

An Analysis of the Inner Workings of Variational Autoencoders

DISSERTATION

der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
URS DOMINIK ZIETLOW
aus Tübingen

Tübingen
2022

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen
Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation: 16.11.2022

Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Dr. Georg Martius
2. Berichterstatter:	Prof. Dr. Philipp Hennig

Summary

Representation learning, the task of extracting meaningful representations of high-dimensional data, lies at the very core of artificial intelligence research. Be it via implicit training of features in a variety of computer vision tasks [1, 2, 3], over more old-school, hand-crafted feature extraction mechanisms for, e.g., eye-tracking [4, 5, 6] or other applications [7, 8, 9, 10], all the way to explicit learning of semantically meaningful data representations [11, 12, 13, 14, 15, 16, 17, 18, 19]. Strictly speaking, any activation of a layer within a neural network can be considered a representation of the input data. This makes the research about achieving explicit control over properties of such representations a fundamentally attractive task. An often desired property of learned representations is called disentanglement [11, 15, 19]. The idea of a disentangled representation stems from the goal of separating sources of variance in the data and consolidates itself in the concept of recovering generative factors. Assuming that every data has its origin in a generative process that produces high-dimensional data given a low-dimensional representation (e.g., rendering images of people given visual attributes, such as hairstyle, camera angle, age, ...), the goal of finding a disentangled representation is to recover those attributes.

The [Variational Autoencoder \(VAE\)](#) is a famous architecture commonly used for disentangled representation learning, and this work summarizes an analysis of its inner workings. [VAEs](#) achieved a lot of attention due to their, at the time, unparalleled performance as both generative models and inference models for learning disentangled representations. However, note that the disentanglement property of a representation is not invariant to

rotations of the learned representation, i.e., rotating a learned representation can change and destroy its disentanglement quality. Given a rotationally symmetric prior over the representations space, the idealized objective function of VAEs is rotationally symmetric. Their success at producing disentangled representations consequently comes as a particular surprise. This thesis discusses why VAEs pursue a particular alignment for their representations and how the chosen alignment is correlated with the generative factors of existing representation learning datasets.

Chapter 3 tackles the first question and sheds light on the connection between VAEs and classic Principal Component Analysis (PCA). It shows theoretically and verified experimentally that the canonical choice of a normal posterior with a diagonal covariance matrix breaks the rotational symmetry of the idealized objective. This choice furthermore leads to a close relation to PCA, as linearizations of the learned projection operations strive for orthogonality – just like the PCA projectors. We can experimentally confirm this finding by introducing a measure for the distance to orthogonality. An extension of the canonical implementation of a VAE to a full covariance matrix posterior functions as an ablation study to this finding. Despite the more general form of the posterior, the disentangling capability of the VAE disappears, which provides additional evidence that the choice of the posterior is crucial. Along the derivation of this behavior, a non-degeneracy assumption on the singular values of the Jacobian of the projection operation arises. We can link violations of this assumption to certain deficiencies of VAEs in practice.

Chapter 4 strengthens the connection between VAEs and PCA further in the linear case and thereby reveals more intricacies of their inner workings. We can show that linear VAEs have much more in common with PCA than just the orthogonality of their projectors: The learned embeddings match up to a signed permutation. This understanding extends to an intuition for the nonlinear case, in which VAEs are typically deployed. By carefully designing dataset perturbations that keep the generating factors intact, their

local structure can be altered so that [VAEs](#) and variations thereof fail to disentangle them. Although this is more of a destructive experiment, the consequent insights carry a high value: (1) It was unclear which type of bias (in the architectural choice or the data) is responsible for the success of [VAEs](#) in producing disentangled representations. This experiment answers how the local structure of datasets plays into this. (2) Various [VAE](#)-based architectures have been proposed, claiming improved disentanglement capabilities. We show that all of them fail to disentangle the altered datasets, indicating that they still rely on the same local structure in the data. (3) Even methods proven to recover the generating factors via weak forms of supervision cannot perform well on those datasets. This indicates that novel architectures should be evaluated on the altered datasets in addition to the originals to quantify their dependence on the local data structure.

Zusammenfassung

Das Lernen aussagekräftiger Repräsentationen aus hochdimensionalen Daten ist ein fundamentales Problem der Erforschung künstlicher Intelligenz. Diverse Methoden beschäftigen sich mit dieser Herausforderung: angefangen von implizit erlernten Repräsentationen [1, 2, 3], über manuell entwickelte Features (beispielsweise für Eye-Tracking [4, 5, 6] oder andere Anwendungen [7, 8, 9, 10]) bis hin zum expliziten Lernen von semantisch sinnvollen Repräsentationen [11, 12, 13, 14, 15, 16, 17, 18, 19]. Streng genommen kann jede Aktivierung innerhalb eines neuronalen Netzes als Repräsentation der Eingabedaten betrachtet werden. Das macht die Erforschung der Beschaffenheit solcher Repräsentationen ausgesprochen wichtig. Eine häufig angestrebte Eigenschaft dieser Repräsentationen wird als Disentanglement bezeichnet [11, 15, 19]. Die Idee einer disentangleten Repräsentation besteht darin, die Varianzen in den Daten zu separieren, um so die generativen Faktoren der Daten zu lernen. Davon ausgehend, dass alle Daten ihren Ursprung in einem generativen Prozess haben, der die hochdimensionalen Daten erzeugt (beispielsweise das Erstellen von Porträts über visuelle Attribute wie Frisur, Kamerawinkel, Alter, ...), dann ist das Ziel der Repräsentation, diese Attribute wiederherzustellen.

Der [VAE](#) ist eine bekannte Architektur, die häufig für das Lernen disentangleter Repräsentationen verwendet wird. Diese Arbeit umfasst eine Analyse ihrer Funktionsweise. [VAEs](#) erlangten aufgrund ihrer anfänglich unvergleichlichen Leistung, sowohl als generative Modelle als auch als Inferenzmodelle für das Lernen disentangleter Repräsentationen, große Aufmerksamkeit. Ob eine Repräsentation disentanglet ist, oder nicht, ist abhängig

von der Ausrichtung der gelernten Repräsentation. Das heißt, dass die Rotation einer gelernten Repräsentation ihre Qualität verändern kann. Bei einem rotationssymmetrischen Prior über die Repräsentationen ist die Kostenfunktion von **VAEs** allerdings rotationssymmetrisch. Ihr Erfolg im Lernen disentangelter Repräsentationen ist folglich überraschend. Diese Arbeit beschäftigt sich mit der Frage, warum **VAEs** eine bestimmte Ausrichtung ihrer Repräsentationen bevorzugen und wie die gewählte Ausrichtung mit den generativen Faktoren zusammenhängt.

Kapitel 3 befasst sich mit der ersten Frage und beleuchtet den Zusammenhang zwischen **VAEs** und der klassischen Hauptkomponentenanalyse (**PCA**). Sowohl in Form theoretischer, als auch experimenteller Ergebnisse wird gezeigt, dass die übliche Wahl eines normalverteilten Posteriors mit diagonaler Kovarianzmatrix die Rotationssymmetrie der Kostenfunktion aufhebt. Diese Wahl führt darüber hinaus zu einer engen Verbindung zu **PCA**: Die Jacobi-Matrizen des erlernten Modells sind orthogonal – genau wie bei **PCA**. Durch die Einführung einer Größe für den Abstand zur Orthogonalität können wir dieses Ergebnis experimentell untermauern. Eine Erweiterung der kanonischen Implementierung eines **VAE** zu einem posterior mit vollständiger Kovarianzmatrix zeigt, dass der **VAE** trotz der allgemeineren Form des Posteriors nicht mehr disentanglet, was einen weiteren Hinweis dafür liefert, dass die Wahl des Posteriors entscheidend ist. Eine formale Voraussetzung für die theoretischen Ergebnisse ist, dass die singulären Werte der Jacobi-Matrix nicht entartet sein dürfen. In Fällen, in denen dies nicht zutrifft, disentanglen **VAEs** nicht mehr, was eine Erklärung für manche, in der Praxis relevanten, Probleme liefert.

In Kapitel 4 wird die Verbindung zwischen **VAEs** und **PCA** im linearen Fall vertieft und weitere Details ihrer Funktionsweise behandelt. Wir können zeigen, dass lineare **VAEs** viel mehr mit **PCA** gemeinsam haben als nur die Orthogonalität ihrer Projektoren. Tatsächlich entsprechen sich die erlernten Repräsentationen bis auf Permutationen und Vorzeichen. Diese Ähnlichkeit erstreckt sich auch auf den nicht-linearen Fall, in dem **VAEs**

üblicherweise eingesetzt werden. Durch sorgfältig entwickelte Störungen existierender Datensätze, bei denen die generativen Faktoren erhalten bleiben, kann ihre lokale Struktur so weit verändert werden, dass VAEs (und Abwandlungen derer) nicht mehr disentanglen können. Die daraus resultierenden Erkenntnisse beantworten mehrere Fragen: (1) Welcher Teil der Architekturen oder der Daten ist für den Erfolg von VAEs verantwortlich? Durch dieses Experiment wird klar, wie die lokale Struktur der Datensätze dabei eine Rolle spielt. (2) Es wurde eine Vielzahl verschiedener VAE-basierter Architekturen entwickelt, die angeblich besser disentanglen. Es zeigt sich, dass keine dieser getesteten Architekturen die veränderten Datensätze disentanglen kann, was darauf hindeutet, dass sie sich immer noch auf dieselbe lokale Struktur in den Daten stützen. (3) Selbst Methoden, die durch zusätzliche Trainingssignale nachweislich die generativen Faktoren wiederherstellen können sollten, scheitern an diesen Datensätzen. Neue Architekturen könnten zukünftig also zusätzlich auf diesen Daten getestet werden, um ihre Abhängigkeit von der lokalen Datenstruktur zu evaluieren.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Technical introduction	2
1.3	Related work	5
2	Background	11
2.1	Principal component analysis	11
2.2	Autoencoders	12
2.3	Variational autoencoders	13
2.4	Singular value decomposition	15
2.5	Disentanglement	17
3	The connection between PCA and VAEs	21
3.1	Motivation	22
3.2	Methods	24
3.2.1	The problem with log-likelihood	24
3.2.2	Reformulating VAE loss	25
3.2.3	VAEs strive for orthogonality	28
3.2.4	Proof outline: A hands-on example	30
3.2.5	Intuitive picture: KL loss as “precision budget”	34
3.2.6	DtO via integer programming	36
3.2.7	β -VAE with full covariance matrix	37
3.3	Experiments	39
3.3.1	Setup	39
3.3.2	Polarized regime	43

3.3.3	Orthogonality and disentanglement	44
3.3.4	Degenerate case	45
3.3.5	Nonlinear VAE eigenfaces	47
3.3.6	Dependence of MIG and DtO on β	48
3.4	Conclusion	50
4	The inductive bias of VAEs and datasets	53
4.1	Motivation	54
4.2	Methods	55
4.2.1	Theoretical support of the connection to PCA	56
4.2.2	The generative process	58
4.2.3	Choice of fostered latent coordinate system	60
4.2.4	Dataset manipulations	60
4.3	Experiments	63
4.3.1	Architecture for perturbation network	64
4.3.2	Effectiveness of manipulations	65
4.3.3	Noisy datasets	66
4.3.4	Robustness over hyperparameters	68
4.3.5	Restart statistics and per factor evaluation	71
4.3.6	Inspection of latent embeddings	72
4.4	Conclusion	75
5	Proofs	79
5.1	Proof of Theorem 1	79
5.2	Proof of Theorem 2	90
6	Discussion and conclusions	95
6.1	Discussion	95
6.2	Limitations	97
6.3	Future work	98
7	Related projects	103
7.1	Deep graph matching via blackbox differentiation	103

7.2	Leveling down: Pareto inefficiencies in fair deep classifiers . . .	106
7.3	Machine learning quantum dynamics	109
7.4	InvGAN: Invertable GANs	113
7.5	Assaying out-of-distribution generalization in transfer learning .	115
7.6	Embrace the gap: VAEs perform independent mechanism analysis	118
List of Tables		121
List of Figures		123
List of Acronyms		125
Acknowledgements		127

Overview of Manuscripts

This thesis comprises the work and results of preceding publications via direct and indirect citations. The citations to these works are not explicitly indicated. For the sake of transparency, a list of all my publications related to the Ph.D. and my contribution to each of them are presented here.

- **Variational Autoencoders Pursue PCA Directions (by Accident) [20];**
Michal Rolínek*, *Dominik Zietlow**, Georg Martius;
CVPR 2019; <https://arxiv.org/abs/1812.06775>
- **Demystifying inductive biases for β -VAE based architectures [21];**
Dominik Zietlow, Michal Rolínek, Georg Martius;
ICML 2021; <https://arxiv.org/abs/2102.06822>
- **Leveling down in computer vision: Pareto inefficiencies in fair deep classifiers [22];**
Dominik Zietlow, Michael Lohaus, Matthaeus Kleindessner, Guha Balakrishnan, Francesco Locatello, Bernhard Schölkopf, Chris Russell;
CVPR 2022; <https://arxiv.org/abs/2203.04913>
- **Deep graph matching via blackbox differentiation of combinatorial solvers [23];**
Michal Rolínek, Paul Swoboda, *Dominik Zietlow*, Anselm Paulus, Vit Musil, Georg Martius;
ECCV 2020; <https://arxiv.org/abs/2003.11657>

* Authors contributed equally

- **Machine learning time-local generators of open quantum dynamics [24];**
Paolo Mazza, *Dominik Zietlow*, Federico Carollo, Sabine Andergassen, Georg Martius, Igor Lesanovsky;
Physical Review Research; <https://arxiv.org/abs/2101.08591>
- **InvGAN: Invertible GANs [25];**
Partha Ghosh, *Dominik Zietlow*, Michael J. Black, Larry S. Davis, Xiaochen Hu
GCPR 2022; <https://arxiv.org/abs/2112.04598>
- **Inferring Markovian quantum master equations of few-body observables in interacting spin chains [26];**
Francesco Carnazza, Federico Carollo, *Dominik Zietlow*, Sabine Andergassen, Georg Martius, Igor Lesanovsky;
New Journal of Physics; <https://arxiv.org/abs/2201.11599>
- **Assaying out-of-distribution generalization in transfer learning [27];**
Florian Wenzel, Andrea Dittadi, Peter Vincent Gehler, Carl-Johann Simon-Gabriel, Max Horn, *Dominik Zietlow*, David Kernert, Chris Russell, Thomas Brox, Bernt Schiele, Bernhard Schölkopf, Francesco Locatello;
NeurIPS 2022; <https://arxiv.org/abs/2207.09239>
- **Embrace the gap: VAEs perform independent mechanism analysis [28];**
Patrik Reizinger, Luigi Gresele, Jack Brady, Julius Von Kügelgen, *Dominik Zietlow*, Bernhard Schölkopf, Georg Martius, Wieland Brendel, Michel Besserve;
NeurIPS 2022; <https://arxiv.org/abs/2206.02416>

The focus of this dissertation is on [20] and [21], forming Chapter 3 and Chapter 4. Chapter 7 summarizes the other published manuscripts in a condensed form.

Title	Scientific ideas	Data generation	Analysis & interpretation	Paper writing
Variational autoencoders pursue PCA directions (by accident) [20]	30 %	80 %	50 %	50 %
Demystifying inductive biases for β -VAE based architectures [21]	80 %	100 %	80 %	80 %
Leveling down in computer vision: Pareto inefficiencies in fair deep classifiers [22]	50 %	90 %	60 %	50 %
Deep graph matching via blackbox differentiation of combinatorial solvers [23]	10 %	30 %	30 %	20 %
Machine learning time-local generators of open quantum dynamics [24]	20 %	30 %	10 %	10 %
InvGAN: Invertible GANs [25]	10 %	20 %	10 %	20 %
Inferring markovian quantum master equations of few-body observables in interacting spin chains [26]	10 %	5 %	10 %	5 %
Assaying out-of-distribution generalization in transfer learning [27]	10 %	10 %	10 %	15 %
Embrace the gap: VAEs perform independent mechanism analysis [28]	10 %	–	10 %	5 %

Chapter 1 | Introduction

1.1 Motivation

Extracting condensed information from complex data, such as from visual input, is, in most cases, trivial for human beings. Imagine walking through the city center of Tübingen and meeting somebody; you would immediately perceive more than just an image, but information such as their size, hair color, clothing, if they carry anything in their hands, and more. We can efficiently use this information to draw conclusions, make predictions, and reason more generally. For example, suppose the person has just finished eating a large ice cream cone. In that case, we can conclude the suggestion of ordering another one is not likely going to elicit a positive response.

As simple as it is for humans to deduce information from visual data, it is an intrinsically complex task for artificially intelligent systems. One approach to this challenge is called unsupervised representation learning and deals with various kinds of data, yet we will primarily focus on images in this work. The goal is to learn a model that can reliably infer a representation of an image that contains semantic and interpretable information. Notably, the model has to be trained on only the images alone, i.e., no additional training information is given.

A desired property of such a representation is called *disentanglement*. There are many ways to motivate and define that term, and here we want to treat it following the highest of its aspirations, namely as the ability to recover the *true generating factors* of the data. The underlying concept is that

1 Introduction

every data, such as a visually perceived image, has its origin in a generative process. In terms of computer graphics, the generative process would be some rendering, and in terms of the physical world we live in, it is defined by how optics and human perception work. Either of the processes produces a type of image given a scene, and the details of that scene are the generating factors. Think about what is necessary to specify the appearance of a simple portrait photograph of a person: The camera angle, lighting conditions, hairstyle, hair color, age, visual attributes of the person, and so on. Although this list is incomplete, just a dozen variables suffice to render a reasonably photo-realistic portrait. The goal of a disentangled representation is to extract and separate the generating factors of a given image such that every dimension of the representation corresponds to one of these factors.

One primary motivation for this research field is to use the learned information in various downstream tasks, such as image manipulation and reinforcement learning. Due to their nature, e.g., being semantically meaningful and disentangled, the representations can hopefully improve modern computer systems' reasoning and planning capabilities. In other words, the hope is to enable artificially intelligent systems to perceive visual input and draw conclusions in a more general, human form.

1.2 Technical introduction

The VAE [12, 29] is one of the foundational architectures in modern-day deep learning. It serves as a generative model and a representation learning technique. The generative model is predominantly exploited in computer vision [30, 31, 32, 33] with notable exceptions such as generating combinatorial graphs [34]. As for representation learning, there is a variety of applications, ranging over image interpolation [35], one-shot generalization [36], language models [37], speech transformation [38], and more. Aside from direct applications, VAEs embody the success of variational methods in deep learning and have inspired a wide range of ongoing research [32, 39].

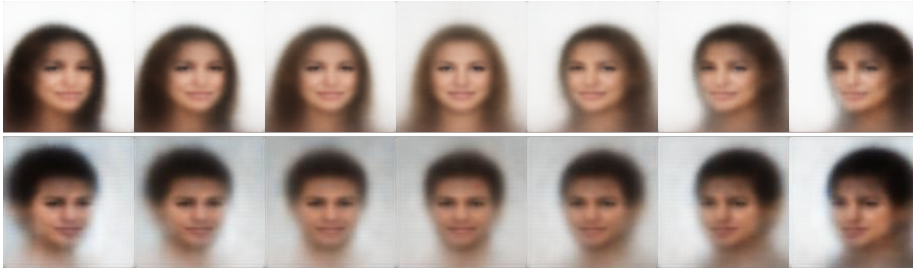


Figure 1.1: **Latent traversals** over a single latent coordinate on two exemplary images from the CelebA dataset [40] for a trained β Variational Autoencoder (β -VAE). The latent coordinate isolates the azimuth angle.

This thesis wants to shed light on the inner workings of VAEs, focusing on their disentanglement capabilities. Under a variety of disentanglement metrics [13, 14, 41, 15], VAE-based architectures (β -VAE [13], Total Correlation Variational Autoencoder (TC-VAE) [41], Factorized Variational Autoencoder (FactorVAE) [14], Disentangled Inferred Prior Variational Autoencoder (DIP-VAE) [16], Slow Variational Autoencoder (SlowVAE) [17]) dominate the benchmarks, leaving behind other approaches such as Information Maximizing Generative Adversarial Network (InfoGAN) [18] and Deep Convolution Inverse Graphics Network (DCIGN) [30]. An illustration of their success is given in Figure 1.1, which shows an example of a latent traversal for a β -VAE in which precisely one generative factor is isolated (face-camera angle).

The success of VAE-based architectures on disentanglement tasks comes as a particular surprise. One astonishing aspect is that VAEs have been challenged on both of their own design functionalities, i.e., as generative models [42, 43] and as log-likelihood optimizers [44, 45]. Yet, no such claims are made in terms of disentanglement. Another surprise stems from disentanglement requiring that the representative low-dimensional manifold is aligned well with the coordinate axes. In other words, learning a representation that recovers the true generating factors necessitates a specific

1 Introduction

alignment of that representation, i.e., it is not invariant under arbitrary rotations. However, the design of the VAE does not suggest any such mechanism. On the contrary, the idealized log-likelihood objective is, for example, invariant to rotational changes in the alignment.

Such observations have planted a suspicion that the inner workings of the VAE are not sufficiently understood. Several recent works approached this issue [46, 47, 48, 49, 43, 50, 41, 51]. However, a mechanistic explanation for the VAE’s unexpected ability to disentangle was still missing.

The first half of this thesis focuses on similarities between VAEs and PCA. We show that the canonical design choices made around VAEs have essential consequences for the representations learned. By choosing a zero mean and unit variance normal prior for the latent space, combined with a diagonal covariance matrix normal posterior, the Jacobian of the decoder model strives for orthogonality. This is a substantial similarity to PCA. We can additionally show that these orthogonal axes align precisely in the case of a linear VAE.

The central hypothesis of the second half of this thesis is that all unsupervised, VAE-based disentanglement architectures are successful because they exploit the same structural bias in the data. The ground truth generating factors align well with the “nonlinear Principal Components (PCs)” that VAEs strive for. This bias can be reduced by introducing a slight change in the local correlation structure of the input data, which, however, perfectly preserves the set of generative factors. We evaluate a set of approaches on slightly modified versions of the two leading datasets in which each image undergoes a modification inducing a small amount of variance. We report drastic drops in disentanglement performance on the altered datasets.

1.3 Related work

The related work can be categorized into three research questions: (i) defining disentanglement and metrics capturing the quality of latent representations; (ii) architecture development for unsupervised learning of disentangled representations; and (iii) understanding the inner workings of existing architectures, as of β -VAEs. This work is built upon results from all three lines of research. After looking into these three branches, this section will summarize published follow-up research built upon our work.

Defining disentanglement. Defining the term *disentangled representation* is an open question [19]. The presence of learned representations in downstream tasks of machine learning, such as object recognition, natural language processing, and others, created the need to “*disentangle the factors of variation*” [11] early on. This vague interpretation of disentanglement is inspired by the existence of a low-dimensional manifold that captures the variance of higher-dimensional data. As such, finding a factorized, statistically independent representation became a core ingredient of disentangled representation learning and dates back to classical **Independent Component Analysis (ICA)** models [52, 53]. For some tasks, the desired feature of a disentangled representation is that it is *semantically meaningful*. Prominent examples can be found in computer vision [54, 55] and in research addressing the interpretability of machine learning models [56, 57]. Based on group theory and symmetry transformations, [19] provides the “*first principled definition of a disentangled representation*”. Closely related to this concept is also the field of causality in machine learning [58, 59], more specifically, the search for causal generative models [60, 61]. In terms of implementable metrics, a variety of quantities have been introduced, such as the β -VAE score [13], **Separated Attribute Predictability (SAP)** score [16], **Disentanglement Completeness Informativeness (DCI)** scores [62] and the **Mutual Information Gap (MIG)** [41].

1 Introduction

Architecture development. The leading architectures for disentangled representation learning are based on VAEs [12]. Although initially developed as a generative modeling architecture, its variants have proven to excel at representation learning tasks. In particular, the β -VAE [13] performs remarkably well, as it exposes the trade-off between reconstruction and regularization via an additional hyperparameter. Other architectures have been proposed that additionally encourage statistical independence in the latent space, e.g., FactorVAE [14] and TC-VAE [41]. The DIP-VAE [16] suggests using moment-matching to close the distribution gap introduced in the original VAE paper. Using data with auxiliary labels, e.g., time indices of time series data, for which the conditional prior latent distribution is factorized, allowed [63] to circumvent the unidentifiability of previous models. Similarly, [17] used a sparse temporal prior for developing an identifiable model that also performs well on natural data. In this work, we also compare to representations learned by Permutation Contrastive Learning (PCL) [64]. This non-variational method conducts nonlinear ICA, also assuming temporal dependencies between the sources of variance. Contrastive methods have shown to be capable of inverting generating processes as desired for representation learning [65]. Another approach utilizes weak supervision on Generative Adversarial Networks (GANs) [42] to achieve disentangled representations in their underlying latent space [66].

Understanding inner workings. With the rising success and development of VAE-based architectures, the question of understanding their inner working principles became dominant in the community. One line of work tries to answer why these models disentangle at all [46]. Another research direction revealed the tight connection between the vanilla β -VAE objective and (probabilistic) PCA [67, 68]. The role of the regularization in β -VAEs was explicitly investigated in [69]. They analyze models under a second-order expansion, retrieving similar results regarding orthogonality for the case of β -VAEs. A broad field study was presented in [15], where they conducted

a set of experiments, questioning the relevance of the specific model architecture compared to the choice of hyperparameters and the variance over restarts. They also formalized the necessity of inductive biases as a strict requirement for unsupervised learning of disentangled representations via an “impossibility result”. Their statement is closely linked to the general unidentifiability theorem for nonlinear ICA [70]. The experiments presented in Chapter 4 are built on their code-base.

Research built upon our work. Building on our findings, novel approaches for model selection were proposed in [71]. They utilize that the local orthogonality of a model can be estimated without access to any labeled data. Consequently, it is possible to perform completely unsupervised model selection based on which model appears to be most orthogonal, thereby reducing the variance of disentanglement quality over restarts.

The work described in [72] makes explicit use of the observation that orthogonality is key to the success of β -VAEs. By restricting the model class to locally orthogonal models, they allow for direct control over that feature rather than promoting it indirectly through the VAE loss. This emphasizes the value of thoroughly understanding the inner working of existing architectures.

The difference between the data’s local and global variance structures is investigated in [73]. Unlike the classical linear directions of variance in a dataset that PCA isolates, the local effects of a generating factor can be manipulated without distorting the dataset much. Their work evaluates the correlation between local structure, global structure, and different types of embeddings. The work concludes with the observation that a discrepancy in the local and global variance structure is detrimental to the disentanglement quality, which is consistent with what is presented in this thesis.

Chapter 2 | Background

This chapter reviews the basics of [PCA](#), [Autoencoders \(AEs\)](#), [VAEs](#), [Singular Value Decomposition \(SVD\)](#) and disentanglement.

2.1 Principal component analysis

Let $\{\mathbf{x}^{(i)}\}_{i=1}^n$ be a dataset consisting of n i.i.d. samples $\mathbf{x}^{(i)} \in \mathcal{X} \subset \mathbb{R}^m$ of a random variable with zero mean. [PCA](#) solves the task of finding an orthogonal matrix that transforms a dataset such that the variance is maximized along one axis, followed by another one, and so on. The solution to this problem is strongly connected to the eigenvectors of the sample covariance matrix C_X , defined as

$$C_X = \frac{X^\top X}{n - 1} \quad (2.1)$$

where the data matrix $X \in \mathbb{R}^{n \times m}$ is

$$X = \begin{pmatrix} \mathbf{x}^{(1)\top} \\ \mathbf{x}^{(2)\top} \\ \vdots \\ \mathbf{x}^{(n)\top} \end{pmatrix}. \quad (2.2)$$

Assuming that C_X has m linearly independent eigenvectors, it can be decomposed into

$$C_X = Q\Lambda Q^{-1}$$

2 Background

where $Q = (\mathbf{q}^{(1)}, \mathbf{q}^{(2)}, \dots, \mathbf{q}^{(m)})^\top$ contains the eigenvectors and Λ is a diagonal matrix containing the sorted corresponding eigenvalues (in descending order): $\Lambda = \text{diag}(\lambda^{(1)}, \lambda^{(2)}, \dots, \lambda^{(m)})$. The *first principal direction* $\mathbf{w}^{(1)} \in \mathbb{R}^m$, i.e., the direction which captures the highest variance in X , is found by optimizing

$$\begin{aligned}\mathbf{w}^{(1)} &= \arg \max_{\|\mathbf{w}\|=1} \left((X\mathbf{w})^\top X\mathbf{w} \right) \\ &= \arg \max_{\|\mathbf{w}\|=1} \left(\mathbf{w}^\top X^\top X \mathbf{w} \right).\end{aligned}$$

Naturally the first principal direction is the normalized eigenvector $\mathbf{w}_{(1)} = \frac{\mathbf{q}^{(1)}}{\|\mathbf{q}^{(1)}\|}$. Similarly, the remaining principal directions are the normalized eigenvectors in ascending order. The requirement on the orthogonality is satisfied as C_X is symmetric by design. When using [PCA](#) for dimensionality reduction, one projects a datapoint only on a certain number of principal directions, which gives a low dimensional representation that covers maximal variance. By forming a linear combination of the normalized eigenvectors (weighted with the projections), the original data point can be reconstructed up to an error induced by the dimensionality reduction. An example of [PCA](#) is illustrated in [Figure 2.1](#). For a more rigorous introduction into [PCA](#), see [\[74\]](#).

2.2 Autoencoders

Similarly to [PCA](#), an [AE](#) operates with two mappings: The encoder $\text{Enc}_\varphi: \mathcal{X} \rightarrow \mathcal{Z}$ (what is the projection operation in [PCA](#)) and the decoder $\text{Dec}_\theta: \mathcal{Z} \rightarrow \mathcal{X}$ (the reconstruction operation), where $\mathcal{Z} \subset \mathbb{R}^d$ is called the *latent space*. Typically, Enc_φ and Dec_θ are modeled using deep neural networks with or without nonlinear activation functions. The parameters θ, φ are optimized according to a reconstruction loss $\mathcal{L}: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$, for example the L^2 distance between the reconstructed and the original datapoint

$$\theta^*, \varphi^* = \arg \min_{\theta, \varphi} \mathbb{E}_{x \in \mathcal{X}} [\mathcal{L}(\text{Dec}_\theta(\text{Enc}_\varphi(x)), x)].$$

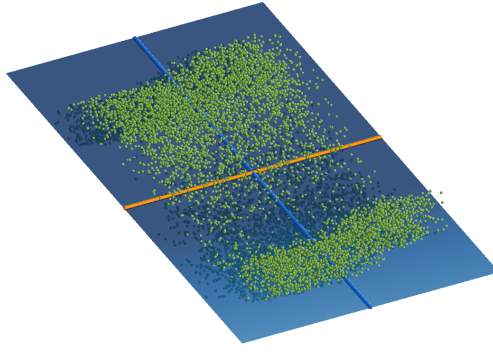


Figure 2.1: **Principal Component Analysis: PCA** with two principal components on the three-dimensional green point cloud isolates the direction of the largest variance (blue line) and the orthogonal direction of the second highest variance (orange line). The resulting plane spanned by **PCA** is illustrated in blue shades.

An example of a linear and a nonlinear **AE** is illustrated in Figure 2.2. For a more rigorous introduction into **AEs** see [75].

2.3 Variational autoencoders

In case of the **VAE**, both **AE** mappings are probabilistic and a fixed *prior distribution* $p(\mathbf{z})$ is assumed. Since the distribution over \mathbf{x} is also fixed (actual data distribution $q(\mathbf{x})$), the mappings Enc_φ and Dec_θ induce joint distributions $q(\mathbf{x}, \mathbf{z}) = q_\varphi(\mathbf{z}|\mathbf{x})q(\mathbf{x})$ and $p(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})$, respectively (omitting the dependencies on parameters θ and φ). The idealized **VAE** objective is then the marginalized log-likelihood

$$\sum_{i=1}^n \log p(\mathbf{x}^{(i)}). \quad (2.3)$$

This objective is, however, in most cases not tractable and therefore approximated by the **Evidence Lower Bound (ELBO)** [12]. For a fixed $\mathbf{x}^{(i)}$,

2 Background

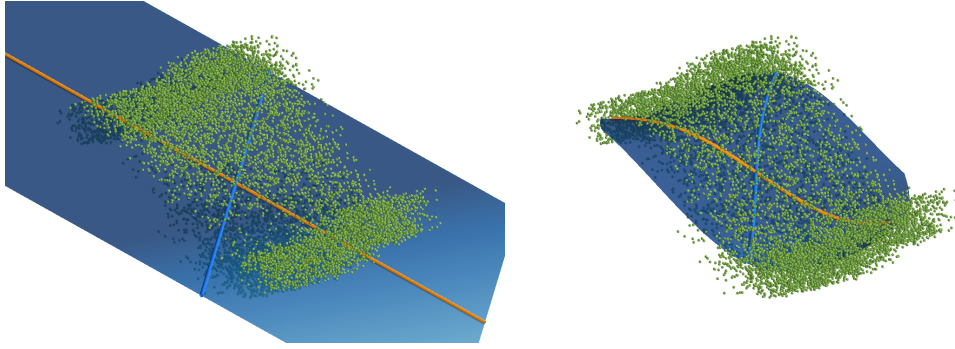


Figure 2.2: **Autencoder**: A linear **AE** with two latent dimensions (left) spans the same space as **PCA** (see Figure 2.1), but has a randomly aligned coordinate system (orange and blue line). A nonlinear **AE** with the same dimensionality (right) finds a much better fitting representation. The surfaces spanned by the **AEs** are illustrated in blue shades. In either case, the two coordinate axes are not necessarily orthogonal.

the log-likelihood $\log p(\mathbf{x}^{(i)})$ is lower bounded by

$$\mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}|\mathbf{x}^{(i)})} [\log p(\mathbf{x}^{(i)} | \mathbf{z}) - D_{\text{KL}}(q(\mathbf{z} | \mathbf{x}^{(i)}) \| p(\mathbf{z}))] , \quad (2.4)$$

where the first term corresponds to the reconstruction loss and the second to the **Kullback–Leibler (KL)** divergence between the latent representation $q(\mathbf{z} | \mathbf{x}^{(i)})$ and the prior distribution $p(\mathbf{z})$. A variant, the **β -VAE** [13], introduces a weighting β on the **KL** term for regulating the trade-off between reconstruction (first term) and the proximity to the prior.

Finally, the prior $p(\mathbf{z})$ is set to $\mathcal{N}(0, \mathcal{I})$ and the encoder is assumed to have the form

$$\text{Enc}_\varphi(\mathbf{x}) \sim q_\varphi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mu_\varphi(\mathbf{x}), \text{diag } \sigma_\varphi^2(\mathbf{x})) , \quad (2.5)$$

where μ_φ and σ_φ are deterministic mappings depending on parameters φ . Note that **the covariance matrix is enforced to be diagonal**. This

turns out to be highly significant for the main result of this work. The KL-divergence in (2.4) can be computed in closed form as

$$L_{\text{KL}} = \frac{1}{2} \sum_{j=1}^d (\mu_j^2(\mathbf{x}^{(i)}) + \sigma_j^2(\mathbf{x}^{(i)}) - \log \sigma_j^2(\mathbf{x}^{(i)}) - 1). \quad (2.6)$$

In practical implementations, the reconstruction term from (2.4) is approximated with either a square loss or a cross-entropy loss.

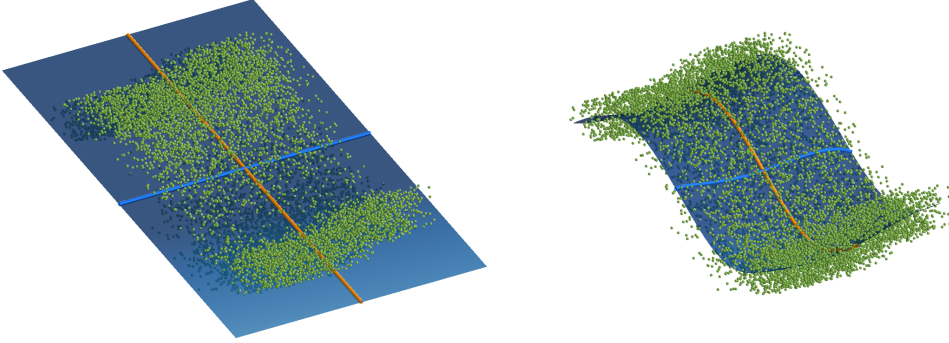


Figure 2.3: **Variational Autoencoder:** A linear VAE with two latent dimensions (left) aligns perfectly with PCA (see Figure 2.1). A nonlinear VAE with the same dimensionality (right) finds a much better fitting representation that is also locally orthogonal. The surfaces spanned by the VAEs are illustrated in blue shades.

For a more rigorous introduction into VAEs, see [76].

2.4 Singular value decomposition

The SVD is a powerful tool to decompose any matrix $M \in \mathbb{R}^{n \times d}$ into a product of a unitary square matrix, a diagonal matrix, and another unitary square matrix.

Theorem (SVD rephrased, [77]). *Let $M: \mathbb{R}^n \rightarrow \mathbb{R}^d$ be a linear transformation (matrix). Then there exist*

2 Background

- $U: \mathbb{R}^n \rightarrow \mathbb{R}^n$, an orthogonal transformation (matrix) of the input space,
- $\Sigma: \mathbb{R}^n \rightarrow \mathbb{R}^d$ a “scale-and-embed” transformation (induced by a diagonal matrix),
- $V: \mathbb{R}^d \rightarrow \mathbb{R}^d$, an orthogonal transformation (matrix) of the output space

such that $M = V\Sigma U^\top$.

Remark 1. For the sake of brevity, orthogonal transformations will be referred to as rotations (with a slight abuse of terminology).

As our results strongly depend on an analysis of the individual components of SVD decompositions, please refer to Figure 2.4 for an intuitive understanding of the decomposition. Colloquially speaking, any linear operator’s effect can be considered a unitary operation (e.g., rotations, sign flips, permutations), followed by an axis-aligned scaling and embedding into the possibly different dimensional space, finished with another unitary operation. The fact that the unitary matrices U and V act isometrically is key to the arguments made in this work.

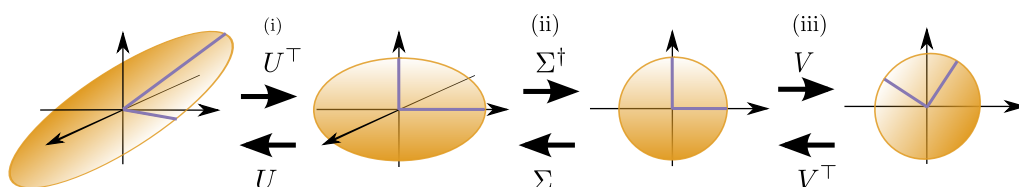


Figure 2.4: **Geometric interpretation of the SVD:** Sequential illustration of the effects of applying the corresponding SVD matrices $V\Sigma^\dagger U^\top$ (left to right) and $U\Sigma V^\top$ (right to left).

2.5 Disentanglement

In the context of learning interpretable representations [11, 13, 46, 47, 78] it is useful to assume that the data originates from a process with some generating factors. For instance, for images of faces, this could be face azimuth, skin tone, hair length, etc.. Disentangled representations can then be defined as ones in which individual latent variables are sensitive to changes in individual generating factors while relatively insensitive to other changes [11]. Although quantifying disentanglement is nontrivial, several metrics have been proposed [14, 13, 41].

In an unsupervised setting, the generating factors are unknown, and the learning has to resort to statistical properties. Linear dimensionality reduction techniques demonstrate the two basic statistical approaches. **PCA** greedily isolates sources of variance in the data, while **ICA** recovers a factorized representation, see [79] for a recent review.

One important point is that **disentanglement is sensitive to rotations of the latent embedding**. Following the example above, let us denote by a , s , and h continuous values corresponding to face azimuth, skin brightness, and hair length. Then, if the ideal latent representation is changed as follows

$$\begin{pmatrix} a \\ s \\ h \end{pmatrix} \mapsto \begin{pmatrix} 0.75a + 0.25s + 0.61h \\ 0.25a + 0.75s - 0.61h \\ -0.61a + 0.61s + 0.50h \end{pmatrix}, \quad (2.7)$$

one obtains an equally expressive representation in terms of reconstruction (in fact, it is only multiplied with an invertible 3D rotation matrix). Still, individual latent variables entirely lost their interpretable meaning.

Quantifying disentanglement

Among the different viewpoints on disentanglement, this thesis follows the recent literature and focuses on the connection between the discovered data representation and a set of *generative factors*.

2 Background

Multiple metrics have been proposed to quantify this connection. Most of them are based on the understanding that, ideally, each generative factor is encoded in precisely one latent variable. This was captured concisely by [41], who proposed the **MIG** – the mean difference (over the n_w generative factors) of the two highest mutual information between a latent coordinate and the single generating factor, normalized by its entropy. For the entropy $H(w_i)$ of a generating factor and the mutual information $I(w_i; z_k)$ between a generating factor and a latent coordinate, the **MIG** is defined as

$$\frac{1}{n_w} \sum_{i=1}^{n_w} \frac{1}{H(w_i)} \left(\max_k I(w_i; z_k) - \max_{k \neq k'} I(w_i; z_k) \right), \quad (2.8)$$

where $k' = \arg \max_k I(w_i, z_k)$. More details about **MIG**, its implementation, and an extension to discrete variables can be found in [41, 20]. Multiple other metrics were proposed such as **SAP** score [16], **FactorVAE** score [14] and **DCI** score [62]. See the supplementary material of [17] for extensive descriptions of each quantity.

Chapter 3 | The connection between PCA and VAEs

This chapter is based on:

**Variational autoencoders pursue PCA directions
(by accident) [20]**

Michal Rolínek*, *Dominik Zietlow**, Georg Martius

Published at CVPR 2019

<https://arxiv.org/abs/1812.06775>

Contributions:

- 30 % Scientific ideas
- 80 % Data generation
- 50 % Analysis & interpretation
- 50 % Paper writing

* Authors contributed equally

3.1 Motivation

In this chapter, we isolate an internal mechanism of the VAE responsible for choosing a particular latent representation and its alignment. We give theoretical analysis covering both the linear and also the nonlinear case and explain the discovered dynamics intuitively. We show that this mechanism promotes local orthogonality of the embedding transformation and clarify how this orthogonality corresponds to good disentanglement. Furthermore, we uncover a strong resemblance between this mechanism and the classical PCA algorithm. We confirm our theoretical findings in experiments.

Our theoretical approach is structured in the following way: (a) we base the analysis on the *implemented* loss function in contrast to the typically considered idealized loss, and (b) we identify a specific regime, prevalent in practice, and utilize it for a vital simplification. This simplification, referred to as *polarized regime*, is the crucial step at the base of our formalization. The polarized regime describes a state in which the latent space is partially suffering from posterior collapse [80, 68, 81]. However, the encoder remains almost deterministic for the remaining non-collapsed dimensions. This behavior is well known to practitioners.

Note that we do not explicitly discriminate between VAEs and β -VAEs, however tuning β is crucial to arrive at the *polarized regime*.

Ambiguous solutions to the reconstruction objective

Before looking into the VAE, let us examine more closely how PCA chooses the alignment of the latent embedding and why it matters. It is well known [82] that for a linear autoencoder with encoder $Y' \in \mathbb{R}^{d \times n}$, decoder $Y \in \mathbb{R}^{n \times d}$, and square error as reconstruction loss, the objective

$$\min_{Y, Y'} \sum_{\mathbf{x}^{(i)} \in X} \|\mathbf{x}^{(i)} - YY'\mathbf{x}^{(i)}\|^2 \quad (3.1)$$

is minimized by the **PCA** decomposition. Specifically, by setting $Y' = P_d$, and $Y = P_d^\top$, where $P_d \in \mathbb{R}^{d \times n}$ is formed by the first d normalized eigenvectors (ordered by the magnitudes of the corresponding eigenvalues) of the sample covariance matrix of X .

However, there are many minimizers of (3.1) that do not induce the same latent representation. It suffices to append Y' with some invertible transformations (e.g., rotations and scaling) and prefix Y with their inverses. This geometrical intuition is well captured using the **SVD** (see also Figure 2.4).

Example 1 (Other minimizers of the **PCA** objective). Define Y and Y' with their **SVDs** as $Y = P^\top \Sigma Q$ and its pseudoinverse $Y' = Y^\dagger = Q^\top \Sigma^\dagger P$ and see that

$$YY' = P^\top \Sigma Q Q^\top \Sigma^\dagger P = P^\top \mathcal{I}_{d \times n} \mathcal{I}_{n \times d} P = P_d^\top P_d \quad (3.2)$$

so they are also minimizers of the objective (3.1) irrespective of our choice of Q and Σ . It is also straightforward to check that the only choices of Q , which respect the coordinate axes given by the **PCA**, are for $|Q|$ to be a permutation matrix.

The takeaway message (also in the nonlinear case) from this example is:

Different rotations of the same latent space are equally suitable for reconstruction.

Following the **PCA** example, we formalize which linear mappings have the desired property.

Proposition 1 (Axes-preserving linear mappings). Assume $M \in \mathbb{R}^{n \times d}$ with $d < n$ has d distinct nonzero singular values. Then the following statements are equivalent:

- (a) The columns of M are (pairwise) orthogonal.
- (b) In every **SVD** of M as $M = U \Sigma V^\top$, $|V|$ is a permutation matrix.

We strongly suggest developing a geometrical understanding for both cases (a) and (b) via Figure 2.4. For an intuitive understanding of the formal requirement of distinct eigenvalues, we refer to Section 3.3.4.

Consider that once the encoder preserves the principal directions of the data, this already ensures an axis-aligned embedding. The same is true if the decoder is axes-preserving, provided the reconstruction of the autoencoder is accurate.

3.2 Methods

3.2.1 The problem with log-likelihood

The message from Example 1 and from the discussion about disentanglement is clear: latent space *rotation* matters. We now look at how the idealized objectives (2.3) and (2.4) handle this.

For a fixed rotation matrix U we will be comparing a baseline encoder-decoder pair $(\text{Enc}_\varphi, \text{Dec}_\theta)$ with a pair $(\text{Enc}_{\varphi,U}, \text{Dec}_{\theta,U})$ defined as

$$\text{Enc}_{\varphi,U}(\mathbf{x}) = U \text{Enc}_\varphi(\mathbf{x}), \quad (3.3)$$

$$\text{Dec}_{\theta,U}(\mathbf{z}) = \text{Dec}_\theta(U^\top \mathbf{z}). \quad (3.4)$$

We summarize the shortcomings of the idealized objective and its lower bound, the [ELBO](#), in the following two propositions.

Proposition 2 (Log-likelihood rotation invariance). *Let φ, θ be any choice of parameters for encoder-decoder pair $(\text{Enc}_{\varphi,U}, \text{Dec}_{\theta,U})$. Then, if the prior $p(\mathbf{z})$ is rotationally symmetric, the value of the log-likelihood objective (2.3) does not depend on the choice of U .*

Note that the standard prior $\mathcal{N}(0, \mathcal{I})$ is rotationally symmetric. This deficiency is not resolved by the [ELBO](#) approximation.

Proposition 3 (ELBO rotation invariance). *Let φ, θ be any choice of parameters for encoder-decoder pair $(\text{Enc}_{\varphi,U}, \text{Dec}_{\theta,U})$. Then, if the prior $p(\mathbf{z})$ is rotationally symmetric, the value of the ELBO objective (2.4) does not depend on the choice of U .*

For better readability, the proofs can be found in Chapter 5. An important point now follows:

Log-likelihood-based methods (with rotationally symmetric priors) cannot claim to be designed to produce disentangled representations.

However, enforcing a diagonal posterior of the VAE encoder (2.5) disrupts the rotational symmetry and the invariance arguments do consequently not hold for the resulting objective (2.6). The breaking of the rotational symmetry is visualized in Figure 3.1, where the rotationally symmetric prior is depicted alongside the non-symmetric, axis-aligned posterior. Moreover, as we are about to see, this diagonalization comes with beneficial effects regarding disentanglement. We assume this diagonalization was primarily introduced for different reasons (tractability, computational convenience).

3.2.2 Reformulating VAE loss

In order to understand which component of VAEs accounts for the orthogonality properties, we follow a bottom-up approach. We consider the implemented loss function and find the right simplifications that allow isolating the effects in question while preserving the original training dynamics.

We start by formalizing the typical situation in which VAE architectures shut down (fill with pure noise) a subset of latent variables and put high precision on the others.

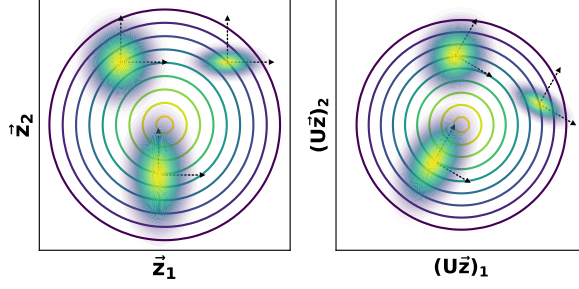


Figure 3.1: **Latent space prior and posterior:** For a rotationally symmetric distribution of the latent space (spherical contour lines), any transformation thereof would be invariant under rotations in latent space (and consequently so the log-likelihood and the ELBO). The rotational symmetry is instead broken by the diagonalization of the normal posterior (illustrated by the local heatmaps), which leads to axis-aligned representations.

Definition 1. We say that parameters φ, θ induce a polarized regime if the latent coordinates $\{1, 2, \dots, d\}$ can be partitioned as $V_a \cup V_p$ (sets of active and passive variables) such that

(a) $\mu_j^2(\mathbf{x}) \ll 1$ and $\sigma_j^2(\mathbf{x}) \approx 1$ for $j \in V_p$,

(b) $\sigma_j^2(\mathbf{x}) \ll 1$ for $j \in V_a$,

(c) The decoder ignores the passive latent components, i.e.,

$$\frac{\partial \text{Dec}_\theta(z)}{\partial z_j} = 0 \quad \forall j \in V_p.$$

The polarized regime simplifies the loss L_{KL} from (2.6); part (a) ensures zero loss for passive variables and part (b) implies that $\sigma_j^2(\mathbf{x}) \ll -\log(\sigma_j^2(\mathbf{x}))$. The per-sample-loss reduces to

$$L_{\approx\text{KL}}(\mathbf{x}^{(i)}) = \frac{1}{2} \sum_{j \in V_a} (\mu_j^2(\mathbf{x}^{(i)}) - \log(\sigma_j^2(\mathbf{x}^{(i)})) - 1). \quad (3.5)$$

We will assume the VAE operates in the polarized regime. In Section 3.3.2, we show on multiple tasks and datasets that the two objectives align very early in the training. This behavior is well-known to practitioners. Also, we approximate the reconstruction term in (2.4), as it is most common, with a square loss

$$L_{\text{rec}}(\mathbf{x}^{(i)}) = \mathbb{E} \|\text{Dec}_\theta(\text{Enc}_\varphi(\mathbf{x}^{(i)})) - \mathbf{x}^{(i)}\|^2 \quad (3.6)$$

where the expectation is over the stochasticity of the encoder. The loss we analyze has the form

$$\sum_{\mathbf{x}^{(i)} \in X} L_{\text{rec}}(\mathbf{x}^{(i)}) + L_{\approx\text{KL}}(\mathbf{x}^{(i)}). \quad (3.7)$$

Moreover, the reconstruction loss can be further decomposed into two parts; deterministic and stochastic. The former is defined by

$$\bar{L}_{\text{rec}}(\mathbf{x}^{(i)}) = \|\text{Dec}_\theta(\mu(\mathbf{x}^{(i)})) - \mathbf{x}^{(i)}\|^2 \quad (3.8)$$

and captures the square loss of the mean encoder. The stochastic loss

$$\hat{L}_{\text{rec}}(\mathbf{x}^{(i)}) = \mathbb{E} \|\text{Dec}_\theta(\mu(\mathbf{x}^{(i)})) - \text{Dec}_\theta(\text{Enc}_\varphi(\mathbf{x}^{(i)}))\|^2 \quad (3.9)$$

is purely induced by the noise injected into the encoder.

Proposition 4. *If the stochastic estimate $\text{Dec}_\theta(\text{Enc}_\varphi(\mathbf{x}^{(i)}))$ is unbiased around $\text{Dec}_\theta(\mu(\mathbf{x}^{(i)}))$, then*

$$L_{\text{rec}}(\mathbf{x}^{(i)}) = \bar{L}_{\text{rec}}(\mathbf{x}^{(i)}) + \hat{L}_{\text{rec}}(\mathbf{x}^{(i)}). \quad (3.10)$$

This decomposition resembles the classical bias-variance decomposition of the square error [83].

3.2.3 VAEs strive for orthogonality

Now, we finally give theoretical evidence for a central claim of this thesis:

**Optimizing the stochastic part of the reconstruction loss
promotes local orthogonality of the decoder.**

On that account, we set up an optimization problem that allows us to optimize the stochastic loss (3.9) independently of the other two. This will isolate its effects on the training dynamics.

In order to make statements about local orthogonality, we introduce for each $\mathbf{x}^{(i)}$ the Jacobian (linear approximation) J_i of the decoder at point $\mu(\mathbf{x}^{(i)})$, i.e.,

$$J_i = \frac{\partial \text{Dec}_\theta(\mu(\mathbf{x}^{(i)}))}{\partial \mu(\mathbf{x}^{(i)})}.$$

According to (2.5), the encoder can be written as

$$\text{Enc}_\varphi(\mathbf{x}^{(i)}) = \mu(\mathbf{x}^{(i)}) + \varepsilon(\mathbf{x}^{(i)}) \quad (3.11)$$

with

$$\varepsilon(\mathbf{x}^{(i)}) \sim \mathcal{N}(0, \text{diag } \sigma^2(\mathbf{x}^{(i)})). \quad (3.12)$$

Therefore, we can approximate the stochastic loss (3.9) with

$$\begin{aligned} & \mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \left\| \text{Dec}_\theta(\mu(\mathbf{x}^{(i)})) - (\text{Dec}_\theta(\mu(\mathbf{x}^{(i)})) + J_i \varepsilon(\mathbf{x}^{(i)})) \right\|^2 \\ &= \mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|J_i \varepsilon(\mathbf{x}^{(i)})\|^2, \end{aligned} \quad (3.13)$$

Although we aim to fix the deterministic loss (3.8), we do not need to freeze the mean encoder and the decoder entirely. Following Example 1, for each J_i and its SVD $J_i = U_i \Sigma_i V_i^\top$, we are free to modify V_i as long as we correspondingly (locally) modify the mean encoder.

Then we state the optimization problem as follows:

$$\min_{V_i, \sigma_j^i > 0} \sum_{\mathbf{x}^{(i)} \in X} \log \mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|J_i \varepsilon(\mathbf{x}^{(i)})\|^2 \quad (3.14)$$

$$\text{s. t.} \quad \sum_{\mathbf{x}^{(i)} \in X} L_{\approx \text{KL}}(\mathbf{x}^{(i)}) = C, \quad (3.15)$$

where $\varepsilon(\mathbf{x}^{(i)})$ are sampled as in (3.12).

A few remarks are now in place.

- This optimization is not over network parameters but rather directly over the values of all V_i, σ_j^i (only constrained by (3.15)).
- Both the objective and the constraint concern *global losses*, not per sample losses.
- None of V_i, σ_j^i interfere with the rest of the VAE objective (3.7).

The presence of the (monotone) log function has one main advantage; we can describe **all global minima** of (3.14) in closed form. This is captured in the following theorem, the technical core of this work.

Theorem 1 (Main result). *The following holds for optimization problem (3.14, 3.15):*

- Every local minimum is a global minimum.*
- In every global minimum, the columns of every J_i are orthogonal.*

The full proof and an explicit description of the minima are given in Section 5.1. However, an outline of the main steps is provided in the next section on the example of a linear decoder.

The presence of the log term in (3.14) admittedly makes our argument indirect. There are, however, a couple of points to make. First, as was mentioned earlier, encouraging orthogonality was *not a design feature* of the VAE. In this sense, it is unsurprising that our results are also mildly indirect. Additionally, and more importantly, the global optimality of Theorem 1 also

3 The connection between PCA and VAEs

implies that local orthogonality is encouraged even for the pure (without the logarithm) stochastic loss.

Corollary 1. For fixed $\mathbf{x}^{(i)} \in X$ consider a subproblem of (3.14) defined as

$$\min_{V_i, \sigma_j^i > 0} \mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|J_i \varepsilon(\mathbf{x}^{(i)})\|^2 \quad (3.16)$$

$$\text{s. t.} \quad L_{\approx KL}(\mathbf{x}^{(i)}) = C_i. \quad (3.17)$$

Also then, the result on the structure of local (global) minima holds:

- (a) Every local minimum is a global minimum.
- (b) In every global minimum, the columns of every J_i are orthogonal.

All in all, Theorem 1 justifies the central message of this chapter. The analogy with PCA is now also clearer. Locally, VAEs optimize a tradeoff between reconstruction and orthogonality. This result is unaffected by the potential β term in Equation (2.4), although an appropriate β might be required to ensure the polarized regime.

3.2.4 Proof outline: A hands-on example

In this section, we sketch the key steps in the proof of Theorem 1 and, more notably, the intuition behind them. The proof can be found in Section 5.1.

We will restrict ourselves to a simplified setting. Consider a linear decoder M with SVD $M = U\Sigma V^T$, which removes the necessity of local linearization. This reduces the objective (3.14) from a “global” problem over all examples $\mathbf{x}^{(i)}$ to an objective where we have the same subproblem for each $\mathbf{x}^{(i)}$. As in optimization problem (3.14, 3.15), we resort to fixing the mean encoder (imagine a well performing one). In the following paragraphs, we separately perform the optimization over the parameters σ and the optimization over the matrix V .

Weighting precision

For this part, we fix the decoder matrix M and optimize over values $\sigma^2 = (\sigma_1^2, \dots, \sigma_d^2)$. The simplified objective is

$$\min_{\sigma} \mathbb{E}_{\varepsilon \sim \mathcal{N}(0, \text{diag}(\sigma^2))} \|M\varepsilon\|^2 \quad (3.18)$$

$$\text{s. t.} \quad \sum_j -\log \sigma_j^2 = C, \quad (3.19)$$

where the $\|\mu\|^2$ terms from (3.5) disappear since the mean encoder is fixed. The values $-\log(\sigma_j)$ can now be thought of as precisions allowed for different latent coordinates. The log function even suggests thinking of the number of significant digits. Problem (3.18) then asks to distribute the “total precision budget” so that the deviation from decoding “uncorrupted” values is minimal.

We will now solve this problem on an example linear decoder $M_1: \mathbb{R}^2 \rightarrow \mathbb{R}^3$ given by

$$M_1: \begin{pmatrix} x \\ y \end{pmatrix} \mapsto \begin{pmatrix} 4x + y \\ -3x + y \\ 5x - y \end{pmatrix}. \quad (3.20)$$

Already here we see, that the latent variable x seems more influential for the reconstruction. We would expect that x receives higher precision than y .

Now, for $\varepsilon = (\varepsilon_x, \varepsilon_y)$, we compute

$$\|M_1\varepsilon\|^2 = \|4\varepsilon_x + \varepsilon_y\|^2 + \|-3\varepsilon_x + \varepsilon_y\|^2 + \|5\varepsilon_x - \varepsilon_y\|^2$$

and after taking the expectation, we can use the fact that ε has zero mean and write

$$\begin{aligned} \mathbb{E} \|M_1\varepsilon\|^2 &= \\ &= \text{var}[4\varepsilon_x + \varepsilon_y] + \text{var}[-3\varepsilon_x + \varepsilon_y] + \text{var}[5\varepsilon_x - \varepsilon_y]. \end{aligned}$$

3 The connection between PCA and VAEs

Finally, for uncorrelated random variables A and B we have that $\text{var}[A + cB] = \text{var}[A] + c^2 \text{var}[B]$. After rearranging we obtain

$$\begin{aligned}\mathbb{E} \|M_1 \varepsilon\|^2 &= \sigma_x^2(4^2 + (-3)^2 + 5^2) + \sigma_y^2(1^2 + 1^2 + (-1)^2) \\ &= 50\sigma_x^2 + 3\sigma_y^2,\end{aligned}$$

where $\sigma = (\sigma_x^2, \sigma_y^2)$. Note that the coefficients are the **squared norms of the column vectors** of M_1 .

This turns the optimization problem (3.18) into a simple exercise, particularly after realizing that (3.19) fixes the value of the product $\sigma_x \sigma_y$. Indeed, we can even set $a^2 = 50\sigma_x^2$ and $b^2 = 3\sigma_y^2$ in the trivial inequality $a^2 + b^2 \geq 2ab$ and find that

$$\mathbb{E} \|M_1 \varepsilon\|^2 = 50\sigma_x^2 + 3\sigma_y^2 \geq 2 \cdot \sqrt{50 \cdot 3} \cdot e^{-C} \approx 24.5e^{-C}, \quad (3.21)$$

with equality achieved when $\sigma_x^2/\sigma_y^2 = 3/50$. This also implies that the precision $-\log \sigma_x^2$ on variable x will be considerably higher than for y , just as expected.

Two remarks regarding the general case follow.

- The full version of inequality (3.21) relies on the concavity of the log function; in particular, on (a version of) Jensen's inequality.
- The minimum value of the objective depends on the product of the column norms. This also carries over to the un-simplified setting.

Isolating sources of variance

Now that we can find optimal values of precision, the focus changes on optimally rotating the latent space. In order to understand how such rotations influence the minimum of objective (3.18), let us consider the following example in which we again resort to decoder matrix $M_2: \mathbb{R}^2 \rightarrow \mathbb{R}^3$.

Imagine, the encoder alters the latent representation by a 45° rotation. Then we can adjust the decoder M_1 by first undoing this rotation. In

particular, we set $M_2 = M_1 R_{45^\circ}^\top$, where R_θ is a 2D rotation matrix, rotating by angle θ . We have

$$M_2: \begin{pmatrix} x' \\ y' \end{pmatrix} \mapsto \begin{pmatrix} \frac{1}{2}\sqrt{2}(3x' + 5y') \\ \sqrt{2}(-2x' - y') \\ \sqrt{2}(3x' + 2y') \end{pmatrix}$$

and performing analogous optimization as before gives

$$\mathbb{E} \|M_2 \varepsilon\|^2 = \frac{61}{2}\sigma_x^2 + \frac{45}{2}\sigma_y^2 \geq 2\sqrt{\frac{61 \cdot 45}{4}}e^{-C} \approx 52.4e^{-C}. \quad (3.22)$$

We see that the minimal value of the objective is more than twice as high, a substantial difference. On a high level, the reason M_1 was a better choice of a decoder is that the variables x and y had a very different impact on the reconstruction. This allowed to save some precision on variable y , as it had a smaller effect, and use it on x , where it is more beneficial.

For a higher number of latent variables, one way to achieve a “maximum stretch” among the impacts of latent variables, is to pick them greedily, always picking the next one so that its impact is maximized. This is, at heart, the greedy algorithm for [PCA](#).

Let us consider a slightly more technical statement. We saw in (3.21) and (3.22) that after finding optimal values of σ the remaining objective is the product of the column norms of matrix M . Let us denote such quantity by $\text{col}_\Pi(M) = \prod_j \|M_{\cdot j}\|$. Then for a fixed matrix M , we optimize

$$\min_V \text{col}_\Pi(MV^\top) \quad (3.23)$$

over orthogonal matrices V . This problem can be interpreted geometrically. The column vectors of MV^\top are the images of base vectors e_j . Consequently, the product gives an upper bound on the volume (the image of the unit cube)

$$\prod_j \|MV^\top e_j\| \geq \text{Vol}(\{MV^\top x: x \in [0, 1]^d\}). \quad (3.24)$$

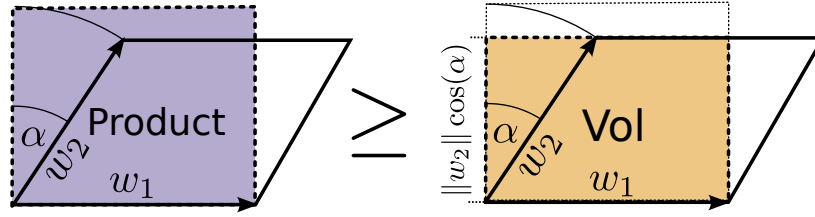


Figure 3.2: **Orthogonality in MV^\top** : The vectors w_1, w_2 are the columns of MV^\top . Minimizing the product $\|w_1\|\|w_2\|$ while maintaining the volume $\|w_1\|\|w_2\|\cos(\alpha)$ results in $w_1 \perp w_2$.

However, as orthogonal matrices V are isometries, they do not change this volume. Also, the bound (3.24) is tight precisely when the vectors $MV^\top e_j$ are orthogonal. Hence, the only way to optimize $\text{col}_\Pi(MV^\top)$ is by tightening the bound by finding V for which the column vectors of MV^\top are orthogonal, see Figure 3.2 for an illustration. In this regard, it is important that M performs a different scaling along each of the axis (using Σ), which allows for changing the angles among the vectors $MV^\top e_j$ (c.f. Figure 2.4).

3.2.5 Intuitive picture: KL loss as “precision budget”

In this subsection we want to provide an intuitive picture for interpreting the optimization problem (Equations (3.18) and (3.19)) as well as Theorem 1. The optimization objective 3.18 comprises the reconstruction error induced by the stochastic nature of the encoding. For the reconstruction objective, the effects of the non-zero $\log \sigma_j^2$ are detrimental, i.e., the reconstruction loss increases with increasing σ_j^2 . Consequently, an intuitive way of thinking about the remainder of the KL loss in constraint 3.19 is to interpret them as causing noise-contamination of the latent space or inversely as a “precision budget”.

Stepping back from the simplified optimization problem in the polarized regime and looking at the vanilla VAE objective, we plot the KL loss 2.6 for one latent dimension (the index is omitted) in Figure 3.3.

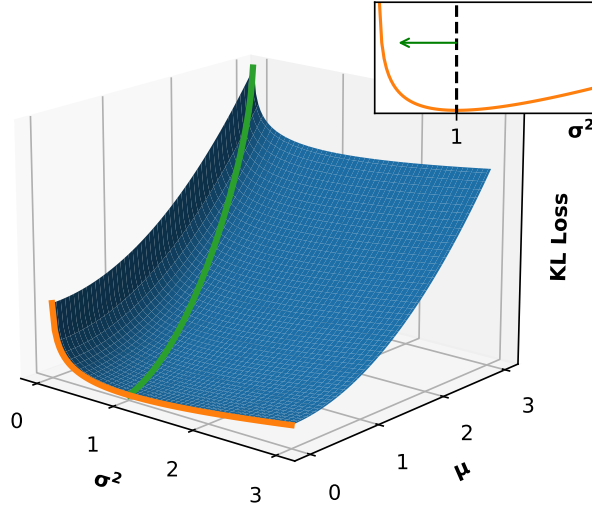


Figure 3.3: **KL loss landscape** for the vanilla VAE implementation with the canonical prior and posterior. The loss is zero for $\mu_i = 0$ and $\sigma_i = 1$, which corresponds to data-point independent noise.

The KL loss is minimal for $\mu = 0$ and $\sigma^2 = 1$, values for which reconstruction beyond the mean of the data is intrinsically impossible as the latent representation does not contain information about the encoded image. To encode information about the input data in the latent representation, the mean has to deviate from zero, and the values of σ^2 have to decrease, for example, along the green line in Figure 3.3. This increase in the KL loss has to be balanced by a decrease in reconstruction loss – the well-known classical trade-off that initially inspired the β -VAE.

The constraint 3.19 tells us how much the model can deviate from $\sigma_j^2 = 1$ on a logarithmic scale, summed over all latent dimensions. As it turns out, the β -VAE distributes the values of σ_j according to the variance in

the data. Like in the linear example of Section 3.2.4 where the variable x induces a much larger variance as y , the best way to distribute the noise on the two variables is by making the representation of x more informative, i.e., have lower noise. Consequently, higher noise on y has to be accepted. The logarithmic scale even allows thinking of the KL loss as a “price per signal-carrying decimal place”.

3.2.6 DtO via integer programming

For measuring the effects of Theorem 1, we introduce a measure of non-orthogonality. As argued in Proposition 1 and Figure 2.4, for a good decoder M and its SVD $M = U\Sigma V^\top$, the matrix V should be trivial (a signed permutation matrix). We measure the deviation with the [Distance to Orthogonality \(DtO\)](#), defined as follows. For each $\mathbf{x}^{(i)}$, $i = 1, \dots, N$, the Jacobian J_i of the decoder at $\mathbf{x}^{(i)}$ and its SVD $J_i = U_i \Sigma_i V_i^\top$, we define

$$\text{DtO} = \frac{1}{N} \sum_{i=1}^N \|V_i - P(V_i)\|_F, \quad (3.25)$$

where $\|\cdot\|_F$ is the Frobenius norm and $P(V_i)$ is a signed permutation matrix that is closest to V (in L^1 sense).

Using [Mixed-Integer Linear Programming \(MILP\)](#) formulation, we find the closest permutation matrix as the optimum P^* of the following optimization problem

$$\begin{aligned} \min_P \quad & \sum_{i,j} |V_{i,j} - P_{i,j}| & (3.26) \\ \text{s.t.} \quad & P_{i,j} \in \{-1, 0, 1\} & \forall (i, j) \\ & \sum_i |P_{i,j}| = 1 & \forall j \\ & \sum_j |P_{i,j}| = 1 & \forall i. \end{aligned}$$

Producing a clean MILP formulation with a purely linear objective and binary integer values can be achieved with a standard technique:

By introducing new variables. In particular, we set

$$\begin{aligned} P_{i,j} &= P_{i,j}^+ - P_{i,j}^- & (3.27) \\ \text{for } P_{i,j}^+, P_{i,j}^- &\in \{0, 1\} \quad \forall (i, j) \end{aligned}$$

and introduce (continuous) variables for the differences $V_{i,j} - P_{i,j}$

$$\begin{aligned} V_{i,j} - P_{i,j} &\leq D_{i,j} & \forall (i, j) & (3.28) \\ P_{i,j} - V_{i,j} &\leq D_{i,j} & \forall (i, j). \end{aligned}$$

The final formulation then is

$$\begin{aligned} \min_P \sum_{i,j} D_{i,j} & & (3.29) \\ \text{s.t. } (P_{i,j}^+ - P_{i,j}^-) - V_{i,j} &\leq D_{i,j} & \forall (i, j) \\ V_{i,j} - (P_{i,j}^+ - P_{i,j}^-) &\leq D_{i,j} & \forall (i, j) \\ \sum_i (P_{i,j}^+ + P_{i,j}^-) &= 1 & \forall j \\ \sum_j (P_{i,j}^+ + P_{i,j}^-) &= 1 & \forall i. \end{aligned}$$

3.2.7 β -VAE with full covariance matrix

In the derivation of the [VAE](#) loss function, the approximate posterior is set to be a multivariate normal distribution with a diagonal covariance matrix. We claim that this diagonalization is responsible for the orthogonalization. As one of the control experiments in [Section 3.3](#) we also implemented [VAE](#) with a full covariance matrix.

Two issues now need to be addressed; computing [KL](#) divergence in closed form and adapting the reparameterization trick. Regarding the former, the sought identity is

$$\begin{aligned} D_{\text{KL}}(\mathcal{N}(\mu, \Sigma) \parallel \mathcal{N}(0, \mathcal{I}_k)) & & (3.30) \\ = \frac{1}{2} (\|\mu\|^2 + \text{tr}(\Sigma) - \log(\det(\Sigma)) - k). \end{aligned}$$

3 The connection between PCA and VAEs

As for the reparameterization trick, if $\varepsilon \sim \mathcal{N}(0, \mathcal{I}_k)$, it is easy to check that

$$\mu + \Sigma^{1/2}\varepsilon \sim \mathcal{N}(\mu, \Sigma), \quad (3.31)$$

where $\Sigma = \Sigma^{1/2} \cdot (\Sigma^{1/2})^\top$ is the unique Cholesky decomposition of the positive definite matrix Σ .

3.3 Experiments

We performed several experiments with different architectures and datasets to validate our results empirically. We show the prevalence of the polarized regime, the strong thrive towards orthogonality in the β -VAE, and the links to disentanglement.

3.3.1 Setup

Architectures: We evaluate the classical VAE, β -VAE, a plain AE, and β -VAE _{Σ} , where the latter removes the critical diagonal approximation (Equation (2.5)) and produces a full covariance matrix $\Sigma(\mathbf{x}^{(i)})$ for every sample. The resulting KL term of the loss is changed accordingly (see Section 3.2.7 for details).

Datasets: We evaluate on the well-known datasets dSprites [84], MNIST [85] and FashionMNIST [86], as well as on two synthetic ones. For both synthetic tasks, the input data X is generated by embedding a unit square $V = [0, 1]^2$ into a higher dimension. The latent representation is then expected to be disentangled with respect to axes of V . In one case (*Synth. Lin.*) we used a linear transformation $f_{\text{lin}}: \mathbb{R}^2 \rightarrow \mathbb{R}^3$ and in the other one a nonlinear (*Synth. Non-Lin.*) embedding $f_{\text{non-lin}}: \mathbb{R}^2 \rightarrow \mathbb{R}^6$.

Disentanglement metric: For quantifying the disentanglement of a representation, the so-called MIG was introduced in [41] (see Section 2.5). As MIG is not well defined for continuous variables, we use an adjusted definition comprising both continuous and discrete variables, referred to as **Disentanglement Score (DS)**. As in the case of MIG, the DS is a number between 0 and 1, where a higher value means stronger disentanglement.

3 The connection between PCA and VAEs

Table 3.1: **Experimental details:** Overview of the used datasets and network architectures. The nonlinearities are only applied in the hidden layers. Biases are used for all datasets.

	Optimizer (LR)	Architecture	Latent Dim.	Epochs	β
dSprites	AdaGrad (10^{-2})	Enc: 1200 – 1200 (ReLU) Dec: 1200 – 1200 – 1200 (tanh)	5	50	4
Synth. Lin.	Adam (10^{-3})	Enc: No hidden Layers (none) Dec: No hidden Layers (none)	2	600	10^{-4}
Synth. Non-Linear.	Adam (10^{-3})	Enc: 60 – 40 – 20 (tanh) Dec: 60 – 40 – 20 (tanh)	2	600	10^{-3}
MNIST	AdaGrad (10^{-2})	Enc: 400 (ReLU) Dec: 500 – 500 (tanh)	6	400	1
fMNIST	AdaGrad (10^{-2})	Enc: 400 (ReLU) Dec: 500 – 500 (tanh)	6	500	1
CelebA	Adam (10^{-4})	Conv/Deconv: [# kernels, kernel size, stride] Enc: [[32, 4, 2], [32, 4, 2], [64, 4, 2], [64, 4, 2]] (ReLU) Dec: [[64], [64, 4, 2], [32, 4, 2], [32, 4, 2], [3, 4, 2]] (ReLU), first layer fully connected	32	50	4

Network details and training

Table 3.1 contains the training parameters used for the different architectures. If applicable, the listed latent dimension is chosen to be the number of independent generating factors and otherwise chosen large enough to ensure decent reconstruction loss on all architectures.

All reported numbers are calculated using a previously unseen test dataset. To facilitate this, we split the whole datasets randomly into three parts for training, evaluation, and test (containing 80 %, 10 % and 10 % of all samples, respectively). During development, we used the evaluation dataset; for the reported numbers, we used the test dataset.

Disentanglement score

For disentangled representations, single latent variables should be sensitive to individual generating factors and insensitive to all others. To quantify this behavior, for each generating factor w_i , all latent variables are evaluated for their sensitivity to w_i . The sensitivity difference between the two most responsive variables then reflects both desired properties; the sensitivity of the associated best matching latent variable and the insensitivity of all others. A set of quantities capturing disentanglement can therefore be described as

$$\text{DS} = \frac{1}{N_{\text{labels}}} \sum_{i=1}^N \left(\frac{A_{i,m(i)} - A_{i,s(i)}}{M_i} \right) \quad (3.32)$$

$$\text{for } m(i) = \arg \max_l (A_{i,l}) \quad (3.33)$$

$$\text{for } s(i) = \arg \max_{k \neq m(i)} (A_{i,k}), \quad (3.34)$$

where $A_{i,j}$ is some sensitivity measure of latent variable z_j concerning the generating factor w_i and M_i is a normalization constant, ensuring the summands fall into the interval $(0, 1)$.

The **MIG** uses mutual information to measure how the latent variables depend on the generating factors. For the normalization, the entropy of the generating factor is used.

$$A_{i,j} = \text{MI}(w_i, z_j) \quad (3.35)$$

$$M_i = H(w_i) \quad (3.36)$$

For discrete generating factors $\{w_i\}$, the normalization with the entropy $H(w_i)$, binds the **MIG** to the $(0, 1)$ interval, as expected. On the other side, this does not hold for continuous generating factors. Differential entropy can be zero or even negative, and no suitable normalization is possible.

To treat this shortcoming, we report the slightly modified **DS** such that it comprises continuous and discrete variables alike. Rather than using mutual information measurements, we employ powerful nonlinear regressors and

3 The connection between PCA and VAEs

classifiers for the two different classes of latent variables. The predictability of a generating factor from a given latent coordinate indirectly reflects how much information the two share.

Accordingly, we define the **DS** as in Equation (3.32) by defining $A_{i,j}$ as the prediction performance of the regressor/classifier for predicting generating factor w_i from the latent coordinate z_j . The normalization factor is then the performance of the best constant classifier/regressor. In the case of regression with mean square error, this is the standard deviation of the generative factor.

More precisely,

$$A_{i,j} = \begin{cases} \sqrt{\text{var}(w_i)} - \sqrt{\text{mse}_{z_j \rightarrow w_i}}, & \text{for regression} \\ \text{accuracy}_{z_j \rightarrow w_i}, & \text{for classification} \end{cases} \quad (3.37)$$

and

$$M_i = \begin{cases} \sqrt{\text{var}(w_i)}, & \text{for regression.} \\ \text{accuracy}_{z_j \rightarrow w_i}^{\text{const}}, & \text{for classification.} \end{cases}$$

We used the SciPy [87] implementation of a k -nearest-neighbors classifier and regressor with default settings ($k = 5$) to measure the **DS**. The regressor/classifier was trained on 80% of the test data and evaluated on the remaining 20%.

Synthetic datasets

The linear synthetic dataset is generated with a transformation $f_{\text{lin}}: \mathbb{R}^2 \rightarrow \mathbb{R}^3$, mapping a unit square $V = [0, 1]^2$ to a 3-dimensional space. The transformation can be decomposed into:

- stretching along one axis by a fixed factor of 2,
- trivial embedding into \mathbb{R}^3 ,
- rotation of 45° along the line containing the vector $(1, -1, 1)$.

For the nonlinear dataset, the transformation $f_{\text{non-lin}}: \mathbb{R}^2 \rightarrow \mathbb{R}^6$ is realized by a random initialization of a **Multilayer Perceptron (MLP)** with one hidden layer (width 10), biases and **tanh** nonlinearities. Both datasets consist of 50000 samples.

3.3.2 Polarized regime

In Section 3.2.2, we assumed **VAEs** operate in a polarized regime and approximated L_{KL} , the **KL** term of the implemented objective (2.6), with $L_{\approx\text{KL}}$ (3.5). In Table 3.2 we show that the polarized regime dominates the training in all examples after a short initial phase. We report the fraction of the training time in which the relative error

$$\Delta_{KL} = \frac{|L_{\text{KL}} - L_{\approx\text{KL}}|}{L_{\text{KL}}} \quad (3.38)$$

stays below 3% continuously until the end (evaluated every 500 batches). Active variables can be selected by $\sqrt{\text{var}(\mu_j(\mathbf{x}^{(i)}))} > 0.5$.

Table 3.2: **Validity of polarized regime:** Percentage of training time where $\Delta_{KL} < 3\%$ (Equation (3.38)) continuously until the end. Reported for **β -VAE** with exact (dataset dependent) and high (10) latent dimension.

	β-VAE (dep.)	β-VAE (10)
dSprites	97.8 %	90.6 %
fMNIST	99.8 %	97.7 %
MNIST	99.8 %	99.5 %
Synth. Lin.	99.8 %	96.7 %
Synth. Non-Lin.	99.9 %	98.5 %

Table 3.3: **Orthogonality and disentanglement:** Results for the distance to orthogonality **DtO** of the decoder (Equation (3.25)) and **DS** for different architectures and datasets. Lower **DtO** values are better and higher **DS** values are better. Random decoders provide a simple baseline for the numbers.

		β -VAE	VAE	AE	β -VAE $_{\Sigma}$	Random Decoder
dSprites	DS \uparrow	0.33 \pm 0.15	0.21 \pm 0.10	0.09 \pm 0.04	0.12 \pm 0.06	1.86 \pm 0.11
	DtO \downarrow	0.76 \pm 0.08	1.08 \pm 0.15	1.62 \pm 0.03	1.73 \pm 0.14	
Synth. Lin.	DS \uparrow	0.99 \pm 0.01	–	0.71 \pm 0.19	0.71 \pm 0.31	0.79 \pm 0.21
	DtO \downarrow	0.00 \pm 0.00	–	0.33 \pm 0.18	0.34 \pm 0.35	
Synth. Non-Linear.	DS \uparrow	0.73 \pm 0.16	–	0.59 \pm 0.30	0.42 \pm 0.24	0.89 \pm 0.16
	DtO \downarrow	0.18 \pm 0.02	–	0.54 \pm 0.13	0.55 \pm 0.02	
MNIST	DtO \downarrow	–	1.59 \pm 0.08	1.83 \pm 0.05	1.93 \pm 0.08	2.11 \pm 0.11
fMNIST	DtO \downarrow	–	1.36 \pm 0.05	1.87 \pm 0.03	2.02 \pm 0.08	2.11 \pm 0.11

3.3.3 Orthogonality and disentanglement

Now, we provide evidence for Theorem 1 by investigating the **DtO** (Equation (3.25)) for a variety of architectures and datasets, see Table 3.3. The results support the claim that the **VAE**-based architectures indeed strive for local orthogonality. By generalizing the β -VAE architecture, such that the approximate posterior is any multivariate Gaussian (β -VAE $_{\Sigma}$), the objective becomes rotationally symmetric (just as the idealized objective). As such, no specific alignment is prioritized. The **AE** also does not favor particular orientations of the latent space.

Some dataset-architecture combinations listed in Table 3.3 are omitted for the following reasons. On the one hand, calculating the **DS** for MNIST and fMNIST does not make sense, as the generating factors are not given (the categorical label cannot serve as a replacement). Consequently, as the values of β are chosen according to this score, we do not report β -VAE numbers for these datasets. On the other hand, for either synthetic task, the regular **VAE** vastly over-prunes, see Figure 3.7, and the values become meaningless.

Another important observation is the clear correlation between **DtO** and **DS**. We show this in Figure 3.4 where we plot results from different restarts of the same β -VAE architecture on the dSprites dataset. We used the literature value $\beta = 4$ [13].

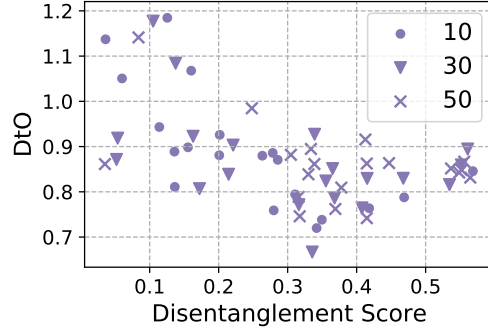


Figure 3.4: **Orthogonality vs. Disentanglement:** Axis alignment of the latent representation (low **DtO**) results in better disentanglement (higher score). Each data point corresponds to an independent run with 10, 30, or 50 epochs.

3.3.4 Degenerate case

Proposition 1 insists that the locally linearized decoder have distinct singular values. Otherwise, the orthogonality of the column vectors does not translate into preserving axes. Here, we design an experiment showing that this condition is relevant in practice.

The dataset in question will be a version of the linear synthetic task where the generating factors have the same scaling, as visualized in the upper plot of Figure 3.5. Note that any linear encoder applying a simple rotation has both orthogonal columns and equal singular values. But it does not respect the alignment of the original square, as it does not meet the assumptions of Proposition 1.

3 The connection between PCA and VAEs

The behavior of the β -VAE with a linear encoder/decoder network is consistent with this. The bottom part of Figure 3.5 shows β -VAE latent representations of four random restarts; they expose random alignments. The same effect results in high variances for both DS and DtO, as shown in Table 3.4. This degeneracy also occurs for PCA. It is easy to check that *any*

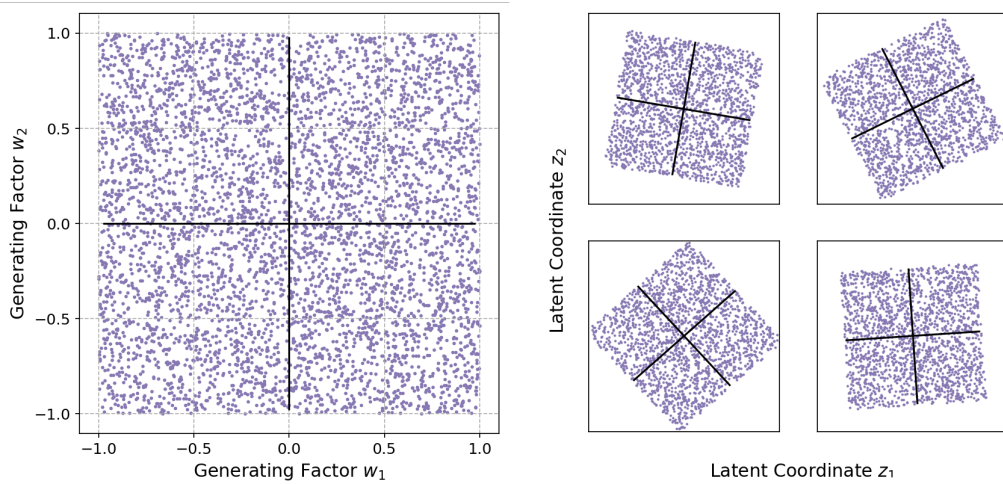


Figure 3.5: **Degenerate singular values:** For strong degeneracy, e.g., in the synthetic dataset with the two generating factors w_1 and w_2 on equal, uniform scale (top), the linear β -VAE generates arbitrarily rotated latent representations (bottom) here for the linear synthetic dataset.

projection of a unit square on a line *has equal variance*. Hence the greedy PCA algorithm has no preference over which alignment to choose, and the practical choice of alignment is implementation dependent.

This insight reinforces our point that β -VAE (just like PCA) looks for sources of variance rather than for statistical independence. We can also see in Table 3.4 that the degeneracy disappears even for small rescaling of the ground truth factors. Since β -VAE promotes normalized latent representations (zero mean, unit variance), the singular values will no longer be equal, and the correct alignment is found. The same is true for PCA.

Table 3.4: **Degenerate singular values:** Overview of DS and DtO for different ratios of importance between the generating factors for the Synth. Lin. task. A ratio of 1.2 means one generating factor is scaled by 1.2.

Ratio	1.0	1.2	1.5
DS	0.51 ± 0.28	0.76 ± 0.25	0.98 ± 0.06
DtO	0.49 ± 0.32	0.20 ± 0.24	0.01 ± 0.06

3.3.5 Nonlinear VAE eigenfaces

In order to highlight the connection with PCA, we use β -VAE to produce a nonlinear version similar to the classical Eigenfaces [88] on the CelebA dataset [40]. Figure 3.6 shows a discrete latent traversal. Starting from the latent representation z_{mean} of the mean face (over 300 randomly selected datapoints) we feed $\{z_{\text{mean}} \pm \alpha e_i\}$ through the decoder, where e_i are the canonical base vectors. Particularly, we chose i covering the first 5 latent coordinates, sorted by the mean σ_j . The parameter $\alpha = 2.5$ was empirically chosen to be on near the tails of the distribution over z^k .

We can see that, unlike classical eigenfaces that mostly reflect photometric properties, the “nonlinear eigenfaces” capture semantic features of the data. Note also that the ordering of the “PCs” by the mean values of σ_j is naturally justified by our work. As was illustrated in Section 3.2.4, the first β -VAE “PCs” also focus on characteristics with high impact on the reconstruction loss (i.e., capture the most variance),

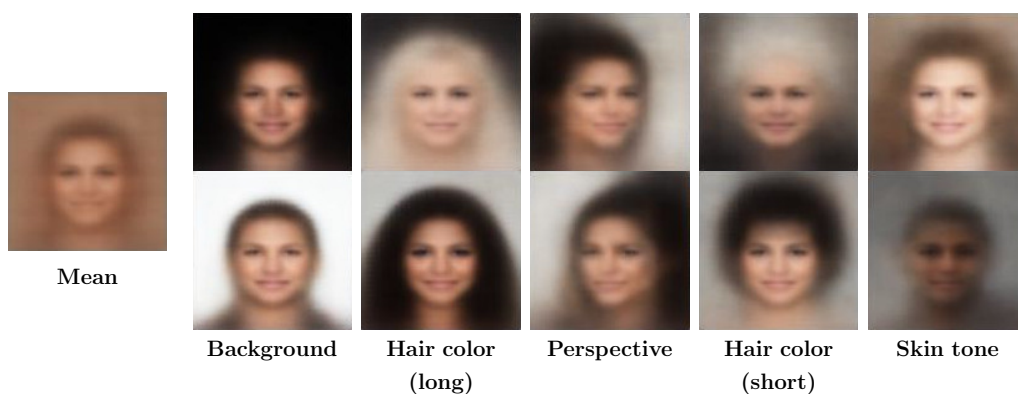


Figure 3.6: **Nonlinear Eigenfaces:** Similarly to the work about Eigenfaces [88], β -VAEs allow for learning their nonlinear counterpart.

3.3.6 Dependence of MIG and DtO on β

The choice of β depends on the achievable DS. Figure 3.7 shows a more thorough analysis of the dependence of both the DS and the DtO.

For too small values of β , the effect of the KL term (and thus the orthogonalization) is negligible. In the other extreme case, too large values of β result in over-pruning, such that the number of active latent coordinates drops below the number of generating factors. This behavior becomes particularly visible for the linear synthetic dataset: By increasing β , the disentanglement score reaches almost 1.0 and forms a plateau. In this region, the latent space is regularized correctly, and the decoder matches the generating function. However, if the regularization strength increases further, the latent space collapses in discrete stages. The optimal hyperparameter should therefore be the one that maximizes the DS (or minimizes the DtO).

The range of interest for β depends on the dataset. As it acts as a regularization to the reconstruction objective, the optimal value of β depends on the gradients of the reconstruction loss. Their magnitude, however, depends, amongst other things, on the data’s variance, potentially the data’s

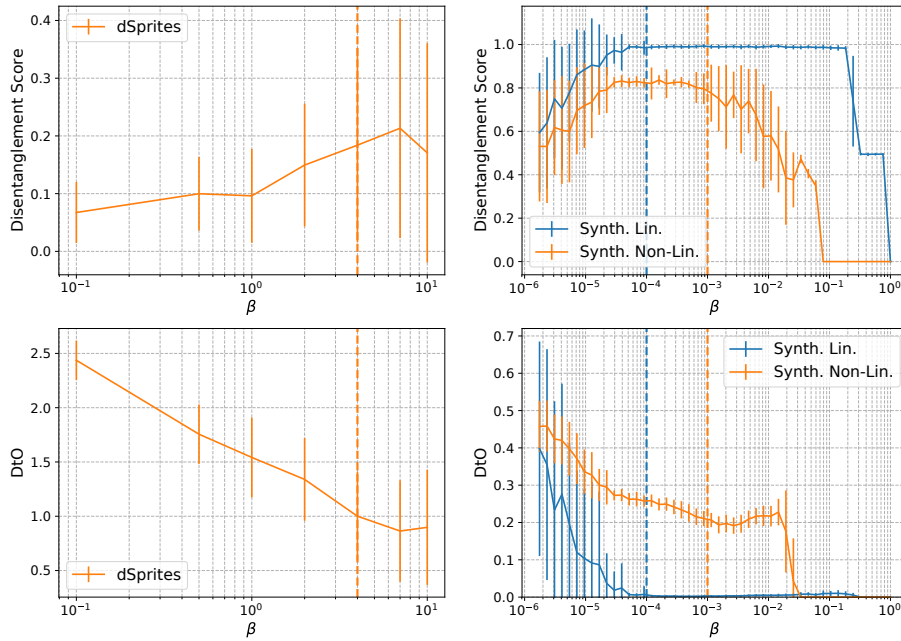


Figure 3.7: **Linesearch over β** : The hyper-parameter in the β -VAE allows to trade off reconstruction error and the KL loss. The plots show the DS (top) and the DtO (bottom) for dSprites (left) and synthetic datasets (right). The dashed lines indicate the parameter chosen for the experiments.

dimensionality, the latent space size, the model capacity, etc.. As a tip for practitioners: It makes sense to keep track of the expectation of the posterior variances over the dataset $\mathbb{E}_{\mathbf{x}} [\sigma_j^2(\mathbf{x})]$ and keep increasing β until the mean collapses to approximately 1 for at least one latent coordinate. This is the value at which the model enters the polarized regime, and the optimal value of β is likely in that order of magnitude, depending on the latent space dimensionality.

3.4 Conclusion

This chapter presented Theorem 1, stating that the columns of the decoder Jacobian strive towards orthogonality. The experimental Section 3.3 provided supporting evidence for the statement, the validity of the assumptions made in its derivation, as well as an ablation study pointing out the relevance of the canonical latent posterior choice. The models were trained using a standard experimental setup and were evaluated using a simple extension of MIG to continuous variables. Aside from the classical datasets dSprites and MNIST, we also tested low-dimensional synthetic datasets.

Starting in Section 3.3.2, we measured the validity of the assumption of the polarized regime. The polarized regime is specified as the situation in which, in the case of a sizeable latent space, some coordinates experience a posterior collapse. At the same time, the encoder is almost deterministic (has a small variance) for the others. The normalized difference between the actual KL loss and the KL loss on only the active variables was below 3% for the vast majority of the training time (> 90.6%). It took longer to prune a larger latent space to the correct number of active latent space dimensions than starting with the ground truth number of generating factors. Yet, in both cases, the assumption of the polarized regime was wholly justified, given the correctly tuned hyperparameters.

With this basic requirement fulfilled, we analyzed the orthogonality of the models using the proposed DtO in Section 3.3.3. We could conclusively show that the β -VAE reliably produces more orthogonal models than other models. In line with the presented theory, a β -VAE with a full covariance matrix latent posterior does not inherit the same strive for orthogonality. It is interesting to note that the DtO was still comparatively large for the dSprites dataset. We believe that this is linked to the degeneracy of the ground truth generating factors *position x* and *position y*, similar to the low dimensional degenerate example in Section 3.3.4.

The applicability of the disentangling abilities of β -VAEs on datasets without a known underlying generative process is equally important. In Section 3.3.5, we tested them on the CelebA dataset and visualized the model similar to the well-known eigenfaces [88]. Due to the orthogonality, the model can retrieve semantically meaningful latent variables, such as hair color, perspective, gender, and more.

The local orthogonality of VAEs is a similarity to classical PCA, a well-known and frequently used tool for dimensionality reduction and factor analysis. The fact that VAE-based architectures behave according to the same variance-isolating mechanism not only shines a light on their inner workings but also advertises them as a nonlinear alternative for data analysis. Particularly if the data structure is assumed to have nonlinear directions of variance that need to be isolated, VAE-based methods can be key architectures. However, it is essential to ensure that the models work in the intended way, which requires decent hyper-parameter choices as shown in Section 3.3.6. In the case of β -VAE, the hyper-parameter β has to be tuned with care to ensure that the polarized regime is reached without over-pruning the latent space. This can be done empirically by increasing β while monitoring the mean standard deviations of the latent space posterior per coordinate. In the next chapter, the connection to PCA and its implications on the nonlinear model are discussed in further detail.

Chapter 4 | The inductive bias of VAEs and datasets

This chapter is based on:

**Demystifying inductive biases for
 β -VAE based architectures [21]**
Dominik Zietlow, Michal Rolínek, Georg Martius

Published at ICML 2021

<https://arxiv.org/abs/2102.06822>

Contributions:

- 80 % Scientific ideas
- 100 % Data generation
- 80 % Analysis & interpretation
- 80 % Paper writing

4.1 Motivation

The performance of VAEs and their variants on learning semantically meaningful, disentangled representations is unparalleled. However, there are theoretical arguments suggesting the impossibility of unsupervised disentanglement [15]. In the previous chapter, we sparked an explanation for this apparent contradiction by elucidating that VAEs share crucial characteristics with PCA, namely the local orthogonality of the decoder. This chapter extends this by shedding light on the inductive bias responsible for the success of VAE-based architectures. We show that the structure of variance induced by the generating factors in classical datasets is conveniently aligned with the directions fostered by the VAE objective. This builds the pivotal bias on which the disentangling abilities of VAEs rely. By small, elaborate perturbations of existing datasets, we hide the convenient correlation structure that is easily exploited by a variety of architectures. To demonstrate this, we construct modified versions of standard datasets in which

- (i) the generative factors are perfectly preserved, e.g., the same generating factors are fully expressive for the modified datasets;
- (ii) each image undergoes only a mild and local transformation causing a small change of variance;
- (iii) the leading VAE-based disentanglement architectures fail to produce disentangled representations while the performance of non-variational methods remains unchanged.

As before, we treat the term disentanglement as the ability to recover the *true generating factors* of data. It was explained by [15] that the concept of generative factors is already compromised from a statistical perspective: Two (in fact, infinitely many) sets of generative factors can generate statistically indistinguishable datasets. Yet, the scores on the disentanglement benchmarks are high and continue to rise. This apparent contradiction stems from biases in used datasets, metrics, and architectures. It was concluded in [89] that

[...] future work on disentanglement learning should be explicit about the role of inductive biases and (implicit) supervision [...].

which did not happen for the majority of existing unsupervised approaches. We close this gap for VAE-based architectures on the two most common datasets, namely dSprites [84] and Shapes3d [90].

The central hypothesis we want to prove in this chapter is that all unsupervised, VAE-based disentanglement architectures are successful because they exploit the same structural bias in the data. The ground truth generating factors are well aligned with “to-be” PCs that VAEs strive for. This bias can be reduced by introducing a slight change in the local correlation structure of the input data, which, however, perfectly preserves the set of generative factors. We evaluate a set of models on slightly modified versions of the two leading datasets in which each image undergoes a modification inducing slight variance. We report drastic drops in disentanglement performance on the altered datasets.

On a technical level, we build on the findings of Chapter 3, where we argued that VAEs share a similarity with PCA. We extend this argument by an additional finding that further strengthens this connection and yields in equivalence between the two methods in the linear setting. In other words, VAEs recover a set of scalars that embody the sources of variance in the data. We propose minor modifications of the datasets that aim to change the leading principal components by adding modest variance to a set of alternative candidates. The “to-be” leading principal components are specific to each dataset, but they are automatically determined in a consistent fashion.

4.2 Methods

We firstly show that linear VAEs indeed fully recover the principal directions of PCA, secondly introduce the general data generation scheme of commonly used disentanglement datasets, and lastly turn this understand-

ing into an experimental setup that allows for empirical confirmation that the success of VAE-based architectures mostly relies on the local structure of the data. By locally perturbing existing datasets, we observe that the “nonlinear PCs” can be tempered without much shifting the dataset’s variance. The resulting dataset can not be disentangled using VAE based architectures. We thereby provide a precise answer to the question of what the inductive biases on the model- and data-side are that allow for VAEs disentanglement properties. Interestingly, the discovered drop in disentanglement performance extends beyond fully unsupervised methods to presumably identifiable models.

4.2.1 Theoretical support of the connection to PCA

We analyze a linear VAE with models $\mu^{(i)} = M_E \mathbf{x}^{(i)}$, $\text{Dec}_\theta(\mathbf{z}^{(i)}) = M_D \mathbf{z}^{(i)}$ and denote the SVD decomposition of M_D as $M_D = U \Sigma V^\top$. We can now state a constrained optimization problem similar to (Equations (3.14) and (3.15)) as

$$\min_{\Sigma, U, V} \mathbb{E}_i (\|U \Sigma V^\top \varepsilon^{(i)}\|^2) \quad (4.1)$$

$$\text{s.t. } \mathbb{E}_i (\mathcal{L}_{\approx \text{KL}}^{(i)}) = c_{\approx \text{KL}}. \quad (4.2)$$

where only the stochastic part of the reconstruction loss is minimized and $c_{\approx \text{KL}}$ is a constant. The term $\mathcal{L}_{\approx \text{KL}}$ again is the KL loss in the polarized regime, defined in Equation (3.5).

As described in Section 2.1, the “decoder matrix” of the classical PCA contains the eigenvectors of the covariance matrix C . By SVD decomposing the zero-mean data matrix $X = U_X \Sigma_X V_X^\top$, we find

$$C = X^\top X = V_X \Sigma_X^2 V_X^\top. \quad (4.3)$$

For encoding data with PCA, the eigenvectors of V_X are typically sorted according to their eigenvalue by a permutation matrix P , which leads to the PCA decoder as

$$M_{\text{PCA}} = V_X^\top \Sigma_X^2 P. \quad (4.4)$$

To tighten the connection between **VAEs** and **PCA**, we compare $M_D = U\Sigma V^\top$ to $M_{\text{PCA}} = V_X^\top \Sigma_X^2 P$.

Theorem 2 (Linear **VAEs** perform **PCA**). *In a setting that precisely isolates the freedom in choosing U , Σ , and V , and under mild non-degeneracy assumptions (full description is available in Chapter 5), the following holds: For any $X \in \mathbb{R}^{n \times m}$, the solution to (Equations (4.1) and (4.2))*

$$\Sigma^*, U^*, V^* = \arg \min_{\Sigma, U, V} \mathbb{E}_i (\|U\Sigma V^\top \varepsilon^{(i)}\|^2), \quad (4.5)$$

satisfies (in a “**PCA-like**” way)

$$\begin{aligned} V^* & \text{ is a signed permutation matrix,} \\ U^* & = V_X^\top. \end{aligned}$$

It was known for long that linear autoencoders, trained on L^2 reconstruction loss, span the same space as **PCA** [91, 92]. The additional similarity that **VAEs** produce orthogonal mappings, like **PCA**, was presented in Chapter 3. With the final connection presented here, even the embedding alignment is shown to be identical. For the sake of better readability, the proofs of the statements can be found in Chapter 5.

Although this does not directly translate to a universal statement about the linearization of a nonlinear model, it also provides an intuition for that case. An important observation is that **the alignment of the latent space is mostly driven by the distribution of the latent noise**. When generalizing this statement to the linearization of a nonlinear decoder, the effect of the noise stays local. Consequently, local changes in the data distribution can potentially lead to a disruptive change in the latent alignments without inducing large global variance. This idea is depicted in Figure 4.1.

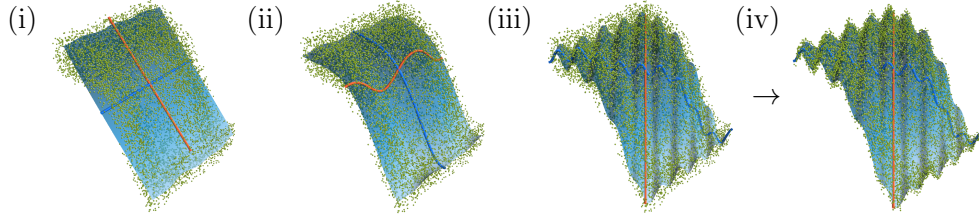


Figure 4.1: **Linear and nonlinear embeddings:** From left to right: (i) a 3-D point cloud and the corresponding 2-D **PCA** manifold (blue surface) with the canonical principal components (red/blue curves), (ii) a nonlinear 2-D manifold with its principal components, (iii) a locally perturbed 2-D manifold with its principal components which are rotated with respect to (ii), (iv) the goal of our modifications is to move each datapoint closer to this *entangled* manifold.

4.2.2 The generative process

The standard datasets for evaluating disentanglement all have an explicit generation procedure. Each data point $\mathbf{x}^{(i)} \in \mathcal{X}$ is an outcome of a generative process g applied to input $\mathbf{w}^{(i)} \in \mathcal{W}$. Imagine that g is a function rendering a simple scene from its specification w containing *as its coordinates* the background color, foreground color, object shape, object size, etc. By design, the individual generative factors are statistically independent in \mathcal{W} . All in all, the dataset $\mathcal{X} = (\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n)})$ is constructed with $\mathbf{x}^{(i)} = g(\mathbf{w}^{(i)})$, where g is a mapping from the generative factors to the corresponding data points.

In this chapter, we design a modification \tilde{g} of the generative procedure g that changes the local structure of the dataset \mathcal{X} while barely distorting each individual data point. In particular, for each $\mathbf{x}^{(i)} \in \mathcal{X}$, we have under some distance measure $d(\cdot, \cdot)$, that

$$d(\mathbf{x}^{(i)}, \tilde{g}(\mathbf{w}^{(i)})) \leq \varepsilon. \quad (4.6)$$

How to design \tilde{g} such that despite an ε -small modification, VAE-based architectures will create an entangled representation? Following the intuition from Chapter 3, Figure 2.3 and Figure 4.1, we *misalign* the local variance with respect to the generating factors in order to promote an alternative (entangled) latent embedding. This is precisely the step from (iii) to (iv) in Figure 4.1.

To avoid hand-crafting this process, we can exploit the following observation. VAE-based architectures suffer from significant performance variance over different random initializations. This hints at an existing ambiguity: Two or more candidates for the latent coordinate system are competing minima of the optimization problem. Some of these solutions perform well, others are “bad” in terms of disentanglement – they correspond to (ii) and (iii) in Figure 4.1 respectively. Below, we elaborate on how to foster the entangling and diminish the disentangling solutions.

Our modifications are not an implementation of [15, Theorem 1]. We **do not modify the set of generative factors, but slightly alter the generating process** to target a specific subtlety in the inner working of VAEs.

Given any dataset, our modification process has three steps:

- (i) Find the most disentangled and the most entangled latent space alignment that a β -VAE produces over multiple restarts.
- (ii) Optimize a generator that manipulates images to foster and diminish their suitability for the entangled and disentangled model respectively.
- (iii) Apply the manipulation to the whole dataset and compare the performance of models trained on the original and the modified dataset.

4.2.3 Choice of fostered latent coordinate system

Over multiple restarts of β -VAE, we pick the model with the lowest MIG score. This gives us an entangled alignment that is expressible by the architecture. Although any choice of metric is valid for this model selection (e.g., Unsupervised Disentanglement Ranking [71]), we chose MIG for the sake of simplicity. The latent variables of each model capture the data’s nonlinear PCs. Similarly to PCA, we can order them according to the variance they induce. The order is inversely reflected by the magnitude of the latent noise values. We find the j ’th principal components $s_j^{(i)}$ as

$$s_j^{(i)}(\mathbf{x}^{(i)}) = \text{enc}(\mathbf{x}^{(i)})_{k^{(j)}} \quad (4.7)$$

$$k^{(j)} = \arg \min_{l \notin \{k^{(0)}, k^{(1)}, \dots, k^{(j-1)}\}} \langle \sigma_l^2 \rangle. \quad (4.8)$$

This procedure of sorting the most *important* latent coordinates is consistent with [13] and [20]. The analogy to PCA is that the mapping $s^{(j)}(\mathbf{x}^{(i)})$ gives the j ’th coordinate of $\mathbf{x}^{(i)}$ in the new (nonlinear) coordinate system.

4.2.4 Dataset manipulations

We will now describe the modification procedure assuming the data points are $r \times r$ images. The manipulated data-point $\mathbf{x}'^{(i)}$ is of the form $\mathbf{x}'^{(i)} = \mathbf{x}^{(i)} + \varepsilon m(\mathbf{w}^{(i)})$ where the mapping $m: \mathbb{R} \rightarrow \mathbb{R}^r \times \mathbb{R}^r$ is constrained by $\|m(\mathbf{w}^{(i)})\|_\infty \leq 1$ for every $\mathbf{w}^{(i)}$. Then inequality 4.6 is naturally satisfied for the maximum norm.

The abstract idea of achieving a change of the latent embedding coordinate systems can be visualized using the intuition following from Equation (4.8). We can think of two VAE latent spaces where one is considered disentangled ($\{\mu_{\text{dis}}^{(i)}, \sigma_{\text{dis}}^{(i)}\}$) and the other is entangled ($\{\mu_{\text{ent}}^{(i)}, \sigma_{\text{ent}}^{(i)}\}$), as two sets of nonlinear principal directions, and the variance each of the dimensions capture is reflected in the magnitude of $\sigma^{(i)}$. We aim to alter the dataset such that its entangled representation is superior to the disentangled representation,

in the sense of being *cheaper* to decode with respect to the reconstruction loss. In other words, projecting the dataset to the manifold supported by $\mathbf{z}_{\text{ent}}^{(i)}$ should result in a lower reconstruction loss than projecting it to the manifold supported by $\mathbf{z}_{\text{dis}}^{(i)}$. A naive way of doing so is by moving each image closer to its projections on the first principal components of the entangled representation and further away from those of the disentangled representation. Instead of hand-crafting this operation, we can optimize for it directly.

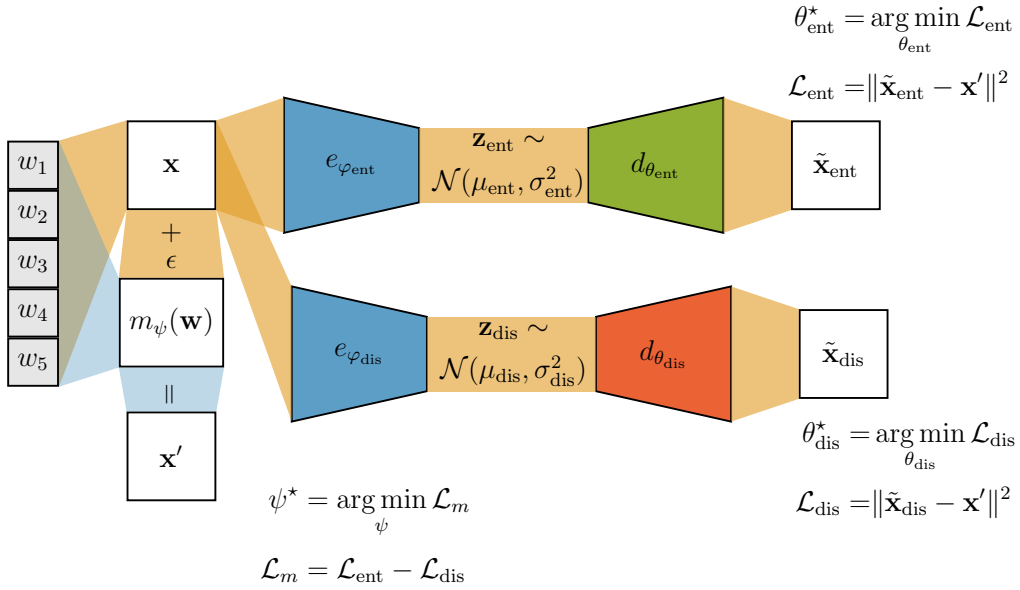


Figure 4.2: **Image perturbation process:** Starting from ground truth generating factors \mathbf{w} , two β -VAE encoder-decoder pairs are initialized such that one (top) produces entangled and the other (bottom) disentangled representations. Another decoder-like network m is trained to produce additive manipulations to the original images x . The encoders are frozen and fed with the original images. The set of ground truth generating factors \mathbf{w} stays untouched by the modification.

4 The inductive bias of VAEs and datasets

This idea can be turned into an end-to-end trainable architecture as depicted in Figure 4.2. We want to change the dataset such that it is more convenient to encode it in an entangled way. Starting with two pre-trained models, we fix their encoders and keep feeding them the original images. This ensures that the latent encoding stays unchanged, as we want to compare their suitability for reconstruction. The decoders are trained to minimize the reconstruction loss given the entangled representation:

$$\begin{aligned}\theta_{\text{ent}}^* &= \arg \min_{\theta_{\text{ent}}} \mathcal{L}_{\text{rec}}^{\text{ent}} \left(\mathbf{x}'^{(i)}, \mathbf{z}^{(i)} \right), \\ \theta_{\text{dis}}^* &= \arg \min_{\theta_{\text{dis}}} \mathcal{L}_{\text{rec}}^{\text{dis}} \left(\mathbf{x}'^{(i)}, \mathbf{z}^{(i)} \right).\end{aligned}$$

We initialize this network with the parameters of the disentangled model $\theta_{\text{dis}}, \varphi_{\text{dis}}$ and the entangled model $\theta_{\text{ent}}, \varphi_{\text{ent}}$ respectively. We introduce a network to learn the additive manipulation, m_ψ with parameters ψ . The parameters are trained to minimize the reconstruction loss of the entangled VAE and to increase the loss of the disentangled VAE via its effect on the dataset:

$$\psi^* = \arg \min_{\psi} \left(\mathcal{L}_{\text{rec}}^{\text{ent}} \left(\mathbf{x}'^{(i)}, \mathbf{z}^{(i)} \right) - \mathcal{L}_{\text{rec}}^{\text{dis}} \left(\mathbf{x}'^{(i)}, \mathbf{z}^{(i)} \right) \right).$$

It is worth noting that both latent spaces were suitable for reconstructing the images of the original dataset. **The major play that the network m_ψ has is to utilize the different ways the noise was distributed across the latent space.**

4.3 Experiments

To experimentally validate the soundness of the manipulations, we need to demonstrate the following:

1. **Effectiveness of manipulations.** Disentanglement metrics should drop on the altered datasets across VAE-based architectures. We do not expect changes in non-variational methods, as the magnitude of the perturbations is fairly small.
2. **Comparison to a trivial modification.** Instead of the proposed method, we modify the data with uniform noise of the same magnitude. The disentanglement scores for the algorithms on the resulting datasets should not drop significantly, as this change does not alleviate the existing bias.
3. **Robustness.** The new datasets should be hard to disentangle even after re-tuning hyperparameters of the original architectures.

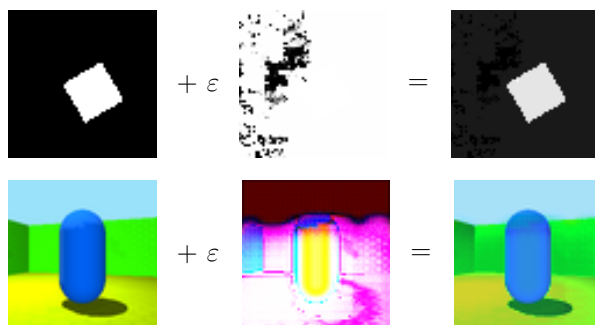


Figure 4.3: **Example perturbations:** From left to right: Original images, manipulations and altered images. Top row shows an example of dSprites, the bottom for Shapes3D.

The primary hyperparameter of each of the models is listed in Table 4.1 and we used the implementations of the Disentanglement Library [15].

Architecture	dSprites	Shapes3D
β -VAE (β)	8	32
TC-VAE (β)	6	32
FactorVAE (γ)	35	7
SlowVAE (β)	1	1

Table 4.1: **Primary hyperparameters**, we used the defaults in the Disentanglement Library or literature values for any other parameter.

4.3.1 Architecture for perturbation network

The model implemented for $m(\mathbf{w})$ has almost the same architecture as the convolutional decoder as it is implemented in the Disentanglement Library. The only difference lies in the input MLP, which was extended by a single neuron hidden layer. This enforces a compression of the generating factors $\mathbf{w}^{(i)}$ to some scalar value based on which the modifications are rendered. Both m and the decoders were trained with Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-7}$) and 10^{-4} learning rate. To ensure training stability, we train the decoders on three times more batches as the manipulation network and reconstruct five latent samples per image to better estimate the stochastic losses. We achieved a better result on Shapes3D when using an ensemble of four disentangling and four entangling encoder-decoder pairs instead of single models. In order to stay in the same value range as the original images, we ensured normalization of the manipulated images $\mathbf{x}'^{(i)} = \mathbf{x}^{(i)} + m(\mathbf{w}^{(i)})$ by $\mathbf{x}'_{\text{norm}}{}^{(i)} = \mathbf{x}^{(i)} - 2\text{ReLU}(\mathbf{x}^{(i)} - 1) + 2\text{ReLU}(-\mathbf{x}^{(i)})$.

4.3.2 Effectiveness of manipulations

We deploy the suggested training for the manipulations on two datasets: Shapes3D and dSprites, leading to manipulations as depicted in Figure 4.3. In terms of models, we trained four VAE-based architectures [13, 14, 41, 17], a regular autoencoder [93], and (as non-variational methods) PCL [64] as well as the weakly supervised GAN from [66] in the full sharing setting. We evaluate on both the original and manipulated datasets. Regularization strengths are used as reported in the literature (or better-tuned values), and other hyperparameters are taken from the Disentanglement Library [15]. For simplicity and clarity, we restricted the latent space dimension to equal the number of ground truth generative factors. Most of the architectures are capable of pruning the latent space as a consequence of their intrinsic regularization [94]. While being a perk in real-world application scenarios, this behavior can lead to over- or under-pruning and thereby cloak the actual difference in the alignment of the latent space.

The resulting MIG scores are listed in Table 4.2, other disentanglement metrics are listed in Tables 4.3 to 4.5. We report the performance on the original dataset, the modified dataset, and a dataset corrupted with noise of equal magnitude as the structured perturbation. Across all variational models, the disentanglement quality is significantly reduced when trained on the perturbed datasets (c.f. left two columns). Interestingly, the disentanglement reduces even for SlowVAE, an architecture that supposedly circumvents the non-identifiability problem by deploying a sparse temporal prior. This indicates that the architecture still builds upon the local data structure more than on the weak supervision induced by temporal sparsity. PCL and the weakly supervised GAN, as non-variational methods, perform similarly well on the original and the modified architecture, which is a strong indicator that due to the constraint (4.6), the primary sources of global variance remain unaltered. The modifications only attack the bias VAEs exploit.

4.3.3 Noisy datasets

In this section, we provide an ablation confirming the necessity of the structure in the dataset perturbations. We replace the proposed manipulation by contaminating each image with uniform pixel-wise noise $[-\varepsilon, \varepsilon]$. The value of ε is fixed to the level of the presented manipulations (0.1 for dSprites and 0.175 for Shapes3D). The results are also listed in Table 4.2. The lack of structure in the contamination does not affect the performance in a guided way and leads to minimal effect on Shapes3D. This shows that the induced drop in disentanglement performance does not stem from a shift in the overall variance. The impact on dSprites is, however, noticeable. Due to the comparatively slight variance among dSprites images, the noise conceals the variance from the less important generating factors (such as orientation).

Table 4.2: **MIG scores** for unmodified, modified, and noisy datasets. We report the mean and standard deviation over ten distinct random seeds for each setting. The regular autoencoder serves as a baseline (random alignment). **PCL** and the weakly supervised **GAN** from [66] are the only disentangling non-variational model. The modification leads to a significant drop in all variational methods.

	dSprites			Shapes3d		
	orig.	mod.	noise	orig.	mod.	noise
AE	0.09 ± 0.06	0.05 ± 0.02	0.06 ± 0.03	0.06 ± 0.03	0.05 ± 0.03	0.07 ± 0.03
β-VAE	0.23 ± 0.08	0.07 ± 0.09	0.14 ± 0.07	0.60 ± 0.31	0.09 ± 0.14	0.66 ± 0.05
FactorVAE	0.27 ± 0.11	0.20 ± 0.12	0.16 ± 0.08	0.27 ± 0.18	0.07 ± 0.05	0.33 ± 0.20
TC-VAE	0.25 ± 0.08	0.14 ± 0.10	0.20 ± 0.04	0.58 ± 0.20	0.24 ± 0.16	0.60 ± 0.11
SlowVAE	0.39 ± 0.08	0.27 ± 0.08	0.37 ± 0.09	0.53 ± 0.19	0.13 ± 0.08	0.60 ± 0.10
PCL	0.21 ± 0.03	0.24 ± 0.07	0.24 ± 0.07	0.44 ± 0.06	0.47 ± 0.08	0.40 ± 0.07
Weak sup. GAN	0.45 ± 0.05	0.36 ± 0.02	0.36 ± 0.01	0.69 ± 0.12	0.66 ± 0.12	0.77 ± 0.13

Table 4.3: **DCI scores** for unmodified, modified and noisy datasets.

We report the mean and standard deviation over ten different random seeds for each setting. **PCL** is the only disentangling non-variational model. The modification leads to a significant drop in all variational methods.

	dSprites			Shapes3d		
	orig.	mod.	noise	orig.	mod.	noise
β-VAE	0.11 \pm 0.03	0.08 \pm 0.11	0.14 \pm 0.07	0.73 \pm 0.14	0.43 \pm 0.06	0.56 \pm 0.06
FactorVAE	0.37 \pm 0.10	0.27 \pm 0.11	0.24 \pm 0.09	0.39 \pm 0.18	0.25 \pm 0.08	0.57 \pm 0.20
TC-VAE	0.34 \pm 0.06	0.19 \pm 0.10	0.27 \pm 0.03	0.67 \pm 0.08	0.41 \pm 0.05	0.59 \pm 0.09
SlowVAE	0.47 \pm 0.07	0.40 \pm 0.07	0.47 \pm 0.08	0.65 \pm 0.10	0.33 \pm 0.08	0.73 \pm 0.09
PCL	0.28 \pm 0.03	0.30 \pm 0.03	0.29 \pm 0.06	0.70 \pm 0.06	0.67 \pm 0.09	0.71 \pm 0.07

Table 4.4: **FactorVAE scores** for unmodified, modified and noisy datasets. We report the mean and standard deviation over ten distinct random seeds for each setting. **PCL** is the only disentangling non-variational model. The modification leads to a significant drop in all variational methods.

	dSprites			Shapes3d		
	orig.	mod.	noise	orig.	mod.	noise
β-VAE	0.47 \pm 0.07	0.38 \pm 0.13	0.50 \pm 0.10	0.80 \pm 0.17	0.54 \pm 0.10	0.71 \pm 0.06
FactorVAE	0.67 \pm 0.11	0.62 \pm 0.14	0.60 \pm 0.11	0.63 \pm 0.15	0.48 \pm 0.05	0.71 \pm 0.15
TC-VAE	0.68 \pm 0.09	0.53 \pm 0.15	0.60 \pm 0.12	0.76 \pm 0.07	0.57 \pm 0.07	0.71 \pm 0.06
SlowVAE	0.77 \pm 0.03	0.77 \pm 0.04	0.76 \pm 0.07	0.87 \pm 0.10	0.62 \pm 0.06	0.85 \pm 0.08
PCL	0.77 \pm 0.09	0.82 \pm 0.05	0.77 \pm 0.08	0.80 \pm 0.06	0.77 \pm 0.07	0.80 \pm 0.06

Table 4.5: **SAP scores** for unmodified, modified and noisy datasets.

We report the mean and standard deviation over ten different random seeds for each setting. **PCL** is the only disentangling non-variational model. The modification leads to a significant drop in all variational methods.

	dSprites			Shapes3d		
	orig.	mod.	noise	orig.	mod.	noise
β-VAE	0.04 \pm 0.01	0.02 \pm 0.02	0.03 \pm 0.03	0.16 \pm 0.08	0.03 \pm 0.03	0.09 \pm 0.02
FactorVAE	0.07 \pm 0.03	0.06 \pm 0.03	0.08 \pm 0.01	0.07 \pm 0.04	0.04 \pm 0.01	0.08 \pm 0.03
TC-VAE	0.08 \pm 0.01	0.06 \pm 0.03	0.05 \pm 0.02	0.08 \pm 0.02	0.04 \pm 0.02	0.06 \pm 0.03
SlowVAE	0.08 \pm 0.01	0.07 \pm 0.01	0.07 \pm 0.01	0.09 \pm 0.04	0.04 \pm 0.01	0.09 \pm 0.05
PCL	0.07 \pm 0.03	0.10 \pm 0.03	0.10 \pm 0.03	0.07 \pm 0.01	0.07 \pm 0.01	0.07 \pm 0.01

4.3.4 Robustness over hyperparameters

We run a line search over the primary hyperparameter for each architecture where we scale the optimal (on the original dataset, according to the literature) parameter by 0.75 to 2.0 and evaluate different disentanglement metrics. The models are trained on the modified datasets. Figure 4.4 shows a violin plot over the **MIG** scores and Figures 4.5 to 4.7 show the other metrics for the sake of completeness. Assuming that our modifications are stable against tuning the hyperparameter, one would expect the disentanglement scores to not recover near the level achieved on the original datasets.

Overall our modifications seem mostly robust for adjusted hyperparameters. A significant increase in the regularization strength allowed for some recovery. A more thorough analysis revealed that this effect starts only once the models reach a level of over-pruning, a behavior well known to practitioners. We discard the runs that over-pruned the latent space (number of active coordinates, i.e., for which $\mathbb{E}(\sigma_i^2) < 0.8$, sinks below the dimensionality of the ground truth generating factors). This effect goes along with decreased reconstruction quality and intrinsically prevents the models from recovering all true generating factors and renders these cases uninteresting.

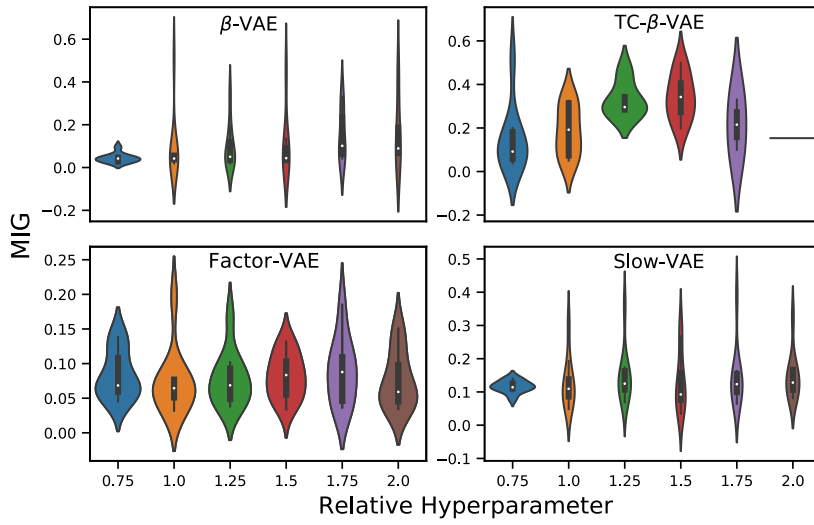


Figure 4.4: **MIG scores** for scaled literature hyperparameters over ten restarts for Shapes3D. Over-pruned models with fewer active units than generating factors were discarded.

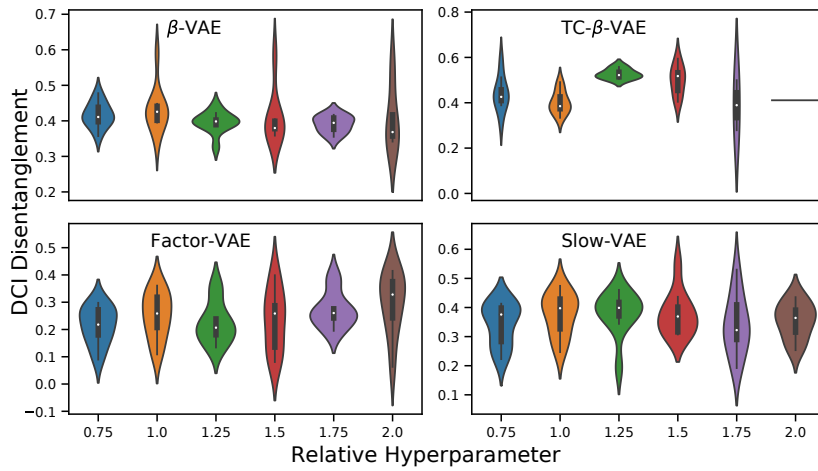


Figure 4.5: **DCI scores** for scaled literature hyperparameters over ten restarts for Shapes3D. Over-pruned models with fewer active units than generating factors were discarded.

4 The inductive bias of VAEs and datasets

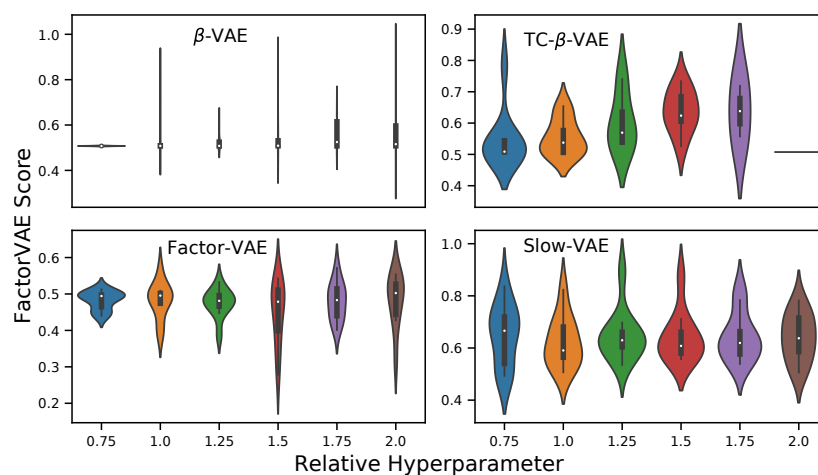


Figure 4.6: **FactorVAE** scores for scaled literature hyperparameters over ten restarts for Shapes3D. Over-pruned models with fewer active units than generating factors were discarded.

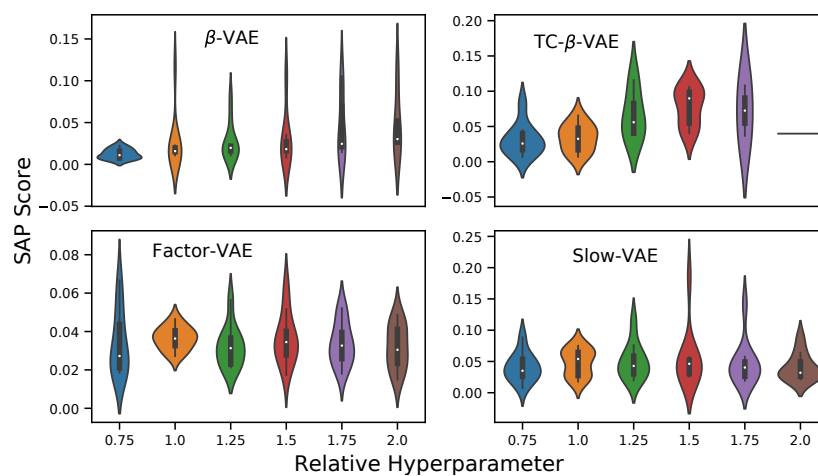


Figure 4.7: **SAP** scores for scaled literature hyperparameters over ten restarts for Shapes3D. Over-pruned models with fewer active units than generating factors were discarded.

4.3.5 Restart statistics and per factor evaluation

Additional information about the distribution of **MIG** scores on the modified datasets on Shapes3D is presented in the histograms of Figure 4.8. Despite the mean **MIG** score dropping significantly when trained on the altered dataset, some models still disentangle reasonably well. One explanation could be that there are two or more nearby solutions for the optimization problem. The manipulations foster the entangled but do not entirely exclude the disentangled solution. We discuss this idea in Section 4.3.6.

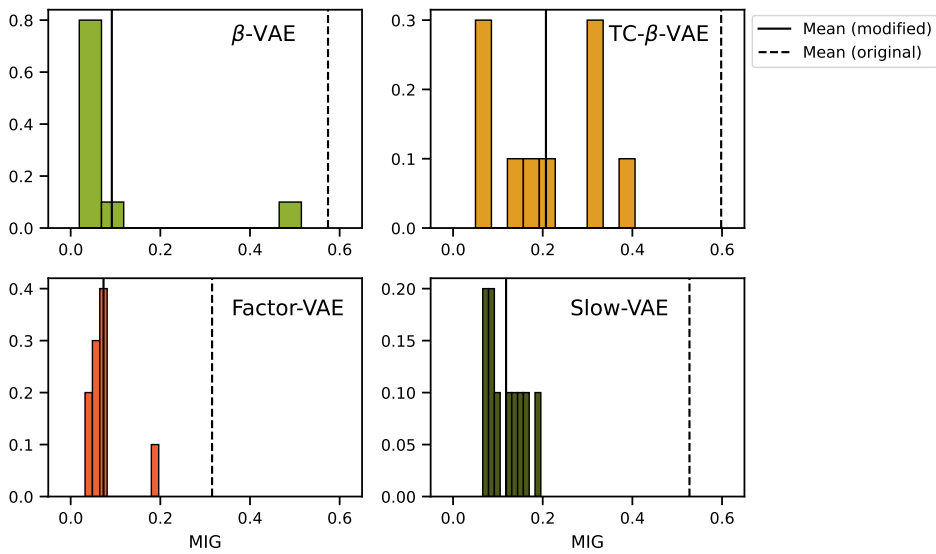


Figure 4.8: **Histogram of MIG scores** for the **VAE**-based methods on the altered Shapes3D dataset. Although the mean **MIG** score is significantly reduced, some models still disentangle reasonably well. We expect that there are two or more nearby solutions for the optimization problem, and the manipulations foster the entangled one but do not fully exclude the disentangled solution.

The individual **MIG** scores per generating factor for the β -VAE on Shapes3D are shown in Figure 4.9. We can see that the **MIG** drops for every generating factor, leading to an overall entanglement.

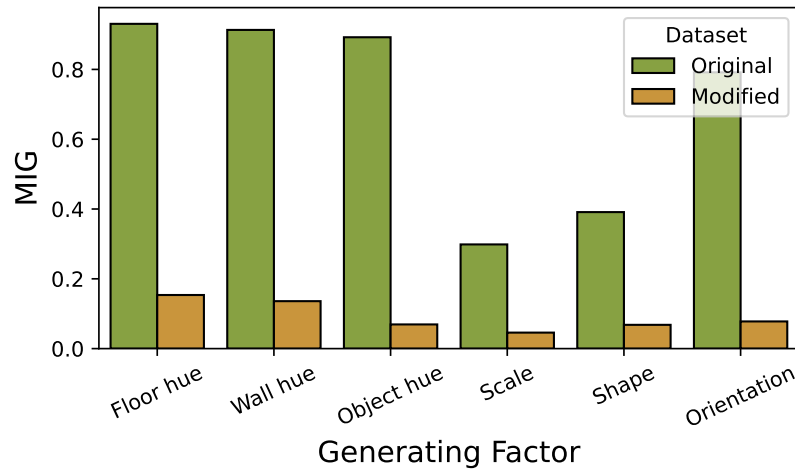


Figure 4.9: **Individual MIG** scores for β -VAEs trained on the original and the altered Shap3D dataset. The **MIG** drops for every generating factor, which leads to entanglement across all of them.

4.3.6 Inspection of entangled and disentangled latent embeddings

Over multiple restarts of β -VAE trainings on the unmodified dataset, we inspect the four runs that achieved the highest and the four that reached the lowest **MIG** scores. Figure 4.10 shows two dimensional latent traversals for the disentangled β -VAE representations. The dimension of the latent traversal was hand-picked to encode the wall hue and the orientation. Interestingly, the models reliably encode the color in the same way (e.g., starting from green to cyan). This is an intriguing finding, as the hue values are uniformly sampled in the dataset, and the hue is by definition a cyclic quantity (i.e., it is the angular component in the **Hue, Saturation, and Lightness (HSL)** color space [95]). Each generative factor is embedded mostly in a single coordinate in those embeddings where the disentanglement scores are high. We refer to this type of embedding as *cartesian*.

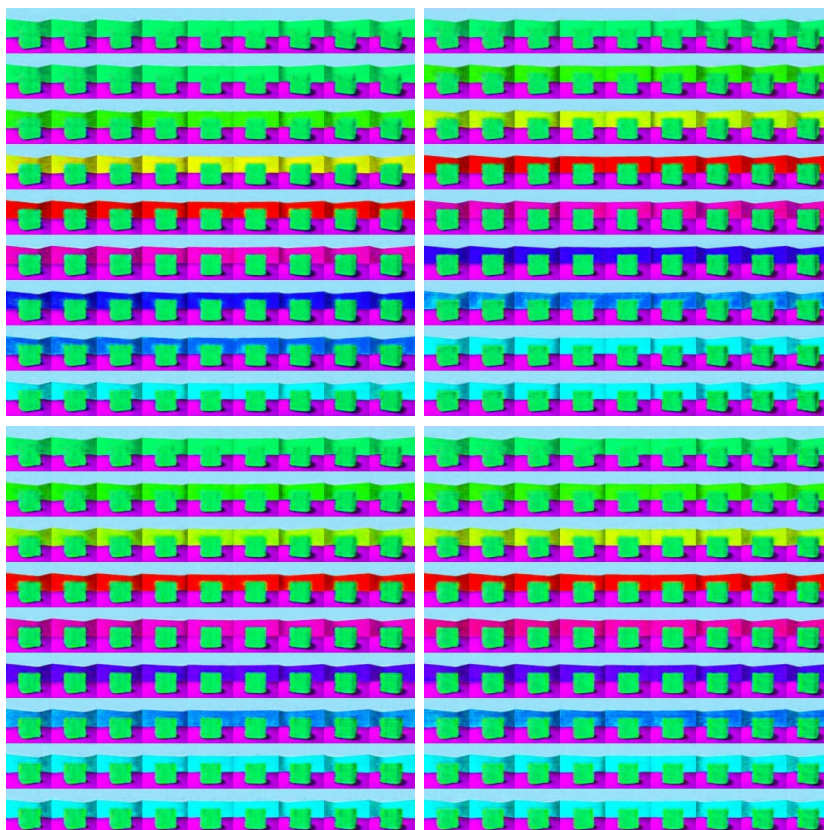


Figure 4.10: **Latent traversals** along two latent dimensions for different disentangled representations from independent β -VAE trainings on the original dataset. They encode the wall hue and orientation separately. We flipped the latent coordinates to match the same alignment.

Figure 4.11, on the other hand, shows latent traversals for the entangled models. Surprisingly, all those embeddings share a very similar structure. They reliably mix the two generating factors in the same way: The color is encoded as the angular component of the two latent dimensions and the orientation as the radial component. This mixing leads to low disentanglement scores, although the representations are very interpretable. The generative factors are captured in a *polar* coordinate system. We found similar behaviors on other datasets, such as NORB [96].

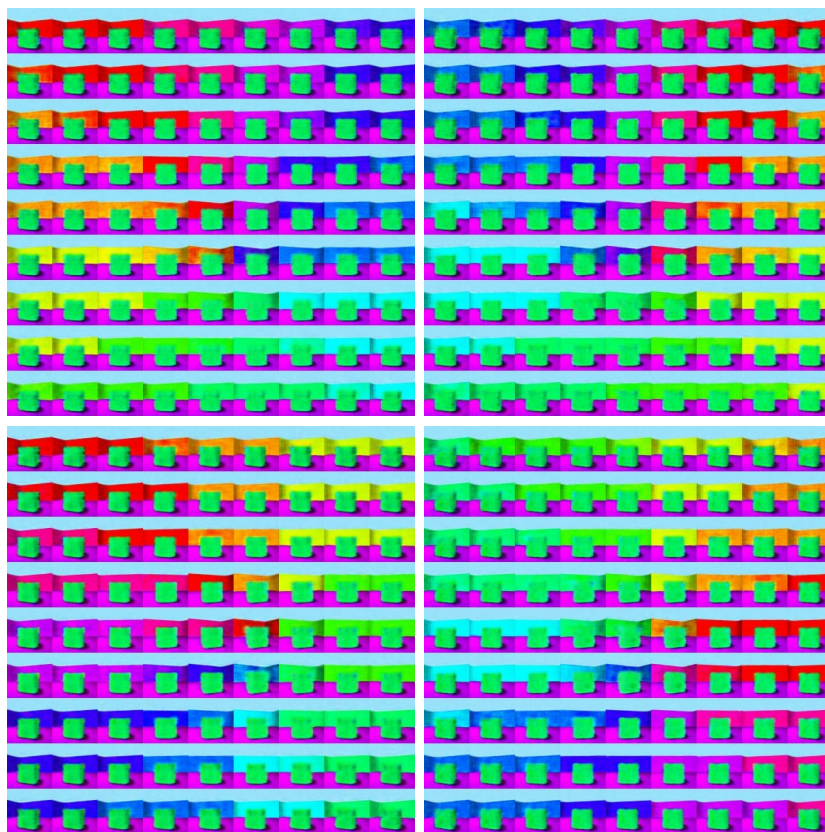


Figure 4.11: **Latent traversals** along two latent dimensions for different entangled representations from independent β -VAE trainings on the original dataset. They encode a mixture of wall hue and orientation.

The two types of encodings (*cartesian* and *polar*) seem to form distinct minima of the β -VAE optimization objective. This also reflects in the bimodality of the histograms or violin plots over disentanglement measures in other work [15].

4.4 Conclusion

The experiments summarized in this chapter showcase the similarity between β -VAEs and PCA beyond the linear case. We designed perturbations of existing datasets that only mildly changed the overall variance induced by the same generating factors as the original datasets. The perturbation only acted within an $\epsilon \ll 1$ range per dimension on each individual data point, thereby only locally changing the variance structure of the dataset (Section 4.3.1). By optimizing the perturbations such that they minimize the reconstruction error on a non-disentangled β -VAE model and maximize it on a disentangled model, we derived a dataset that can no longer be disentangled by any tested VAE-based architecture (Section 4.3.2).

As an ablation study, we prepared another version of a perturbed dataset where the perturbations lack any particular structure, i.e., we apply noise to the dataset with the same variance as the local perturbations. The resulting change in disentangling quality was significantly smaller than the optimized manipulation (Section 4.3.3). This gives additional evidence that the effect of the proposed perturbation relies on the explicit adversarial optimization of the stochastic reconstruction losses.

Since the local structure was changed, one can expect the reconstruction error of any trained model to be different between the original and the perturbed dataset. We therefore conducted line-searches across the primary hyper-parameter of each model (Section 4.3.4). Those conclusively showed that the drop in disentangling capabilities does not stem from a shift in the trade-off captured via the primary hyper-parameter of each model.

We evaluated where the drop in disentangling performance originates from, i.e., which generating factor is the primary source of decreased performance (Section 4.3.5). The analysis revealed that the MIG scores decrease for all generating factors alike. Surprisingly, an analysis of multiple restarts revealed that there are still some that perform well.

4 The inductive bias of VAEs and datasets

A very intriguing observation, linked to the fragility of VAE-based methods, was discussed in Section 4.3.6. Based on multiple selected best- and worst-performing runs, it was shown that there are at least two ways in which a β -VAE can encode generating factors. One is the traditional cartesian way, where one generating factor is embedded in one latent dimension. The other one can be described as a polar embedding, where two generating factors are encoded as the radial and angular components of two latent representations (it is unclear if there are more angular components). Whether the one or the other embedding is more desirable depends on the task at hand, whereas most disentanglement metrics favor the cartesian representation. This is an example of a shortcoming of most existing disentanglement metrics and an indicator of the human bias involved in disentangled representation learning task description. The observation that there might be two or more nearby solutions aligns well with the observation in Section 4.3.5, namely that across multiple restarts, some models still perform well on the modified dataset. The proposed perturbations favor the minimum corresponding to entangled representations but do not exclusively prohibit the disentangled minimum.

Chapter 5 | Proofs

5.1 Proof of Theorem 1

Proof strategy: For part (b), we aim to derive a lower bound on the objective (3.14), that is independent from the optimization variables $\sigma_j^2(\mathbf{x}^{(i)})$ and V_i . Moreover, we show that this lower bound is tight for some specific choices of $\sigma_j^2(\mathbf{x}^{(i)})$ and V_i , i.e., the global optima. For these choices, all J_i will have orthogonal columns.

The strategy for part (a) is to show that whenever $\sigma_j^2(\mathbf{x}^{(i)})$ and V_i do not induce a global optimum, we can find a small perturbation that decreases the objective function. Thereby showing that local minima do not exist.

Technical lemmas: We begin with introducing a few useful statements. First is the inequality between arithmetic and geometric mean; a consequence of Jensen's inequality.

Lemma 1 (AM-GM inequality). *Let a_1, \dots, a_N be nonnegative real numbers. Then*

$$\frac{1}{N} \sum_{i=1}^N a_i \geq \left(\prod_{i=1}^N a_i \right)^{1/N} \quad (5.1)$$

with equality occurring if and only if $a_1 = a_2 = \dots = a_n$.

The second bound to be used is the classical Hadamard's inequality.

5 Proofs

Lemma 2 (Hadamard’s inequality [97]). *Let $M \in \mathbb{R}^{k \times k}$ be non-singular matrix with column vectors c_1, \dots, c_k . Then*

$$\prod_{i=1}^k \|c_i\| \geq |\det M| \quad (5.2)$$

with equality if and only if the vectors c_1, \dots, c_k are pairwise orthogonal.

And finally a simple lemma for characterizing matrices with orthogonal columns.

Lemma 3 (Column orthogonality). *Let $M \in \mathbb{R}^{n \times d}$ be a matrix and let $M = U\Sigma V^\top$ be its singular value decomposition. Then the following statements are equivalent:*

- (a) *The columns of M are (pairwise) orthogonal.*
- (b) *The matrix $M^\top M$ is diagonal.*
- (c) *The columns of ΣV^\top are (pairwise) orthogonal.*

Proof. The equivalence of (a) and (b) is immediate. For equivalence of (a) and (c) it suffices to notice that if we set $M' = \Sigma V^\top$, then

$$M'^\top M' = V\Sigma^\top \Sigma V^\top = M^\top M. \quad (5.3)$$

The equivalence of (a) and (b) now implies that M has orthogonal columns if and only if M' does. \square

Initial considerations: First, without loss of generality, we will ignore all passive latent variables (in the sense of definition 1). Formally speaking, we will restrict to the case when the local decoder mappings J_i are non-degenerate (i.e., have non-zero singular values). Now d denotes the dimensionality of the latent space with $d = |V_a|$.

Next, we simplify the loss $L_{\approx \text{KL}}$, Equation (3.5). Up to additive and multiplicative constants, this loss can be, for a fixed sample $\mathbf{x}^{(i)} \in X$, written

as

$$\|\mu(\mathbf{x}^{(i)})\|^2 + \sum_{j=1}^d -\log(\sigma_j^2(\mathbf{x}^{(i)})). \quad (5.4)$$

In the optimization problem Equations (3.14) and (3.15) the values $\mu(\mathbf{x}^{(i)})$ can only be affected via applying an orthogonal transformation V_i . But such transformations are norm-preserving (isometric) and hence the values $\|\mu(\mathbf{x}^{(i)})\|^2$ do not change in the optimization. As a result, we can restate the constraint (3.15) as

$$\sum_{\mathbf{x}^{(i)} \in X} \sum_{j=1}^d -\log(\sigma_j^2(\mathbf{x}^{(i)})) = C_1 \quad (5.5)$$

for some constant C_1 .

Proof of theorem 1(b): Here, we explain how Theorem 1(b) follows from the following two propositions.

Proposition 5. *For a fixed sample $\mathbf{x}^{(i)} \in X$ let us denote by c_1, \dots, c_d the column vectors of J_i . Then*

$$\mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|J_i \varepsilon(\mathbf{x}^{(i)})\|^2 \geq d \left(\prod_{j=1}^d \|c_j\|^2 \sigma_j^2(\mathbf{x}^{(i)}) \right)^{1/d} \quad (5.6)$$

with equality if and only if $\|c_j\|^2 \sigma_j^2(\mathbf{x}^{(i)}) = \|c_k\|^2 \sigma_k^2(\mathbf{x}^{(i)})$ for every $j, k \in \{1, \dots, d\}$.

Proposition 6. *Let $M \in \mathbb{R}^{n \times d}$, where $d < n$, be a matrix with column vectors c_1, \dots, c_d and nonzero singular values s_1, \dots, s_d . Then*

$$\prod_{j=1}^d \|c_j\| \geq \det^\dagger(M), \quad (5.7)$$

where by $\det^\dagger(M)$ we denote the product of the singular values of M . Equality occurs if and only if c_1, \dots, c_d are pairwise orthogonal.

5 Proofs

First, Proposition 6 allows making further estimates in the inequality from Proposition 5. Indeed, we get

$$\mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|J_i \varepsilon(\mathbf{x}^{(i)})\|^2 \geq d \left((\det^\dagger(J_i))^2 \prod_{j=1}^d \sigma_j^2(\mathbf{x}^{(i)}) \right)^{1/d} \quad (5.8)$$

and after applying the (monotonous) log function we are left with

$$\log \mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|J_i \varepsilon(\mathbf{x}^{(i)})\|^2 \geq \quad (5.9)$$

$$\log(d) + \frac{2}{d} \log(\det^\dagger(J_i)) + \frac{1}{d} \sum_{j=1}^d \log(\sigma_j^2(\mathbf{x}^{(i)})). \quad (5.10)$$

Finally, we sum over the samples $\mathbf{x}^{(i)} \in X$ and simplify via (5.5) as

$$\begin{aligned} \sum_{\mathbf{x}^{(i)} \in X} \log \mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|J_i \varepsilon(\mathbf{x}^{(i)})\|^2 &\geq \\ N \log(d) - \frac{C_1}{d} + \frac{2}{d} \sum_{\mathbf{x}^{(i)} \in X} \log(\det^\dagger(J_i)). \end{aligned} \quad (5.11)$$

The right-hand side of this inequality is independent of the values of $\sigma_j^2(\mathbf{x}^{(i)})$, as well as from the orthogonal matrices V_i , since these do not influence the singular values of any J_i .

Moreover, it is possible to make inequality (5.11) tight (i.e., reach the global minimum), by setting $\sigma_j^2(\mathbf{x}^{(i)})$ as hinted by Proposition 5 and by choosing the matrices V_i such that every J_i has orthogonal columns (this is clearly possible as seen in Proposition 1).

This yields the desired description of the global minima of (3.14). \square

Proof of proposition 5: We further denote by r_1, \dots, r_n the row vectors of J_i , and by $a_{r,c}$ the element of J_i at r -th row and c -th column. With sampling $\varepsilon(\mathbf{x}^{(i)})$ according to

$$\varepsilon(\mathbf{x}^{(i)}) \sim \mathcal{N}(0, \text{diag } \sigma^2(\mathbf{x}^{(i)})), \quad (5.12)$$

we begin simplifying the objective (3.14) with

$$\mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|J_i \varepsilon(\mathbf{x}^{(i)})\|^2 = \mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \sum_{k=1}^n \|r_k^\top \varepsilon(\mathbf{x}^{(i)})\|^2 \quad (5.13)$$

$$= \sum_{k=1}^n \mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|r_k^\top \varepsilon(\mathbf{x}^{(i)})\|^2. \quad (5.14)$$

Now, as the samples $\varepsilon(\mathbf{x}^{(i)})$ are zero mean, we can further write

$$\sum_{k=1}^n \mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|r_k^\top \varepsilon(\mathbf{x}^{(i)})\|^2 = \sum_{k=1}^n \text{var}(r_k^\top \varepsilon(\mathbf{x}^{(i)})). \quad (5.15)$$

Now we use the fact that for uncorrelated random variables A and B we have $\text{var}(A + cB) = \text{var} A + c^2 \text{var} B$. This allows expanding the variance of the inner product as

$$\begin{aligned} \text{var}(r_k^\top \varepsilon(\mathbf{x}^{(i)})) &= \text{var}\left(\sum_{j=1}^d a_{k,j} \varepsilon_j(\mathbf{x}^{(i)})\right) \\ &= \sum_{j=1}^d a_{k,j}^2 \text{var} \varepsilon_j(\mathbf{x}^{(i)}) = \sum_{j=1}^d a_{k,j}^2 \sigma_j^2(\mathbf{x}^{(i)}). \end{aligned} \quad (5.16)$$

Now, we can regroup the terms via

$$\begin{aligned} \sum_{k=1}^n \text{var}(r_k^\top \varepsilon(\mathbf{x}^{(i)})) &= \sum_{k=1}^n \sum_{j=1}^d a_{k,j}^2 \sigma_j^2(\mathbf{x}^{(i)}) \\ &= \sum_{j=1}^d \sum_{k=1}^n a_{k,j}^2 \sigma_j^2(\mathbf{x}^{(i)}) \\ &= \sum_{j=1}^d \|c_j\|^2 \sigma_j^2(\mathbf{x}^{(i)}). \end{aligned} \quad (5.17)$$

All in all, we obtain

$$\mathbb{E}_{\varepsilon(\mathbf{x}^{(i)})} \|J_i \varepsilon(\mathbf{x}^{(i)})\|^2 = \sum_{j=1}^d \|c_j\|^2 \sigma_j^2(\mathbf{x}^{(i)}). \quad (5.18)$$

5 Proofs

from which the desired inequality follows via setting $a_j = \|c_j\|^2 \sigma_j^2(\mathbf{x}^{(i)})$ for $j = 1, \dots, d$ in Lemma 1. Indeed, then we have

$$\sum_{j=1}^d \|c_j\|^2 \sigma_j^2(\mathbf{x}^{(i)}) \geq d \left(\prod_{j=1}^d \|c_j\|^2 \sigma_j^2(\mathbf{x}^{(i)}) \right)^{1/d} \quad (5.19)$$

as required. \square

Proof of proposition 6: As the first step, we show that both sides of the desired inequality are invariant to multiplying the matrix M from the left with an orthogonal matrix $U \in \mathbb{R}^{n \times n}$.

For the right-hand side, this is clear as the singular values of UM are identical to those of M . As for the left-hand side, we first need to realize that the vectors c_j are the images of the canonical basis vectors e_j , i.e., $c_j = Me_j$ for $j = 1, \dots, d$. But since U is an isometry, we have $\|UMe_j\| = \|Me_j\| = \|c_j\|$ for every j , and hence also the column norms are intact by prepending U to M .

This allows us to restrict to matrices M for which the SVD has a simplified form $M = \Sigma V^\top$. Next, let us denote by $\Sigma_{d \times d}$ the $d \times d$ top-left submatrix of Σ . Note that $\Sigma_{d \times d}$ contains all nonzero elements of Σ . As a result, the matrix $M' = \Sigma_{d \times d} V^\top$ contains precisely the nonzero rows of the matrix M . This implies

$$M^\top M = M'^\top M'. \quad (5.20)$$

In particular, the column vectors c'_j of M' have the same norms as those of M . Now we can write

$$\prod_{j=1}^d \|c_j\| = \prod_{j=1}^d \|c'_j\| \geq |\det(M')| = \det^\dagger(M), \quad (5.21)$$

where the inequality follows from Lemma 2 applied to nonsingular matrix M' . Equality in Lemma 2 occurs precisely if the columns of M' are orthogonal. However, according to Lemma 3 and (5.20), it also follows that the

columns of M' are orthogonal if and only if the columns of M are. Note that Lemma 3(c) is needed for covering the reduction performed in the first two paragraphs. \square

Proof of Theorem 1(a): We show the nonexistence of local minima as follows. For any values of $\sigma_j^2(\mathbf{x}^{(i)})$ and V_i that do not minimize the objective function (3.14), we find a small perturbation that improves this objective.

All estimates involved in establishing inequality (5.11) rely on either Lemma 1 or Lemma 2, where in both cases, the right-hand side was kept fixed. We show that both of these inequalities can be tightened in such fashion by small perturbations in their parameters.

Lemma 4 (Locally improving AM-GM). *For any non-negative values a_1, \dots, a_N for which*

$$\frac{1}{N} \sum_{i=1}^N a_i > \left(\prod_{i=1}^N a_i \right)^{1/N} \quad (5.22)$$

there exists a small perturbation a'_i of a_i for $i = 1, \dots, N$ such that

$$\frac{1}{N} \sum_{i=1}^N a_i > \frac{1}{N} \sum_{i=1}^N a'_i \geq \quad (5.23)$$

$$\left(\prod_{i=1}^N a'_i \right)^{1/N} = \left(\prod_{i=1}^N a_i \right)^{1/N} \quad (5.24)$$

Proof. Since (5.22) is a sharp inequality, we have $a_i > a_j$ for some $i \neq j$. Then setting $a'_i = a_i/(1 + \delta)$, $a'_j = a_j(1 + \delta)$, and $a'_k = a_k$ otherwise, will do the trick. Indeed, we have $a_i a_j = a'_i a'_j$ as well as $a_i + a_j > a'_i + a'_j$ for small enough δ . This ensures both 5.23 and 5.24. \square

An analogous statement for Lemma 2 has the following form.

5 Proofs

Lemma 5 (Locally improving Hadamard's inequality). *Let $M \in \mathbb{R}^{k \times k}$ be a non-singular matrix with *SVD* $M = U\Sigma V^\top$, and column vectors c_1, \dots, c_k , for which*

$$\prod_{i=1}^k \|c_i\| > |\det M|. \quad (5.25)$$

Then there exists an orthogonal matrix V' , a small perturbation of V , such that if we denote by c'_1, \dots, c'_k the column vectors of $M' = U\Sigma V'^\top$, we have

$$\prod_{i=1}^k \|c_i\| > \prod_{i=1}^k \|c'_i\|. \quad (5.26)$$

Proof. We proceed by induction on k . For $k = 2$, it can be verified directly that for some small δ (in absolute value) setting $V' = VR_\delta$, where R_δ is a 2D rotation matrix by angle δ , achieves what is required.

For the general case, the sharp inequality (5.25) implies that $c_i^\top c_j \neq 0$ for some pair of $i \neq j$. Without loss of generality, let $i = 1, j = 2$. In such case, we consider $V' = VR_\delta^{2D}$, where

$$R_\delta^{2D} = \begin{pmatrix} R_\delta & \\ & \mathcal{I}_{k-2} \end{pmatrix} \quad (5.27)$$

is a block diagonal matrix, in which R_δ is again a 2×2 rotation matrix. By design, we have $c_i = c'_i$ for $i > 2$. This, along with the fact that U can be set to \mathcal{I}_k (isometry does not influence either side of (5.25)), allows for a full reduction to the discussed two-dimensional case. \square

It is easy to see that the performed perturbations continuously translate into perturbations of the parameters $\sigma_j^2(\mathbf{x}^{(i)})$ and V_i in estimates (5.19) and (5.21). Consequently, any non-optimal values of $\sigma_j^2(\mathbf{x}^{(i)})$ and V_i can be locally improved. This concludes the proof.

Rotational invariances

Let us start by fleshing out the common elements of the proofs of Proposition 2 and Proposition 3. In both cases, the encoder and decoder mappings $\text{Enc}_{\varphi,U}$, $\text{Dec}_{\theta,U}$ induce joint distributions $p_U(\mathbf{x}, \mathbf{z})$, $q_U(\mathbf{x}, \mathbf{z})$ described as

$$p_U(\mathbf{x}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{x} | U^\top \mathbf{z}) \quad (5.28)$$

$$q_U(\mathbf{x}, \mathbf{z}) = q(\mathbf{x})q(U^\top \mathbf{z} | \mathbf{x}) \quad (5.29)$$

Lemma 6. *For every $\mathbf{x}^{(i)} \in X$ we have $p(\mathbf{x}^{(i)}) = p_U(\mathbf{x}^{(i)})$.*

Proof. We simply compute

$$\begin{aligned} p_U(\mathbf{x}^{(i)}) &= \int p_U(\mathbf{x}^{(i)}, \mathbf{z}) \, d\mathbf{z} \\ &= \int p(\mathbf{z})p(\mathbf{x}^{(i)} | U^\top \mathbf{z}) \, d\mathbf{z} \\ &= \int p(U\mathbf{z})p(\mathbf{x}^{(i)} | \mathbf{z}) \, d\mathbf{z} \\ &= \int p(\mathbf{z})p(\mathbf{x}^{(i)} | \mathbf{z}) \, d\mathbf{z} = p(\mathbf{x}^{(i)}), \end{aligned}$$

where in the third equality we used the Change of Variable Theorem to substitute $U\mathbf{z}$ for \mathbf{z} (keep in mind that $|\det(U)| = 1$ as U is an orthogonal matrix). In the fourth equality, we used the rotational symmetry of the prior $p(\mathbf{z})$. \square

Proof of Proposition 2. This immediately follows from Lemma 6. \square

Proof of Proposition 3. We utilize the full identity from ELBO derivation. For fixed $\mathbf{x}^{(i)} \in X$ we have [12]

$$\text{ELBO} = D_{\text{KL}}(q_U(\mathbf{z} | \mathbf{x}^{(i)}) \| p_U(\mathbf{z} | \mathbf{x}^{(i)})) + \log p_U(\mathbf{x}^{(i)}) \quad (5.30)$$

In order to prove the invariance of ELBO to the choice of U , it suffices to prove the invariance of the right-hand side of (5.30). Due to Proposition

(3) we only need to focus on the KL term. Similarly, as in the proof of Lemma 6, we calculate

$$\begin{aligned}
& D_{\text{KL}}(q_U(\mathbf{z} \mid \mathbf{x}^{(i)}) \parallel p_U(\mathbf{z} \mid \mathbf{x}^{(i)})) \\
&= \int q_U(\mathbf{z} \mid \mathbf{x}^{(i)}) \log \frac{q_U(\mathbf{z} \mid \mathbf{x}^{(i)})}{p_U(\mathbf{z} \mid \mathbf{x}^{(i)})} d\mathbf{z} \\
&= \int q_U(\mathbf{z} \mid \mathbf{x}^{(i)}) \log \frac{q_U(\mathbf{z} \mid \mathbf{x}^{(i)}) \cdot p_U(\mathbf{x}^{(i)})}{p_U(\mathbf{z}) \cdot p_U(\mathbf{x}^{(i)} \mid \mathbf{z})} d\mathbf{z} \\
&\stackrel{(3)}{=} \int q(U^\top \mathbf{z} \mid \mathbf{x}^{(i)}) \log \frac{q(U^\top \mathbf{z} \mid \mathbf{x}^{(i)}) \cdot p(\mathbf{x}^{(i)})}{p(\mathbf{z}) \cdot p(\mathbf{x}^{(i)} \mid U^\top \mathbf{z})} d\mathbf{z} \\
&\stackrel{(4)}{=} \int q(\mathbf{z} \mid \mathbf{x}^{(i)}) \log \frac{q(\mathbf{z} \mid \mathbf{x}^{(i)}) \cdot p(\mathbf{x}^{(i)})}{p(U\mathbf{z}) \cdot p(\mathbf{x}^{(i)} \mid \mathbf{z})} d\mathbf{z} \\
&\stackrel{(5)}{=} \int q(\mathbf{z} \mid \mathbf{x}^{(i)}) \log \frac{q(\mathbf{z} \mid \mathbf{x}^{(i)}) \cdot p(\mathbf{x}^{(i)})}{p(\mathbf{z}) \cdot p(\mathbf{x}^{(i)} \mid \mathbf{z})} d\mathbf{z} \\
&= \int q(\mathbf{z} \mid \mathbf{x}^{(i)}) \log \frac{q(\mathbf{z} \mid \mathbf{x}^{(i)})}{p(\mathbf{z} \mid \mathbf{x}^{(i)})} d\mathbf{z} \\
&= D_{\text{KL}}(q(\mathbf{z} \mid \mathbf{x}^{(i)}) \parallel p(\mathbf{z} \mid \mathbf{x}^{(i)})),
\end{aligned}$$

where we again used the change of variable theorem in Equality (4), rotational symmetry of $p(\mathbf{z})$ in Equality (5), and Lemma 6 in Equality (3). \square

Proof of auxiliary statements

Proof of Proposition 1. Recall from Lemma 3 that column orthogonality of M is equivalent to $M^\top M$ being a diagonal matrix.

(b) \Rightarrow (a): Let $M = U\Sigma V^\top$ where $|V|$ is a permutation matrix. Then

$$M^\top M = V\Sigma^\top U^\top U\Sigma V^\top = V\Sigma'V^\top \quad (5.31)$$

where $\Sigma' = \Sigma^\top \Sigma$ is a diagonal matrix. But then $V\Sigma'V^\top$ only permutes the diagonal entries of Σ' (and possibly flips their signs). In particular, $V\Sigma'V^\top$ is also diagonal.

5.1 Proof of Theorem 1

(a) \Rightarrow (b): Let again $M = U\Sigma V^\top$ be some SVD of M and assume $M^\top M = D$ for some diagonal matrix D . Since M has d distinct nonzero singular values, $M^\top M$ has d distinct nonzero eigenvalues (diagonal elements). Moreover, these eigenvalues are precisely the squares of the singular values captured by Σ . Next, if we denote by P the permutation matrix for which PDP^{-1} has decreasing diagonal elements, we can write

$$PDP^{-1} = \Sigma^\top \Sigma \quad (5.32)$$

Then using (5.32) and the SVD of M similarly as in (5.31), we obtain

$$D = M^\top M = V\Sigma^\top \Sigma V^\top = VPDP^{-1}V^\top. \quad (5.33)$$

Further, the resulting identity $(VP)D = D(VP)$ implies that columns of VP are eigenvectors of D , i.e., the canonical basis vectors. Since VP is additionally orthogonal, these eigenvectors are normalized. It follows that $|VP|$ is a permutation matrix and the conclusion follows.

□

Proof of Proposition 4. First, note that for any random variable $\mathbf{X} \in \mathbb{R}^k$ with $\mathbb{E}\mathbf{X} = \mu$ and a constant $\mathbf{b} \in \mathbb{R}^k$, the following identity holds

$$\mathbb{E} \|\mathbf{X} - \mathbf{b}\|^2 = \mathbb{E} \|\mathbf{X} - \mu\|^2 + \|\mu - \mathbf{b}\|^2. \quad (5.34)$$

In our case, we set $\mathbf{X} = \text{Dec}_\theta(\text{Enc}_\varphi(\mathbf{x}^{(i)}))$, the unbiasedness assumption translates to $\mathbb{E}\mathbf{X} = \text{Dec}_\theta(\mu(\mathbf{x}^{(i)}))$, and finally we set $\mathbf{b} = \mathbf{x}^{(i)}$.

The identity we obtain is exactly what was required to prove.

□

5.2 Proof of Theorem 2

Proof strategy: We start with the optimization problem stated in (Equations (4.1) and (4.2)) and repeated here for improved readability:

$$\min_{\Sigma, U, V} \mathbb{E}_i \left(\|U \Sigma V^\top \varepsilon^{(i)}\|^2 \right) \quad (5.35)$$

$$\text{s.t.} \quad \mathbb{E}_i \left(\mathcal{L}_{\approx \text{KL}}^{(i)} \right) = c_{\approx \text{KL}}. \quad (5.36)$$

Based on Theorem 1 and Proposition 1 and without loss of generality, we assume $V = \mathcal{I}$ and rearrange the elements of Σ in ascending order and those of $\varepsilon^{(i)}$ in descending order with respect to $\sigma^{(i)2}$.

In the setting of Theorem 2, we consider the mean latent representation Z to be constrained only by the condition $\text{diag}(Z^\top Z) = \mathbf{1}$, which reads as “each active latent variable has unit variance”. Even though this statement is unsurprising in the context of VAEs, we offer a quick proof of how this follows directly from the KL loss in Lemma 7. Additionally, we fully fix the matrix \hat{X} , which contains the reconstruction of all data points. The remaining freedom in U and Σ has the following nature: for each fixed U^\top (which rotates \hat{X}), the nonzero singular values of Σ (scaling factors along individual axes in the latent space) are fully determined by the $\text{diag}(Z^\top Z) = \mathbf{1}$ requirement. We minimize objective (5.35) under these constraints.

Remark Notice that fixing the reconstructed data points ensures that the observed effect is entirely independent of the deterministic loss. The deterministic loss is known to have some PCA-like effects, as it is basically an MSE loss of a deterministic autoencoder. The additional (and, in fact, stronger) effects of the stochastic loss are precisely the novelty of the following theoretical derivations.

For technical reasons regarding the uniqueness of SVD, we additionally inherit the assumption of Proposition 1 that the random variables $\varepsilon^{(i)}$ have distinct variances.

Finally, the orthonormal matrix U acts isometrically and can be removed from the objective (Equation (5.35)), even though it still plays a vital role in how the problem is constrained. The reduced objective is further conveniently rewritten as a trace as:

$$\min_{\Sigma} \mathbb{E}_i \|\Sigma \varepsilon^{(i)}\|^2 = \min_{\Sigma} \mathbb{E}_i \operatorname{tr} (E \Sigma^\top \Sigma E), \quad (5.37)$$

where E is the diagonal matrix induced by the vector ε .

A visualization of the role of U , Σ , and V in the decoding process is illustrated in Figure 2.4.

Proof

We rewrite the objective in order to introduce U , \hat{X} , and Z and make use of the constraints $\operatorname{diag}(Z^\top Z) = \mathbf{1}$ and $\hat{X} = Z \Sigma U$. We have

$$E \Sigma^\top \Sigma E = E \Sigma^\top (Z^\top Z + M) \Sigma E, \quad (5.38)$$

where $M = \mathcal{I} - Z^\top Z$ is a matrix with $\operatorname{diag}(M) = 0$. Also, we can expand

$$\Sigma^\top Z^\top Z \Sigma = U (U^\top \Sigma^\top Z^\top) (Z \Sigma U) U^\top = U \hat{X}^\top \hat{X} U^\top \quad (5.39)$$

By combining Equation (5.38) and Equation (5.39), we learn that

$$E \Sigma^\top \Sigma E - E U \hat{X}^\top \hat{X} U^\top E = E \Sigma^\top M \Sigma E. \quad (5.40)$$

By repeating Lemma 8, we learn that $\operatorname{diag}(E \Sigma^\top M \Sigma E) = 0$, which allows us to use Lemma 8 yet again, this time on the left-hand side of (5.40) and obtain a key intermediate conclusion:

$$\operatorname{tr} (E \Sigma^\top \Sigma E) = \operatorname{tr} (E U \hat{X}^\top \hat{X} U^\top E) \quad (5.41)$$

This has a lower bound according to a classical trace inequality (see Proposition 7), as $E U \hat{X}^\top \hat{X} U^\top E$ is positive semi-definite.

$$\operatorname{tr} (E U \hat{X}^\top \hat{X} U^\top E) \geq n \det (E U \hat{X}^\top \hat{X} U^\top E)^{1/n} \quad (5.42)$$

$$= n \det (E \hat{X}^\top \hat{X} E)^{1/n} \quad (5.43)$$

5 Proofs

with equality if and only if

$$EU\hat{X}^\top\hat{X}U^\top E = \lambda\mathcal{I}. \quad (5.44)$$

For the **SVD** $\hat{X} = U_X\Sigma_XV_X^\top$, we see that $\hat{X}^\top\hat{X} = V_X\Sigma_X^2V_X^\top$ and with $U' = UV_X$ we arrive at

$$U'\Sigma_X^2U'^\top = \lambda E^{-2}. \quad (5.45)$$

The left-hand side gives an **SVD** of the diagonal matrix E^{-2} . The **SVD** of a diagonal matrix is unique up to a signed permutation matrix. The conclusion of Theorem 1 now follows.

Proof of auxiliary statements

In the following lemma, the vectors \mathbf{x} and \mathbf{y} correspond to the mean latent μ and the noise standard deviation σ respectively. We allow for scaling the latent space and find that the **KL** loss is minimal for unit standard deviation of the means.

Lemma 7. For vectors $\mathbf{x} = (x_0, \dots, x_n) \in \mathbb{R}^n$, $\mathbf{y} = (y_0, \dots, y_n) \in \mathbb{R}^n$ and

$$c = \arg \min_{c \in \mathbb{R}} \sum_i (c^2 x_i^2 - \log(c^2 y_i^2)),$$

it holds that

$$c = \sqrt{\sum_i (x_i^2)} \quad (5.46)$$

Proof. It is easy to inspect that the minimum of $\sum_i (c^2 x_i^2 - \log(c^2 y_i^2))$ with respect to c fulfils the statement. \square

Proposition 7 (Trace Inequality). For a positive semi-definite $M \in \mathbb{R}^{n \times n}$, that is $M \succcurlyeq 0$, it holds that

$$\text{tr}(M) \geq n \det(M)^{1/n} \quad (5.47)$$

with equality if and only if $M = \lambda \cdot \mathcal{I}$ for some $\lambda \geq 0$.

5.2 Proof of Theorem 2

Proof. Let $\lambda_1, \dots, \lambda_n$ denote the eigenvalues of M , then $\text{tr}(M) = \sum_i \lambda_i$ and $\det(M) = \prod_i \lambda_i$. Since $M \succcurlyeq 0$, we have $\lambda_i \geq 0$ for every $i = 1, \dots, n$. Then, due to the classical AM-GM inequality, we have

$$\text{tr}(M) = \sum_i \lambda_i \geq n \cdot \left(\prod_i \lambda_i \right)^{1/n} = n \det(M)^{1/n}, \quad (5.48)$$

with equality precisely if all eigenvalues are equal to the same value $\lambda \geq 0$. Then by the definition of eigenvalues, the $M - \lambda \mathcal{I}$ has zero rank and equals zero as required. \square

Lemma 8 (“Empty diagonal absorbs”). *Let $D \in \mathbb{R}^{m \times m}$ be a diagonal matrix and let $M \in \mathbb{R}^{m \times m}$ be a matrix with zero elements on the diagonal, that is $\text{diag}(M) = 0$. Then $\text{diag}(MD) = \text{diag}(DM) = 0$ and consequently also $\text{tr}(MD) = \text{tr}(DM) = 0$.*

Proof. Follows immediately from the definition of matrix multiplication. \square

Chapter 6 | Discussion and conclusions

6.1 Discussion

This work summarized results that answer the question of why VAE-based architectures disentangle. It was made evident in [15] that unsupervised learning of disentangled representations can only be possible by choosing the right type of inductive bias. We provided a retrospective answer to which biases are responsible for the disentangling capabilities of β -VAEs, both from the model and data perspectives.

In the first part, we isolated the effect of the particular choice of the prior and posterior in the canonical implementation of VAEs. This leads to a mechanism that fosters local orthogonalization of the learned models. Orthogonality itself is an inherent similarity between VAE-based models and PCA. We provide a theoretical framework in which we can derive this behavior directly from the loss function of the canonical implementation of a VAE. Furthermore, we demonstrated the functionality of this mechanism in intuitive terms via a hands-on linear example and an intuitive picture. The introduced measure for the distance to orthogonality allowed us to provide extensive experimental evidence to support our findings. Orthogonality, being an attractive geometric characteristic, however, distinguishes the classical PCA from ICA. This results in a new question: How effective are novel VAE-based architectures in terms of nonlinear ICA and why?

6 Discussion and conclusions

In which cases can an orthogonal model act as a de-mixing operation in terms of nonlinear ICA? This work does not answer this question which we leave open for future work.

In the second part, we showed that β -VAEs use the differences in variance in the data to form the representation in the latent space. This tightens the connection further and concludes an equivalence between VAEs and PCA for the linear case. We have also shown that the success of VAE-based architectures stems mainly from the structured nature of the datasets on which they are being trained and evaluated. Small perturbations of the dataset can reduce this structure and decrease the bias such architectures exploit. Interestingly, even architectures that are proven to be identifiable, like the SlowVAE, still owe their success to the same bias. PCL and the weakly supervised GAN, however, as non-variational methods, were unaffected by the small perturbation. This naturally leads to the question of whether novel methods that combine β -VAEs with subtle supervision (e.g., sparse transition priors or auxiliary observables) owe their success to the additional supervision information or the existing biases. In conclusion, the success of VAE-based architectures can largely be explained by two components: (1) The choice of the prior and posterior results in PCA-like behavior, and (2) In existing datasets, the ground truth generating factors seem to align with the principal components. The second component, however, is still insufficiently answered for the nonlinear setting.

6.2 Limitations

There are three main technical limitations of this work: (i) We restrict our statements to the so-called polarized regime, which is itself a not completely understood but highly discussed phenomenon, (ii) Our analysis works with the vanilla β -VAE loss and only generalizes to other VAE-based architectures experimentally and (iii) We do not provide an extension of the VAE-PCA equivalence for the nonlinear case.

Our analysis of the β -VAE loss is based on the simplification that arises within the polarized regime, i.e., in the situation in which the latent dimensions can be categorized into active (non-zero mean, smaller than one standard deviation) and inactive (zero mean, unit variance). This phenomenon is well-known to practitioners and has been discussed under different terms, such as the posterior collapse [68, 81]. We did not observe any shortcomings of this simplification during our experiments (which are conducted optimizing the whole, un-simplified VAE loss). This type of behavior is another hidden, practical perk of the β -VAE, as it allows for smooth pruning of the latent space. However, this work misses out on analyzing the exact origin of this almost discrete and intriguing behavior.

Although we evaluate our methods on various VAE-based architectures, the theory is based only on the classical VAE loss. The fact that the empirical evidence suggests that the primary mechanisms discovered translate across derivatives of the VAE is insightful. Every architecture was designed with a certain innovative change in mind. Still, the underlying loss function often mildly deviates from the original β -VAE or is extended by additional regularization terms. However, providing equivalent statements to the ones in this work would require a detailed analysis of each architecture.

We extended the statements about the orthogonality of β -VAEs to linearizations thereof (orthogonality of the Jacobians). However, we did not provide a similar extension for the specific alignment in the nonlinear case, which is attributed to the complexity of the question and left open for future work.

6.3 Future work

Many opportunities and questions arose while discovering the insights summarized in this work. The most obvious and pressing question is to which extent one could utilize the novel understanding to improve the disentangling quality of β -VAE-based architectures. It is unsatisfying that β -VAEs promote orthogonality somewhat indirectly, and being able to control this feature explicitly through the architecture design would be beneficial. Along with this almost accidental perk of β -VAEs, the choice of prior and posterior might not come exclusively as an advantage but may hide more intricacies. Perhaps achieving explicit control over orthogonality in other architectures would allow for better overall representation quality.

One of the downsides that come along with the β -VAE inner workings are the significant variances over restarts. They can be partially explained by the degeneracy of the datasets (Section 3.3.4) and partially attributed to ambiguous solutions to the optimization objective (Section 3.3.2). The latter problem is associated with the highly-debated question of how to measure disentanglement and whether there are better ways to define it in the first place [19]. However, the example of the β -VAE choosing a Cartesian or a polar coordinate system is fascinating. It is not immediately intuitive why the polar representation is more suitable than any other (invertible) coordinate transformation. One particular feature that comes to mind is the cyclic behavior of the angular coordinates, which links to the cyclic nature of generating factors that cover, e.g., hues of objects in an image. However, the symmetry of the hues is typically broken to some extent, which leads to the reproducible alignment of the color coordinates in the cartesian representations of Figure 4.10. To sum it up, it comes as a certain surprise that the polar representations form a distinct optimum to the optimization problem, thereby fueling the discussion about a unified, general definition of disentanglement.

While we could showcase that small perturbations in the local structure of the data alleviate the particular features of synthetic datasets that make β -VAEs disentangle, it remains an open question whether the same local structure can reliably be found in real-world data on which such architectures could be deployed. If so, fostering the sensitivity of future architectures towards the natural alignment of data could result in a transparent advance of unsupervised representation learning. Similarly to the question discussed in the previous paragraph, it would be interesting to follow the leads of clearly distinct local minima of the optimization problem since their suitability for downstream applications remains unexplored.

Last but not least, formalizing a theoretical identifiability statement for β -VAEs could help to broaden the understanding of their peculiarities. The problem that completely unsupervised recovery of the ground truth generating factors for arbitrary generative processes is an ill-posed task means that any such claim can, if at all, only be made for a particular set of mappings. This could perhaps be linked to [98], as their work deals with generative processes that share the local orthogonality condition that β -VAEs strive for. Connecting the lines of research on nonlinear ICA, causal mechanism analysis, and β -VAEs will hopefully yield intriguing insights that could help the field to make a large leap forward.

Chapter 7 | Related projects

7.1 Deep graph matching via blackbox differentiation

Abstract

Building on recent progress at the intersection of combinatorial optimization and deep learning, we propose an end-to-end trainable architecture for deep graph matching that contains unmodified combinatorial solvers. Using the presence of heavily optimized combinatorial solvers together with some improvements in architecture design, we advance state-of-the-art on deep graph matching benchmarks for keypoint correspondence. In addition, we highlight the conceptual advantages of incorporating solvers into deep learning architectures, such as the possibility of post-processing with a strong multi-graph matching solver or the indifference to changes in the training setting. Finally, we propose two new challenging experimental setups.

Michal Rolínek, Paul Swoboda, *Dominik Zietlow*, Anselm Paulus,
Vit Musil, Georg Martius
ECCV 2020; <https://arxiv.org/abs/2003.11657>

Summary and connection to this thesis

The task of semantic keypoint correspondence matching deals with a setting in which two or more images with the locations of a set of key points are provided. Every pair of matching key points has to be linked to each other by the model. For example, the two images could show sheep with keypoint annotations at a hoof, the ears, the eyes, the nose, and the mouth. The model has to match key points accordingly, as illustrated in Figure 7.1.

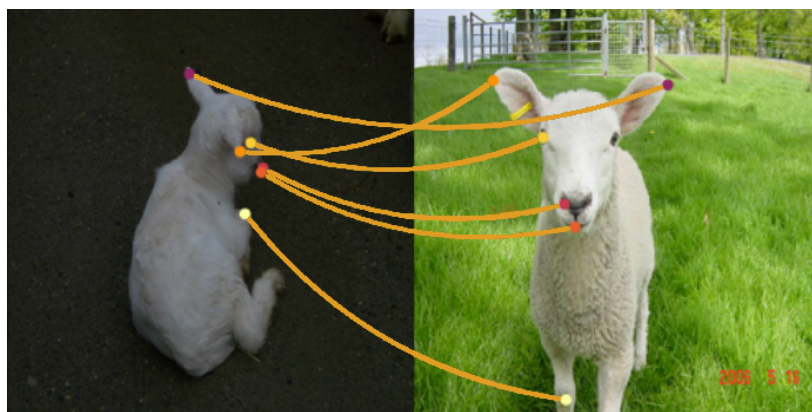


Figure 7.1: **Keypoint matching:** The points in either image are annotated solely by their position and have to be matched according to their semantics.

We propose using BlackboxBackprop, a method developed to provide linear gradient interpolations for piece-wise constant loss functions, to connect state-of-the-art combinatorial solvers with deep feature extractors. The combinatorial solver works on an instance specified by one graph per image and resulting edge- as well as vertex-affinities. We use a deep neural network to extract a representation for each keypoint (vertex in the graph) and calculate the edge features as differences of the vertex features, as well as the affinities as weighted inner products. The resulting matching loss is a piece-wise constant function, in other words: Mildly perturbing the affinities has likely no effect on the resulting keypoint matching. This is where BlackboxBackprop comes into play, as it allows for computing a

7.1 Deep graph matching via blackbox differentiation

linearized gradient interpolation between areas of constant losses, thereby allowing for backpropagating through the combinatorial solver.

The whole representation of an image as a graph and the combination of multiple graphs as affinity matrices can be seen as representation learning. Although the desired property is not clearly specified as it is in terms of disentanglement, we use modern machine learning techniques to end-to-end train deep models for finding the right representation that combinatorial solvers can work on.

7.2 Leveling down: Pareto inefficiencies in fair deep classifiers

Abstract

Algorithmic fairness is frequently motivated in terms of a trade-off in which overall performance is decreased so as to improve performance on disadvantaged groups where the algorithm would otherwise be less accurate. Contrary to this, we find that applying existing fairness approaches to computer vision improve fairness by degrading the performance of classifiers across all groups (with increased degradation on the best performing groups).

Extending the bias-variance decomposition for classification to fairness, we theoretically explain why the majority of fairness methods designed for low capacity models should not be used in settings involving high-capacity models, a scenario common to computer vision. We corroborate this analysis with extensive experimental support that shows that many of the fairness heuristics used in computer vision also degrade performance on the most disadvantaged groups. Building on these insights, we propose an adaptive augmentation strategy that, uniquely of all methods tested, improves performance for the disadvantaged groups.

Dominik Zietlow, Michael Lohaus, Matthaeus Kleindessner, Guha Balakrishnan, Francesco Locatello, Bernhard Schölkopf, Christopher Russell
CVPR 2022; <https://arxiv.org/abs/2203.04913>

Summary and connection to this thesis

We show that existing fairness methods on classification tasks typically reduce the accuracy across protected groups when deployed in a computer vision setting. We propose an explanation that links to the classical bias-variance loss decomposition and postulates that in the presence of high dimensional data and high capacity models, fairness can be improved alongside accuracy by fostering the generalization on the worse performing group.

Generalization in deep learning, however, is intrinsically hard to achieve. There are two main approaches to improving the generalization of a model: Via regularization mechanisms or data augmentation strategies. We deploy a [GAN](#) based augmentation method to improve fairness and classification accuracy across groups.

Without additional measures, the [GAN](#) latent space is not disentangled. Although their latent spaces have shown to be structured and allow for meaningful interpolations, they are, however, not axis aligned. In other words, a latent traversal along a Cartesian axis would not lead to a change in a single semantic quantity, such as ,e.g., hair color but change multiple characteristics at once.

We address this shortcoming of [GANs](#) by extending the [Synthetic Minority Oversampling Strategy \(SMOTE\)](#) [99] to [GAN](#) latent spaces and extend the sampling strategy beyond simple linear interpolations ([g-SMOTE](#)). [SMOTE](#) is an augmentation-like approach proposed to improve training on biased datasets. The under-represented group is therein extended by generating new data via linear interpolations in the feature space of a data point and one of its nearest neighbors. By using the [GAN](#) latent space as a feature space and extending the augmentation process to uniform sampling within a simplex formed by more than one of the nearest neighbors, we can improve the generalization performance of the trained models. [Figure 7.2](#) shows two examples of data augmentations for sampling within the simplex formed of a datapoint (green) and two of its nearest neighbors (orange).

7 Related projects

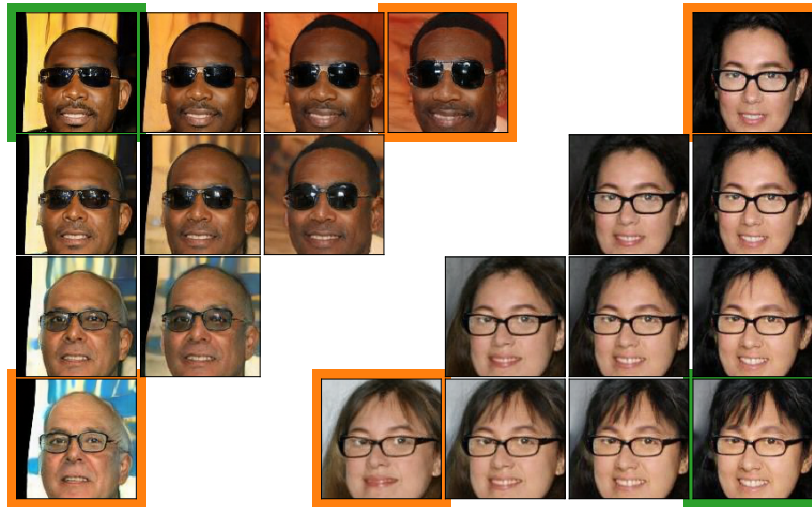


Figure 7.2: **g-SMOTE augmentations** with $k=3$. Given a data-point (green) and two neighbors (orange), linear interpolations in **GAN** latent space yield diverse images. Nearest neighbors are chosen to share the target attribute (“*eyeglasses*”). We give all interpolated images the same attribute label value.

The required property of the **GAN** latent space for this to work is not as strict as the one for disentanglement. We assign target labels to the augmented images by choosing consistent nearest neighbors, e.g., if the datapoint corresponds to a person wearing glasses, we sample in the simplex formed by some of its nearest neighbors who also wear glasses. For this to work, the volume covered by the simplex must be semantically consistent: Every image generated from that volume in latent space should correspond to a picture of a person wearing glasses.

7.3 Machine learning quantum dynamics

Machine learning generators of open quantum dynamics

Abstract

In the study of closed many-body quantum systems, one is often interested in the evolution of a subset of degrees of freedom. On many occasions it is possible to approach the problem by performing an appropriate decomposition into a bath and a system. In the simplest case the evolution of the reduced state of the system is governed by a quantum master equation with a time-independent, i.e., Markovian, generator. Such evolution is typically emerging under the assumption of a weak coupling between the system and an infinitely large bath. Here we are interested in understanding to which extent a neural network function approximator can predict open quantum dynamics—described by time-local generators—from an underlying unitary dynamics. We investigate this question using a class of spin models, which is inspired by recent experimental setups. We find that indeed time-local generators can be learned. In certain situations they are even time independent and allow to extrapolate the dynamics to unseen times. This might be useful for situations in which experiments or numerical simulations do not allow to capture long-time dynamics and for exploring thermalization occurring in closed quantum systems.

Paolo Mazza, *Dominik Zietlow*, Frederico Carollo, Sabine Andergassen, Georg Martius, Igor Lesanovsky
Physical Review Research; <https://arxiv.org/abs/2101.08591>

Inferring Markovian quantum master equations of few-body observables in interacting spin chains

Abstract

Full information about a many-body quantum system is usually out-of-reach due to the exponential growth - with the size of the system - of the number of parameters needed to encode its state. Nonetheless, in order to understand the complex phenomenology that can be observed in these systems, it is often sufficient to consider dynamical or stationary properties of local observables or, at most, of few-body correlation functions. These quantities are typically studied by singling out a specific subsystem of interest and regarding the remainder of the many-body system as an effective bath. In the simplest scenario, the subsystem dynamics, which is in fact an open quantum dynamics, can be approximated through Markovian quantum master equations. Here, we show how the generator of such a dynamics can be efficiently learned by means of a fully interpretable neural network which provides the relevant dynamical parameters for the subsystem of interest. Importantly, the neural network is constructed such that the learned generator implements a physically consistent open quantum time-evolution. We exploit our method to learn the generator of the dynamics of a subsystem of a many-body system subject to a unitary quantum dynamics. We explore the capability of the network to predict the time-evolution of a two-body subsystem and exploit the physical consistency of the generator to make predictions on the stationary state of the subsystem dynamics.

Francesco Carnazza, Federico Carollo, *Dominik Zietlow*, Sabine Andergassen, Georg Martius, Igor Lesanovsky
New Journal of Physics; <https://arxiv.org/abs/2201.11599>

Summary and connection to this thesis

The simulation of physical processes is often computationally complex, even infeasible for large systems, and typically only possible in an iterative fashion. In the first paper ([24]), we conducted two sets of experiments. We analyzed how well a linear model generalizes as a time evolution generator for a specific physical system, as depicted in Figure 7.3. This is particularly interesting as the typical iterative simulation procedure conceals the closed-form analytical solution, which could otherwise be analyzed directly. In the second experiment, we deviate from the time-global linear model to a time-local linear model (given by a hypermodel that predicts the linear time generator at any point in time). We compute the time dependence of the resulting generators, which yields insights into the dynamics of the physical system under investigation.

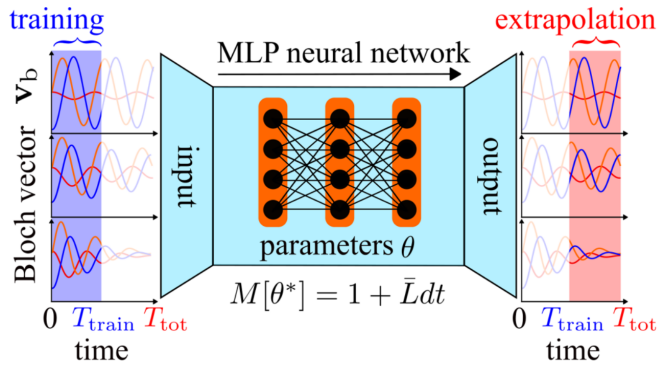


Figure 7.3: **Time-independent generator:** We train a linear model to obtain a time-independent generator. The training data are the time-dependent expectation values of the reduced system observables in a given time window (blue-shaded region). When the reduced dynamics is Markovian, this learned generator allows the network to make predictions for unseen times (red-shaded region).

7 *Related projects*

In the second paper ([26]), we train a neural network to approximate a parameterization of the so-called Lindblad operator. As the true physical dynamic generators are positive and trace-preserving, we choose a parameterization that enforces those properties by design. The learned dynamics are therefore constrained to be physically valid, which is not the case if the generator is learned in a less- or unconstrained way.

Although there is no direct connection between these papers and the work presented in this thesis, there are follow-up plans to investigate the curse of dimensionality in physical systems through the lens of deep representation learning. Quantum spin systems suffer from the same curse of dimensionality that is well known to deep learning practitioners. The state space of such systems scales exponentially with the number of spins and becomes impossible to model precisely (without approximations) for a comparatively small system size already. We want to investigate if and how representation learning can be used to predict the dynamics of high dimensional systems in a lower dimensional space. The grand goal of this project is to generalize beyond the system sizes used to train the representation learning models, thereby extending the realm of investigable system sizes.

7.4 InvGAN: Invertible GANs

Abstract

Generation of photo-realistic images, semantic editing and representation learning are a few of many potential applications of high resolution generative models. Recent progress in GANs have established them as an excellent choice for such tasks. However, since they do not provide an inference model, image editing or downstream tasks such as classification can not be done on real images using the GAN latent space. Despite numerous efforts to train an inference model or design an iterative method to invert a pre-trained generator, previous methods are dataset (e.g. human face images) and architecture (e.g. StyleGAN) specific. These methods are nontrivial to extend to novel datasets or architectures. We propose a general framework that is agnostic to architecture and datasets. Our key insight is that, by training the inference and the generative model together, we allow them to adapt to each other and to converge to a better quality model. Our **InvGAN**, short for Invertible GAN, successfully embeds real images to the latent space of a high quality generative model. This allows us to perform image inpainting, merging, interpolation and online data augmentation. We demonstrate this with extensive qualitative and quantitative experiments.

Partha Ghosh, *Dominik Zietlow*, Michael J. Black, Larry S. Davis, Xiaochen Hu

GCPR 2022; <https://arxiv.org/abs/2112.04598>

Summary and connection to this thesis

Despite their unparalleled performance in generating photo-realistic images, **GANs** typically lack an inference module that allows embedding images in their latent space. Existing approaches using **GANs** for ,e.g., image editing, up-scaling, sharpening, and so on often rely on directly optimizing the latent representation such that the generated image resembles the input. This optimization is computationally inefficient and can not be used in downstream tasks requiring high data throughput. Model-based inversion typically deals with pre-trained generators: They train an inversion model given a fixed generator. In this work, we propose an architecture that jointly trains a **GAN** and an inversion module with basically any given **GAN** backbone architecture. We achieve high-fidelity image reconstructions and only a minor decrease in the overall image quality of random latent samples. An example on the CelebA dataset is illustrated in Figure 7.4. This project tackles the task of closing the gap between **GANs** and **VAEs**.

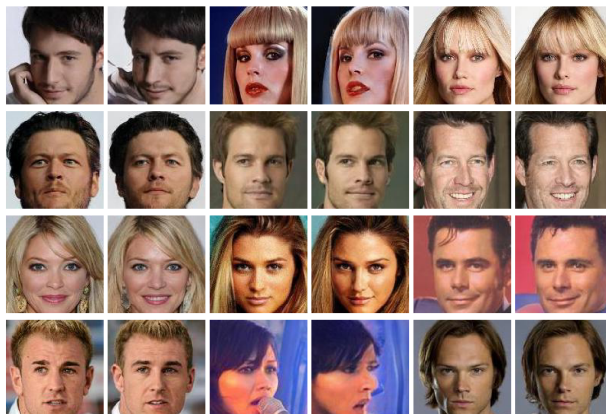


Figure 7.4: **InvGAN** reconstructions of CelebA. Alternating (from left to right) original and reconstructed images.

The inversion of the decoder is inherently part of the **VAE** training, but **VAEs** are vastly outperformed by **GANs** in terms of visual quality. By combining the training of the inversion module with the **GAN** training, we could combine the advantages of either architecture.

7.5 Assaying out-of-distribution generalization in transfer learning

Abstract

Since out-of-distribution generalization is a generally ill-posed problem, various proxy targets (e.g., calibration, adversarial robustness, algorithmic corruptions, invariance across shifts) were studied across different research programs resulting in different recommendations. While sharing the same aspirational goal, these approaches have never been tested under the same experimental conditions on real data. In this paper, we take a unified view of previous work, highlighting message discrepancies that we address empirically, and providing recommendations on how to measure the robustness of a model and how to improve it. To this end, we collect 172 publicly available dataset pairs for training and out-of-distribution evaluation of accuracy, calibration error, adversarial attacks, environment invariance, and synthetic corruptions. We fine-tune over 31k networks, from nine different architectures in the many- and few-shot setting. Our findings confirm that in- and out-of-distribution accuracies tend to increase jointly, but show that their relation is largely dataset-dependent, and in general more nuanced and more complex than posited by previous, smaller-scale studies.

Florian Wenzel, Andrea Dittadi, Peter Vincent Gehler, Carl-Johann Simon-Gabriel, Max Horn, ***Dominik Zietlow***, David Kernert, Chris Russell, Thomas Brox, Bernt Schiele, Bernhard Schölkopf, Francesco Locatello
NeurIPS 2022; <https://arxiv.org/abs/2207.09239>

Summary and connection to this thesis

Training models that are robust against distributional shifts is a core challenge in computer vision. The problem that models do not achieve generalization beyond the training dataset, limits their application to domains in which either vast amounts of (labeled) data are available, or to applications in which the target domain is particularly narrow. This paper analyzes how out-of-distribution accuracy can be estimated based on various other metrics (in- and out-of-distribution).

Figure 7.5 shows the results of a factor analysis with four factors on those metrics. We observed that they aggregate different combinations of metrics, indicated by the four differently colored bars. The **blue** factor captures classification error, adversarial error, log-likelihood, and their corrupted variants whereas the **green** factor contains almost only OOD metrics. The **yellow** factor represents the expected calibration error and the **red** factor the demographic disparity.

The desire for computer vision models that generalize beyond the training distribution also fuelled the research for disentangled representations. If a representation learning model were to disentangle perfectly, i.e., it recovered the generating factors, it could foster the generalization of models that ingest those representations. The reason is that the representation would be consistent independently of the distribution shift, e.g., a coordinate representing an object's position. However, it is unclear if a model's disentangling ability would trivially generalize out-of-distribution [100].

7.5 Assaying out-of-distribution generalization in transfer learning

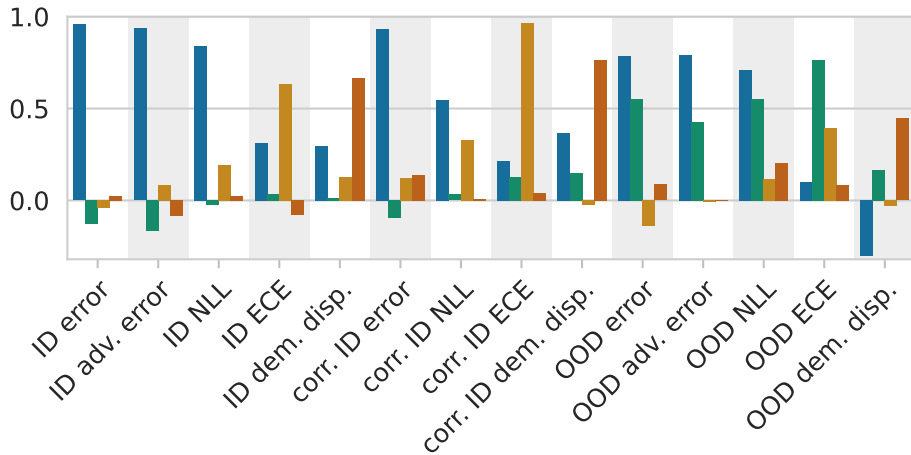


Figure 7.5: **Factor analysis:** Factor loadings (contributions) of different metrics based on a factor analysis with four orthogonal factors (color-coded), highlighting similarities between the metrics. The factor **Blue**: captures classification error, adversarial error, log-likelihood, and their corrupted variants. **Green**: only in OOD metrics. **Yellow**: expected calibration error. **Red**: demographic disparity.

7.6 Embrace the gap: VAEs perform independent mechanism analysis

Abstract

VAEs are a popular framework for modeling complex data distributions; they can be efficiently trained via variational inference by maximizing the ELBO, at the expense of a gap to the exact (log-) marginal likelihood. While VAEs are commonly used for representation learning, it is unclear why ELBO maximization would yield useful representations, since unregularized maximum likelihood estimation cannot invert the data-generating process. Yet, VAEs often succeed at this task. We seek to elucidate this apparent paradox by studying nonlinear VAEs in the limit of near-deterministic decoders. We first prove that, in this regime, the optimal encoder approximately inverts the decoder—a commonly used but unproven conjecture—which we refer to as *self-consistency*. Leveraging self-consistency, we show that the ELBO converges to a regularized log-likelihood. This allows VAEs to perform what has recently been termed **Independent Mechanism Analysis (IMA)**: it adds an inductive bias towards decoders with column-orthogonal Jacobians, which helps recover the true latent factors. The gap between ELBO and log-likelihood is therefore welcome since it bears unanticipated benefits for nonlinear representation learning. In experiments on synthetic and image data, we show that VAEs uncover the true latent factors when the data generating process satisfies the IMA assumption.

Patrik Reizinger, Luigi Gresele, Jack Brady, Julius Von Kügelgen, **Dominik Zietlow**, Bernhard Schölkopf, Georg Martius, Wieland Brendel, Michel Besserve;
NeurIPS 2022; <https://arxiv.org/abs/2206.02416>

Summary and connection to this thesis

This paper presents two major findings: Firstly, it proves that in the near-deterministic regime, the so-called *self-consistency* of VAEs holds. That means that the decoder inverts the encoder. Secondly, it shows that the ELBO converges to a regularized log-likelihood and more precisely matches the IMA objective. This result only holds in the case of a self-consistent model. As shown in Figure 7.6, with increasing γ^2 (the decoder’s precision), the difference between the self-consistent ELBO and the IMA objective vanishes. The remaining gap between the log-likelihood and the ELBO is exactly the IMA regularizer.

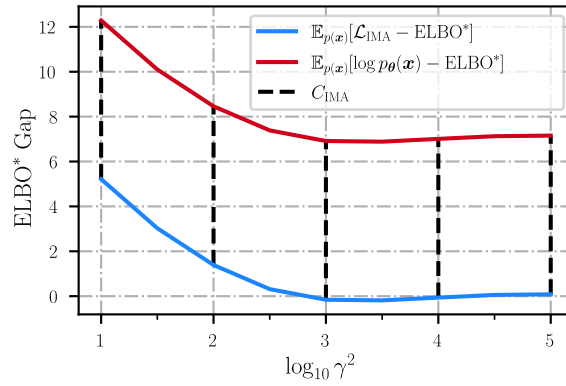


Figure 7.6: **ELBO gaps:** Comparison of the ELBO^* , the IMA-regularized and unregularized log-likelihoods over different γ^2 .

Those findings are strongly related to this thesis, where we investigated the effects of the implemented loss function of β -VAEs. In contrast, this paper provides a theoretical analysis of VAEs from the variational inference perspective. The IMA objective was motivated by formalizing signal independence via column orthogonality of the model’s Jacobian [98]. This is at heart the key property of VAEs, as discussed in Chapter 3. The required self-consistency property of the model reflects in the assumption of the polarized regime.

List of Tables

3.1	Experimental details: Overview of the used datasets and network architectures. The nonlinearities are only applied in the hidden layers. Biases are used for all datasets.	40
3.2	Validity of polarized regime: Percentage of training time where $\Delta_{KL} < 3\%$ (Equation (3.38)) continuously until the end. Reported for β -VAE with exact (dataset dependent) and high (10) latent dimension.	43
3.3	Orthogonality and disentanglement: Results for the distance to orthogonality DtO of the decoder (Equation (3.25)) and DS for different architectures and datasets. Lower DtO values are better and higher DS values are better. Random decoders provide a simple baseline for the numbers.	44
3.4	Degenerate singular values: Overview of DS and DtO for different ratios of importance between the generating factors for the Synth. Lin. task. A ratio of 1.2 means one generating factor is scaled by 1.2.	47
4.1	Primary hyperparameters, we used the defaults in the Disentanglement Library or literature values for any other parameter.	64

List of Tables

4.2	MIG scores for unmodified, modified, and noisy datasets. We report the mean and standard deviation over ten distinct random seeds for each setting. The regular autoencoder serves as a baseline (random alignment). PCL and the weakly supervised GAN from [66] are the only disentangling non-variational model. The modification leads to a significant drop in all variational methods.	66
4.3	DCI scores for unmodified, modified and noisy datasets. We report the mean and standard deviation over ten different random seeds for each setting. PCL is the only disentangling non-variational model. The modification leads to a significant drop in all variational methods.	67
4.4	FactorVAE scores for unmodified, modified and noisy datasets. We report the mean and standard deviation over ten distinct random seeds for each setting. PCL is the only disentangling non-variational model. The modification leads to a significant drop in all variational methods.	67
4.5	SAP scores for unmodified, modified and noisy datasets. We report the mean and standard deviation over ten different random seeds for each setting. PCL is the only disentangling non-variational model. The modification leads to a significant drop in all variational methods.	68

List of Figures

1.1	Latent traversals over a single latent coordinate on two exemplary images from the CelebA dataset [40] for a trained β -VAE. The latent coordinate isolates the azimuth angle.	3
2.1	Principal Component Analysis: PCA with two principal components on the three-dimensional green point cloud isolates the direction of the largest variance (blue line) and the orthogonal direction of the second highest variance (orange line). The resulting plane spanned by PCA is illustrated in blue shades.	13
2.2	Autencoder: A linear AE with two latent dimensions (left) spans the same space as PCA (see Figure 2.1), but has a randomly aligned coordinate system (orange and blue line). A nonlinear AE with the same dimensionality (right) finds a much better fitting representation. The surfaces spanned by the AEs are illustrated in blue shades. In either case, the two coordinate axes are not necessarily orthogonal.	14
2.3	Variational Autoencoder: A linear VAE with two latent dimensions (left) aligns perfectly with PCA (see Figure 2.1). A nonlinear VAE with the same dimensionality (right) finds a much better fitting representation that is also locally orthogonal. The surfaces spanned by the VAEs are illustrated in blue shades.	15

List of Figures

2.4	Geometric interpretation of the SVD: Sequential illustration of the effects of applying the corresponding SVD matrices $V\Sigma^\dagger U^\top$ (left to right) and $U\Sigma V^\top$ (right to left).	16
3.1	Latent space prior and posterior: For a rotationally symmetric distribution of the latent space (spherical contour lines), any transformation thereof would be invariant under rotations in latent space (and consequently so the log-likelihood and the ELBO). The rotational symmetry is instead broken by the diagonalization of the normal posterior (illustrated by the local heatmaps), which leads to axis-aligned representations.	26
3.2	Orthogonality in MV^\top: The vectors w_1, w_2 are the columns of MV^\top . Minimizing the product $\ w_1\ \ w_2\ $ while maintaining the volume $\ w_1\ \ w_2\ \cos(\alpha)$ results in $w_1 \perp w_2$.	34
3.3	KL loss landscape for the vanilla VAE implementation with the canonical prior and posterior. The loss is zero for $\mu_i = 0$ and $\sigma_i = 1$, which corresponds to data-point independent noise.	35
3.4	Orthogonality vs. Disentanglement: Axis alignment of the latent representation (low DtO) results in better disentanglement (higher score). Each data point corresponds to an independent run with 10, 30, or 50 epochs.	45
3.5	Degenerate singular values: For strong degeneracy, e.g., in the synthetic dataset with the two generating factors w_1 and w_2 on equal, uniform scale (top), the linear β -VAE generates arbitrarily rotated latent representations (bottom) here for the linear synthetic dataset.	46
3.6	Nonlinear Eigenfaces: Similarly to the work about Eigenfaces [88], β -VAEs allow for learning their nonlinear counterpart.	48

- 3.7 **Line search over β :** The hyper-parameter in the β -VAE allows to trade off reconstruction error and the KL loss. The plots show the DS (top) and the DtO (bottom) for dSprites (left) and synthetic datasets (right). The dashed lines indicate the parameter chosen for the experiments. 49
- 4.1 **Linear and nonlinear embeddings:** From left to right: (i) a 3-D point cloud and the corresponding 2-D PCA manifold (blue surface) with the canonical principal components (red/blue curves), (ii) a nonlinear 2-D manifold with its principal components, (iii) a locally perturbed 2-D manifold with its principal components which are rotated with respect to (ii), (iv) the goal of our modifications is to move each data-point closer to this *entangled* manifold. 58
- 4.2 **Image perturbation process:** Starting from ground truth generating factors \mathbf{w} , two β -VAE encoder-decoder pairs are initialized such that one (top) produces entangled and the other (bottom) disentangled representations. Another decoder-like network m is trained to produce additive manipulations to the original images x . The encoders are frozen and fed with the original images. The set of ground truth generating factors \mathbf{w} stays untouched by the modification. 62
- 4.3 **Example perturbations:** From left to right: Original images, manipulations and altered images. Top row shows an example of dSprites, the bottom for Shapes3D. 63
- 4.4 **MIG scores** for scaled literature hyperparameters over ten restarts for Shapes3D. Over-pruned models with fewer active units than generating factors were discarded. 69
- 4.5 **DCI scores** for scaled literature hyperparameters over ten restarts for Shapes3D. Over-pruned models with fewer active units than generating factors were discarded. 69

List of Figures

4.6	FactorVAE scores for scaled literature hyperparameters over ten restarts for Shapes3D. Over-pruned models with fewer active units than generating factors were discarded.	70
4.7	SAP scores for scaled literature hyperparameters over ten restarts for Shapes3D. Over-pruned models with fewer active units than generating factors were discarded.	70
4.8	Histogram of MIG scores for the VAE-based methods on the altered Shapes3D dataset. Although the mean MIG score is significantly reduced, some models still disentangle reasonably well. We expect that there are two or more nearby solutions for the optimization problem, and the manipulations foster the entangled one but do not fully exclude the disentangled solution.	71
4.9	Individual MIG scores for β -VAEs trained on the original and the altered Shapes3D dataset. The MIG drops for every generating factor, which leads to entanglement across all of them.	72
4.10	Latent traversals along two latent dimensions for different disentangled representations from independent β -VAE trainings on the original dataset. They encode the wall hue and orientation separately. We flipped the latent coordinates to match the same alignment.	73
4.11	Latent traversals along two latent dimensions for different entangled representations from independent β -VAE trainings on the original dataset. They encode a mixture of wall hue and orientation.	74
7.1	Keypoint matching: The points in either image are annotated solely by their position and have to be matched according to their semantics.	104

- 7.2 **g-SMOTE augmentations** with $k=3$. Given a datapoint (green) and two neighbors (orange), linear interpolations in GAN latent space yield diverse images. Nearest neighbors are chosen to share the target attribute (“*eyeglasses*”). We give all interpolated images the same attribute label value. 108
- 7.3 **Time-independent generator:** We train a linear model to obtain a time-independent generator. The training data are the time-dependent expectation values of the reduced system observables in a given time window (blue-shaded region). When the reduced dynamics is Markovian, this learned generator allows the network to make predictions for unseen times (red-shaded region). 111
- 7.4 **InvGAN reconstructions** of CelebA. Alternating (from left to right) original and reconstructed images. 114
- 7.5 **Factor analysis:** Factor loadings (contributions) of different metrics based on a factor analysis with four orthogonal factors (color-coded), highlighting similarities between the metrics. The factor **Blue:** captures classification error, adversarial error, log-likelihood, and their corrupted variants. **Green:** only in OOD metrics. **Yellow:** expected calibration error. **Red:** demographic disparity. 117
- 7.6 **ELBO gaps:** Comparison of the *ELBO**, the *IMA*-regularized and unregularized log-likelihoods over different γ^2 119

List of Acronyms

- VAE** Variational Autoencoder. [i–iii](#), [v–vii](#), [2–4](#), [6–8](#), [11](#), [13](#), [15](#), [22](#), [25](#), [27](#), [29](#), [30](#), [35](#), [37](#), [39](#), [43](#), [44](#), [51](#), [54–57](#), [59–61](#), [63](#), [65](#), [71](#), [75](#), [76](#), [90](#), [95–97](#), [114](#), [118](#), [119](#)
- PCA** Principal Component Analysis. [ii](#), [vi](#), [vii](#), [4](#), [7](#), [8](#), [11–15](#), [17](#), [22](#), [23](#), [30](#), [33](#), [46](#), [47](#), [51](#), [54–58](#), [60](#), [75](#), [90](#), [95–97](#)
- β -VAE** β Variational Autoencoder. [3](#), [5–7](#), [14](#), [22](#), [35](#), [39](#), [43–51](#), [59](#), [60](#), [64](#), [66–68](#), [71–76](#), [95–98](#), [100](#), [119](#)
- TC-VAE** Total Correlation Variational Autoencoder. [3](#), [6](#), [64](#), [66–68](#)
- FactorVAE** Factorized Variational Autoencoder. [3](#), [6](#), [18](#), [64](#), [66–68](#), [70](#)
- DIP-VAE** Disentangled Inferred Prior Variational Autoencoder. [3](#), [6](#)
- SlowVAE** Slow Variational Autoencoder. [3](#), [64–68](#), [96](#)
- InfoGAN** Information Maximizing Generative Adversarial Network. [3](#)
- DCIGN** Deep Convolution Inverse Graphics Network. [3](#)
- PC** Principal Component. [4](#), [47](#), [55](#), [56](#), [60](#)
- ICA** Independent Component Analysis. [5–7](#), [17](#), [95](#), [96](#), [100](#)
- SAP** Separated Attribute Predictability. [6](#), [18](#), [68](#), [70](#)
- DCI** Disentanglement Completeness Informativeness. [6](#), [18](#), [67](#), [69](#)
- MIG** Mutual Information Gap. [6](#), [18](#), [39](#), [41](#), [50](#), [60](#), [65](#), [66](#), [68](#), [69](#), [71](#), [72](#), [75](#)
- PCL** Permutation Contrastive Learning. [6](#), [65–68](#), [96](#)

List of Acronyms

- GAN** Generative Adversarial Network. 7, 65, 66, 96, 107, 108, 113, 114
- AE** Autoencoder. 11–14, 39, 44, 66
- SVD** Singular Value Decomposition. 11, 15, 16, 23, 24, 28, 30, 36, 56, 84, 86, 88–90, 92
- ELBO** Evidence Lower Bound. 13, 24–26, 118, 119
- KL** Kullback–Leibler. 14, 34–37, 39, 43, 48–50, 56, 88, 90, 92
- DtO** Distance to Orthogonality. 36, 44–50
- MILP** Mixed-Integer Linear Programming. 36
- DS** Disentanglement Score. 39, 41, 42, 44–49
- MLP** Multilayer Perceptron. 43, 64
- HSL** Hue, Saturation, and Lightness. 72
- SMOTE** Synthetic Minority Oversampling Strategy. 107, 108
- IMA** Independent Mechanism Analysis. 118, 119

Acknowledgments

First and foremost, I want to thank Michal and Georg for your supervision over the past four years. Your scientific rigor and curiosity had a lasting impression on me. I am thankful that you provided me with the chance to work in such a great environment around the MPI-IS; it was a pleasure! Thanks to the Autonomous Learning group (including but not limited to Andrii, Anselm, Christian, Cristina, Georg, Huanbo, Marin, Max, Michal, and Sebastian) for providing an excellent and comfortable environment throughout the whole time. For their academic guidance, I thank the members of my advisory committee, Georg Martius, Philipp Hennig, and Andreas Geiger. In this context, I also want to thank Leila and Sara as representatives of the IMPRS-IS program and for their individual efforts, particularly the organization of the annual boot camps. For reading and providing feedback on this thesis, I thank Alessandro, Michael, Cristina and Judith.

It is impossible to extensively list all those at and around the MPI that contributed to this work in one way or another, so I am not even attempting it. Thanks for countless hours of conversations, scientific discussions, card games, climbing, coffees, beers, hiking, pool, biking, foosball, skiing, concerts, festivals, binge-watching, gaming, and whatnot.

Weiterhin möchte ich mich bei Stefan und Lisa bedanken, für unsere Gespräche, euren Zuspruch und Rat. Ebenso gilt mein Dank Alessandro, als langjährigem Freund und “Leidensgenossen”, über die Schulzeit, das Studium und die Promotion hinweg. Auch Dir, Judith, danke ich für deine Unterstützung, deine verständnisvolle Art und dafür, dass Du mir hin- und wieder hilfst meine Prioritäten richtig zu setzen.

Acknowledgements

Zu guter Letzt möchte ich mich noch bei meiner Familie bedanken, allen voran bei meinen Eltern für die wortwörtlich bislang lebenslange Unterstützung! Ihr habt mir das alles ermöglicht: Das Studium, die Promotion aber vor allem auch die Freiheiten die ich währenddessen und darüber hinaus genießen konnte. Ganz besonderer Dank gilt an dieser Stelle natürlich auch meinen Brüdern. Durch Euch entstand meine Begeisterung für die Informatik, auch wenn ihr von künstlicher Intelligenz noch immer nicht viel haltet.

Bibliography

- [1] Zhuoyao Zhong, Lianwen Jin, and Zecheng Xie. “High performance of-line handwritten chinese character recognition using googlenet and directional feature maps”. In: *International Conference on Document Analysis and Recognition*. 2015.
- [2] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. “Understanding of a convolutional neural network”. In: *International Conference on Engineering and Technology*. 2017.
- [3] Athanasios Voulodimos et al. “Deep learning for computer vision: A brief review”. In: *Computational intelligence and neuroscience*. 2018.
- [4] David Beymer and Myron Flickner. “Eye gaze tracking using an active stereo head”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2003.
- [5] Moritz Kassner, William Patera, and Andreas Bulling. “Pupil: an open source platform for pervasive eye tracking and mobile gaze-based interaction”. In: *Proceedings of the ACM international joint conference on pervasive and ubiquitous computing: Adjunct publication*. 2014.
- [6] David Geisler, Dieter Fox, and Enkelejda Kasneci. “Real-time 3d glint detection in remote eye tracking based on bayesian inference”. In: *IEEE International Conference on Robotics and Automation*. 2018.
- [7] John Canny. “A computational approach to edge detection”. In: *IEEE Transactions on pattern analysis and machine intelligence*. 1986.
- [8] David G Lowe. “Object recognition from local scale-invariant features”. In: *Proceedings of the seventh IEEE international conference on computer vision*. 1999.



Bibliography

- [9] Paul Viola and Michael Jones. “Rapid object detection using a boosted cascade of simple features”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2001.
- [10] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. “Surf: Speeded up robust features”. In: *European Conference on Computer Vision*. 2006.
- [11] Yoshua Bengio, Aaron Courville, and Pascal Vincent. “Representation learning: A review and new perspectives”. In: *IEEE transactions on pattern analysis and machine intelligence*. 2013.
- [12] Diederik P. Kingma and Max Welling. “Auto-encoding variational bayes”. In: *Proceedings of the International Conference on Learning Representations*. 2014.
- [13] Irina Higgins et al. “ β -VAE: Learning basic visual concepts with a constrained variational framework”. In: *Proceedings of the International Conference on Learning Representations*. 2017.
- [14] Hyunjik Kim and Andriy Mnih. “Disentangling by factorising”. In: *Proceedings of the International Conference on Machine Learning*. 2018.
- [15] Francesco Locatello et al. “Challenging common assumptions in the unsupervised learning of disentangled representations”. In: *Proceedings of the International Conference on Machine Learning*. 2019.
- [16] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. “Variational inference of disentangled latent concepts from unlabeled observations”. In: *Proceedings of the International Conference on Learning Representations*. 2018.
- [17] David A. Klindt et al. “Towards nonlinear disentanglement in natural data with temporal sparse coding”. In: *Proceedings of the International Conference on Learning Representations*. 2021.
- [18] Xi Chen et al. “Infogan: Interpretable representation learning by information maximizing generative adversarial nets”. In: *Advances in Neural Information Processing Systems*. 2016.
- [19] Irina Higgins et al. “Towards a definition of disentangled representations”. In: *ArXiv e-prints*. 2018.




- [20] Michal Rolinek, Dominik Zietlow, and Georg Martius. “Variational autoencoders pursue PCA directions (by accident)”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019. 
- [21] Dominik Zietlow, Michal Rolinek, and Georg Martius. “Demystifying inductive biases for (Beta-)VAE based architectures”. In: *Proceedings of the International Conference on Machine Learning*. 2021. 
- [22] Dominik Zietlow et al. “Leveling down in computer vision: Pareto inefficiencies in fair deep classifiers”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022. 
- [23] Michal Rolinek et al. “Deep graph matching via blackbox differentiation of combinatorial solvers”. In: *European Conference on Computer Vision*. 2020. 
- [24] Paolo P. Mazza et al. “Machine learning time-local generators of open quantum dynamics”. In: *Phys. Rev. Research*. 2021. 
- [25] Partha Ghosh et al. “InvGAN: Invertible GANs”. In: *ArXiv e-prints*. 2021. 
- [26] Francesco Carnazza et al. “Inferring Markovian quantum master equations of few-body observables in interacting spin chains”. In: *New Journal of Physics*. 2022. 
- [27] Florian Wenzel et al. “Assaying out-of-distribution generalization in transfer learning”. In: *ArXiv e-prints*. 2022. 
- [28] Patrik Reizinger et al. “Embrace the Gap: VAEs Perform Independent Mechanism Analysis”. In: *ArXiv e-prints*. 2022. 
- [29] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. “Stochastic backpropagation and approximate inference in deep generative models”. In: *Proceedings of the International Conference on Machine Learning*. 2014. 

Bibliography

- [30] Tejas D Kulkarni et al. “Deep convolutional inverse graphics network”. In: *Advances in Neural Information Processing Systems*. 2015.
- [31] Karol Gregor et al. “Towards conceptual compression”. In: *Advances in Neural Information Processing Systems*. 2016.
- [32] Diederik P Kingma et al. “Improved variational inference with inverse autoregressive flow”. In: *Advances in Neural Information Processing Systems*. 2016.
- [33] Karol Gregor et al. “Draw: A recurrent neural network for image generation”. In: *Proceedings of the International Conference on Machine Learning*. 2015.
- [34] Matt J Kusner, Brooks Paige, and José Miguel Hernández-Lobato. “Grammar variational autoencoder”. In: *Proceedings of the International Conference on Machine Learning*. 2017.
- [35] X. Hou et al. “Deep feature consistent variational autoencoder”. In: *IEEE Winter Conference on Applications of Computer Vision (WACV)*. 2017.
- [36] Danilo Rezende et al. “One-shot generalization in deep generative models”. In: *Proceedings of the International Conference on Machine Learning*. 2016.
- [37] Zichao Yang et al. “Improved variational autoencoders for text modeling using dilated convolutions”. In: *Proceedings of the International Conference on Machine Learning*. 2017.
- [38] Merlijn Blaauw and Jordi Bonada. “Modeling and transforming speech using variational autoencoders”. In: *Interspeech*. 2016.
- [39] Cheng Zhang et al. “Advances in variational inference”. In: *IEEE transactions on pattern analysis and machine intelligence*. 2018.
- [40] Ziwei Liu et al. “Deep learning face attributes in the wild”. In: *Proceedings of the IEEE international conference on computer vision*. 2015.



- [41] Ricky TQ Chen et al. “Isolating sources of disentanglement in variational autoencoders”. In: *Advances in Neural Information Processing Systems*. 2018. 
- [42] Ian Goodfellow et al. “Generative adversarial nets”. In: *Advances in Neural Information Processing Systems*. 2014. 
- [43] Partha Ghosh et al. “From variational to deterministic autoencoders”. In: *Proceedings of the International Conference on Learning Representations*. 2020. 
- [44] Alireza Makhzani et al. “Adversarial autoencoders”. In: *Proceedings of the International Conference on Learning Representations*. 2016. 
- [45] L. Mescheder, S. Nowozin, and A. Geiger. “Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks”. In: *Proceedings of the International Conference on Machine Learning*. 2017. 
- [46] Christopher P Burgess et al. “Understanding disentangling in β -VAE”. In: *ArXiv e-prints*. 2018. 
- [47] Jan Stühmer, Richard Turner, and Sebastian Nowozin. “ISA-VAE: Independent subspace analysis with variational autoencoders”. In: *Proceedings of the International Conference on Learning Representations*. 2019. 
- [48] B. Dai et al. “Hidden talents of the variational autoencoder”. In: *ArXiv e-prints*. 2018. 
- [49] Alexander Alemi et al. “Fixing a broken ELBO”. In: *Proceedings of the International Conference on Machine Learning*. 2018. 
- [50] Emile Mathieu et al. “Disentangling disentanglement in variational autoencoders”. In: *Proceedings of the International Conference on Machine Learning*. 2019. 
- [51] Bin Dai and David P. Wipf. “Diagnosing and enhancing VAE models”. In: *Proceedings of the International Conference on Learning Representations*. 2019. 

Bibliography

- [52] Pierre Comon. “Independent component analysis, A new concept?” In: *Signal Processing*. 1994.
- [53] Anthony J Bell and Terrence J Sejnowski. “An information-maximization approach to blind separation and blind deconvolution”. In: *Neural computation*. 1995.
- [54] Zhixin Shu et al. “Neural face editing with intrinsic image disentangling”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2017.
- [55] Yiyi Liao et al. “Towards unsupervised learning of generative models for 3d controllable image synthesis”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020.
- [56] Tameem Adel, Zoubin Ghahramani, and Adrian Weller. “Discovering interpretable representations for both deep generative and discriminative models”. In: *Proceedings of the International Conference on Machine Learning*. 2018.
- [57] Hyunjik Kim. “Interpretable models in probabilistic machine learning”. University of Oxford, 2019.
- [58] Bernhard Schölkopf. “Causality for machine learning”. In: *Probabilistic and Causal Inference: The Works of Judea Pearl*. 2022.
- [59] Raphael Suter et al. “Robustly disentangled causal mechanisms: Validating deep representations for interventional robustness”. In: *Proceedings of the International Conference on Machine Learning*. 2019.
- [60] Michel Besserve et al. “Group invariance principles for causal generative models”. In: *Proceedings of the International Conference on Artificial Intelligence and Statistics*. 2018.
- [61] Michel Besserve et al. “A theory of independent mechanisms for extrapolation in generative models”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. 2021.
- [62] Cian Eastwood and Christopher KI Williams. “A framework for the quantitative evaluation of disentangled representations”. In: *Proceedings of the International Conference on Learning Representations*. 2018.



- [63] Ilyes Khemakhem et al. “Variational autoencoders and nonlinear ICA: A unifying framework”. In: *Proceedings of the International Conference on Artificial Intelligence and Statistics*. 2020. 
- [64] Aapo Hyvarinen and Hiroshi Morioka. “Nonlinear ICA of temporally dependent stationary sources”. In: *Proceedings of the International Conference on Artificial Intelligence and Statistics*. 2017. 
- [65] Roland S Zimmermann et al. “Contrastive learning inverts the data generating process”. In: *Proceedings of the International Conference on Machine Learning*. 2021. 
- [66] Rui Shu et al. “Weakly supervised disentanglement with guarantees”. In: *Proceedings of the International Conference on Learning Representations*. 2020. 
- [67] Michael E Tipping and Christopher M Bishop. “Probabilistic principal component analysis”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*. 1999. 
- [68] James Lucas et al. “Don’t blame the ELBO! A linear VAE perspective on posterior collapse”. In: *Advances in Neural Information Processing Systems*. 2019. 
- [69] Abhishek Kumar and Ben Poole. “On implicit regularization in β -VAEs”. In: *Proceedings of the International Conference on Machine Learning*. 2020. 
- [70] Aapo Hyvärinen and Petteri Pajunen. “Nonlinear independent component analysis: Existence and uniqueness results”. In: *Neural Networks*. 1999. 
- [71] Sunny Duan et al. “Unsupervised model selection for variational disentangled representation learning”. In: *Proceedings of the International Conference on Learning Representations*. 2020. 
- [72] Arun Pandey et al. “Disentangled representation learning and generation with manifold optimization”. In: *Neural Computation*. 2022. 
- [73] Alexander Rakowski and Christoph Lippert. “Disentanglement and local directions of variance”. In: *Machine Learning and Knowledge Discovery in Databases. Research Track*. Cham, 2021. ISBN: 978-3-030-86523-8. 

Bibliography

- [74] Rasmus Bro and Age K Smilde. “Principal component analysis”. In: *Analytical methods*. 2014.
- [75] Walter Hugo Lopez Pinaya et al. “Chapter 11 - Autoencoders”. In: *Machine Learning*. 2020. ISBN: 978-0-12-815739-8.
- [76] Diederik P. Kingma and Max Welling. “An introduction to variational autoencoders”. In: *Foundations and Trends® in Machine Learning*. 2019.
- [77] G. H. Golub and W. Kahan. “Calculating the singular values and pseudo-inverse of a Matrix”. In: *Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis*. 1965.
- [78] Jürgen Schmidhuber. “Learning factorial codes by predictability minimization”. In: *Neural Computation*. 1992.
- [79] Karl Ridgeway. “A survey of inductive biases for factorial representation-learning”. In: *ArXiv e-prints*. 2016.
- [80] Ali Razavi et al. “Preventing posterior collapse with delta-VAEs”. In: *Proceedings of the International Conference on Learning Representations*. 2019.
- [81] James Lucas et al. *Understanding posterior collapse in generative latent variable models*. 2019.
- [82] H. Bourlard and Y. Kamp. *Auto-association by multilayer perceptrons and singular value decomposition*. Brussels, Belgium: Philips Research Laboratory, 1987.
- [83] Gareth James et al. *An introduction to statistical learning: With applications in R*. 2014. ISBN: 1461471370.
- [84] Loic Matthey et al. *dSprites: Disentanglement testing Sprites dataset*. 2017.



- [85] Y. LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE*. 1998.
- [86] Han Xiao, Kashif Rasul, and Roland Vollgraf. *Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms*. 2017.
- [87] Pauli Virtanen et al. “SciPy 1.0: Fundamental algorithms for scientific computing in python”. In: *Nature Methods*. 2020.
- [88] Matthew Turk and Alex Pentland. “Eigenfaces for recognition”. In: *Journal of cognitive neuroscience*. 1991.
- [89] Francesco Locatello et al. “A sober look at the unsupervised learning of disentangled representations and their evaluation”. In: *Journal of Machine Learning Research*. 2020.
- [90] Chris Burgess and Hyunjik Kim. *3D shapes dataset*. 2018.
- [91] Hervé Bourlard and Yves Kamp. “Auto-association by multilayer perceptrons and singular value decomposition”. In: *Biological cybernetics*. 1988.
- [92] Pierre Baldi and Kurt Hornik. “Neural networks and principal component analysis: Learning from examples without local minima”. In: *Neural networks*. 1989.
- [93] Geoffrey E Hinton and Ruslan R Salakhutdinov. “Reducing the dimensionality of data with neural networks”. In: *Science*. 2006.
- [94] Jan Stuehmer, Richard Turner, and Sebastian Nowozin. “Independent subspace analysis for unsupervised learning of disentangled representations”. In: *Aistats*. 2020.
- [95] George H Joblove and Donald Greenberg. “Color spaces for computer graphics”. In: *Proceedings of the 5th annual conference on Computer graphics and interactive techniques*. 1978.



Bibliography

- [96] Yann LeCun, Fu Jie Huang, and Leon Bottou. “Learning methods for generic object recognition with invariance to pose and lighting”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2004.
- [97] Lloyd N Trefethen and David Bau III. *Numerical linear algebra*. 1997.
- [98] Luigi Gresele et al. “Independent mechanism analysis, a new concept?” In: *Advances in Neural Information Processing Systems*. 2021.
- [99] Nitesh V Chawla et al. “SMOTE: synthetic minority over-sampling technique”. In: *Journal of artificial intelligence research*. 2002.
- [100] Lukas Schott et al. “Visual representation learning does not generalize strongly within the same domain”. In: *Proceedings of the International Conference on Learning Representations*. 2022.

