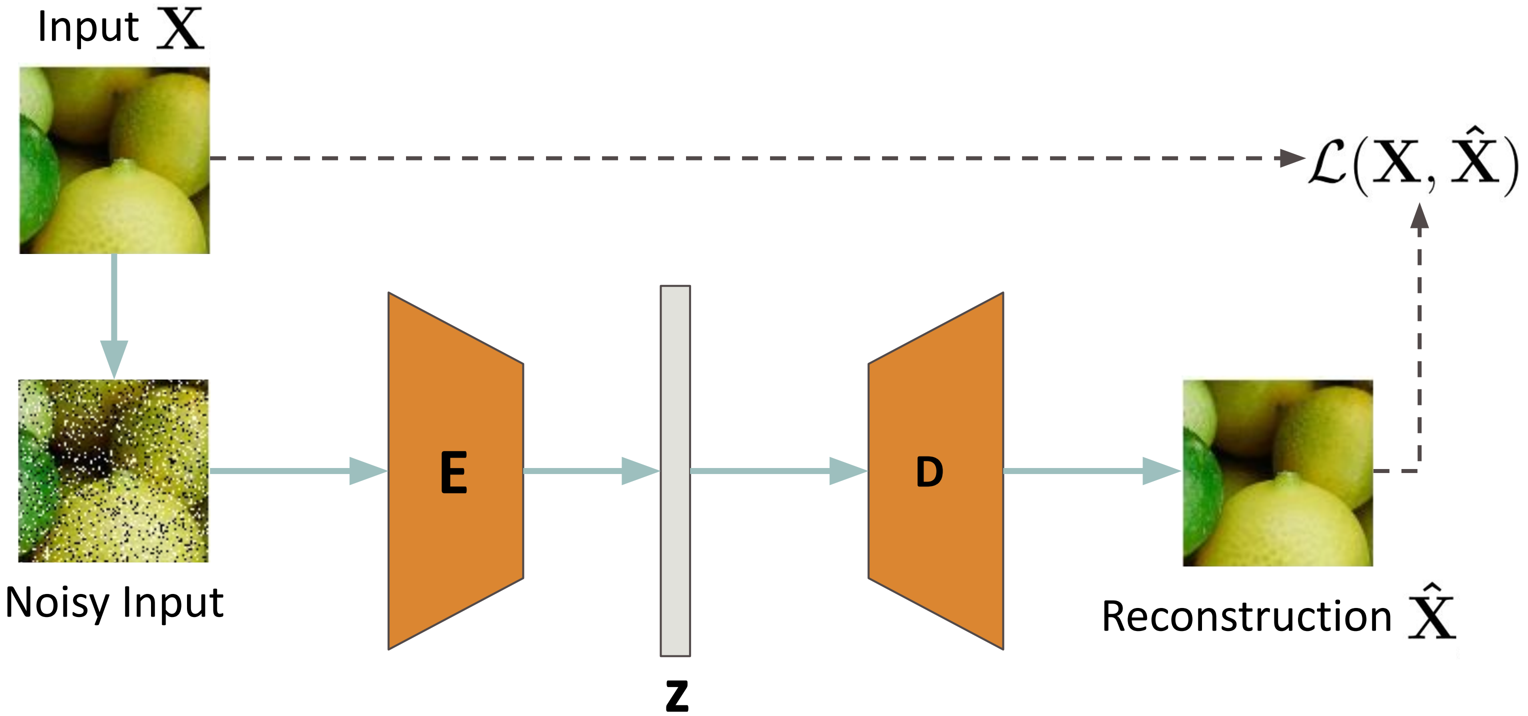


Denoising Autoencoders

Denoising Autoencoder (DAE)

- **Denoising**: *removing noise from a signal, keeping 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
- Pretext task for learning robust representations

DAE Pipeline

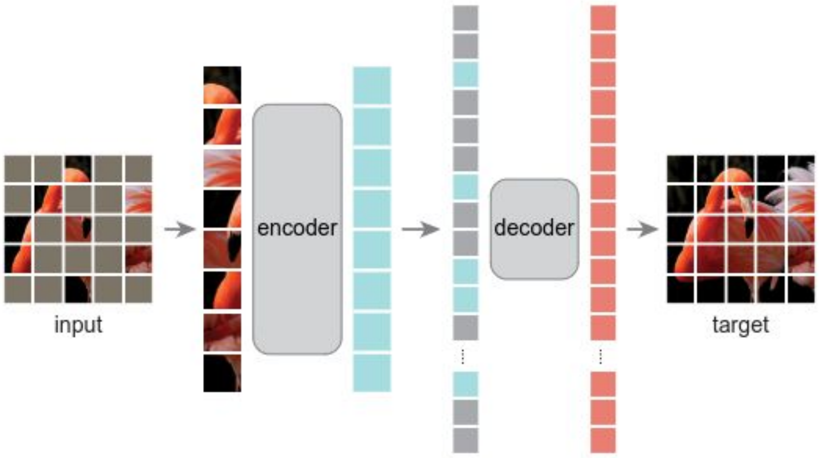


Masked Autoencoders (MAE)

- State-of-the-art self-supervised pretraining algorithm
- Pretrain large transformers using images only

method	pre-train data	ViT-B	ViT-L	ViT-H
scratch, our impl.	-	82.3	82.6	83.1
DINO [5]	IN1K	82.8	-	-
MoCo v3 [9]	IN1K	83.2	84.1	-
BEiT [2]	IN1K+DALLE	83.2	85.2	-
MAE	IN1K	83.6	85.9	86.9

1. Split image into patches
2. Throw away many patches (e.g. 80%)
3. Encode only the unmasked patches with deep transformer encoder (very efficient)
4. Decode embeddings of masked & unmasked patches with shallow transformer decoder
5. Compute recons. loss on masked patches



Let’s try it!

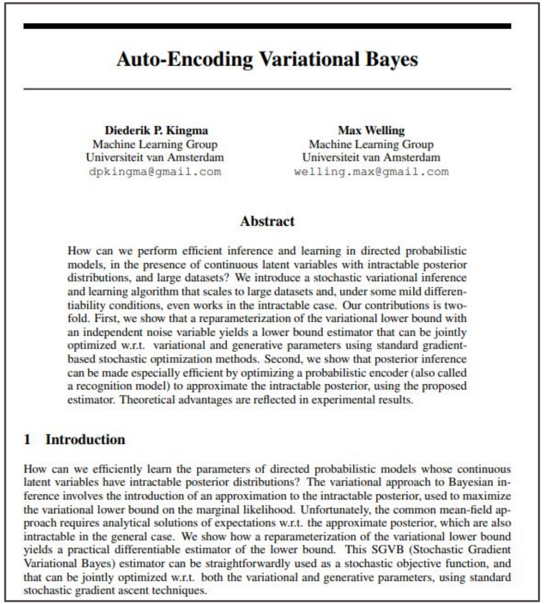
Variational Autoencoders

Variational Autoencoder (VAE)

- AEs compute a **deterministic** latent vector for the input
  - Unstructured latent space
  - Deterministic mapping
  - ➢ Cannot be used to generate new data!

- 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 decoding 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

Image recons. from latent code

Approx.distribution

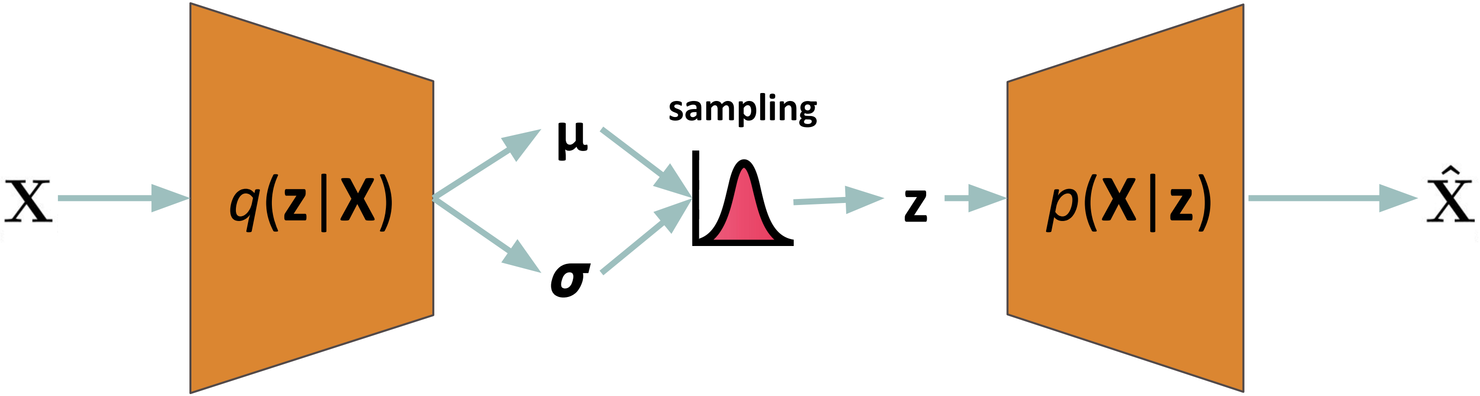
Actual distribution of the latent space

ELBO

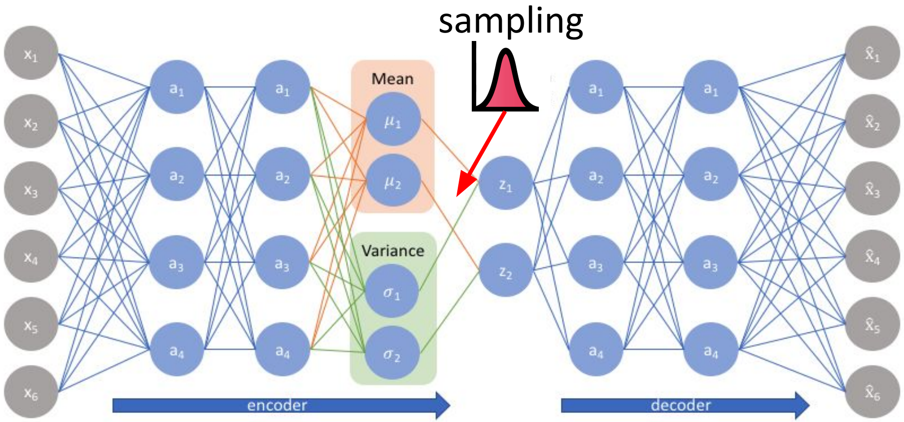
$$\max \ \underbrace{\mathrm{E}_{q(\mathbf{z}|\mathbf{X})} \log p(\mathbf{X}|\mathbf{z})}_{\text{Reconstruction likelihood}} - \underbrace{\mathrm{KL}(q(\mathbf{z}|\mathbf{X})||p(\mathbf{z}))}_{\text{Difference between } q(z|X) \text{ and true prior}}$$

VAE: Statistical Motivation

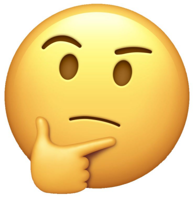
- $p(\mathbf{z})$ is often assumed to be an Isotropic Gaussian distribution
  - For determining $q(\mathbf{z}|\mathbf{X})$ we just need $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$
  - We use neural networks to estimate $q(\mathbf{z}|\mathbf{X})$ and $p(\mathbf{X}|\mathbf{z})$



VAE Training



Loss requires sampling
How do we solve this?

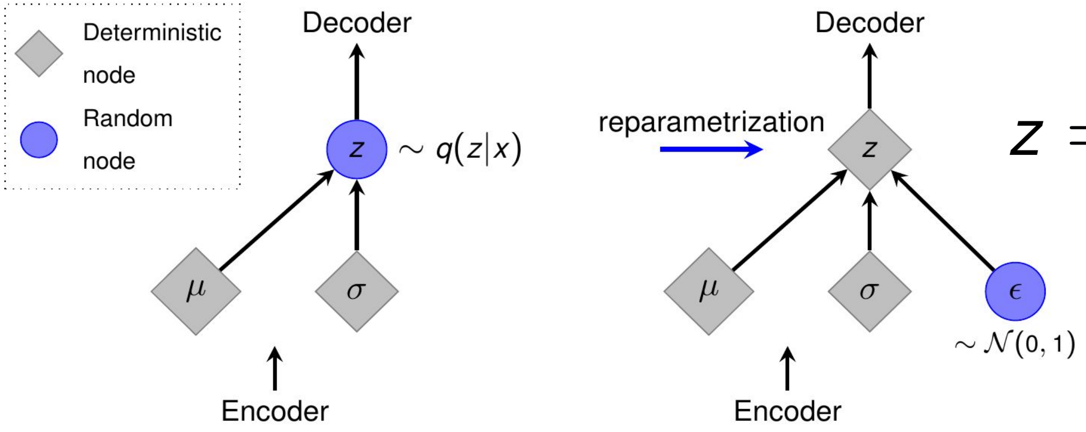


$$\mathcal{L}_{\text{VAE}} = \mathcal{L}_{\text{Recons}} - \mathcal{L}_{\text{KL}}$$

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

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

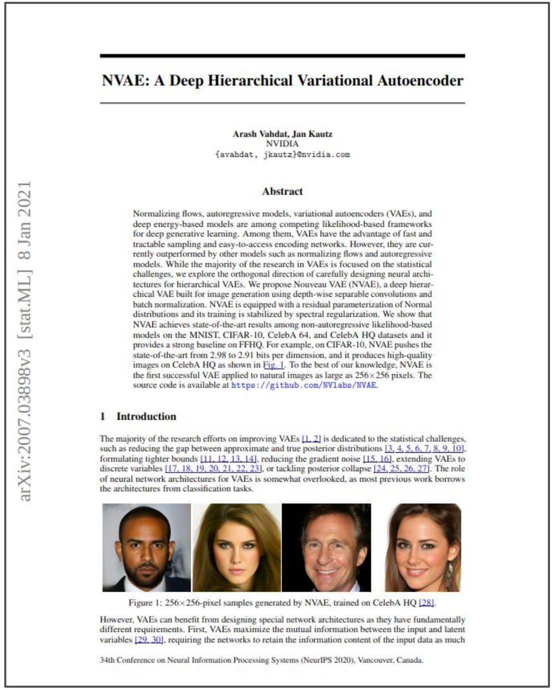
- New data can be generated by sampling from latent space distribution
  - Use learned mean and covariance
  - Sample from distribution
  - Reconstruct using decoder

- Diagonal covariance enforces independent latent variables

- Smooth latent space can be transversed

What can we do with modern VAEs?

Image Generation



arXiv:2007.03898v3 [stat.ML] 8 Jan 2021

NVAE: A Deep Hierarchical Variational Autoencoder

Arash Vahdat, Jan Kautz
NVIDIA
{avahdat, jkautz}@nvidia.com

Abstract

Normalizing flows, autoregressive models, variational autoencoders (VAEs), and deep energy-based models are among competing likelihood-based frameworks for deep generative learning. Among them, VAEs have the advantage of fast and tractable sampling and easy-to-access encoding networks. However, they are currently outperformed by other models such as normalizing flows and autoregressive models. While the majority of the research in VAEs is focused on the statistical challenges, we explore the orthogonal direction of carefully designing neural architectures for hierarchical VAEs. We propose Nouveau VAE (NVAE), a deep hierarchical VAE built for image generation using depth-wise separable convolutions and batch normalization. NVAE is equipped with a residual parameterization of Normal distributions and its training is stabilized by spectral regularization. We show that NVAE achieves state-of-the-art results among non-autoregressive likelihood-based models on the MNIST, CIFAR-10, CelebA 64, and CelebA HQ datasets and it provides a strong baseline on FFHQ. For example, on CIFAR-10, NVAE pushes the state-of-the-art from 2.98 to 2.91 bits per dimension, and it produces high-quality images on CelebA HQ as shown in Fig. 1. To the best of our knowledge, NVAE is the first successful VAE applied to natural images as large as 256×256 pixels. The source code is available at https://github.com/NVlabs/NVAE.

1 Introduction

The majority of the research efforts on improving VAEs [1, 2] is dedicated to the statistical challenges, such as reducing the gap between approximate and true posterior distributions [3, 4, 5, 6, 7, 8, 9, 10], formulating tighter bounds [11, 12, 13, 14], reducing the gradient noise [15, 16], extending VAEs to discrete variables [17, 18, 19, 20, 21, 22, 23], or tackling posterior collapse [24, 25, 26, 27]. The role of neural network architectures for VAEs is somewhat overlooked, as most previous work borrows the architectures from classification tasks.

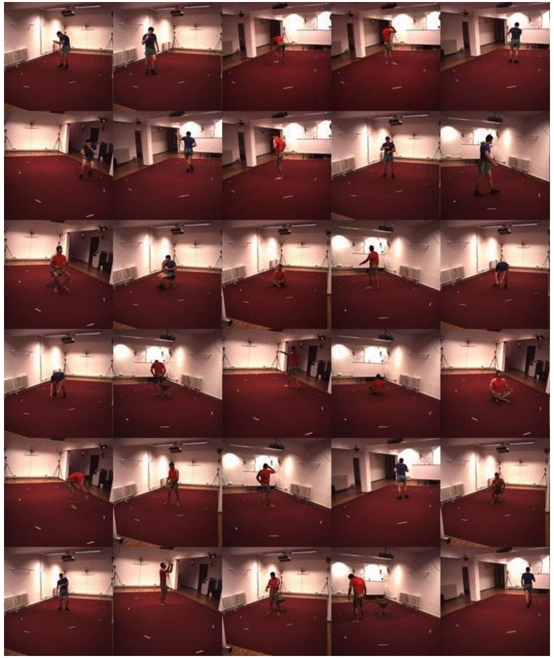
Figure 1: 256×256-pixel samples generated by NVAE, trained on CelebA HQ [28].

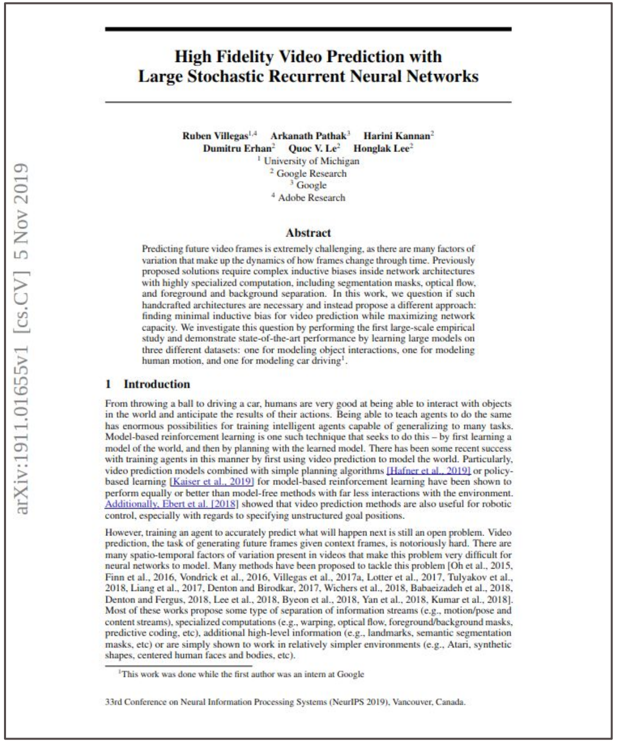
However, VAEs can benefit from designing special network architectures as they have fundamentally different requirements. First, VAEs maximize the mutual information between the input and latent variables [29, 30], requiring the networks to retain the information content of the input data as much

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

What can we do with modern VAEs?

Video Prediction





arXiv:1911.01655v1 [cs.CV] 5 Nov 2019

High Fidelity Video Prediction with Large Stochastic Recurrent Neural Networks

Ruben Villegas[1,4] Arkanath Pathak[3] Harini Kannan[2]
Dumitru Erhan[2] Quoc V. Le[2] Honglak Lee[2]
[1] University of Michigan
[2] Google Research
[3] Google
[4] Adobe Research

Abstract

Predicting future video frames is extremely challenging, as there are many factors of variation that make up the dynamics of how frames change through time. Previously proposed solutions require complex inductive biases inside network architectures with highly specialized computation, including segmentation masks, optical flow, and foreground and background separation. In this work, we question if such handcrafted architectures are necessary and instead propose a different approach: finding minimal inductive bias for video prediction while maximizing network capacity. We investigate this question by performing the first large-scale empirical study and demonstrate state-of-the-art performance by learning large models on three different datasets: one for modeling object interactions, one for modeling human motion, and one for modeling car driving[1].

1 Introduction

From throwing a ball to driving a car, humans are very good at being able to interact with objects in the world and anticipate the results of their actions. Being able to teach agents to do the same has enormous possibilities for training intelligent agents capable of generalizing to many tasks. Model-based reinforcement learning is one such technique that seeks to do this – by first learning a model of the world, and then by planning with the learned model. There has been some recent success with training agents in this manner by first using video prediction to model the world. Particularly, video prediction models combined with simple planning algorithms [Hafner et al., 2019] or policy-based learning [Kaiser et al., 2019] for model-based reinforcement learning have been shown to perform equally or better than model-free methods with far less interactions with the environment. Additionally, Ebert et al. [2018] showed that video prediction methods are also useful for robotic control, especially with regards to specifying unstructured goal positions.

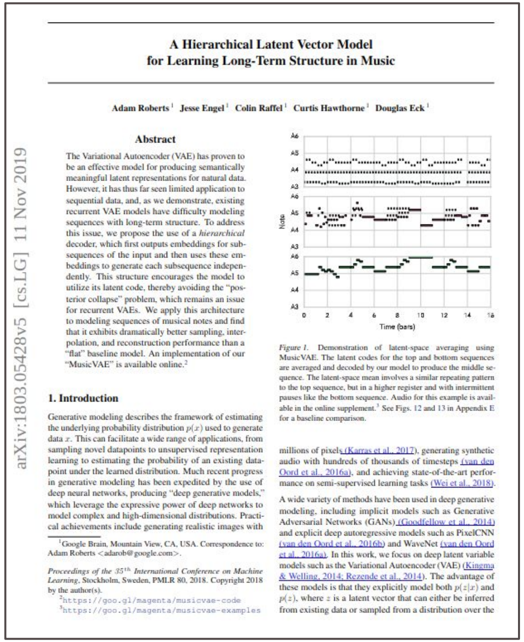
However, training an agent to accurately predict what will happen next is still an open problem. Video prediction, the task of generating future frames given context frames, is notoriously hard. There are many spatio-temporal factors of variation present in videos that make this problem very difficult for neural networks to model. Many methods have been proposed to tackle this problem [Oh et al., 2015, Finn et al., 2016, Vondrick et al., 2016, Villegas et al., 2017a, Lotter et al., 2017, Tulyakov et al., 2018, Liang et al., 2017, Denton and Birodkar, 2017, Wichers et al., 2018, Babaeizadeh et al., 2018, Denton and Fergus, 2018, Lee et al., 2018, Byeon et al., 2018, Yan et al., 2018, Kumar et al., 2018]. Most of these works propose some type of separation of information streams (e.g., motion/pose and content streams), specialized computations (e.g., warping, optical flow, foreground/background masks, predictive coding, etc), additional high-level information (e.g., landmarks, semantic segmentation masks, etc) or are simply shown to work in relatively simpler environments (e.g., Atari, synthetic shapes, centered human faces and bodies, etc).

[1]This work was done while the first author was an intern at Google

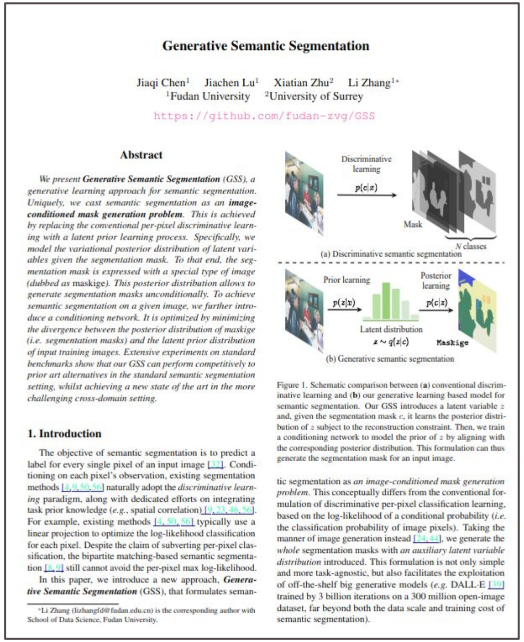
33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

What can we do with modern VAEs?

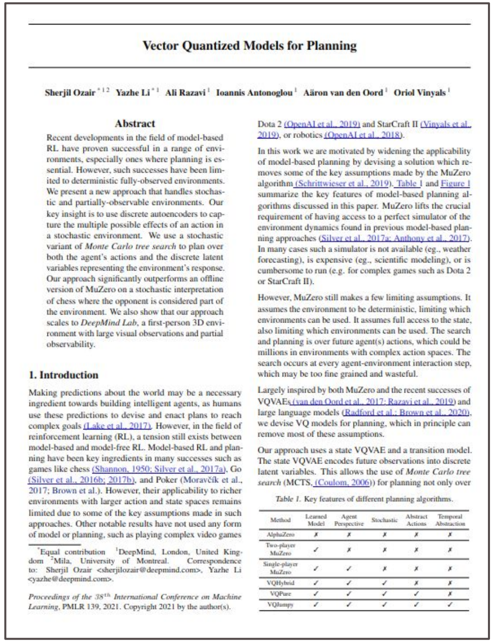
Audio/Music Generation



Vision Tasks



Planning



and much more!

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/