A brief outlook

Leander Kurscheidt

May 4, 2018

We normally learn in batches

GANs

Introduction to generative models

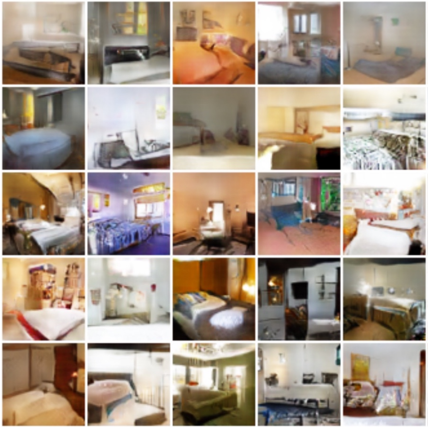


Figure: Samples from a Wasserstein-GAN, Selection from a figure from [**ArjovskyCB17**]

Introduction to generative models

In our context an (optimal) generative model is a function

$$G : P_z \to P_{data}$$

where P_z is an arbitrary distribution, often called noise-distribution. P_{data} is the distribution we want to sample from.

Generative Adversarial Networks

as formulated by [**goodfellow14**]:

discriminative model $D(x; \theta_d)$ tries to:

- assign 1 to elements of the original data
- assign 0 to elements produced by the generator
- *"tries distinguish real and generated samples"*

generative model $G(z; \theta_g)$ tries to:

- maximize $D(G(z))$
- *"tries to generate samples that fool the discriminator"*

GANs converge to a Nash-Equilibrium, as shown by [**HeuselRUNKH17**].

Generative Adversarial Networks

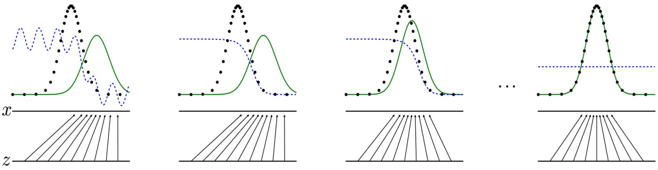


Figure: Generative Adversarial Networks over time, Figure from [**goodfellow14**]

Generative Adversarial Networks

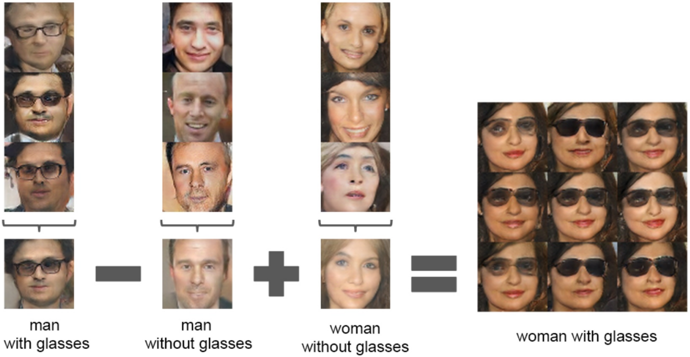


Figure: vector arithmetic on the generators input, Figure from [**Radford15**]

Generative Adversarial Networks

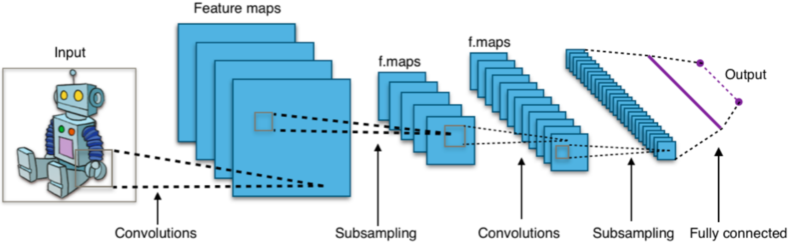
Figure: Learning cross domain relations between shoes and handbags, Figure from [**Kim17**]

Comparison to usual neural networks

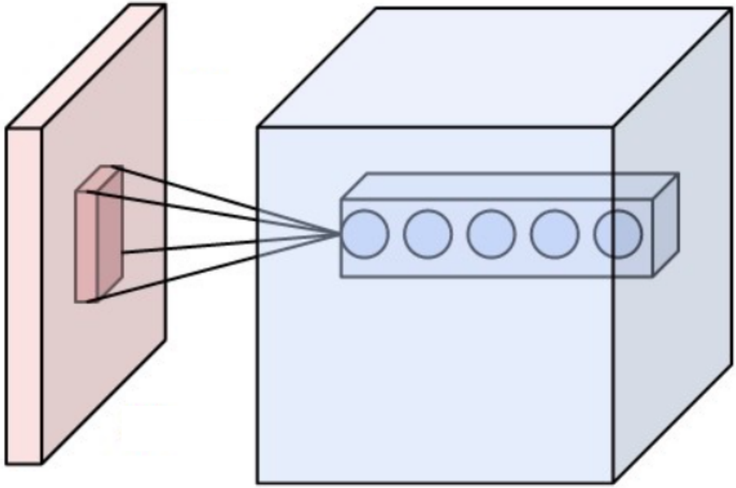
the traditional approach:
While other approaches exist (for example variational autoencoders [**KingmaW13**]), most of the applications of neural networks are of an discriminative nature, for example classification. Generative models open up exciting new possibilites.

Convolutional Neural Networks

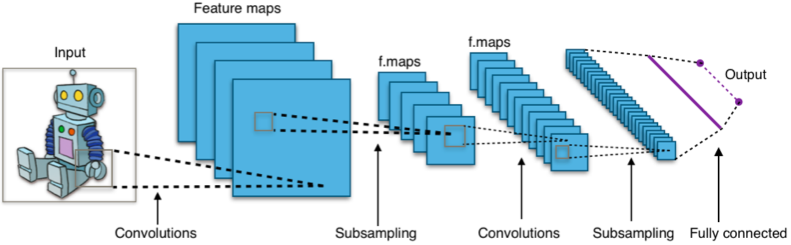
Convolutional Neural Networks



Convolutional Neural Networks



Convolutional Neural Networks



References I