# Contents

# 1 Prerequisites on Learning Disentanglement

## 1.1 Learning from Data

Learning from data is commonly understood as the ability of algorithms to improve their performance on a task with experience accumulated from the observation of data [1]. The source of data is usually a dataset - set of data points $X = \{x_i | i \in \{1 \dots n\}\}$, which are sampled from a probability distribution $x_i \sim p(x)$.

### 1.1.1 Supervised

The term supervised learning denotes the task to learn a mapping from data points $x_i$ to target labels $y_i$. A supervised algorithm has access to data-label pairs $(y_i, x_i) \sim p(y, x)$, in order to estimate the connection between data points and labels, either in form of a conditional probability $p(y|x)$, or in form of a deterministic function $y = f(x)$. The label $y$ can be either discrete (*e.g.* information about an object class) or continuous (*e.g.* the location of an object part in an image). Recent advances, in particular the effectiveness of neural network models (cf. sec. 1.1.3) on big datasets, have led to huge progress on problems that can be formulated as regression or classification. That is why on many traditional computer vision problems, such as *e.g.* object recognition, image classification or human pose estimation, machines are now performing on a superhuman level; hence, these problems are now considered to be essentially solved.
The Achilles' heel of supervised learning lies in the need for a viable supervision signal. To get labels, it is usually required to manually annotate the data. The human effort in this is costly, error-prone and not scalable to the ever-growing vast amounts of raw data.

### 1.1.2 Unsupervised

Unsupervised learning is the endeavour to learn about structures and patterns in unlabelled data. In this paradigm, the learning algorithm has access to the samples of the data distribution $x \sim p(x)$. The task is usually framed as a form of density estimation, *i.e.* to model the entire distribution in a probabilistic generative model (cf. sec. 1.2). Unsupervised learning is considered much harder than supervised learning. There are several complication in the design of unsupervised algorithms:

- Naturally, without supervision the goal of learning is not specified, hence surrogate objectives have to be formulated. The lack of specification renders the evaluation often arbitrary and subjective.

- It is a priori not clear, how much a priori knowledge should be embedded. To introduce no artificial bias, the goal is often to be purely data-driven. Others argue for the importance of certain inductive priors to guide learning [2]. A related modeling choice is, if the algorithm should be model-free or model-based.

### 1.1.3 Artificial Neural Networks

Artificial neural networks are a powerful and flexible tool for function approximation. In their principles they are inspired by biological neural networks. An artificial network is a model for a function $y = f(x)$ with vector input $x = \{x_i | i = 1 \dots n\}$ and vector output $y = \{y_j | j = 1 \dots m\}$:

$$\begin{aligned} h_j &= a(\sum_i w_{ji} x_i + b_i) \\ y_j &= a'(\sum_i w'_{ji} h_i + b'_i) \end{aligned} \tag{1.1}$$

, with weight matrices $w, w'$, non-linear so-called activation functions $a, a'$ (*e.g.* $a(x) = 0$ for $x < 0$, $a(x) = x$ for $x >= 0$) and bias vectors $b, b'$. The components $h_j$ are called hidden units or neurons. Neural networks can also comprise multiple hidden layers via $h_j = a(\sum_i w_{ji} h_i + b_i)$. It can be shown theoretically, that in the limit of infinite hidden units $h_j$ they can approximate any (continuous) function arbitrarily close [3, 4] In practice, however, networks with more that one layer, referred to as deep neural networks, seem to work better.

For processing image data, one constrains the weight matrices to be only locally connected and to share weights across locations to enforce translation invariance, resulting in *convolutional* neural networks.

longstanding model gained hype-status as working together optimization via gradient descent has proven successful (for deep networks called backpropagation) differentiable

## 1.2 Generative Models

> What I cannot create, I do not understand. - R. Feynman

Learning and understanding structure in data by being able to generate, is the rationale behind generative modelling. Generative models are mostly applied for unsupervised learning and can be contrasted to discriminative models. While discriminative models are used to model posterior conditionals $p(y|x)$ (*e.g.* for supervised learning (cf. sec. 1.1.1), generative models capture the complete data distribution $p(x)$ in an estimate $\hat{p}(x)$. Thus, after estimation, one can generate samples from this model $\hat{p}$, hence the name generative model. The currently predominant formulations for learning generative models are built on either autoencoding or adversarial formulations:

### 1.2.1 Autoencoding Formulations

An autoencoding model is learning by reconstructing samples of data, $\hat{x} = f(x)$. To enforce data compression (otherwise the identity function is a trivial solution of autoencoding) the function has an information bottleneck, namely an inferred latent code $z$ of reduced dimension. The autoencoder is then the chain of an encoding function $z = e(x)$ and a decoding function $\hat{x} = d(z) = d(e(x))$.

Whereas the conventional autoencoder consists of deterministic mappings $e, d$, the *variational autoencoder* models the probability distribution $p(x)$. More specifically, it maximizes a lower bound to the logarithmic likelihood $\log p(x)$ of data $x$. This so-called variational lower bound $\mathcal{L}$ is given by:

$$\mathcal{L} = \mathbb{E}_{z \sim q(z|x)} \log p(x|z) - \mathrm{KL}(q(z|x)||p(z)) \tag{1.2}$$

Where $z$ introduces latent variables, with a prior distribution $p(z)$. The approximation to the posterior $q(z|x)$ of the latent variables and the posterior of the data given the latent variables $p(x|z)$. If one wants to model the distributions with neural networks, one typically uses Gaussian distributions and lets the networks predict the parameters (mean $\mu$ and variance $\Sigma$) based on the image. In the current machine learning contexts, all functions ($e, d$) and or moments ($\mu, \Sigma$) are modelled with neural networks.

### 1.2.2 Adversarial Formulations

*Generative Adversarial Networks* (GAN) [5] consist of two neural networks competing in a zero-sum game. A generator network $G$ is generating images based on a latent code $z$ sampled from a distribution $p(z)$. The discriminator network $D$ is a binary classifier with the task to classify an image as originating from the data distribution $p_{data}$ or from the distribution produced by $G$. The loss function of $G$ is the negative of the loss of $D$, such that one can formulate the optimization in a minmax form:

$$\min_D \max_G -\frac{1}{2}\mathbb{E}_{x \sim p_{data}}[\log D(x)] - \frac{1}{2}\mathbb{E}_{z \sim p(z)}[\log(1 - D(G(z)))] \tag{1.3}$$

The discriminator can also be interpreted as a learned similarity metric to measure the closeness of an image to the data distribution [6]. The generator is then optimized to make the output indiscriminable from the data distribution.

There are many variants and extensions to this basic principle of learning with an adversarial task. For example, one can learn a discriminator on for a set of image patches [7].

## 1.3 Disentangling Representations

In supervised learning, a performance measure is naturally induced by the metric that is being optimized. In the unsupervised setting, judging the performance of a model is less straightforward. For example, when modelling an image domain, one could subjectively rate the quality of the generated image. But even for a qualitative assessment the question arises, how to rate the quality of the latent representation?

### 1.3.1 Learning Representations

> Disentangle as many factors as possible, discarding as little information about the data as is practical. - [8]

According to [8] a representation is useful, if it can be applied to many - in advance unknown - different tasks, while being trained on only one particular task. As the downstream tasks can be multifarious, the essential *information* should be contained in the representation. For some tasks only a subset of aspects of the data will be necessary, that is why *disentangled factors* make a representation particularly practical - so goes their reasoning.

The latent representation $z$ learned by generative models captures the essential *information* of the data distribution. That is made sure by requiring the ability to generate samples from the original data distribution from it. How then to reach the second goal, the *disentanglement* of generative factors:

### 1.3.2 Disentangling as Equivariance and Invariance

The definition of factor by change static ... factors should represent elements of real world - change in element -> corresponding change in representational factor - leave other factors representing other elements invariant

Formally, this can be posed as an inference problem: a number of latent variables $\mathbf{z_1} \ldots \mathbf{z_N}$ has interacted in certain ways to cause the existence of the observed image $\mathbf{x}$. An inference algorithm aims at recovering these latent variables from the observation, *i.e*. the image. These recoveries can be seen as estimates $\mathbf{\hat{z}_i}$ for - or a representation of - the true latent variables $\mathbf{z_i}$. A graphical model of the process is shown in figure 1.1. A disentangled representation should then represent each causal element and its state independently: A change in the real causal element $\mathbf{z_i}$ should correspond to an equivalent change in the abstract representational factor $\mathbf{\hat{z}_i}$, while leaving the other factors $\mathbf{\hat{z}_j}, j \neq i$, that represent other causes, unchanged.

mathematically,.. $f \circ g(x) = ...$

## 1.4 Theoretical Impediments from Causality

As outlined earlier, the type of knowledge that can be gained by learning from "raw" data is limited. With raw data we mean data $x$ sampled from a $p(x)$. so far fitting curve p(x) to data manifold what is missing to human-level intelligence? (cite lake 2016)

causal learning is a hard problem: instead of only learning statistical measures from data, model also needs to be learned ([9])

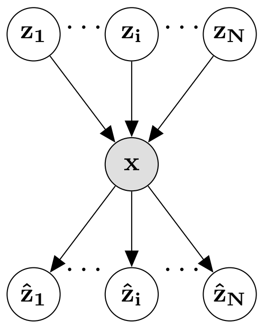
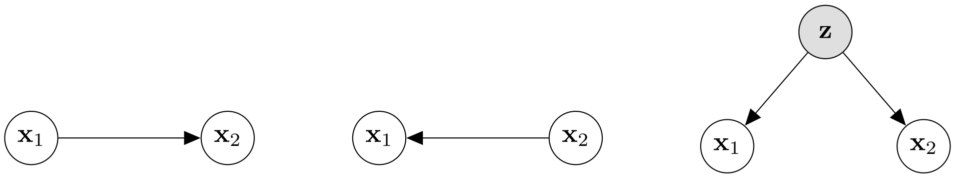



Figure 1.1: Disentangling causal factors means to infer an estimate - *i.e.* a representation - from an image




(a) (b) (c)

Figure 1.2: Correlation implies causation - if $x_1$ and $x_2$ correlate, a) $x_1$ may cause $x_2$, b) $x_1$ may be caused by $x_2$ or c) both are contingent on a latent cause $z$

Hypothesis: disentangling factors = estimating causal factors -> needs causal for estimation of causal factors "raw data" insufficient -> need interventional data or model assumptions. we do both: 1. intervene with changes to an image which are assumed to change only one factor. 2. model the causal process of the image generation in the theme of analysis-by-synthesis

### 1.4.1 Causal Learning requires Interventions or Assumptions

What does the causality literature have to say? Statistic background $\rightarrow$ correlation is not causation. Reichenbachs principle [10] $\rightarrow$ barometer example: How to find out the causal connection between a barometer and the weather. Highly capable machine learning algorithm that learns only with access to an image dataset showing the barometer and the weather. -> will be able to capture the correlation between needle position and weather condition, but never understand causal direction, since it is not in the data. Imagine how a human would go about solving this problem. Having a mechanistic model of the world he could reason about the precise causal mechanism relating weather to humidity to needle position. - model of influences (humidity -> barometer) What if no prior knowledge? A child-level simple solution is to force the needle to move with a finger. The weather will not change. Hence causality has to go other way or third latent variable influencing both. - intervening: move barometer needle by hand -> no change in weather, hence causality has to go the other way, (example from [11]) There cannot be an abstract intelligence, which finds out about the world purely by observation. The intelligence has to interact with the world, it has to be in the world. before this becomes too philosophical infer causation from correlation RCT

lacking the tools to accurately estimate causality, researchers shied away from making causal statement. Developing machines with human-like abilities requires discovery and reasoning in terms of causal models. Recently (in the past 30 years), overshadowed by the prominent success of data-driven deep learning, the field of causality has emerged to mathematical rigor.

- ladder of causation: association, intervention, counter-factual - current machine learning mostly on level of association (correlations estimated from "pure" data) -> purely data-driven approach can only go so far humans seem to have innate assumptions on coherence, causality, physics etc. introducing inductive biases

measure: p(x) assume causal model: p(x | a, s) want: p(s) and p(a)

encoding $p(s) = p(s|x)$ $p(a) = p(a|x) = p(a|s, x)$

decoding $p(x) = p(x|a, s)p(a)p(s)$

p(x| do(s), do(a))

example: Gaussian only with access to p(x) hopes to find factors p(a, b) = p(a) p(b) ([12], [13]) what if not full-filled? two-dimensional Gaussian: axis x and y are independent factors. in general any superposition of x and y which is orthogonal, can be found imagine a perfect dimensionality reduction yielding a two-dimensional latent space one can find the axes that correlate most with human understanding of independent factors i.e. pose and appearance. But how can a machine find these axes automatically from raw data? it cant, neither can anyone (including humans) (Pearl). Humans know these factors are independent from observing that they can change independently e.g. from observing someone undressing or changing his pose (i.e. harnessing temporal information, with the assumption of temporal coherence) or by changing the factors themselves e.g. what happens to the image of me if I change my pullover? It can be proven mathematically (Pearl) that interventional data or at least certain (which) causal assumptions about the world are necessary to estimate certain quantities.

### 1.4.2 Interventions are Transformations

we harness intervention p(x| do(a), b) in computer vision an intervention is an image transformation if ..

### 1.4.3 Assumptions in Analysis-by-Synthesis

Inverse graphics Capsules, Tieleman [14] make model as good as we can implementing as many assumptions as we can and only leave the rest to powerful model Synthesis known, analysis only indirectly by observing cognition

leaving synthesis to learning from scratch, can meet practical/computational limits *e.g.* convolutional neural networks better than fully connected neural models. But can also be ultimately impossible. Modelling synthesis explicitly with a causal model about image generation, by knowledge about the physical world enables answering interventional and counter-factual questions. (mathematically impossible to learn from ”pure” data alone)

## 1.5 Object Shape and Appearance

# Part I

# Appendix

# A Datasets

**CelebA** [15] contains ca. 200k celebrity faces of 10k identities. We resize all images to $128 \times 128$ and exclude the training and test set of the MAFL subset, following [16]. As [16, 17], we train the regression (to 5 ground truth landmarks) on the MAFL training set (19k images) and test on the MAFL test set (1k images).

**Cat Head** [18] has nearly 9k images of cat heads. We use the train-test split of [17] for training (7,747 images) and testing (1,257 images). We regress 5 of the 7 (same as [17]) annotated landmarks. The images are cropped by bounding boxes constructed around the mean of the ground truth landmark coordinates and resized to $128 \times 128$.

**CUB-200-2011** [19] comprises ca. 12k images of birds in the wild from 200 bird species. We excluded bird species of seabirds, roughly cropped using the provided landmarks as bounding box information and resized to $128 \times 128$. We aligned the parity with the information about the visibility of the eye landmark. For comparing with [17] we used their published code.

**BBC Pose** [20] contains videos of sign-language signers with varied appearance in front of a changing background. Like [21] we loosely crop around the signers. The test set includes 1000 frames and the test set signers did not appear in the train set. For evaluation, as [21], we utilized the provided evaluation script, which measures the PCK around $d = 6$ pixels in the original image resolution.

**Human3.6M** [22] features human activity videos. We adopt the training and evaluation procedure of [17]. For proper comparison to [17] we also removed the background using the off-the-shelf unsupervised background subtraction method provided in the dataset.

**Penn Action** [23] contains 2326 video sequences of 15 different sports categories. For this experiment we use 6 categories (tennis serve, tennis forehand, baseball pitch, baseball swing, jumping jacks, golf swing). We roughly cropped the images around the person, using the provided bounding boxes, then resized to $128 \times 128$.

**Dogs Run** is made from dog videos from YouTube totaling in 1250 images under similar conditions as in Penn Action. The dogs are running in one direction in front of varying backgrounds. The 17 different dog breeds exhibit widely varying appearances.

**Deep Fashion** [24, 25] consists of ca. 53k in-shop clothes images in high-resolution of $256 \times 256$. We selected the images which are showing a full body (all keypoints visible, measured with the pose estimator by [26]) and used the provided train-test split. For comparison with Esser *et al*. [27] we used their published code.

# B Lists

## B.1 List of Figures



## B.2 List of Tables

# C Bibliography

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. 2

[2] Josh Tenenbaum. Building machines that learn and think like people. In *AAMAS*, 2018. 3

[3] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 1989. 3

[4] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 1991. 3

[5] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 4

[6] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. *arXiv*, 2015. 4

[7] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv*, 2017. 4

[8] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *TPAMI*, 2013. 5

[9] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. MIT press, 2017. 5

[10] H. Reichenbach. *The Direction of Time*. University of California Press, 1956. 6

[11] Judea Pearl and Dana Mackenzie. *The Book of Why*. Hachette Book Group, 2018. 6

[12] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *NIPS*, 2016. 7

[13] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. Beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR*, 2017. 7

[14] Tijmen Tieleman. *Optimizing neural networks that generate images*. University of Toronto (Canada), 2014. 7

[15] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 10

[16] James Thewlis, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks by factorized spatial embeddings. In *ICCV*, 2017. 10

[17] Yuting Zhang, Yijie Guo, Yixin Jin, Yijun Luo, Zhiyuan He, and Honglak Lee. Unsupervised discovery of object landmarks as structural representations. In *CVPR*, 2018. 10

[18] Weiwei Zhang, Jian Sun, and Xiaoou Tang. Cat head detection - how to effectively exploit shape and texture features. In *ECCV*, 2008. 10

[19] C Wah, S Branson, P Welinder, P Perona, and S Belongie. The caltech-ucsd birds-200-2011 dataset. Technical report, California Institute of Technology, 2011. 10

[20] James Charles, Tomas Pfister, Derek R Magee, David C Hogg, and Andrew Zisserman. Domain adaptation for upper body pose tracking in signed tv broadcasts. In *BMVC*, 2013. 10

[21] Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Conditional image generation for learning the structure of visual objects. *NIPS*, 2018. 10

[22] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 2014. 10

[23] Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *ICCV*, 2013. 10

[24] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *CVPR*, 2016. 10

[25] Ziwei Liu, Sijie Yan, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Fashion landmark detection in the wild. In *ECCV*, 2016. 10

[26] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. 10

[27] Patrick Esser, Ekaterina Sutter, and Björn Ommer. A variational u-net for conditional appearance and shape generation. *CVPR*, 2018. 10