


# A Sober Look at the Unsupervised Learning of Disentangled Representations and their Evaluation

**Journal Article****Author(s):**

Locatello, Francesco; Bauer, Stefan; Lucic, Mario; [Rätsch, Gunnar](#) ; Gelly, Sylvain; Schölkopf, Bernhard; Bachem, Olivier

**Publication date:**

2020-09

**Permanent link:**

<https://doi.org/10.3929/ethz-b-000450167>

**Rights / license:**

[Creative Commons Attribution 4.0 International](#)

**Originally published in:**

Journal of Machine Learning Research 21

# A Sober Look at the Unsupervised Learning of Disentangled Representations and their Evaluation

**Francesco Locatello**

FRANCESCO.LOCATELLO@INF.ETHZ.CH

*Department of Computer Science*

*ETH Zurich*

*Universitätsstrasse 6, 8092 Zürich, Switzerland*

**Stefan Bauer**

STEFAN.BAUER@TUEBINGEN.MPG.DE

*Department of Empirical Inference*

*Max Planck Institute for Intelligent Systems*

*Max-Planck-Ring 4, 72076 Tübingen, Germany*

**Mario Lucic**

LUCIC@GOOGLE.COM

*Google Research, Brain Team*

*Brandschenkestrasse 110, 8002 Zürich, Switzerland*

**Gunnar Rätsch**

RATSCH@INF.ETHZ.CH

*Department of Computer Science*

*ETH Zurich*

*Universitätsstrasse 6, 8092 Zürich, Switzerland*

**Sylvain Gelly**

SYLVAINGELLY@GOOGLE.COM

*Google Research, Brain Team*

*Brandschenkestrasse 110, 8002 Zürich, Switzerland*

**Bernhard Schölkopf**

BS@TUEBINGEN.MPG.DE

*Department of Empirical Inference*

*Max Planck Institute for Intelligent Systems*

*Max-Planck-Ring 4, 72076 Tübingen, Germany*

**Olivier Bachem**

BACHEM@GOOGLE.COM

*Google Research, Brain Team*

*Brandschenkestrasse 110, 8002 Zürich, Switzerland*

**Editor:** Kilian Weinberger

## Abstract

The idea behind the *unsupervised* learning of *disentangled* representations is that real-world data is generated by a few explanatory factors of variation which can be recovered by unsupervised learning algorithms. In this paper, we provide a sober look at recent progress in the field and challenge some common assumptions. We first theoretically show that the unsupervised learning of disentangled representations is fundamentally impossible without inductive biases on both the models and the data. Then, we train over 14 000 models covering most prominent methods and evaluation metrics in a reproducible large-scale experimental study on eight data sets. We observe that while the different methods successfully enforce properties “encouraged” by the corresponding losses, well-disentangled models seemingly cannot be identified without supervision. Furthermore, different evaluation metrics do not always agree on what should be considered “disentangled” and exhibit systematic differences in the estimation. Finally, increased disentanglement does not seem to

necessarily lead to a decreased sample complexity of learning for downstream tasks. Our results suggest that future work on disentanglement learning should be explicit about the role of inductive biases and (implicit) supervision, investigate concrete benefits of enforcing disentanglement of the learned representations, and consider a reproducible experimental setup covering several data sets.

**Keywords:** Disentangled representations, impossibility, evaluation, reproducibility, large scale experimental study.

## 1. Introduction

In representation learning it is often assumed that real-world observations  $\mathbf{x}$  (such as images or videos) are generated by a two-step generative process. First, a multivariate latent random variable  $\mathbf{z}$  is sampled from a distribution  $P(\mathbf{z})$ . Intuitively,  $\mathbf{z}$  corresponds to semantically meaningful factors of variation of the observations (such as content and position of objects in an image). Then, in a second step, the observation  $\mathbf{x}$  is sampled from the conditional distribution  $P(\mathbf{x}|\mathbf{z})$ . The key idea behind this model is that the high-dimensional data  $\mathbf{x}$  can be explained by the substantially lower dimensional and semantically meaningful latent variable  $\mathbf{z}$  which is mapped to the higher-dimensional space of observations  $\mathbf{x}$ . Informally, the goal of representation learning is to find useful transformations  $r(\mathbf{x})$  of  $\mathbf{x}$  that “make it easier to extract useful information when building classifiers or other predictors” (Bengio et al., 2013).

A recent line of work has argued that representations that are *disentangled* are an important step towards a better representation learning (Bengio et al., 2013; Peters et al., 2017; LeCun et al., 2015; Bengio et al., 2007; Schmidhuber, 1992; Lake et al., 2017; Tschannen et al., 2018). They should contain all the information present in  $\mathbf{x}$  in a compact and interpretable structure (Bengio et al., 2013; Kulkarni et al., 2015; Chen et al., 2016) while being independent from the task at hand (Goodfellow et al., 2009; Lenc and Vedaldi, 2015). They should be useful for (semi-)supervised learning of downstream tasks, transfer and few shot learning (Bengio et al., 2013; Schölkopf et al., 2012; Peters et al., 2017). They should enable to integrate out nuisance factors (Kumar et al., 2018), to perform interventions, and to answer counterfactual questions (Pearl, 2009; Spirtes et al., 2000; Peters et al., 2017).

While there is no single formalized notion of disentanglement (yet) which is widely accepted, the key intuition is that a disentangled representation should separate the distinct, informative *factors of variations* in the data (Bengio et al., 2013). A change in a single underlying factor of variation  $z_i$  should lead to a change in a single factor in the learned representation  $r(\mathbf{x})$ . This assumption can be extended to groups of factors as, for instance, in the work of Bouchacourt et al. (2018) or Suter et al. (2019). Based on this idea, a variety of disentanglement evaluation protocols have been proposed leveraging the statistical relations between the learned representation and the ground-truth factor of variations. Disentanglement is then measured as a particular structural property of these relations (Higgins et al., 2017a; Kim and Mnih, 2018; Eastwood and Williams, 2018; Kumar et al., 2018; Chen et al., 2018; Ridgeway and Mozer, 2018). We can group the disentanglement scores in two categories. The scores proposed by Higgins et al. (2017a), Kim and Mnih (2018) and Suter et al. (2019) all require interventions. The first two involve intervening on a factor of variation for each batch and then predicting which factor was intervened on and the third one measures deviations in the latent space after performing the intervention. The scores proposed by Eastwood and Williams (2018); Kumar et al. (2018); Chen et al. (2018); Ridgeway and Mozer (2018) first construct a matrix of relation between factors of variation and codes (for example pairwise mutual information) and then aggregate this matrix into a single final number. Typically, this step involves computing some normalized gap between the largest and second largest entries either row or column-wise.

State-of-the-art approaches for unsupervised disentanglement learning are largely based on *Variational Autoencoders (VAEs)* (Kingma and Welling, 2014): One assumes a specific prior  $P(\mathbf{z})$  on the latent space and then uses a deep neural network to parameterize the conditional probability  $P(\mathbf{x}|\mathbf{z})$ . Similarly, the distribution  $P(\mathbf{z}|\mathbf{x})$  is approximated using a variational distribution  $Q(\mathbf{z}|\mathbf{x})$ , again parametrized using a deep neural network. The model is then trained by minimizing a suitable approximation to the negative log-likelihood. The representation for  $r(\mathbf{x})$  is usually taken to be the mean of the approximate posterior distribution  $Q(\mathbf{z}|\mathbf{x})$ . Several variations of VAEs were proposed with the motivation that they lead to better disentanglement (Higgins et al., 2017a; Burgess et al., 2018; Kim and Mnih, 2018; Chen et al., 2018; Kumar et al., 2018). The common theme behind all these approaches is that they try to enforce a factorized aggregated posterior  $\int_{\mathbf{x}} Q(\mathbf{z}|\mathbf{x})P(\mathbf{x})d\mathbf{x}$ , which should encourage disentanglement.

## 1.1 Our Contributions

In this paper, we challenge commonly held assumptions in this field in both theory and practice. Our key contributions can be summarized as follows:

- We theoretically prove that (perhaps unsurprisingly) the unsupervised learning of disentangled representations is fundamentally impossible without inductive biases both on the considered learning approaches and the data sets.
- We investigate current approaches and their inductive biases in a reproducible large-scale experimental study<sup>1</sup> with a sound experimental protocol for unsupervised disentanglement learning. We implement six recent unsupervised disentanglement learning methods as well as seven disentanglement measures from scratch and train more than 14 000 models on eight data sets.
- We release `disentanglement_lib`<sup>2</sup>, a new library to train and evaluate disentangled representations. As reproducing our results requires substantial computational effort, we also release more than 10 000 trained models which can be used as baselines for future research.
- We analyze our experimental results and challenge common beliefs in unsupervised disentanglement learning: (i) While all considered methods prove effective at ensuring that the individual dimensions of the aggregated posterior (which is sampled) are not correlated, we observe that the dimensions of the representation (which is taken to be the mean) are correlated. (ii) We do not find any evidence that the considered models can be used to reliably learn disentangled representations in an *unsupervised* manner as random seeds and hyperparameters seem to matter more than the model choice. Furthermore, good trained models seemingly cannot be identified without access to ground-truth labels even if we are allowed to transfer good hyperparameter values across data sets. (iii) We observe systematic differences in the evaluation of disentangled representations. These differences arise both from how disentanglement is “defined” and how the relations between factors of variation and the dimensions of the representation are estimated. (iv) For the considered models and data sets, we cannot validate the assumption that disentanglement is useful for downstream tasks, for example through a decreased sample complexity of learning.
- Based on these empirical evidence, we suggest three critical areas of further research: (i) The role of inductive biases and implicit and explicit supervision should be made explicit: unsupervised

1. Reproducing these experiments requires approximately 2.92 GPU years (NVIDIA P100).

2. [https://github.com/google-research/disentanglement\\_lib](https://github.com/google-research/disentanglement_lib)



model selection persists as a key question. (ii) The concrete practical benefits of enforcing a specific notion of disentanglement of the learned representations should be demonstrated. (iii) Experiments should be conducted in a reproducible experimental setup on data sets of varying degrees of difficulty and with a clear evaluation protocol.

## 1.2 Roadmap

In Section 2 we briefly discuss other related works. In Section 3, we present our theoretical result with extensive discussion of its implications. In Section 4, we discuss our experimental design. In Sections 5, 6, and 7 we present the results of our experimental studies concerning the training, evaluation metrics, and downstream performance respectively. In Section 8, we summarize the implications of our findings and highlight directions for future research.

## 2. Other Related Work

In a similar spirit to disentanglement, (non-)linear independent component analysis (Comon, 1994; Bach and Jordan, 2002; Jutten and Karhunen, 2003; Hyvarinen and Morioka, 2016) studies the problem of recovering independent components of a signal. The underlying assumption is that there is a generative model for the signal composed of the combination of statistically independent non-Gaussian components. While the identifiability result for linear ICA (Comon, 1994) proved to be a milestone for the classical theory of factor analysis, similar results are in general not obtainable for the nonlinear case and the underlying sources generating the data cannot be identified (Hyvärinen and Pajunen, 1999). The lack of almost any identifiability result in non-linear ICA has been a main bottleneck for the utility of the approach (Hyvärinen et al., 2019) and partially motivated alternative machine learning approaches (Desjardins et al., 2012; Schmidhuber, 1992; Cohen and Welling, 2014b). Given that unsupervised algorithms did not initially perform well on realistic settings most of the other works have considered some more or less explicit form of supervision (Reed et al., 2014; Zhu et al., 2014; Yang et al., 2015; Kulkarni et al., 2015; Cheung et al., 2014; Mathieu et al., 2016; Narayanaswamy et al., 2017; Suter et al., 2019). (Hinton et al., 2011; Cohen and Welling, 2014a) assume some knowledge of the effect of the factors of variations even though they are not observed. One can also exploit known relations between factors in different samples (Karaletsos et al., 2015; Goroshin et al., 2015; Whitney et al., 2016; Fraccaro et al., 2017; Denton and Birodkar, 2017; Hsu et al., 2017; Yingzhen and Mandt, 2018b; Locatello et al., 2018). This is not a limiting assumption especially in sequential data like for videos. There is for example a rich literature in disentangling pose from content in 3D objects and content from motion in videos or time series in general (Yang et al., 2015; Yingzhen and Mandt, 2018a; Hsieh et al., 2018; Fortuin et al., 2019; Deng et al., 2017; Goroshin et al., 2015). Similarly, the non-linear ICA community recently shifted to non-iid data types exploiting time dependent or grouped observations (Hyvärinen and Morioka, 2016; Hyvärinen et al., 2019; Gresele et al., 2019)

We focus our study on the setting where factors of variations are not observable at all, that is, we only observe samples from  $P(\mathbf{x})$ .

### 3. Impossibility Result

The first question that we investigate is whether unsupervised disentanglement learning is even possible for arbitrary generative models. Theorem 1 essentially shows that without inductive biases both on models and data sets the task is fundamentally impossible. The proof is provided in Appendix A.

**Theorem 1** *For  $d > 1$ , let  $\mathbf{z} \sim P$  denote any distribution which admits a density  $p(\mathbf{z}) = \prod_{i=1}^d p(z_i)$ . Then, there exists an infinite family of bijective functions  $f : \text{supp}(\mathbf{z}) \rightarrow \text{supp}(\mathbf{z})$  such that  $\frac{\partial f_i(\mathbf{u})}{\partial u_j} \neq 0$  almost everywhere for all  $i$  and  $j$  (implying that  $\mathbf{z}$  and  $f(\mathbf{z})$  are completely entangled) and  $P(\mathbf{z} \leq \mathbf{u}) = P(f(\mathbf{z}) \leq \mathbf{u})$  for all  $\mathbf{u} \in \text{supp}(\mathbf{z})$  (they have the same marginal distribution).*

Consider the commonly used “intuitive” notion of disentanglement which advocates that a change in a single ground-truth factor should lead to a single change in the representation. In that setting, Theorem 1 implies that unsupervised disentanglement learning is *impossible* for arbitrary generative models with a factorized prior<sup>3</sup> in the following sense: Assume we have  $p(\mathbf{z})$  and some  $P(\mathbf{x}|\mathbf{z})$  defining a generative model. Consider any unsupervised disentanglement method and assume that it finds a representation  $r(\mathbf{x})$  that is perfectly disentangled with respect to  $\mathbf{z}$  in the generative model. Then, Theorem 1 implies that there is an equivalent generative model with the latent variable  $\hat{\mathbf{z}} = f(\mathbf{z})$  where  $\hat{\mathbf{z}}$  is completely *entangled* with respect to  $\mathbf{z}$  and thus also  $r(\mathbf{x})$ : as all the entries in the Jacobian of  $f$  are non-zero, a change in a single dimension of  $\mathbf{z}$  implies that all dimensions of  $\hat{\mathbf{z}}$  change. Furthermore, since  $f$  is deterministic and  $p(\mathbf{z}) = p(\hat{\mathbf{z}})$  almost everywhere, both generative models have the same marginal distribution of the observations  $\mathbf{x}$  by construction, that is,  $P(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z} = \int p(\mathbf{x}|\hat{\mathbf{z}})p(\hat{\mathbf{z}})d\hat{\mathbf{z}}$ . Since the (unsupervised) disentanglement method only has access to observations  $\mathbf{x}$ , it hence cannot distinguish between the two equivalent generative models and thus has to be entangled to at least one of them.

This may not be surprising to readers familiar with the causality and ICA literature as it is consistent with the following argument: After observing  $\mathbf{x}$ , we can construct infinitely many generative models which have the same marginal distribution of  $\mathbf{x}$ . Any one of these models could be the true causal generative model for the data, and the right model cannot be identified given only the distribution of  $\mathbf{x}$  (Peters et al., 2017). Similar results have been obtained in the context of non-linear ICA (Hyvärinen and Pajunen, 1999). The main novelty of Theorem 1 is that it allows the explicit construction of latent spaces  $\mathbf{z}$  and  $\hat{\mathbf{z}}$  that are completely *entangled* with each other in the sense of (Bengio et al., 2013). We note that while this result is very intuitive for multivariate Gaussians it also holds for distributions which are not invariant to rotation, for example multivariate uniform distributions.

While Theorem 1 shows that unsupervised disentanglement learning is fundamentally impossible for arbitrary generative models, this does not necessarily mean it is an impossible endeavour in practice. After all, real world generative models may have a certain structure that could be exploited through suitably chosen inductive biases. However, Theorem 1 clearly shows that inductive biases are required both for the models (so that we find a specific set of solutions) and for the data sets (such that these solutions match the true generative model). We hence argue that the role of inductive biases should be made explicit and investigated further as done in the following experimental study.

3. Theorem 1 only applies to factorized priors; however, we expect that a similar result can be extended to non-factorizing priors.

## 4. Experimental Design

In this section, we discuss the methods, evaluation metrics, data sets and overall experimental conditions of our study.

### 4.1 Considered Methods

All the considered methods augment the VAE loss with a regularizer: The  $\beta$ -VAE (Higgins et al., 2017a), introduces a hyperparameter in front of the KL regularizer of vanilla VAEs to constrain the capacity of the VAE bottleneck. The AnnealedVAE (Burgess et al., 2018) progressively increase the bottleneck capacity so that the encoder can focus on learning one factor of variation at the time (the one that most contribute to a small reconstruction error). The FactorVAE (Kim and Mnih, 2018) and the  $\beta$ -TCVAE (Chen et al., 2018) penalize the total correlation (Watanabe, 1960) with adversarial training (Nguyen et al., 2010; Sugiyama et al., 2012) or with a tractable but biased Monte-Carlo estimator respectively. The DIP-VAE-I and the DIP-VAE-II (Kumar et al., 2018) both penalize the mismatch between the aggregated posterior and a factorized prior. Implementation details can be found in Appendix E.

#### 4.1.1 UNSUPERVISED LEARNING OF DISENTANGLED REPRESENTATIONS WITH VAEs

Variants of variational autoencoders Kingma and Welling (2014) are considered the state-of-the-art for unsupervised disentanglement learning. One assumes a specific prior  $P(\mathbf{z})$  on the latent space and then parameterizes the conditional probability  $P(\mathbf{x}|\mathbf{z})$  with a deep neural network. Similarly, the distribution  $P(\mathbf{z}|\mathbf{x})$  is approximated using a variational distribution  $Q(\mathbf{z}|\mathbf{x})$ , again parametrized using a deep neural network. One can then derive the following approximation to the maximum likelihood objective,

$$\max_{\phi, \theta} \mathbb{E}_{p(\mathbf{x})} [\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z}))] \quad (1)$$

which is also known as the evidence lower bound (ELBO). By carefully considering the KL term, one can encourage various properties of the resulting presentation. We will briefly review the main approaches. We now briefly categorize the different approaches.

#### 4.1.2 BOTTLENECK CAPACITY

Higgins et al. (2017a) propose the  $\beta$ -VAE, introducing a hyperparameter in front of the KL regularizer of vanilla VAEs. They maximize the following expression:

$$\mathbb{E}_{p(\mathbf{x})} [\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - \beta D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z}))]$$

By setting  $\beta > 1$ , the encoder distribution will be forced to better match the factorized unit Gaussian prior. This procedure introduces additional constraints on the capacity of the latent bottleneck, encouraging the encoder to learn a disentangled representation for the data. Burgess et al. (2018) argue that when the bottleneck has limited capacity, the network will be forced to specialize on the factor of variation that most contributes to a small reconstruction error. Therefore, they propose to progressively increase the bottleneck capacity, so that the encoder can focus on learning one factor of variation at the time:

$$\mathbb{E}_{p(\mathbf{x})} [\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - \gamma |D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})) - C|]$$

where  $C$  is annealed from zero to some value which is large enough to produce good reconstruction. In the following, we refer to this model as AnnealedVAE.

#### 4.1.3 PENALIZING THE TOTAL CORRELATION

Let  $I(\mathbf{x}; \mathbf{z})$  denote the mutual information between  $\mathbf{x}$  and  $\mathbf{z}$  and note that the second term in equation 1 can be rewritten as

$$\mathbb{E}_{p(\mathbf{x})}[D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))] = I(\mathbf{x}; \mathbf{z}) + D_{\text{KL}}(q(\mathbf{z})||p(\mathbf{z})).$$

Therefore, when  $\beta > 1$ ,  $\beta$ -VAE penalizes the mutual information between the latent representation and the data, thus constraining the capacity of the latent space. Furthermore, it pushes  $q(\mathbf{z})$ , the so called *aggregated posterior*, to match the prior and therefore to factorize, given a factorized prior. Kim and Mnih (2018) argues that penalizing  $I(\mathbf{x}; \mathbf{z})$  is neither necessary nor desirable for disentanglement. The FactorVAE (Kim and Mnih, 2018) and the  $\beta$ -TCVAE (Chen et al., 2018) augment the VAE objective with an additional regularizer that specifically penalizes dependencies between the dimensions of the representation:

$$\mathbb{E}_{p(\mathbf{x})}[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))] - \gamma D_{\text{KL}}(q(\mathbf{z})||\prod_{j=1}^d q(z_j)).$$

This last term is also known as *total correlation* (Watanabe, 1960). The total correlation is intractable and vanilla Monte Carlo approximations require marginalization over the training set. (Kim and Mnih, 2018) propose an estimate using the density ratio trick (Nguyen et al., 2010; Sugiyama et al., 2012) (FactorVAE). Samples from  $\prod_{j=1}^d q(z_j)$  can be obtained shuffling samples from  $q(\mathbf{z})$  (Arcones and Gine, 1992). Concurrently, Chen et al. (2018) propose a tractable biased Monte-Carlo estimate for the total correlation ( $\beta$ -TCVAE).

#### 4.1.4 DISENTANGLED PRIORS

Kumar et al. (2018) argue that a disentangled generative model requires a disentangled prior. This approach is related to the total correlation penalty, but now the aggregated posterior is pushed to match a factorized prior. Therefore

$$\mathbb{E}_{p(\mathbf{x})}[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))] - \lambda D(q(\mathbf{z})||p(\mathbf{z})),$$

where  $D$  is some (arbitrary) divergence. Since this term is intractable when  $D$  is the KL divergence, they propose to match the moments of these distribution. In particular, they regularize the deviation of either  $\text{Cov}_{p(\mathbf{x})}[\mu_\phi(\mathbf{x})]$  or  $\text{Cov}_{q_\phi}[\mathbf{z}]$  from the identity matrix in the two variants of the DIP-VAE. This results in maximizing either the DIP-VAE-I objective

$$\begin{aligned} \mathbb{E}_{p(\mathbf{x})}[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))] - \lambda_{od} \sum_{i \neq j} [\text{Cov}_{p(\mathbf{x})}[\mu_\phi(\mathbf{x})]]_{ij}^2 \\ - \lambda_d \sum_i \left( [\text{Cov}_{p(\mathbf{x})}[\mu_\phi(\mathbf{x})]]_{ii} - 1 \right)^2 \end{aligned}$$

or the DIP-VAE-II objective

$$\begin{aligned} \mathbb{E}_{p(\mathbf{x})}[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))] - \lambda_{od} \sum_{i \neq j} [\text{Cov}_{q_\phi}[\mathbf{z}]]_{ij}^2 \\ - \lambda_d \sum_i \left( [\text{Cov}_{q_\phi}[\mathbf{z}]]_{ii} - 1 \right)^2. \end{aligned}$$

## 4.2 Considered Metrics

The *BetaVAE* metric (Higgins et al., 2017a) measures disentanglement as the accuracy of a linear classifier that predicts the index of a fixed factor of variation. Kim and Mnih (2018) address several issues with this metric in their *FactorVAE* metric by using a majority vote classifier on a different feature vector which accounts for a corner case in the BetaVAE metric. The *Mutual Information Gap (MIG)* (Chen et al., 2018) measures for each factor of variation the normalized gap in mutual information between the highest and second highest coordinate in  $r(\mathbf{x})$ . Instead, the *Modularity* (Ridgeway and Mozer, 2018) measures if each dimension of  $r(\mathbf{x})$  depends on at most a factor of variation using their mutual information. The metrics of Eastwood and Williams (2018) compute the entropy of the distribution obtained by normalizing the importance of each dimension of the learned representation for predicting the value of a factor of variation. Their disentanglement score (which we call *DCI Disentanglement* for clarity) penalizes multiple factors of variation being captured by the same code and their completeness score (which we call *DCI Completeness*) penalizes a factor of variation being captured by multiple codes. The *SAP score* (Kumar et al., 2018) is the average difference of the prediction error of the two most predictive latent dimensions for each factor. The *Interventional Robustness Score (IRS)* (Suter et al., 2019) measures whether the representation is robustly disentangled by performing interventions on the factors of variations and measuring deviations in the latent space. Finally, we note that MIG, DCI Disentanglement, Modularity and SAP scores all involves the estimation of a matrix relating the factors of variation to the latent codes. Then, this matrix is aggregated into a score following some different disentanglement notion. In order to understand the role of each of these two steps we separate them and consider blends of these scores. For example, we compute the mutual information matrix as in the MIG or Modularity but compute the score using the DCI Disentanglement aggregation. We call this score *MIG-DCI Disentanglement*. In our experiments, we consider all possible pairs of matrix and aggregation.

All our metrics consider the expected representation of training samples (except total correlation for which we also consider the sampled representation as described in Section 5).

### 4.2.1 BETAVAE METRIC

Higgins et al. (2017a) suggest to fix a random factor of variation in the underlying generative model and to sample two mini batches of observations  $\mathbf{x}$ . Disentanglement is then measured as the accuracy of a linear classifier that predicts the index of the fixed factor based on the coordinate-wise sum of absolute differences between the representation vectors in the two mini batches. We sample two batches of 64 points with a random factor fixed to a randomly sampled value across the two batches and the others varying randomly. We compute the mean representations for these points and take the absolute difference between pairs from the two batches. We then average these 64 values to form the features of a training (or testing) point. We train a Scikit-learn logistic regression with default parameters on 10 000 points. We test on 5000 points.

#### 4.2.2 FACTORVAE METRIC

Kim and Mnih (2018) address several issues with this metric by using a majority vote classifier that predicts the index of the fixed ground-truth factor based on the index of the representation vector with the least variance. First, we estimate the variance of each latent dimension by embedding 10 000 random samples from the data set and we exclude collapsed dimensions with variance smaller than 0.05. Second, we generate the votes for the majority vote classifier by sampling a batch of 64 points, all with a factor fixed to the same random value. Third, we compute the variance of each dimension of their latent representation and divide by the variance of that dimension we computed on the data without interventions. The training point for the majority vote classifier consists of the index of the dimension with the smallest normalized variance. We train on 10 000 points and evaluate on 5000 points.

#### 4.2.3 MUTUAL INFORMATION GAP

Chen et al. (2018) argue that the BetaVAE metric and the FactorVAE metric are neither general nor unbiased as they depend on some hyperparameters. They compute the mutual information between each ground truth factor and each dimension in the computed representation  $r(\mathbf{x})$ . For each ground-truth factor  $z_k$ , they then consider the two dimensions in  $r(\mathbf{x})$  that have the highest and second highest mutual information with  $z_k$ . The *Mutual Information Gap (MIG)* is then defined as the average, normalized difference between the highest and second highest mutual information of each factor with the dimensions of the representation. The original metric was proposed evaluating the sampled representation. Instead, we consider the mean representation, in order to be consistent with the other metrics. We estimate the discrete mutual information by binning each dimension of the representations obtained from 10 000 points into 20 bins. Then, the score is computed as follows:

$$\frac{1}{K} \sum_{k=1}^K \frac{1}{H_{z_k}} \left( I(v_{j_k}, z_k) - \max_{j \neq j_k} I(v_j, z_k) \right),$$

where  $z_k$  is a factor of variation,  $v_j$  is a dimension of the latent representation,  $H_{z_k}$  is the entropy of  $z_k$  (using again 20 bins), and  $j_k = \arg \max_j I(v_j, z_k)$ .

#### 4.2.4 MODULARITY

Ridgeway and Mozer (2018) argue that two different properties of representations should be considered: *Modularity* and *Explicitness*. In a modular representation each dimension of  $r(\mathbf{x})$  depends on at most a single factor of variation. In an explicit representation, the value of a factor of variation is easily predictable (for example with a linear model) from  $r(\mathbf{x})$ . They propose to measure the Modularity as the average normalized squared difference of the mutual information of the factor of variations with the highest and second-highest mutual information with a dimension of  $r(\mathbf{x})$ . They measure Explicitness as the ROC-AUC of a one-versus-rest logistic regression classifier trained to predict the factors of variation. In this study, we focus on Modularity as it is the property that corresponds to disentanglement. For the modularity score, we sample 10 000 points for which we obtain the latent representations. We discretize these points into 20 bins and compute the mutual information between representations and the values of the factors of variation. These values are



stored in a matrix  $\mathbf{m}$ . For each dimension of the representation  $i$ , we compute a vector  $\mathbf{t}_i$  as:

$$t_{i,f} = \begin{cases} \theta_i & \text{if } f = \arg \max_g m_{i,g} \\ 0 & \text{otherwise} \end{cases}$$

where  $\theta_i = \max_g m_{i,g}$ . The modularity score is the average over the dimensions of the representation of  $1 - \delta_i$  where:

$$\delta_i = \frac{\sum_f (m_{if} - t_{if})^2}{\theta_i^2 (N - 1)}$$

and  $N$  is the number of factors.

#### 4.2.5 DCI DISENTANGLEMENT

Eastwood and Williams (2018) consider three properties of representations: *Disentanglement*, *Completeness* and *Informativeness*. First, Eastwood and Williams (2018) compute the importance of each dimension of the learned representation for predicting a factor of variation. The predictive importance of the dimensions of  $r(\mathbf{x})$  can be computed with a Lasso or a Random Forest classifier. Disentanglement is the average of the difference from one of the entropy of the probability that a dimension of the learned representation is useful for predicting a factor weighted by the relative importance of each dimension. Completeness, is the average of the difference from one of the entropy of the probability that a factor of variation is captured by a dimension of the learned representation. Finally, the Informativeness can be computed as the prediction error of predicting the factors of variations. We sample 10 000 and 5000 training and test points respectively. For each factor, we fit gradient boosted trees from Scikit-learn with the default setting. From this model, we extract the importance weights for the feature dimensions. We take the absolute value of these weights and use them to form the importance matrix  $R$ , whose rows correspond to factors and columns to the representation. To compute the disentanglement score, we first subtract from 1 the entropy of each column of this matrix (we treat the columns as a distribution by normalizing them). This gives a vector of length equal to the dimensionality of the latent space. Then, we compute the relative importance of each dimension by  $\rho_i = \sum_j R_{ij} / \sum_{ij} R_{ij}$  and the disentanglement score as  $\sum_i \rho_i (1 - H(R_i))$ .

#### 4.2.6 SAP SCORE

Kumar et al. (2018) propose to compute the  $R^2$  score of the linear regression predicting the factor values from each dimension of the learned representation. For discrete factors, they propose to train a classifier. The *Separated Attribute Predictability (SAP)* score is the average difference of the prediction error of the two most predictive latent dimensions for each factor. We sample 10 000 points for training and 5000 for testing. We then compute a score matrix containing the prediction error on the test set for a linear SVM with  $C = 0.01$  predicting the value of a factor from a single latent dimension. The SAP score is computed as the average across factors of the difference between the top two most predictive latent dimensions.

#### 4.2.7 INTERVENTIONAL ROBUSTNESS SCORE

Suter et al. (2019) introduce a causality perspective and measure the robustness of a representation after interventions on the factors of variation: the *Interventional Robustness Score (IRS)*. For two



factors of variation  $\mathbf{z}_i$  and  $\mathbf{z}_j$ , they define the *post interventional disagreement* as the distance between the representation with an intervention on  $\mathbf{z}_i$  and on both  $\mathbf{z}_i$  and  $\mathbf{z}_j$ . Then, they take the supremum of this distance with respect to the values of  $\mathbf{z}_j$  and average with respect to the distribution of  $\mathbf{z}_i$ . This value is normalized by the maximum post interventional disagreement with no fixed  $\mathbf{z}_i$  and subtracted from 1. This score measure essentially how well  $\mathbf{z}_i$  is robustly disentangled from  $\mathbf{z}_j$ . The disentanglement of  $\mathbf{z}_i$  can be computed by taking its maximum disagreement with all other factors of variation for each dimension of the representation.

#### 4.2.8 DOWNSTREAM TASK

We sample training sets of different sizes: 10, 100, 1000 and 10 000 points. We always evaluate on 5000 samples. We consider as a downstream task the prediction of the values of each factor from  $r(\mathbf{x})$ . For each factor we fit a different model and report then report the average test accuracy across factors. We consider two different models. First, we train a cross validated logistic regression from Scikit-learn with 10 different values for the regularization strength ( $Cs = 10$ ) and 5 folds. Finally, we train a gradient boosting classifier from Scikit-learn with default parameters.

#### 4.2.9 TOTAL CORRELATION BASED ON FITTED GAUSSIAN

We sample 10 000 points and obtain their latent representation  $r(\mathbf{x})$  by either sampling from the encoder distribution or by taking its mean. We then compute the mean  $\mu_{r(\mathbf{x})}$  and covariance matrix  $\Sigma_{r(\mathbf{x})}$  of these points and compute the total correlation of a Gaussian with mean  $\mu_{r(\mathbf{x})}$  and covariance matrix  $\Sigma_{r(\mathbf{x})}$ :

$$D_{\text{KL}} \left( \mathcal{N}(\mu_{r(\mathbf{x})}, \Sigma_{r(\mathbf{x})}) \parallel \prod_j \mathcal{N}(\mu_{r(\mathbf{x})_j}, \Sigma_{r(\mathbf{x})_{jj}}) \right),$$

where  $j$  indexes the dimensions in the latent space. We choose this approach for the following reasons. In this study, we compute statistics of  $r(\mathbf{x})$  which can be either sampled from the probabilistic encoder or taken to be its mean. We argue that estimating the total correlation as in (Kim and Mnih, 2018) is not suitable for this comparison as it consistently underestimates the true value (see Figure 7 in (Kim and Mnih, 2018)) and depends on a non-convex optimization procedure (for fitting the discriminator). The estimate of (Chen et al., 2018) is also not suitable as the mean representation is a deterministic function for the data, therefore we cannot use the encoder distribution for the estimate. Furthermore, we argue that the total correlation based on the fitted Gaussian provides a simple and robust way to detect if a representation is not factorizing based on the first two moments. In particular, if it is high, it is a strong signal that the representation is not factorizing (while a low score may not imply the opposite). We note that this procedure is similar to the penalty of DIP-VAE-I.

### 4.3 Data Sets

We consider five data sets in which  $\mathbf{x}$  is obtained as a deterministic function of  $\mathbf{z}$ : *dSprites* (Higgins et al., 2017a), *Cars3D* (Reed et al., 2015), *SmallNORB* (LeCun et al., 2004), *Shapes3D* (Kim and Mnih, 2018) and *MPI3D* (Gondal et al., 2019). We also introduce three data sets where the observations  $\mathbf{x}$  are stochastic given the factor of variations  $\mathbf{z}$ : *Color-dSprites*, *Noisy-dSprites* and *Scream-dSprites*. In *Color-dSprites*, the shapes are colored with a random color. In *Noisy-dSprites*, we consider white-colored shapes on a noisy background. Finally, in *Scream-dSprites* the background

is replaced with a random patch in a random color shade extracted from the famous *The Scream* painting (Munch, 1893). The dSprites shape is embedded into the image by inverting the color of its pixels. Further details on the preprocessing of the data can be found in Appendix F.

#### 4.4 Inductive Biases

To fairly evaluate the different approaches, we separate the effect of regularization (in the form of model choice and regularization strength) from the other inductive biases (for example, the choice of the neural architecture). Each method uses the same convolutional architecture, optimizer, hyperparameters of the optimizer and batch size. All methods use a Gaussian encoder where the mean and the log variance of each latent factor is parametrized by the deep neural network, a Bernoulli decoder and latent dimension fixed to 10. We note that these are all standard choices in prior work (Higgins et al., 2017a; Kim and Mnih, 2018).

We choose six different regularization strengths, that is, hyperparameter values, for each of the considered methods. The key idea was to take a wide enough set to ensure that there are useful hyperparameters for different settings for each method and not to focus on specific values known to work for specific data sets. However, the values are partially based on the ranges that are prescribed in the literature (including the hyperparameters suggested by the authors).

We fix our experimental setup in advance and we run all the considered methods on each data set for 50 different random seeds and evaluate them on the considered metrics. The full details on the experimental setup are provided in the Appendix E. Our experimental setup, the limitations of this study, and the differences with previous implementations are extensively discussed in Appendices B-D.

### 5. Can We Learn Disentangled Representations Without Supervision?

In this section, we provide a sober look at the performances of state-of-the-art approaches and investigate how effectively we can learn disentangled representations without looking at the labels. We focus our analysis on key questions for practitioners interested in learning disentangled representations reliably and without supervision.

#### 5.1 Can One Achieve a Good Reconstruction Error Across Data Sets and Models?

First, we check for each data set that we manage to train models that achieve reasonable reconstructions. Therefore, for each data set we sample a random model and show real samples next to their reconstructions. The results are depicted in Figure 1. As expected, the additional variants of dSprites with continuous noise variables are harder than the original data set. On Noisy-dSprites and Color-dSprites the models produce reasonable reconstructions with the noise on Noisy-dSprites being ignored. Scream-dSprites is even harder and we observe that the shape information is lost. On the other data sets, we observe that reconstructions are blurry but objects are distinguishable. Since in MPI3D the objects are small, their shape appear sometime difficult to distinguish. The other factors of variation however are clearly captured. SmallNORB seems to be the most challenging data set.

#### 5.2 Can Current Methods Enforce a Uncorrelated Aggregated Posterior and Representation?

We investigate whether the considered unsupervised disentanglement approaches are effective at enforcing a factorizing and thus uncorrelated aggregated posterior. For each trained model, we

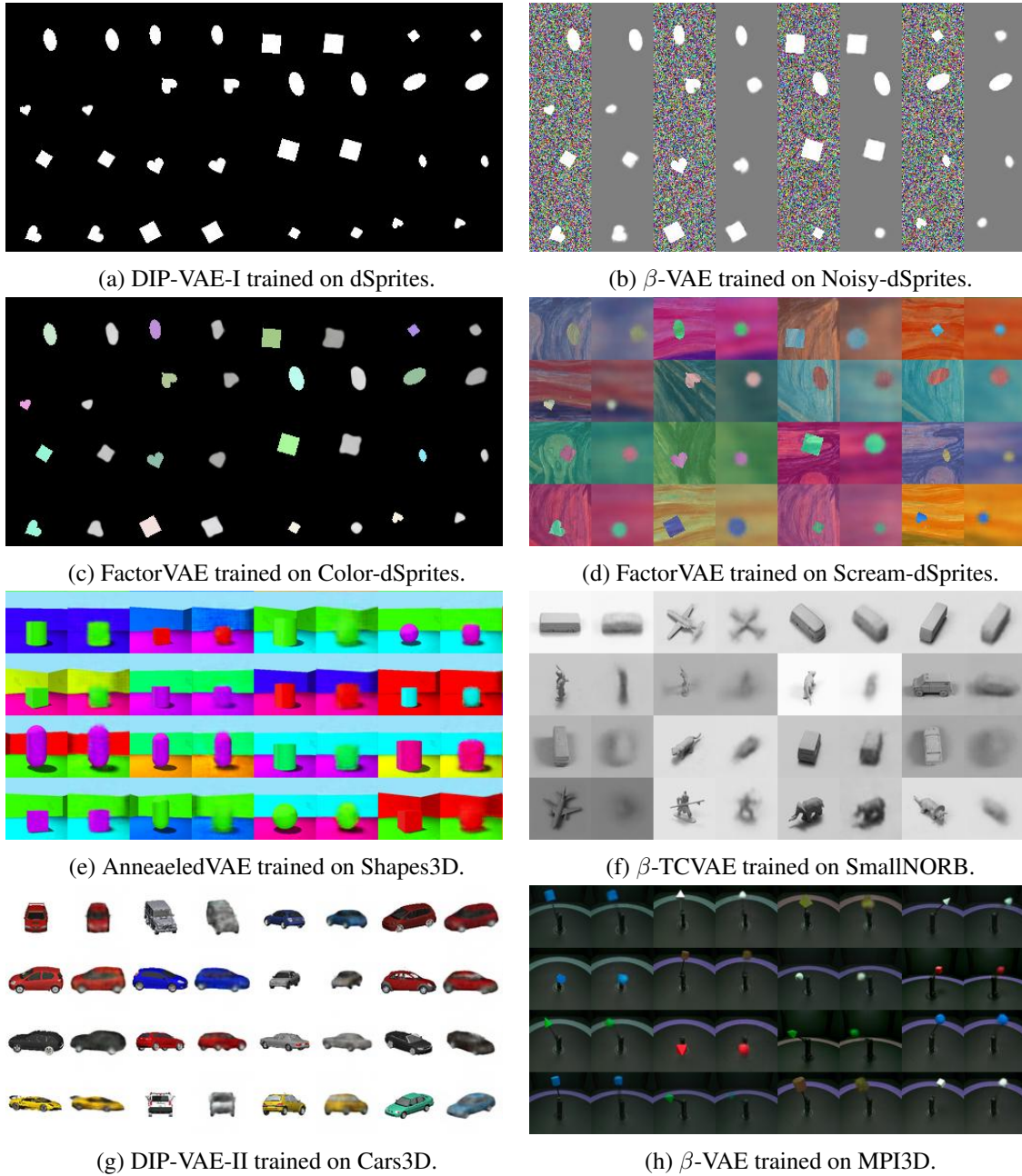


Figure 1: Reconstructions for different data sets and methods. Odd columns show real samples and even columns their reconstruction. As expected, the additional variants of dSprites with continuous noise variables are harder than the original data set. On Noisy-dSprites and Color-dSprites the models produce reasonable reconstructions with the noise on Noisy-dSprites being ignored. Scream-dSprites is even harder and we observe that the shape information is lost. On the other data sets, we observe that reconstructions are blurry but objects are distinguishable. The MPI3D Dataset consists of real images of a robotic arm.

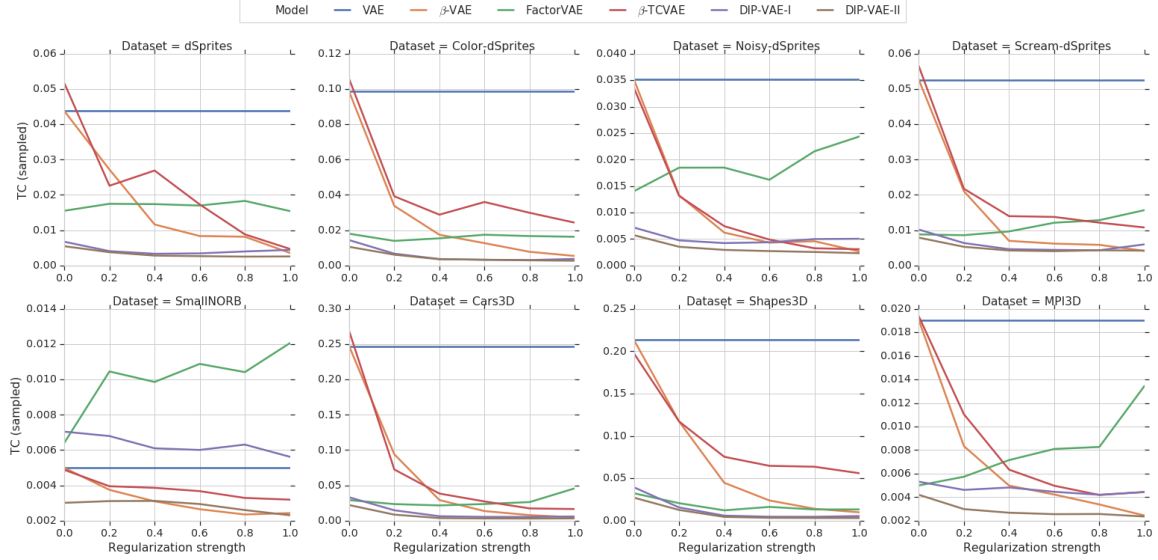


Figure 2: Total correlation of sampled representation plotted against regularization strength for different data sets and approaches (except AnnealedVAE). The total correlation of the sampled representation decreases as the regularization strength is increased.

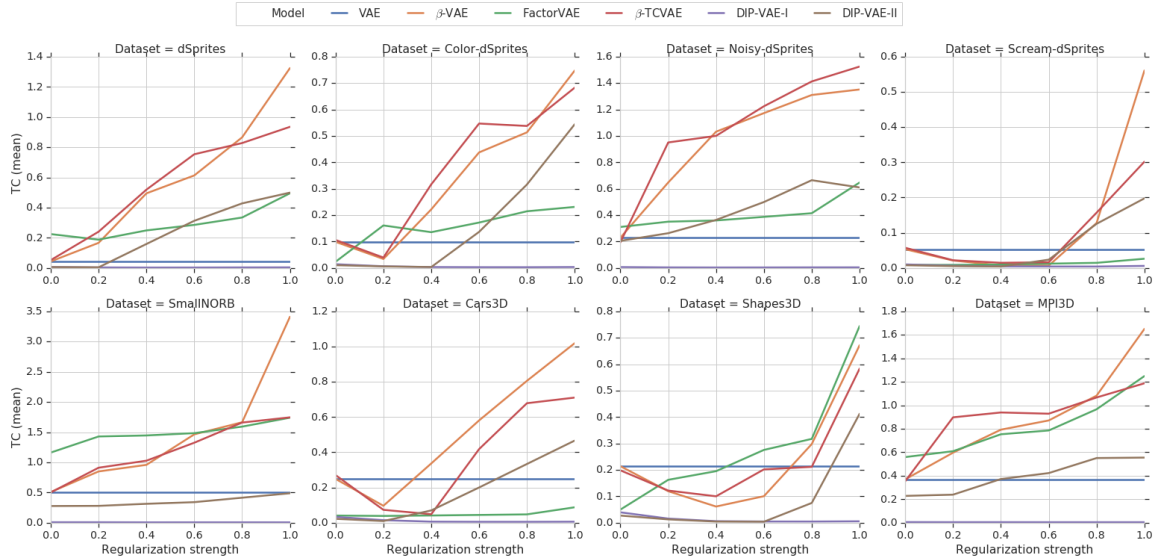


Figure 3: Total correlation of mean representation plotted against regularization strength for different data sets and approaches (except AnnealedVAE). The total correlation of the mean representation does not necessarily decrease as the regularization strength is increased.

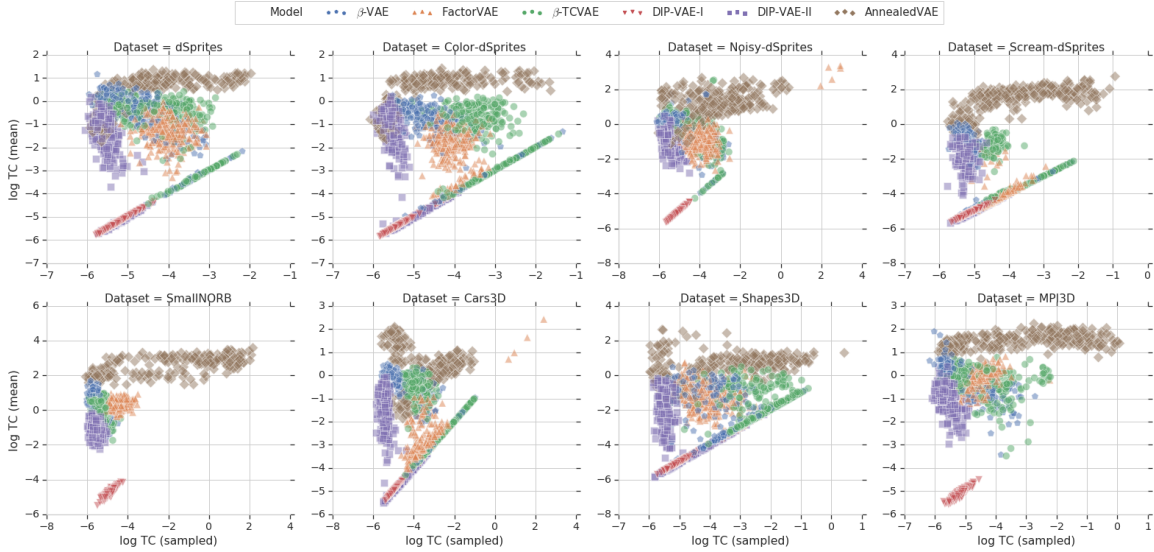


Figure 4: Log total correlation of mean vs sampled representations. For a large number of models, the total correlation of the mean representation is higher than that of the sampled representation.

sample 10 000 images and compute a sample from the corresponding approximate posterior. We then fit a multivariate Gaussian distribution over these 10 000 samples by computing the empirical mean and covariance matrix. Finally, we compute the total correlation of the fitted Gaussian and report the median value for each data set, method and hyperparameter value.

Figure 2 shows the total correlation of the sampled representation plotted against the regularization strength for each data set and method except AnnealedVAE. On all data sets except SmallNORB, we observe that plain vanilla variational autoencoders (the  $\beta$ -VAE model with  $\beta = 1$ ) exhibit the highest total correlation. For  $\beta$ -VAE and  $\beta$ -TCVAE, it can be clearly seen that the total correlation of the sampled representation decreases on all data sets as the regularization strength (in the form of  $\beta$ ) is increased. The two variants of DIP-VAE exhibit low total correlation across the data sets except DIP-VAE-I which incurs a slightly higher total correlation on SmallNORB compared to a vanilla VAE. Increased regularization in the DIP-VAE objective also seems to lead a reduced total correlation, even if the effect is not as pronounced as for  $\beta$ -VAE and  $\beta$ -TCVAE. While FactorVAE achieves a low total correlation on all data sets except on SmallNORB, we observe that the total correlation does not seem to decrease with increasing regularization strength. We further observe that AnnealedVAE (shown in Figure 29 in the Appendix) is much more sensitive to the regularization strength. However, on all data sets except Scream-dSprites (on which AnnealedVAE performs poorly), the total correlation seems to decrease with increased regularization strength.

While many of the considered methods aim to enforce a factorizing aggregated posterior, they use the mean vector of the Gaussian encoder as the representation and not a sample from the Gaussian encoder. This may seem like a minor, irrelevant modification; however, it is not clear whether a factorizing aggregated posterior also ensures that the dimensions of the mean representation are uncorrelated. To test whether this is true, we compute the mean of the Gaussian encoder for the same 10 000 samples, fit a multivariate Gaussian and compute the total correlation of that fitted Gaussian. Figure 3 shows the total correlation of the mean representation plotted against the regularization



strength for each data set and method except AnnealedVAE. We observe that, for  $\beta$ -VAE and  $\beta$ -TCVAE, increased regularization leads to a substantially increased total correlation of the mean representations. This effect can also be observed for FactorVAE, albeit in a less extreme fashion. For DIP-VAE-I, we observe that the total correlation of the mean representation is consistently low. This is not surprising as the DIP-VAE-I objective directly optimizes the covariance matrix of the mean representation to be diagonal which implies that the corresponding total correlation (as we compute it) is low. The DIP-VAE-II objective which enforces the covariance matrix of the sampled representation to be diagonal seems to lead to a factorized mean representation on some data sets (for example Shapes3D), but also seems to fail on others (dSprites, MPI3D). For AnnealedVAE (shown in Figure 30 in the Appendix), we overall observe mean representations with a very high total correlation. In Figure 4, we further plot the log total correlations of the sampled representations versus the mean representations for each of the trained models. It can be clearly seen that for a large number of models, the total correlation of the mean representations is much higher than that of the sampled representations. The same trend can be seen computing the average discrete mutual information of the representation. In this case, the DIP-VAE-I exhibit increasing mutual information in both the mean and sampled representation. This is to be expected as DIP-VAE-I enforces a variance of one for the mean representation. We remark that as the regularization terms and hyperparameter values are different for different losses, one should not draw conclusions from comparing different models at nominally the same regularization strength. From these plots one can only compare the effect of increasing the regularization in the different models.

### 5.2.1 IMPLICATIONS

Overall, these results lead us to conclude with minor exceptions that the considered methods are effective at enforcing an aggregated posterior whose individual dimensions are not correlated but that this does not seem to imply that the dimensions of the mean representation (usually used for representation) are uncorrelated.

## 5.3 How Important Are Different Models and Hyperparameters for Disentanglement?

The primary motivation behind the considered methods is that they should lead to improved disentanglement scores. This raises the question how disentanglement is affected by the model choice, the hyperparameter selection and randomness (in the form of different random seeds). To investigate this, we compute all the considered disentanglement metrics for each of our trained models. In Figure 5, we show the range of attainable disentanglement scores for each method on each data set varying the regularization strength and the random seed. We observe that these ranges are heavily overlapping for different models leading us to (qualitatively) conclude that the choice of hyperparameters and the random seed seems to be substantially more important than the choice of objective function. While certain models seem to attain better maximum scores on specific data sets and disentanglement metrics, we do not observe any consistent pattern that one model is consistently better than the other. DIP-VAE-I consistently gets lower IRS score, but is comparable to the other methods with all the other scores. Furthermore, we note that in our study we have fixed the range of hyperparameters a priori to six different values for each model and did not explore additional hyperparameters based on the results (as that would bias our study). However, this also means that specific models may have performed better than in Figure 5 if we had chosen a different set of hyperparameters.

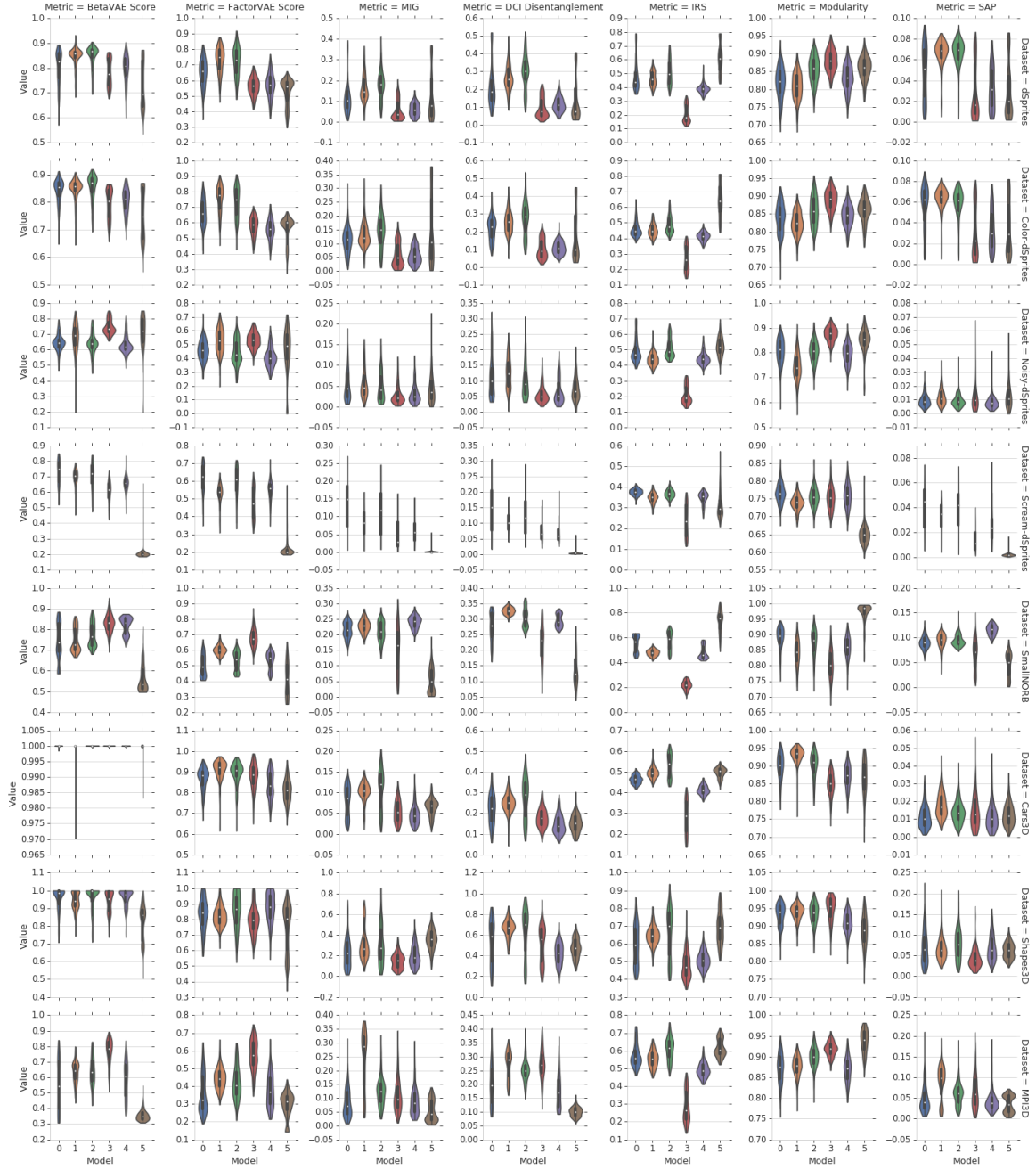


Figure 5: Score for each method for each score (column) and data set (row) with different hyperparameters and random seed. Models are abbreviated (0= $\beta$ -VAE, 1=FactorVAE, 2= $\beta$ -TCVAE, 3=DIP-VAE-I, 4=DIP-VAE-II, 5=AnnealedVAE). The scores are heavily overlapping and we do not observe a consistent pattern. We conclude that hyperparameters and random seed matter more than the model choice.



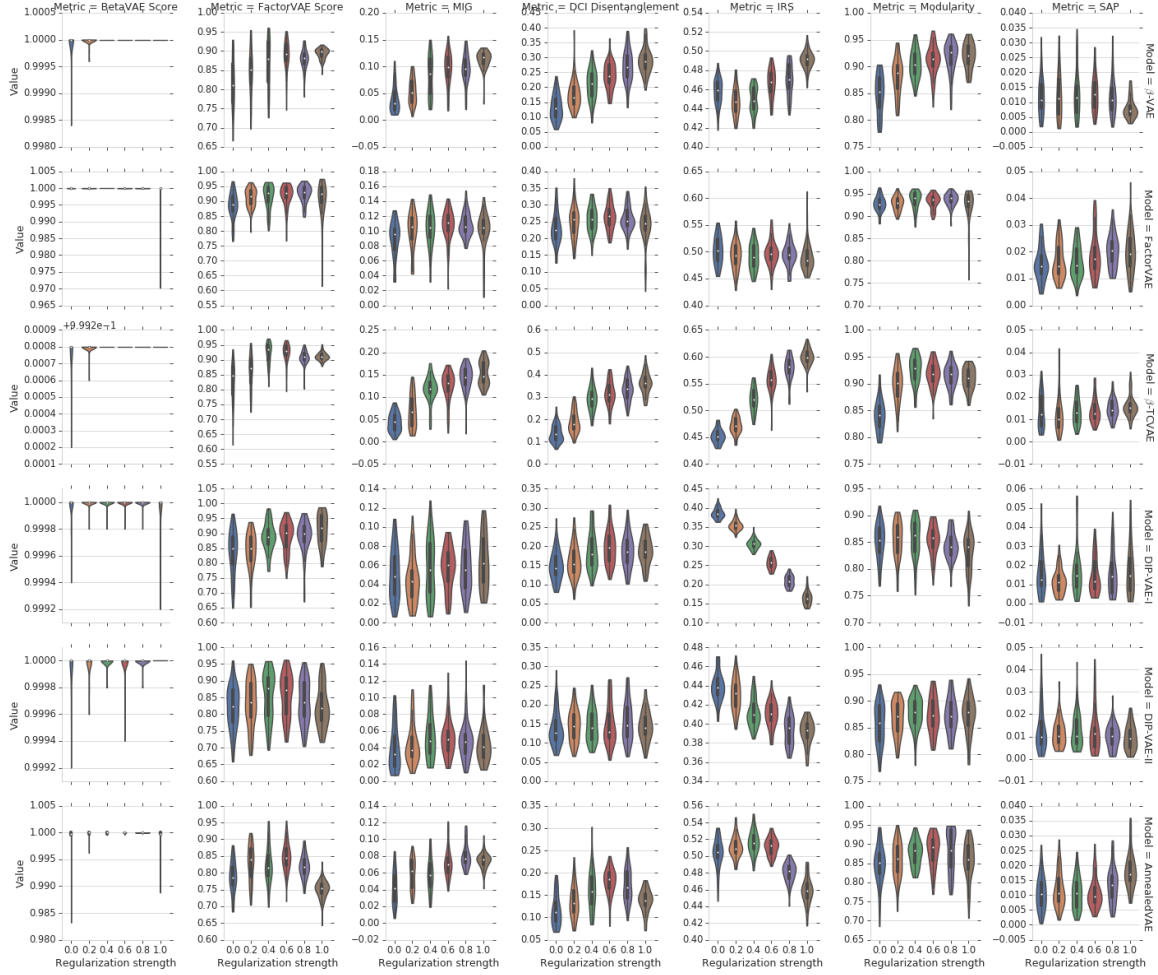


Figure 6: Distribution of scores for different models, hyperparameters and regularization strengths on Cars3D. We clearly see that randomness (in the form of different random seeds) has a substantial impact on the attained result and that a good run with a bad hyperparameter can beat a bad run with a good hyperparameter in many cases. IRS seem to be an exception on some data sets.

In Figure 6, we further show the impact of randomness in the form of random seeds on the disentanglement scores. Each violin plot shows the distribution of the disentanglement metric across all 50 trained models for each model and hyperparameter setting on Cars3D. We clearly see that randomness (in the form of different random seeds) has a substantial impact on the attained result and that a good run with a bad hyperparameter can beat a bad run with a good hyperparameter in many cases. We note that IRS seem to exhibit a clear trend on some data sets.

Finally, we perform a variance analysis by trying to predict the different disentanglement scores using ordinary least squares for each data set: If we allow the score to depend only on the objective function (categorical variable), we are only able to explain 37% of the variance of the scores on average. Similarly, if the score depends on the Cartesian product of objective function and regularization strength (again categorical), we are able to explain 59% of the variance while the rest is due to the random seed. In Table 5 in the Appendix, we report the percentage of variance explained for the different metrics in each data set considering the regularization strength or not.

### 5.3.1 IMPLICATIONS

The disentanglement scores of unsupervised models are heavily influenced by randomness (in the form of the random seed) and the choice of the hyperparameter (in the form of the regularization strength). The objective function appears to have less impact.

## 5.4 Are There Reliable Recipes for Model Selection?

In this section, we investigate how good hyperparameters can be chosen and how we can distinguish between good and bad training runs. In this paper, we advocate that model selection *should not* depend on the considered disentanglement score for the following reasons: The point of unsupervised learning of disentangled representation is that there is no access to the labels as otherwise we could incorporate them and would have to compare to semi-supervised and fully supervised methods. All the disentanglement metrics considered in this paper require a substantial amount of ground-truth labels or the full generative model (for example for the BetaVAE and the FactorVAE metric). Hence, one may substantially bias the results of a study by tuning hyperparameters based on (supervised) disentanglement metrics. Furthermore, we argue that it is not sufficient to fix a set of hyperparameters *a priori* and then show that one of those hyperparameters and a specific random seed achieves a good disentanglement score as it amounts to showing the existence of a good model, but does not guide the practitioner in finding it. Finally, in many practical settings, we might not even have access to adequate labels as it may be hard to identify the true underlying factor of variations, in particular, if we consider data modalities that are less suitable to human interpretation than images.

In the remainder of this section, we hence investigate and assess different ways how hyperparameters and good model runs could be chosen. In this study, we focus on choosing the learning model and the regularization strength corresponding to that loss function. However, we note that in practice this problem is likely even harder as a practitioner might also want to tune other modeling choices such architecture or optimizer.

### 5.4.1 GENERAL RECIPES FOR HYPERPARAMETER SELECTION

We first investigate whether we may find generally applicable “rules of thumb” for choosing the hyperparameters. For this, we plot in Figure 7 different disentanglement metrics against different regularization strengths for each model and each data set. The values correspond to the median

	Random different data set	Same data set
Random different metric	52.7%	62.1%
Same metric	59.6%	81.9%

Table 1: Probability of outperforming random model selection on a different random seed. A random disentanglement metric and data set is sampled and used for model selection. That model is then compared to a randomly selected model: (i) on the same metric and data set, (ii) on the same metric and a random different data set, (iii) on a random different metric and the same data set, and (iv) on a random different metric and a random different data set. The results are averaged across 10 000 random draws.

obtained values across 50 random seeds for each model, hyperparameter and data set. There seems to be no model dominating all the others and for each model there does not seem to be a consistent strategy in choosing the regularization strength to maximize disentanglement scores. Furthermore, even if we could identify a good objective function and corresponding hyperparameter value, we still could not distinguish between a good and a bad training run.

#### 5.4.2 MODEL SELECTION BASED ON UNSUPERVISED SCORES

Another approach could be to select hyperparameters based on unsupervised scores such as the reconstruction error, the KL divergence between the prior and the approximate posterior, the Evidence Lower Bound or the estimated total correlation of the sampled representation. This would have the advantage that we could select specific trained models and not just good hyperparameter settings whose median trained model would perform well. To test whether such an approach is fruitful, we compute the rank correlation between these unsupervised metrics and the disentanglement metrics and present it in Figure 8. While we do observe some correlations, no clear pattern emerges which leads us to conclude that this approach is unlikely to be successful in practice.

#### 5.4.3 HYPERPARAMETER SELECTION BASED ON TRANSFER

The final strategy for hyperparameter selection that we consider is based on transferring good settings across data sets. The key idea is that good hyperparameter settings may be inferred on data sets where we have labels available (such as dSprites) and then applied to novel data sets. To test this idea, we plot in Figure 10 the different disentanglement scores obtained on dSprites against the scores obtained on other data sets. To ensure robustness of the results, we again consider the median across all 50 runs for each model, regularization strength, and data set. We observe that the scores on Color-dSprites seem to be strongly correlated with the scores obtained on the regular version of dSprites. Figure 9 further shows the rank correlations obtained between different data sets for each disentanglement scores. This confirms the strong and consistent correlation between dSprites and Color-dSprites. While these result suggest that some transfer of hyperparameters is possible, it does not allow us to distinguish between good and bad random seeds on the target data set.

To illustrate this, we compare such a transfer based approach to hyperparameter selection to random model selection as follows: We first randomly sample one of our 50 random seeds and consider the set of trained models with that random seed. First, we sample one of our 50 random seeds, a random disentanglement metric and a data set and use them to select the hyperparameter

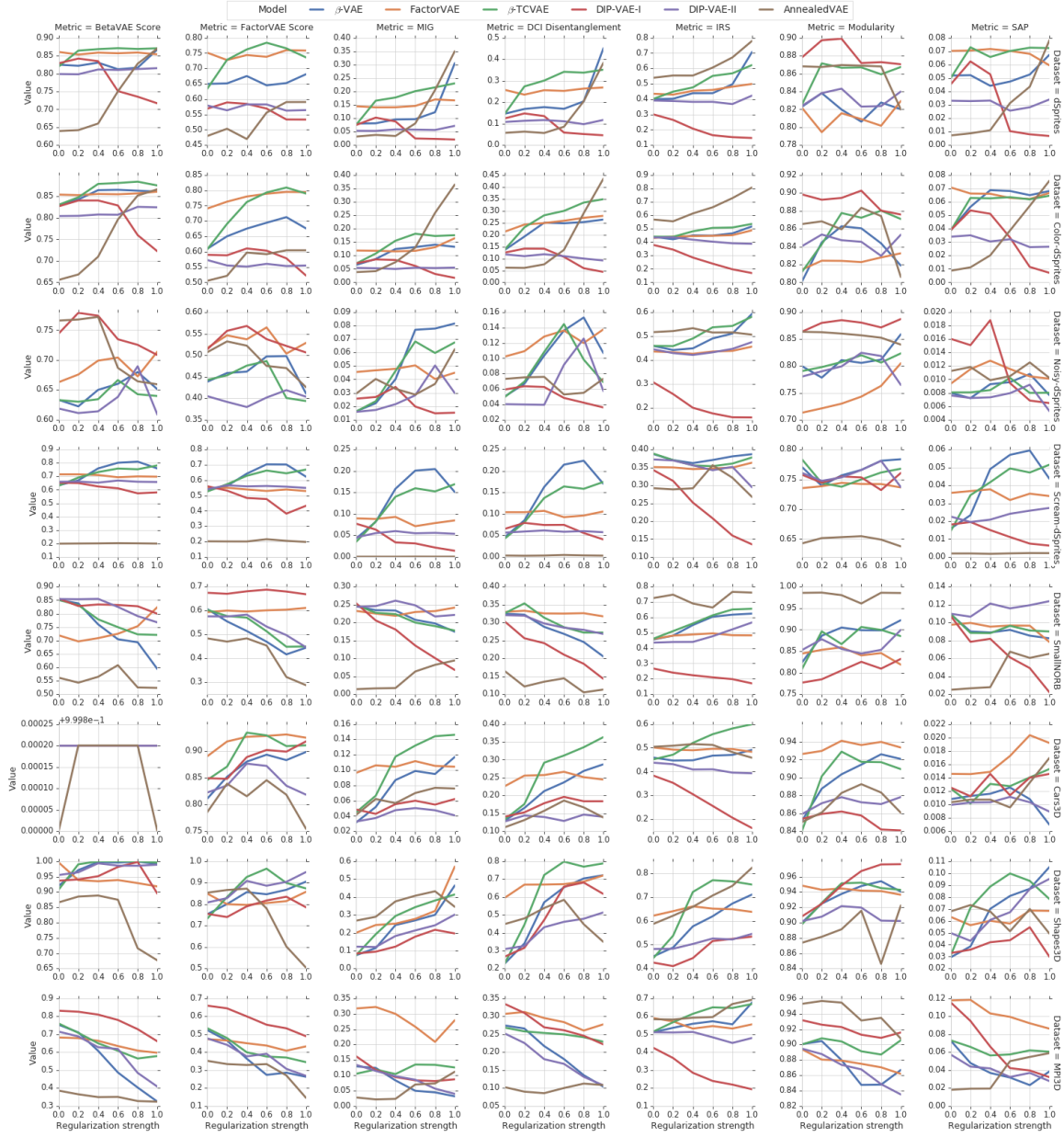


Figure 7: Score vs hyperparameters for each score (column) and data set (row). There seems to be no model dominating all the others and for each model there does not seem to be a consistent strategy in choosing the regularization strength.

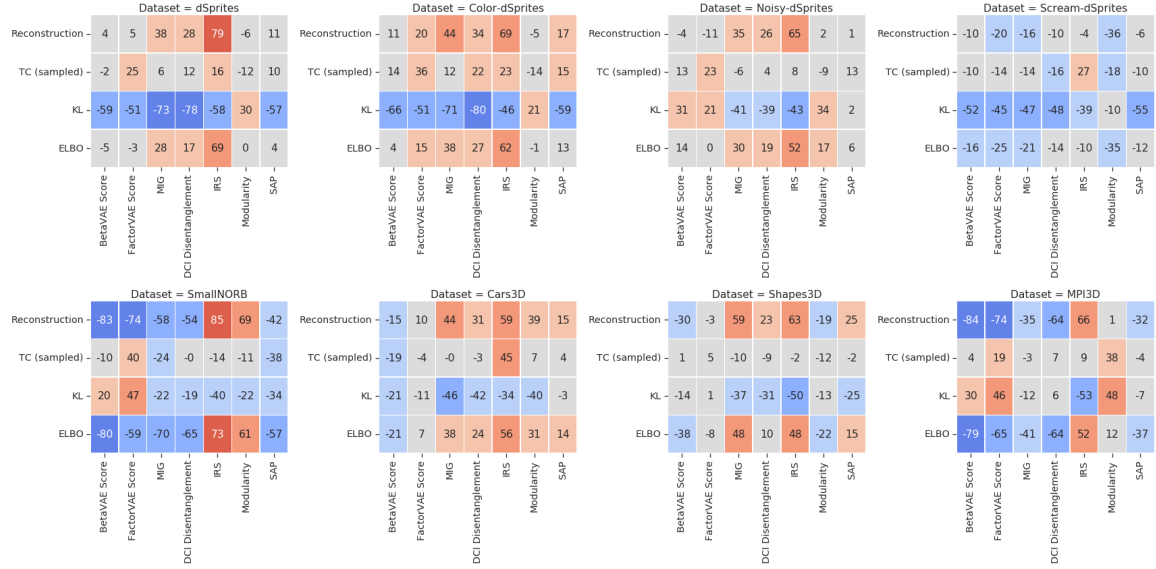


Figure 8: Rank correlation between unsupervised scores and supervised disentanglement metrics. The unsupervised scores we consider do not seem to be useful for model selection.

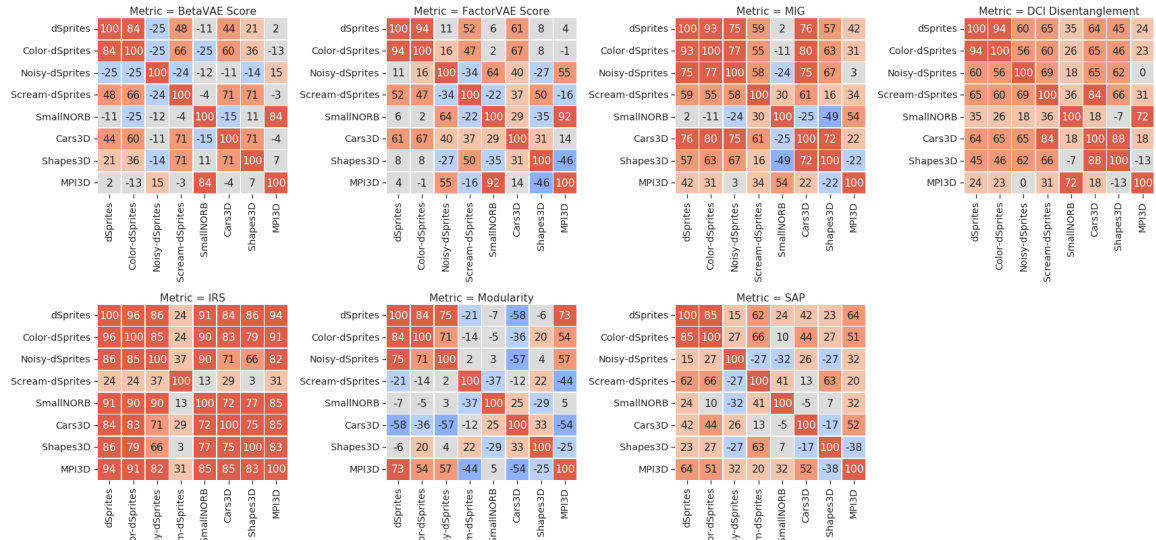


Figure 9: Rank-correlation of different disentanglement metrics across different data sets. Good hyperparameters only seem to transfer between dSprites and Color-dSprites but not in between the other data sets.

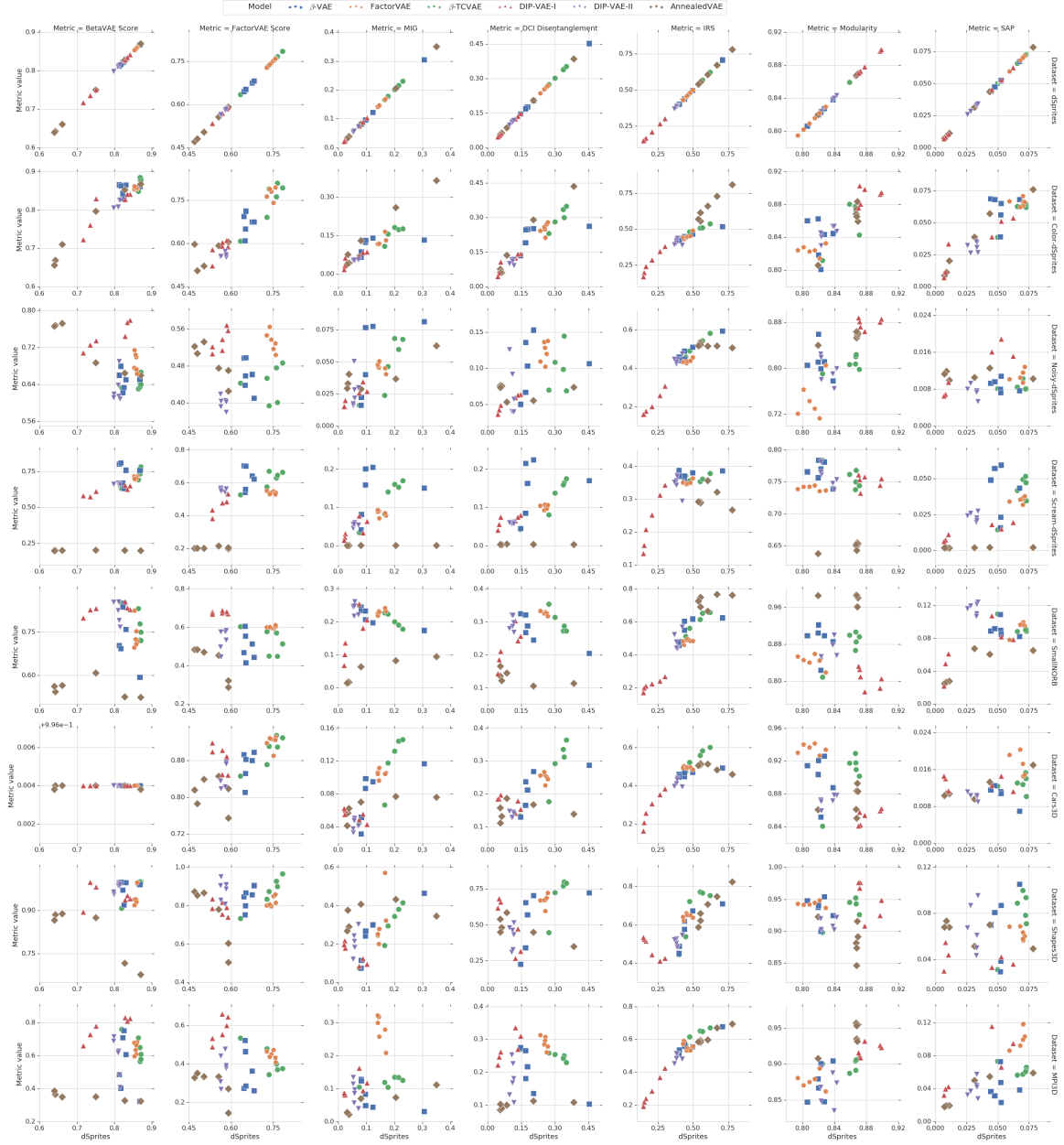


Figure 10: Disentanglement scores on dSprites vs other data sets. Good hyperparameters only seem to transfer consistently from dSprites to Color-dSprites.



setting with the highest attained score. Then, we compare that selected hyperparameter setting to a randomly selected model on either the same or a random different data set, based on either the same or a random different metric and for a randomly sampled seed. Finally, we report the percentage of trials in which this transfer strategy outperforms or performs equally well as random model selection across 10 000 trials in Table 1. If we choose the same metric and the same data set (but a different random seed), we obtain a score of 81.9%. If we aim to transfer for the same metric across data sets, we achieve around 59.6%. Finally, if we transfer both across metrics and data sets, our performance drops to 52.7%. The drop in performance transferring hyperparameters across different metrics may be interpreted in light of the results of Section 6.1.1.

#### 5.4.4 IMPLICATIONS

Unsupervised model selection remains an unsolved problem. Transfer of good hyperparameters between metrics and data sets does not seem to work as there appears to be no unsupervised way to distinguish between good and bad random seeds on the target task. Recent work (Duan et al., 2019) may be used to select stable hyperparameter configurations. The IRS score seem to be more correlated with the unsupervised training metrics on most data set and generally transfer the hyperparameters better. However, as we shall see in Section 6.1, IRS is not very correlated with the other disentanglement metrics.

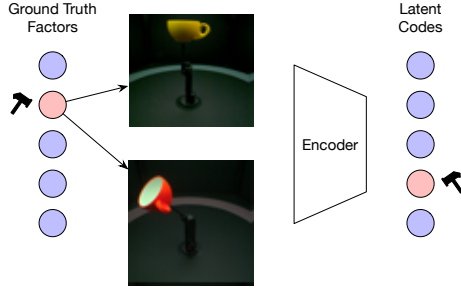
## 6. What Are the Differences Between the Disentanglement Metrics?

The disentanglement of a learned representation can be seen as a certain structural property of the statistical relations between the latent space of the VAE with that of the ground truth factors. Therefore, when evaluating disentangled representations several metrics typically estimate these statistical dependencies first and then compute how well this structure encodes the desired properties. As quantifying statistical dependencies through independence testing is a challenging task (Shah and Peters, 2018) several approaches have been proposed. We identify two prevalent settings: using interventional (Higgins et al., 2017a; Kim and Mnih, 2018; Suter et al., 2019) and observational data (Chen et al., 2018; Ridgeway and Mozer, 2018; Eastwood and Williams, 2018).

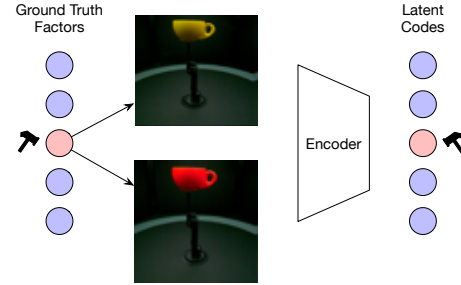
For interventional data, the two main properties a disentangled representation should have are *consistency* and *restrictiveness* (Shu et al., 2020). Examples can be seen in Figures 11a and 11b. Both can be interpreted in the context of independent mechanisms (Peters et al., 2017): interventions on a ground-truth factor should manifest in a localized way in the representation. For example, fixing a certain factor of variation and sampling twice all others should result in a subset of dimensions being constant in the representation of the two points (consistency). This notion is used in the metrics of Higgins et al. (2017a); Kim and Mnih (2018). On the other hand, changing the value of a factor of variation while keeping the others constant should result in a single change in the representation. This fact was used in the evaluation metric proposed by Suter et al. (2019). While (Shu et al., 2020) argue that both aspects are necessary for disentangled representations, when the ground-truth factors are independent and unconfounded the two definitions are equivalent.

On the observational data, which is arguably the most practical case, there are several ways of estimating the relationship between factors and codes. For example, Chen et al. (2018); Ridgeway and Mozer (2018) use the mutual information while Eastwood and Williams (2018); Kumar et al. (2018) rely on predictability with a random forest classifier and a SVM respectively. The practical impact of these low-level and seemingly minor differences is not yet understood.

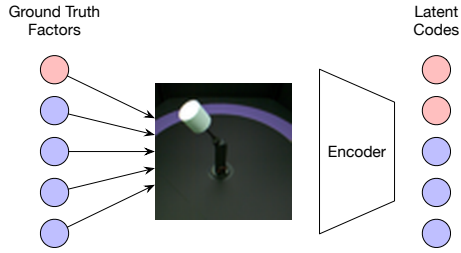




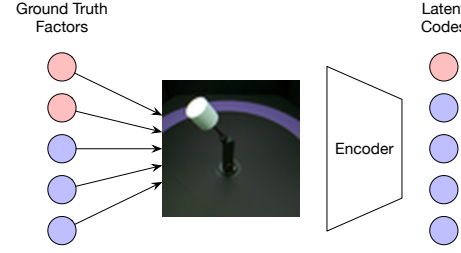
(a) Example of encoder *consistency* (Shu et al., 2020) for one factor of variation: intervening (↗) on a ground-truth factor (or subset of factors) by *fixing* its value corresponds to *fixing* a dimension (or subset of dimensions) in the representation. In this example the object shape is constant and everything else is changing.



(b) Example of encoder *restrictiveness* (Shu et al., 2020) for one factor of variation: intervening (↗) on a ground-truth factor (or subset of factors) by *changing* its value corresponds to *changing* a dimension (or subset of dimensions) in the representation. In this example only the color is changing.



(c) Example of *disentangled* encoder for one factor of variation in the sense of (Eastwood and Williams, 2018): a few dimensions are capturing a single factor.



(d) Example of encoder *compactness* for one factor of variation in the sense of (Eastwood and Williams, 2018): a factor of variation should be captured in a single dimension. However multiple factors can still be encoded in the same dimension.

Figure 11: Examples of different notions of disentanglement being captured by the scores. Further, different scores *measure* the same notion in different ways, which can introduce systematic differences in the evaluation. For the encoder to be consistent (a), restrictive (b), disentangled (c), or compact (d) the property highlighted in the each example should hold for each factor.

Once the relation between the factors and the codes is known for a given model, we need to evaluate the properties of the structure in order to measure its “disentanglement”. Since a generally accepted formal definition for disentanglement is missing (Eastwood and Williams, 2018; Higgins et al., 2018a; Ridgeway and Mozer, 2018), the desired structure of the latent space compared to the ground truth factors is a topic of debate. Eastwood and Williams (2018) (and in part Ridgeway and Mozer (2018)) proposed three properties of representations: *disentanglement*, *compactness*, and *informativeness*. A representation is disentangled if each dimension only captures a single factor of variation and compact if each factor is encoded in a single dimension, see Figures 11c and 11d. Note that disentangled representations do not need to be compact nor compact representations need to be disentangled. Combining the two implies that a representation implements a one-to-one mapping between factors of variation and latent codes. Informativeness measures how well the information about the factors of variation is accessible in the latent representations with linear models. The degree of informativeness captured by any of the disentanglement metrics is unclear. In particular, as discussed in Section 7, it is not clear whether the correlation between disentanglement metrics and downstream performance is an artifact of the linear model used to estimate the relations between factors and code (Eastwood and Williams, 2018; Kumar et al., 2018). Maintaining the terminology, the disentanglement scores in (Higgins et al., 2017a; Kim and Mnih, 2018; Ridgeway and Mozer, 2018; Eastwood and Williams, 2018; Suter et al., 2019) focus on disentanglement in the sense of (Eastwood and Williams, 2018) and (Chen et al., 2018; Kumar et al., 2018) on compactness. Note that all these scores implement their own “notion of disentanglement”. Theoretically, we can characterize existing metrics in these two groups. On the other hand, observing the latent traversal of top performing models, it is not clear what the differences between the scores are and whether compactness and disentanglement are essentially equivalent on representations learned by VAEs (a compact representation is also disentangled and vice-versa).

As a motivating example for this section consider the two models in Figure 14. While visually we may say that they are similarly disentangled, they achieve significantly different MIG scores, making the first model twice as good as the second one. Artifacts like this clearly impact the conclusions one may draw from a quantitative evaluation. Further, the structure of the representation may influence its usefulness downstream, and different properties may be useful for different tasks. For example, the applications in fairness (Locatello et al., 2019), abstract reasoning (van Steenkiste et al., 2019) and strong generalization (Locatello et al., 2020a) all conceptually rely on the disentanglement notion of (Eastwood and Williams, 2018).

In this section, we first question how much the metrics agree with each other in terms of how the models are ranked. Second, we focus on the metrics that can be estimated from observational data, as we anticipate they will be more generally applicable in practice. There, we question the impact of different choices in the estimation of the factor-code matrices as well as in the aggregation. This latter step encodes which notion of disentanglement is measured. Finally, we investigate the sample efficiency of the different metrics in order to provide practical insights on which scores may be used in practical settings where labelled data is scarce.

## 6.1 How Much Do Existing Disentanglement Metrics Agree?

As there exists no single, commonly accepted definition of disentanglement, an interesting question is to see how much the different metrics agree. Figure 12 shows pairwise scatter plots of the different considered metrics on dSprites where each point corresponds to a trained model, while Figure 13

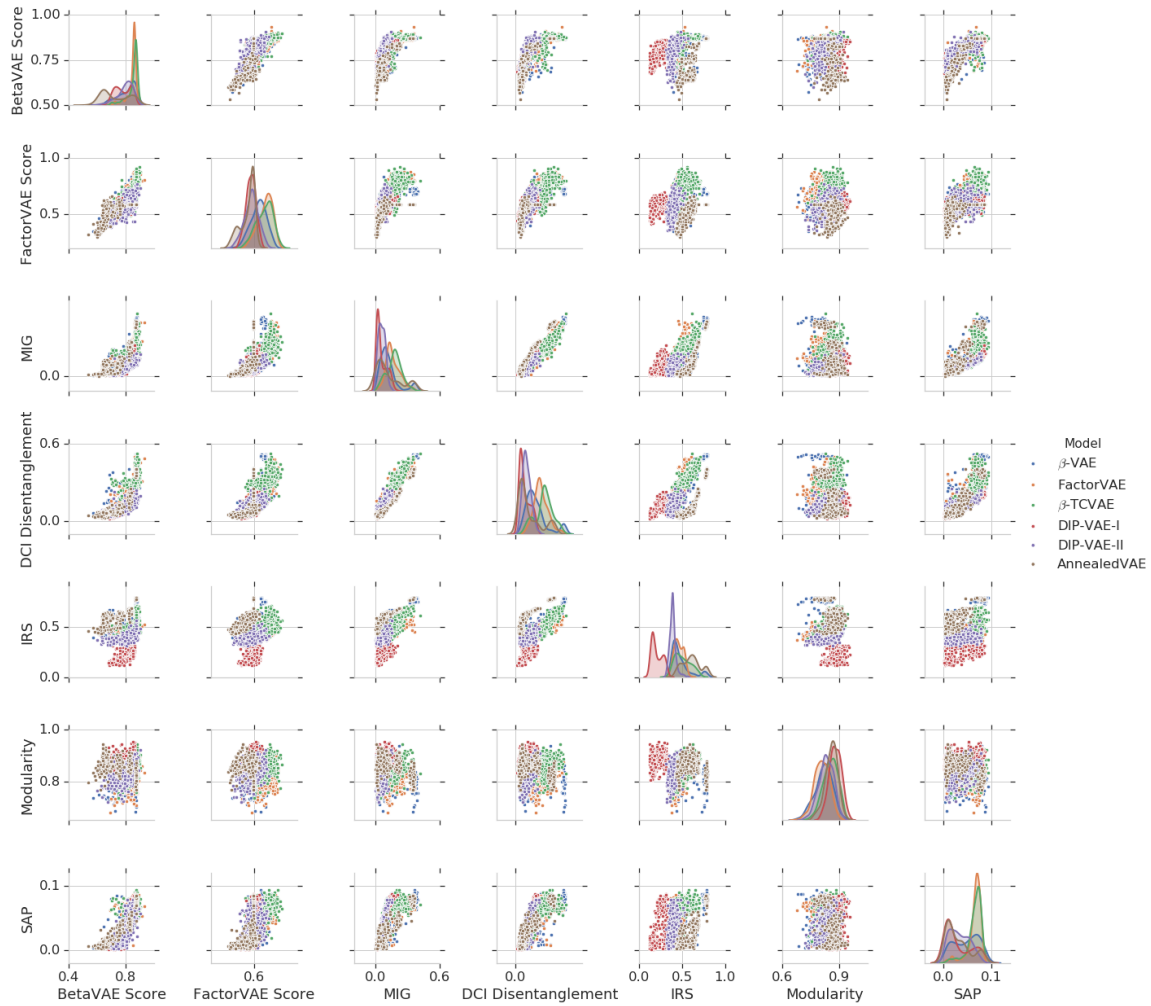


Figure 12: Pairwise scatter plots of different disentanglement metrics on dSprites. All the metrics except Modularity appear to be correlated. The strongest correlation seems to be between MIG and DCI Disentanglement.

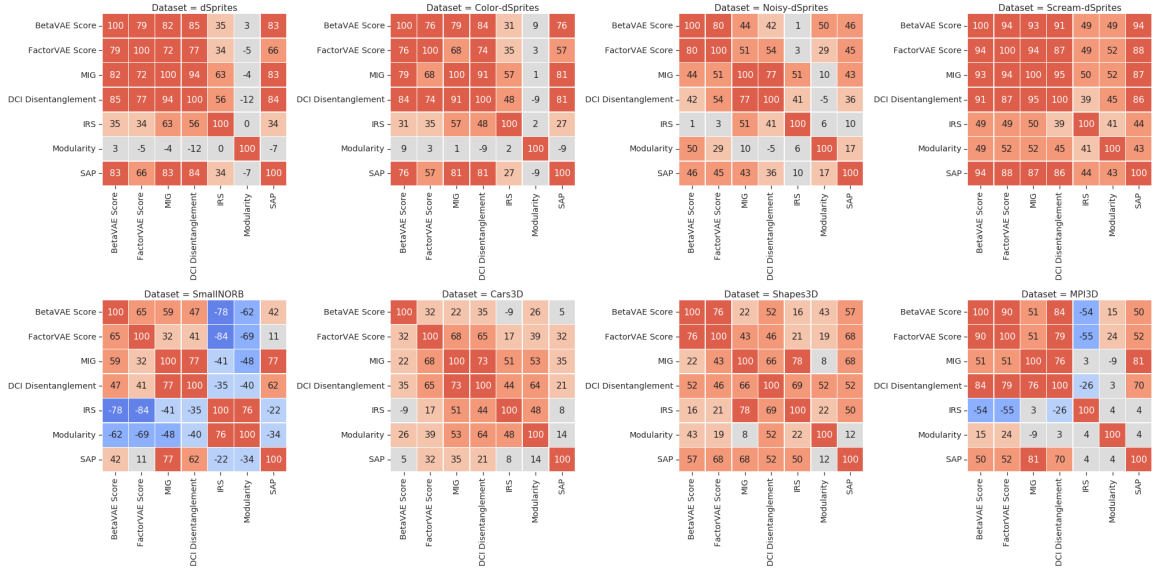


Figure 13: Rank correlation of different metrics on different data sets. Overall, we observe that all metrics except Modularity seem to be strongly correlated on the data sets dSprites, Color-dSprites and Scream-dSprites and mildly on the other data sets. There appear to be two pairs among these metrics that capture particularly similar notions: the BetaVAE and the FactorVAE score as well as the Mutual Information Gap and DCI Disentanglement.

shows the Spearman rank correlation between different disentanglement metrics on different data sets. Overall, we observe that all metrics except Modularity and, in part, IRS seem to be correlated strongly on the data sets dSprites, Color-dSprites and Scream-dSprites and mildly on the other data sets. There appear to be two pairs among these metrics that correlate well: the BetaVAE and the FactorVAE scores as well as the Mutual Information Gap and DCI Disentanglement. Note that this positive correlation does not necessarily imply that these metrics are measuring the same notion of disentanglement.

Indeed, we visualize in Figure 14 the latent traversals of two models that visually achieve similar disentanglement. Arguably, the model on the bottom may even be more disentangled than the one on the top (the shape in dimension 0 of the top model is not perfectly constant). However, the top model received a MIG of 0.66 while the model at the bottom just 0.33. We remark that similar examples can be found for other disentanglement metrics as well by looking for models with large disagreement between the scores. The two models in Figure 14 have DCI Disentanglement of 0.77 and 0.94 respectively.

The scores that require interventions and measure disentanglement computing consistency versus restrictiveness are not strongly correlated although they should be theoretically equivalent. On the other hand, we notice that the IRS is not very correlated with the other scores either, indicating that the difference may arise from how the IRS is computed.

We now investigate the differences on the scores that are computed from purely observational data: DCI Disentanglement, MIG, Modularity and SAP Score. These scores are composed of two stages. First they estimate a matrix relating factors of variation and latent codes. DCI Disentanglement considers the feature importance of a GBT predicting each factor of variation from the latent codes.

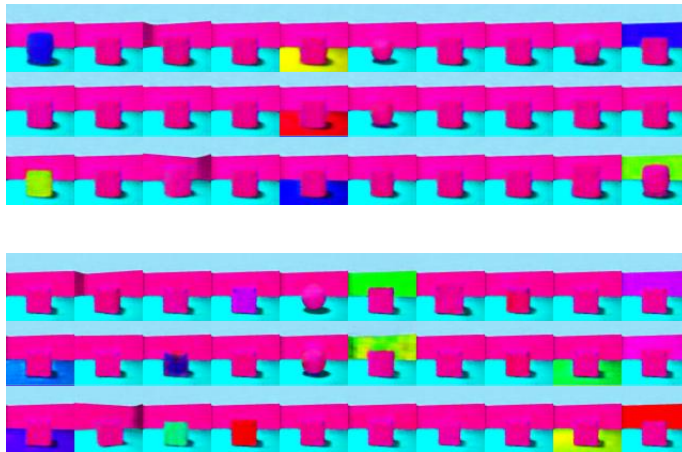


Figure 14: Latent traversal of a FactorVAE model (top) and a DIP-VAE-I (bottom) trained on Shapes3D. Despite dimensions 0, 5, and 8 not being perfectly disentangled (see Figure 17), the model at the top achieves a MIG of 0.66 while the model at the bottom 0.33. Each column corresponds to a latent dimension.

MIG and Modularity compute the pairwise mutual information matrix between factors and codes. The SAP Score computes the predictability of each factor of variation from each latent code using a SVM. Second, they aggregate this matrix into a score measuring some of its structural properties. This is typically implemented as a normalized gap between largest and second largest entries in the factor-code matrix either row or column wise. We argue that this second step is the one that most encodes the “notion of disentanglement” being measured by the score. However, the correlation between the scores may also be influenced by how the matrix is estimated. In the remainder of this section, we put under scrutiny these two steps, systematically analyzing their similarities, robustness, and biases.

#### 6.1.1 WHAT IS THE DIFFERENCE BETWEEN THE AGGREGATIONS? IS COMPACTNESS EQUIVALENT TO DISENTANGLEMENT IN PRACTICE?

In this section, we focus on the metrics that can be computed from observational data. We question the “notion of disentanglement” which is implemented by the second step of DCI Disentanglement, MIG, Modularity and SAP Score and look for differences between disentanglement and compactness in practice. These aggregations measure some structural properties of the statistical relation between factors and codes. In order to empirically understand similarities and differences of these aggregations, we compare their result when evaluating the same input matrix in Figure 15 for dSprites and the GBT feature importance matrix. We observe that the different aggregations seem to correlate well but we note that this correlation is not always consistent across different matrices and data sets as can be seen in Figure 16. We note that MIG, SAP and DCI Completeness are always strongly correlated with each other when the matrix is the same. On the contrary, MIG/SAP and DCI Disentanglement are consistently less correlated on the same matrix. The correlation between Modularity and the other scores varies dramatically depending on the matrix. This is not in contrast with Figure 13 where we observed MIG being more correlated with DCI Disentanglement rather than SAP Score. Indeed, the

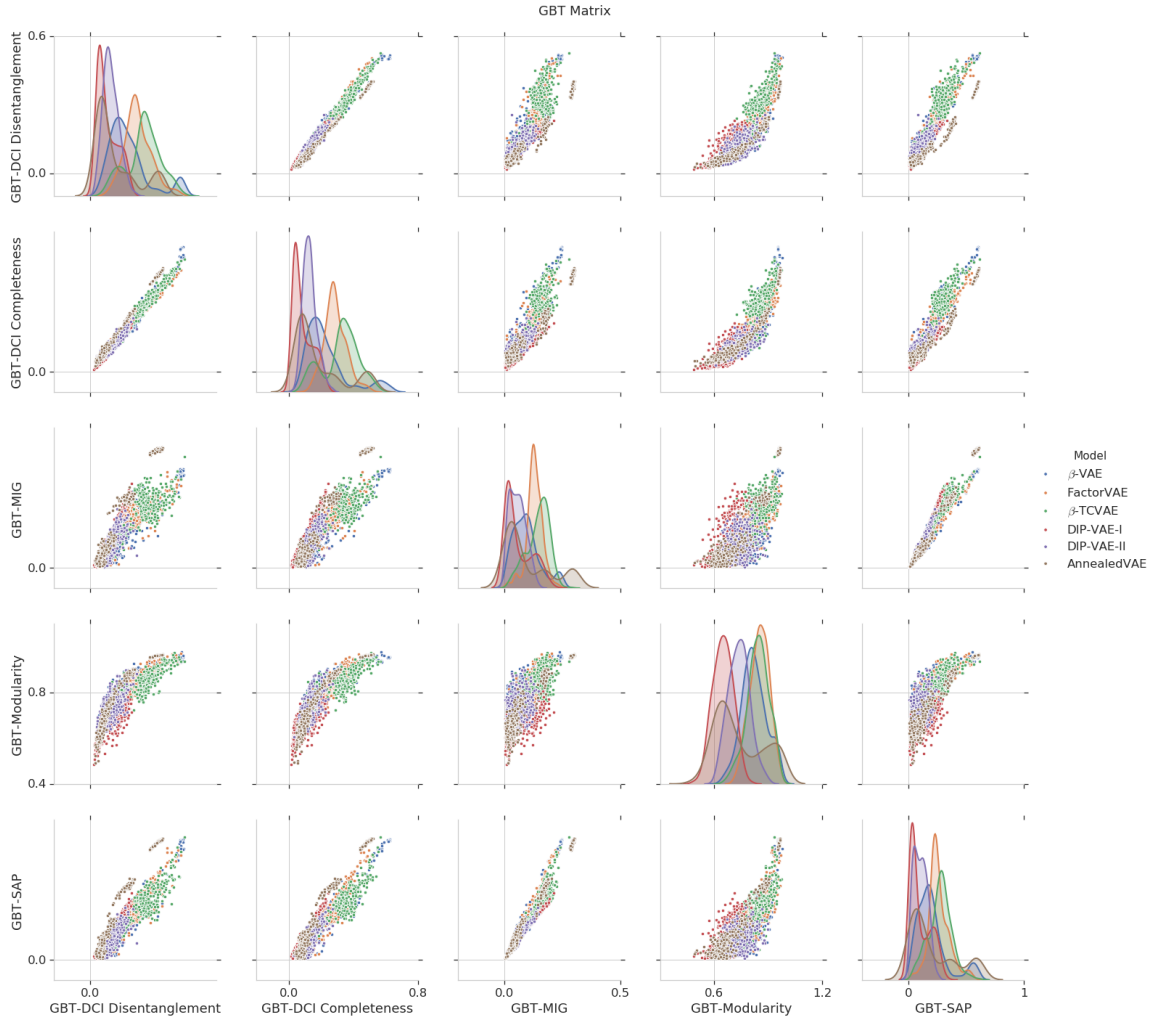


Figure 15: Aggregations computed on the same matrix (GBT feature importance) correlate well on dSprites.

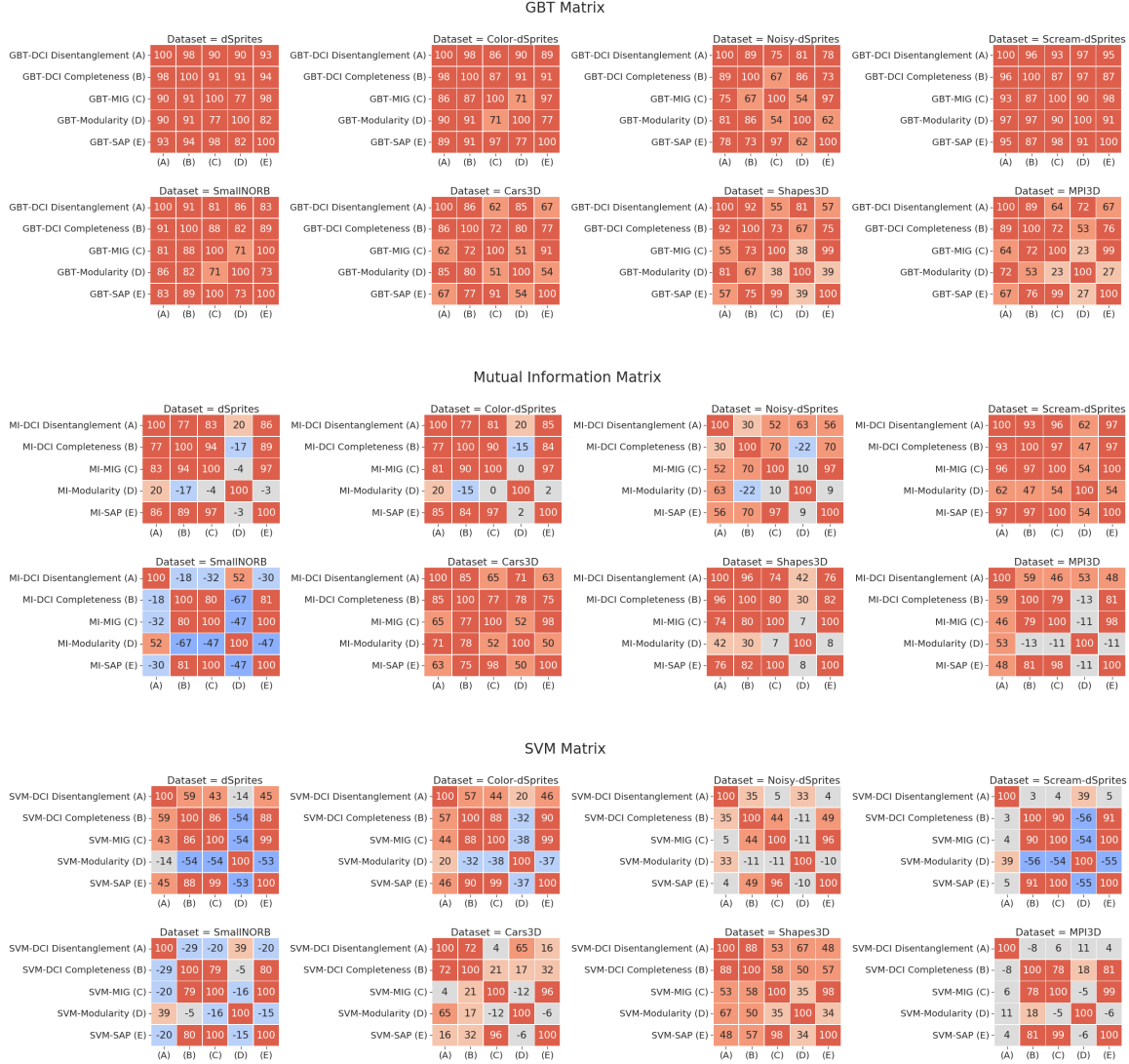


Figure 16: Rank correlation of the different aggregations computed on the same matrix (GBT feature importance, mutual information, and predictability with a SVM). When the matrix is the same, MIG, SAP and DCI Completeness are significantly more correlated while the correlation with DCI Disentanglement decreases, highlighting the difference between completeness and disentanglement (Eastwood and Williams, 2018).



dissimilarity between MIG and SAP depends on differences in the estimation of the matrix as we show in Section 6.1.2.

These results may not be surprising given the insights presented by [Eastwood and Williams \(2018\)](#). MIG and SAP computes the gap between the entries of the matrix per factor and therefore penalize compactness rather than disentanglement. In other words, they penalize whether a factor of variation is embedded in multiple codes but do not penalize the same code capturing multiple factors. DCI Disentanglement instead penalizes whether a code is related to multiple factors. Observing these differences in a large pool of trained models is challenging. First, the representations are not evenly distributed across the possible configurations (one-to-one, one-to-many, many-to-one and many-to-many) and for some of these relations (such as one-to-one and many-to-many) the scores behave similarly. Second, when comparing aggregations computed on different matrices it is typically unclear where the difference is coming from. However, we believe it is important to understand these practical differences as enforcing different notions of disentanglement may not result in the same benefits downstream.

We conclude that the similarity between the scores in Section 6.1 is confounded by how the statistical relations are computed. Further, we note that one-to-one or many-to-many mappings are preferred to one-to-many in the models we train, partially supporting the insights from [Rolinek et al. \(2019\)](#).

### 6.1.2 DOES THE ESTIMATION FACTOR-CODE MATRICES IMPACT THE EVALUATION?

In this section, we continue to investigate the metrics that can be computed from observational data and focus on the different matrices estimating the statistical relations between factors of variation and latent codes. First, we build new visualization tools that allow us to understand both what a model has learned and how its been evaluated by the factor-code matrices.

In Figure 17 we visualize the model at the bottom of Figure 14. On the first row, we plot the factor-codes matrices as learned by GBT feature importance, pairwise mutual information and SVM predictability respectively. We observe that for the GBT features and the mutual information matrix the largest entries are the same but the latter underestimates the effect of some dependencies, for example object size and type in dimensions number five and eight. The SVM feature importance, also agrees on some of the large values but exhibit a longer tail compared to the other matrices.

In order to further analyze the differences between the matrices we view them as weights on the edges of a bipartite graph encoding the statistical relation between each factor of variation and code. We can now delete all edges with weight smaller than some threshold and count (i) how many factors of variation are connected with at least a latent code and (ii) the number of connected components with size larger than one. In Figure 17 (middle row), we plot these two curves computed on the respective matrices, and, in Figure 17 (bottom row), we record which factors are merged at which threshold. Factors that are merged at lower threshold are more entangled in the sense that are more statistically related to a shared latent dimension.

The long tail of the SVM importance matrix explains why we observed a weaker correlation between MIG and SAP Score in Figure 13 even though the scores are measuring a similar concept. Indeed, we can observe in the middle row of Figure 17 that the largest entries of the three matrices are distributed differently, in particular for the SVM predictability. Similarly, we can read in the dendrogram plot that the factors are merged in a significantly different order for the SVM predictability compared to the other two matrices. We hypothesize that the long tail of the SVM

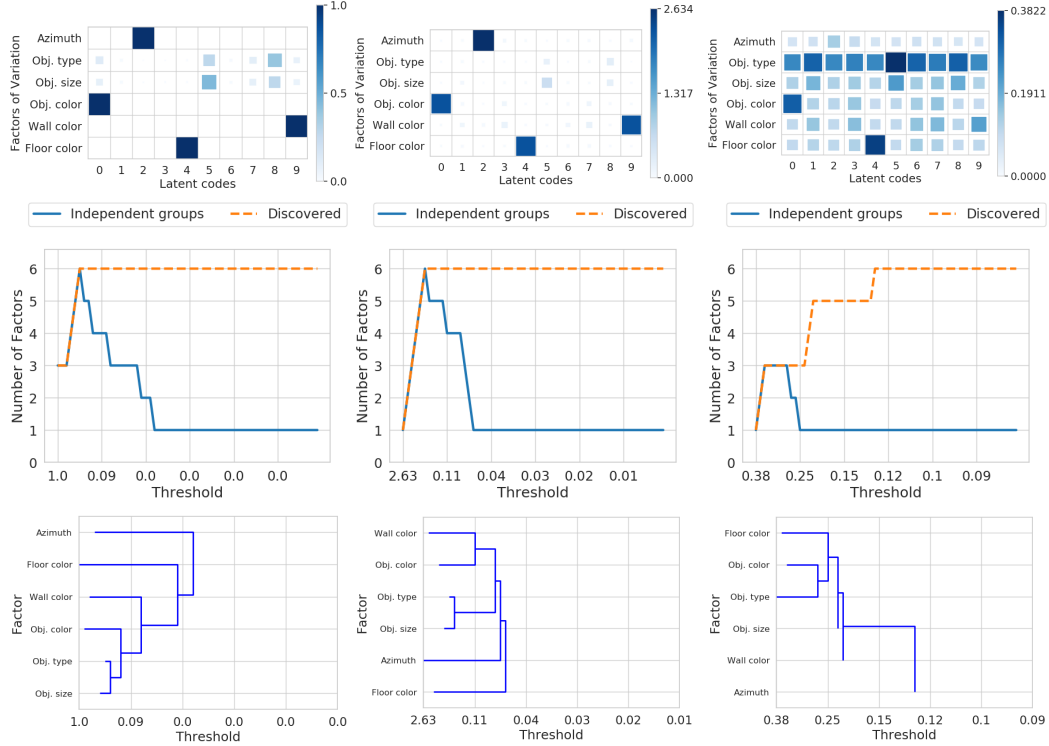


Figure 17: Visualization of the relation between factors of variations and latent codes using for the model at the top of Figure 14: (left) GBT feature importance as in the DCI Disentanglement score, (center) the mutual information as computed in the MIG and Modularity and, (right) SVM predictability as computed by the SAP Score. Top row: factor-code matrix. Middle row: independent-groups curve recording how many connected components of size larger than one there are in the factor-code bipartite graph defined by the matrix at a given threshold. Bottom row: dendrogram plot recording which factors are merged at which threshold. The long tail of the SVM importance matrix explains the weaker correlation between MIG and SAP Score in Figure 13 even though the scores are measuring a similar concept. The dendrogram plots computed from the independent-groups curve can be used to systematically analyze which factors are merged at which threshold by the different estimation techniques (e.g. SVM, GBT feature importance and mutual information).

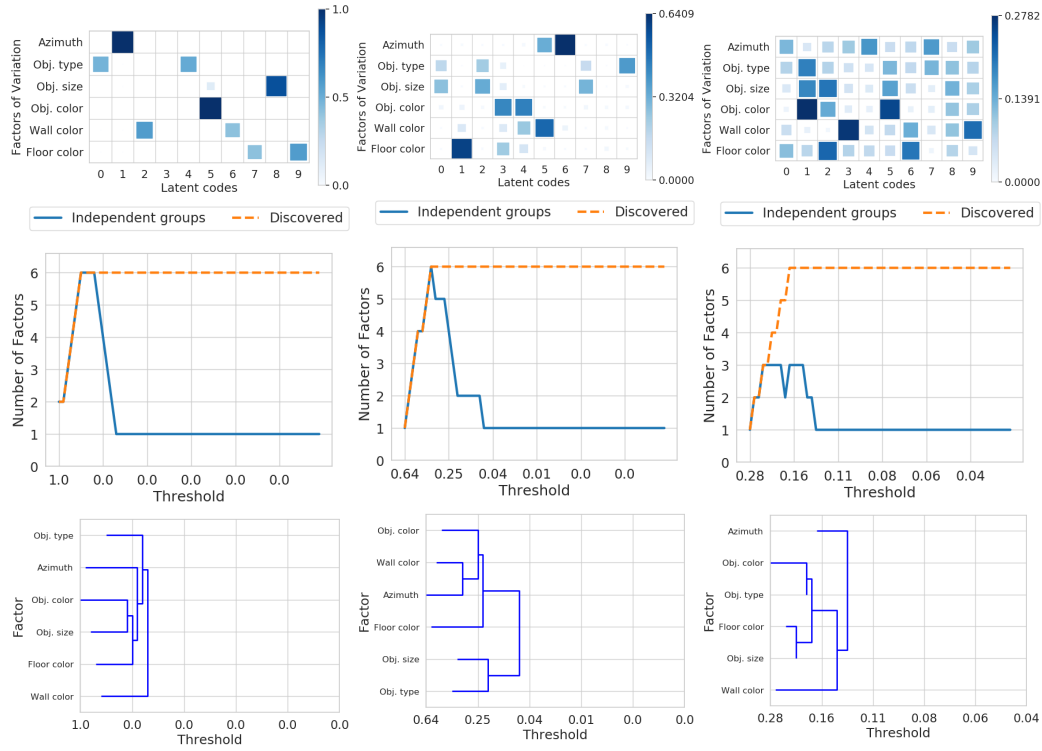


Figure 18: (top row) Visualization of the GBT importance matrix used in the DCI Disentanglement score for models with top (left), average (center), and worse (right) DCI Disentanglement on Shapes3D. (middle row) Independent-groups curves of the GBT importance matrix. (bottom row) Dendrogram plot recording when factors are merged. Comparing these plots with the ones in Figure 19, we note that there are differences in the factor-code matrices. In particular, they disagree on which factors are most entangled.

predictability is a consequence of spurious correlations and optimization issues that arise from how the score is computed (fitting a threshold separately on each code predicting each factor).

In Figures 18 and 19 we compare the factor-code matrices, independent-groups curves, and dendrograms for the best, average and worse model in terms of DCI Disentanglement. Figure 18 shows the plots for the GBT (Gradient Boosted Trees) feature importance matrix used by the DCI Disentanglement score and Figure 19 the mutual information matrix of MIG and Modularity. By comparing these plots, we can clearly distinguish which model is the most disentangled but we again note differences in how the factors of variation are captured by the different matrices. In particular, we again observe that the two matrices may disagree on which factors are most entangled in the same model. For example, the GBT features computed on the model on the left suggest that object color and size are more entangled while the mutual information matrix suggest azimuth and wall color. These differences appear to be systematic. From the dendrogram plot of each model and estimation matrix, we can compute at which threshold each pair of factors is merged on average. This allow us to systematically analyze the differences in terms of which factors are found more

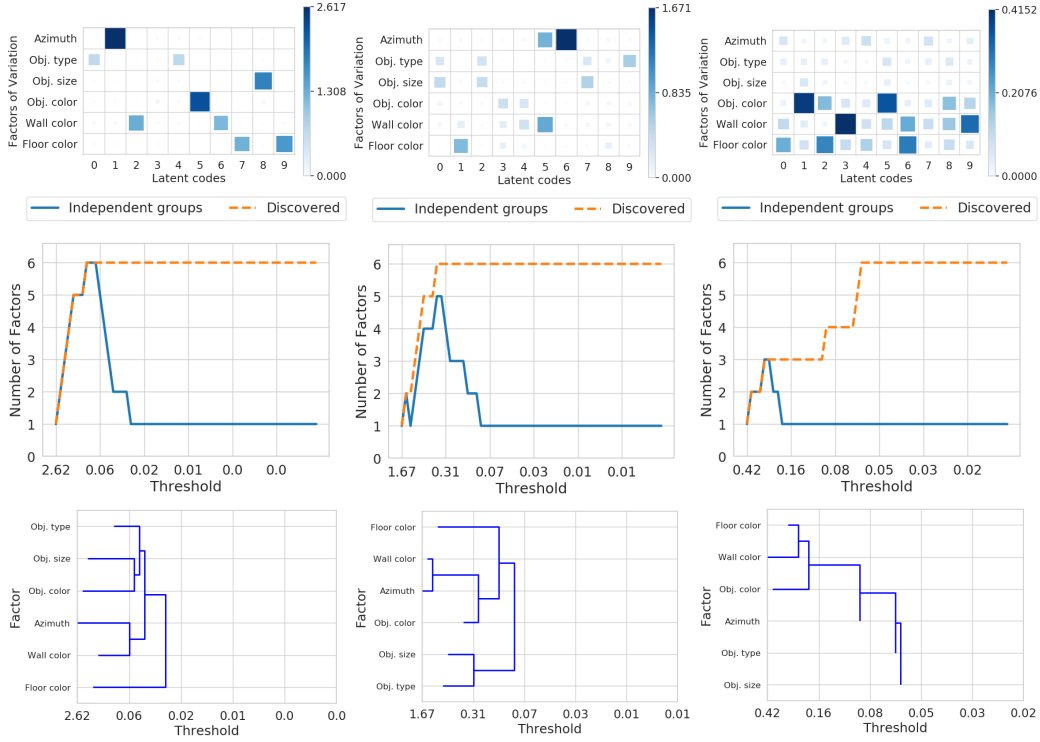


Figure 19: (top row) Visualization of the mutual information matrix used in the MIG and Modularity scores for the same models of Figure 18. (middle row) Independent-groups curves of the mutual information matrix. (bottom row) Dendrogram plot recording when factors are merged. Comparing these plots with the ones in Figure 18, we note that there are differences in the factor-code matrices. In particular, they disagree on which factors are most entangled.

entangled by the different matrices. In Figure 34 in the Appendix, we can see that on dSprites and Color-dSprites some factors of variation are consistently entangled across different data set and estimation matrices indicating that they are hardest to disentangle. On the other variants, the different matrices significantly disagree and similar results can be observed in Figure 35 in the Appendix for the other data sets. This indicates systematic differences in the structure found by different estimation techniques which may impact the final computation of the scores.

Finally, we test whether the differences in the factor-code matrix impact the computation of the disentanglement scores. To do so, we compare the ranking produced by each aggregation computed on the different matrices. If the different matrices encode the same statistical relations, the ranking should also be similar. We observe in Figure 20 that the ranking seem to be generally different and the level of correlation appears to depend on the data set. Overall, the aggregation of SAP Score and MIG seem to be more robust to changes in the estimation matrix compared to Modularity and DCI Disentanglement.

Based on this result, we conclude that systematic differences in the estimation matrix may indeed impact the evaluation of disentanglement. It seem important for the evaluation that the statistical

relations between factors and codes are robustly and consistently estimated. We observed that changing the estimation technique may produce different rankings of the models. It appears therefore important to not bias the evaluation by considering a single estimation technique, unless reliability guarantees are also given.

### 6.1.3 IMPLICATIONS

We conclude that the different disentanglement scores are not measuring the same concept: they measure different notions of disentanglement (compactness versus disentanglement) that are generally correlated in practice but not equivalent.

In particular, MIG and SAP Score intend disentanglement differently than DCI Disentanglement as they are rather measure completeness: they do not penalize multiple factors of variation being captured by a single latent dimension. Modularity seem to be more dependent on the estimation matrix as its correlation with the other scores changes significantly with different matrices. Furthermore, there are systematic differences between the different techniques to estimate the relation between factors of variation and latent codes that influence the correlation of the scores: the ranking of the models is different depending on the chosen estimation technique.

We argue that future works advancing the state-of-the-art in disentanglement, with or without any form of supervision, should reflect upon which notion of disentanglement they consider and how it is measured in the chosen evaluation protocol.

Not all the properties that are generally associated with the term “disentanglement” are necessarily related to all the scores considered in this paper and specific downstream tasks may require specific notions (Locatello et al., 2019; van Steenkiste et al., 2019; Locatello et al., 2020a). Further, separating the estimation of the statistical dependencies between factors of variation and codes from what the score is measuring may help clarify the properties that are being evaluated. As robustly capturing these statistical dependencies is a crucial step of the evaluation metrics that do not rely on interventions, we argue that future work on disentanglement scores should specifically highlight (i) how this estimation is performed precisely, (ii) its sample complexity/variance and (iii) biases (for example do they work well with coarse grained as opposed to fine grained factors of variation). Future research is necessary to understand both how estimation metrics overestimate or underestimate the amount of disentanglement and how to robustly aggregate this information into a score. Among the scores tested in this paper, we recommend to use the DCI aggregation, either with the GBT feature importance or the mutual information matrix, ideally both.

## 6.2 Is the Computation of the Disentanglement Scores Reliable?

The computation of the disentanglement scores require supervision and having access to a large number of observations of  $\mathbf{z}$  may be unreasonable. On the other hand, for the purpose of this study we are interested in a stable and reproducible experimental setup. In Figure 21, we observe that running the disentanglement scores twice yields comparable results with 10 000 examples. Using just 100 examples may be feasible in practice as suggested by Locatello et al. (2020b) but has less stable results as depicted in Figure 22. We observe that not every score is equally sample efficient. The FactorVAE scores and the IRS seem to be the most efficient ones, followed by DCI Disentanglement and MIG.

# UNSUPERVISED LEARNING OF DISENTANGLED REPRESENTATIONS AND THEIR EVALUATION

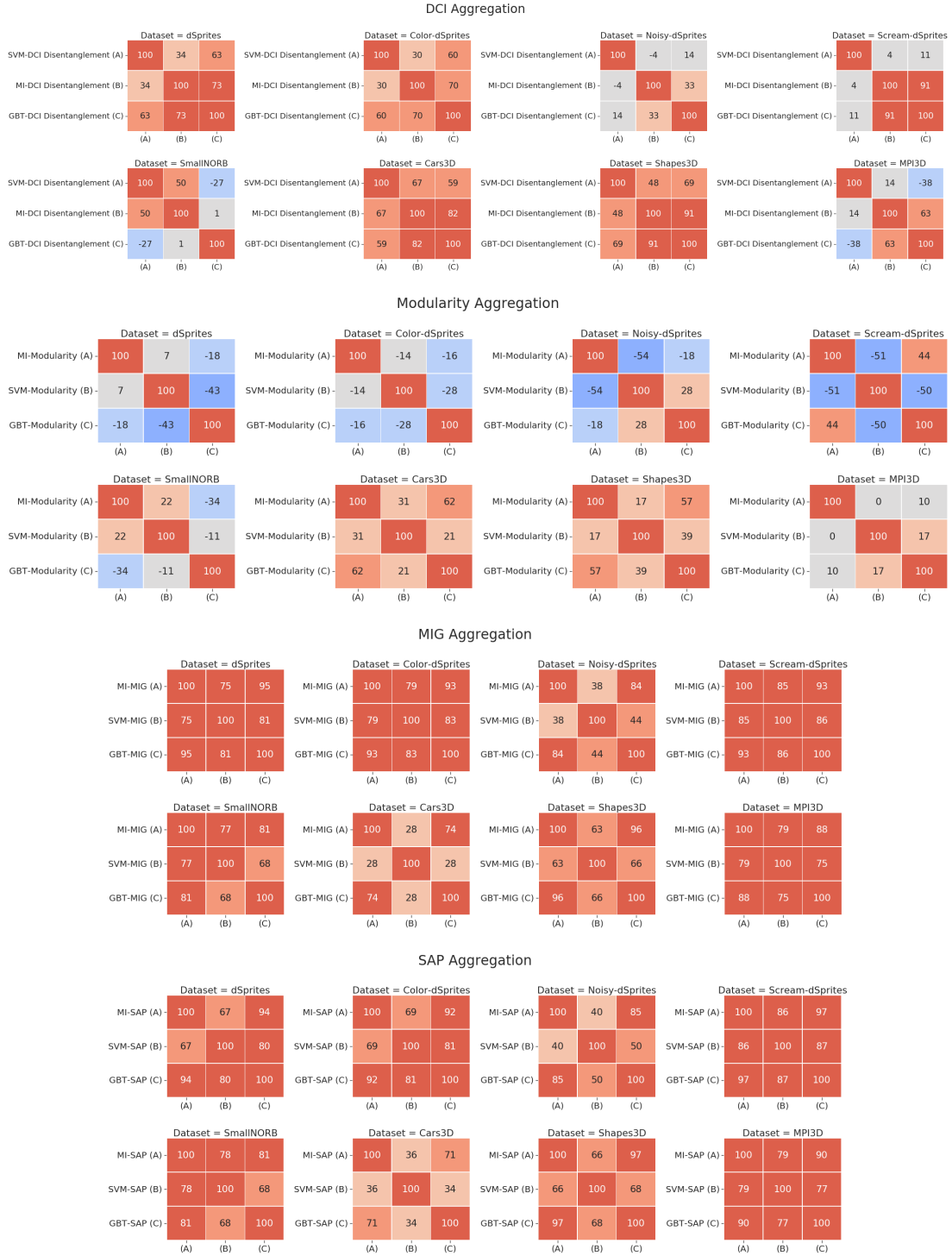


Figure 20: Rank correlation of DCI Disentanglement, Modularity, SAP Score and MIG aggregations on different matrices. The ranking seem to be generally different and data set dependant indicating that systematic differences in the estimation matrix may impact the evaluation of disentanglement. MIG and SAP aggregations appear to be more robust to changes in the estimation matrix.



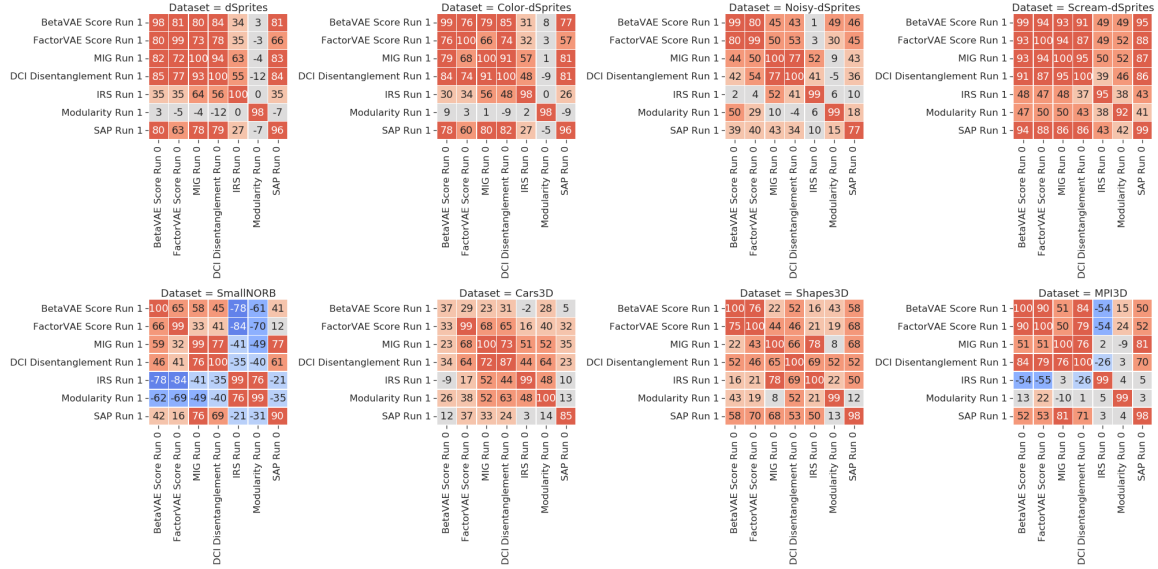


Figure 21: Rank correlation of different metrics on different data sets across two runs. Overall, we observe that the disentanglement scores computed with 10 000 examples are relatively stable.

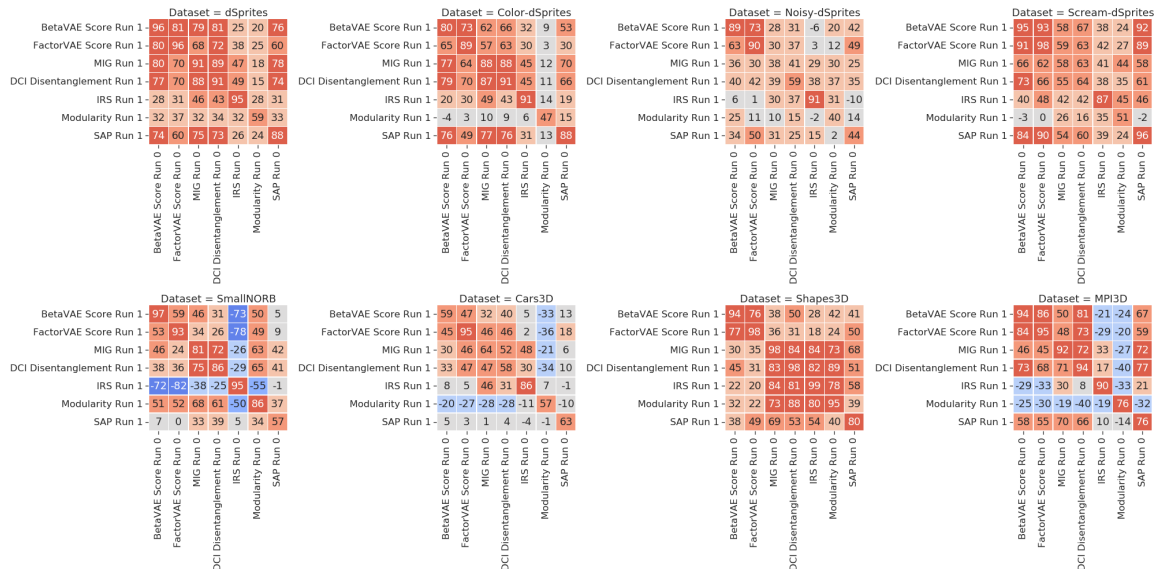


Figure 22: Rank correlation of different metrics on different data sets across two runs. Overall, we observe that with fewer examples the disentanglement scores are significantly less stable.

Dataset = dSprites									
LR10	21	24	22	29	13	-13	20		
LR100	61	43	55	59	25	-13	55		
LR1000	55	35	40	43	7	5	43		
LR10000	38	21	9	15	-37	15	22		
GBT10	46	43	51	55	39	-2	44		
GBT100	78	73	87	95	59	-17	78		
GBT1000	77	72	86	94	58	-18	75		
GBT10000	75	71	87	94	58	-12	75		
Efficiency (LR)	24	21	41	39	53	-20	31		
Efficiency (GBT)	45	41	44	50	34	-21	47		
BetaVAE Score									
FactorVAE Score									
IMG									
DC Disentanglement									
IRS									
Modularity									
SAP									

Dataset = Color-dSprites									
LR10	20	20	15	21	2	-1	17		
LR100	44	19	33	37	6	5	30		
LR1000	28	5	7	11	-22	16	13		
LR10000	8	-14	-19	-15	-50	22	-7		
GBT10	45	38	45	49	26	-1	41		
GBT100	79	68	85	93	51	-9	76		
GBT1000	75	65	84	90	56	-4	68		
GBT10000	69	60	81	83	56	4	63		
Efficiency (LR)	26	29	45	44	52	-16	29		
Efficiency (GBT)	38	35	29	40	3	-26	44		
BetaVAE Score									
FactorVAE Score									
IMG									
DC Disentanglement									
IRS									
Modularity									
SAP									

Dataset = Noisy-dSprites									
LR10	18	24	25	36	-1	-11	11		
LR100	56	36	8	-1	2	50	28		
LR1000	63	42	2	-8	-11	59	28		
LR10000	63	44	-2	-11	-18	58	28		
GBT10	30	29	30	42	22	11	17		
GBT100	26	37	64	85	48	-16	28		
GBT1000	-2	22	54	75	32	-35	12		
GBT10000	-9	17	45	65	20	-34	5		
Efficiency (LR)	-14	-14	13	13	35	-18	-4		
Efficiency (GBT)	31	18	28	32	36	7	27		
BetaVAE Score									
FactorVAE Score									
IMG									
DC Disentanglement									
IRS									
Modularity									
SAP									

Dataset = Scream-dSprites									
LR10	66	70	66	62	34	43	62		
LR100	69	65	64	65	31	39	69		
LR1000	86	85	80	76	54	47	82		
LR10000	91	85	83	81	49	44	87		
GBT10	49	53	53	52	20	42	44		
GBT100	85	77	84	93	30	36	82		
GBT1000	82	75	84	94	24	36	78		
GBT10000	80	72	82	92	21	34	77		
Efficiency (LR)	-51	-50	-49	-45	-32	-39	-47		
Efficiency (GBT)	-66	-59	-70	-80	-8	-29	-63		
BetaVAE Score									
FactorVAE Score									
IMG									
DC Disentanglement									
IRS									
Modularity									
SAP									

Dataset = SmallNORB									
LR10	-22	-34	-40	-53	24	37	-29		
LR100	-13	-10	-48	-60	9	22	-53		
LR1000	20	16	-18	-34	-24	0	-32		
LR10000	58	50	13	1	-60	-34	-9		
GBT10	-32	-29	-36	-35	36	44	-36		
GBT100	55	38	80	91	-38	40	62		
GBT1000	72	58	78	86	-60	-55	55		
GBT10000	76	64	75	81	-68	-61	53		
Efficiency (LR)	-56	-53	-60	-65	57	50	44		
Efficiency (GBT)	-72	-70	-36	-34	67	68	-16		
BetaVAE Score									
FactorVAE Score									
IMG									
DC Disentanglement									
IRS									
Modularity									
SAP									

Dataset = Cars3D									
LR10	4	-18	-27	-25	-30	-26	-12		
LR100	-4	-22	-57	-48	-64	-55	-17		
LR1000	9	-10	-44	-30	-67	-51	-17		
LR10000	14	-6	-40	-24	-64	-43	-17		
GBT10	-4	-10	-0	1	0	-7	-3		
GBT100	13	24	39	45	28	16	0		
GBT1000	37	57	63	79	31	48	9		
GBT10000	34	34	29	49	4	26	-2		
Efficiency (LR)	-27	-16	-7	-22	19	0	9		
Efficiency (GBT)	-20	-8	10	-2	21	-11	1		
BetaVAE Score									
FactorVAE Score									
IMG									
DC Disentanglement									
IRS									
Modularity									
SAP									

Dataset = Shapes3D									
LR10	45	31	-9	9	-8	15	15		
LR100	72	52	-4	29	-2	30	30		
LR1000	45	9	-46	2	-40	38	-12		
LR10000	41	5	-53	-9	-48	31	-15		
GBT10	60	50	51	76	54	48	53		
GBT100	39	32	69	95	76	51	44		
GBT1000	50	39	54	97	62	54	44		
GBT10000	57	44	48	95	54	54	44		
Efficiency (LR)	20	46	66	41	56	-8	49		
Efficiency (GBT)	19	17	75	79	83	39	38		
BetaVAE Score									
FactorVAE Score									
IMG									
DC Disentanglement									
IRS									
Modularity									
SAP									

Dataset = MPI3D									
LR10	-6	-2	-25	-18	8	40	-4		
LR100	79	76	37	66	-38	26	47		
LR1000	77	73	32	63	-44	18	42		
LR10000	86	80	41	72	-49	13	48		
GBT10	35	34	19	38	1	21	30		
GBT100	78	68	66	90	-12	-1	59		
GBT1000	87	76	59	87	-29	1	53		
GBT10000	90	79	54	83	-40	3	49		
Efficiency (LR)	-18	-12	-11	-19	21	33	1		
Efficiency (GBT)	-4	-6	-42	-33	60	-3	39		
BetaVAE Score									
FactorVAE Score									
IMG									
DC Disentanglement									
IRS									
Modularity									
SAP									

Figure 23: Rank-correlation between the metrics and the performance on downstream task on different data sets. We observe some correlation between most disentanglement metrics and downstream performance. However, the correlation varies across data sets.

### 6.2.1 IMPLICATIONS

Computing the disentanglement scores on these data sets with 10 000 examples yields stable results and is appropriate for the purpose of this study. Finding sample efficient disentanglement scores is an important research direction for practical semi-supervised disentanglement (Locatello et al., 2020b).

## 7. Are These Disentangled Representations Useful for Downstream Tasks in Terms of the Sample Complexity of Learning?

One of the key motivations behind disentangled representations is that they are assumed to be useful for later downstream tasks. In particular, it is argued that disentanglement should lead to a better sample complexity of learning (Bengio et al., 2013; Schölkopf et al., 2012; Peters et al., 2017). In this section, we consider the simplest downstream classification task where the goal is to recover the true factors of variations from the learned representation using either multi-class logistic regression (LR) or gradient boosted trees (GBT). Our goal is to investigate the relationship between disentanglement and the average classification accuracy on these downstream tasks as well as whether better disentanglement leads to a decreased sample complexity of learning.

To compute the classification accuracy for each trained model, we sample true factors of variations and observations from our ground truth generative models. We then feed the observations into our trained model and take the mean of the Gaussian encoder as the representations. Finally, we predict each of the ground-truth factors based on the representations with a separate learning algorithm. We consider both a 5-fold cross-validated multi-class logistic regression as well as gradient boosted trees of the Scikit-learn package. For each of these methods, we train on 10, 100, 1000 and 10 000 samples. We compute the average accuracy across all factors of variation using an additional set 10 000 randomly drawn samples.

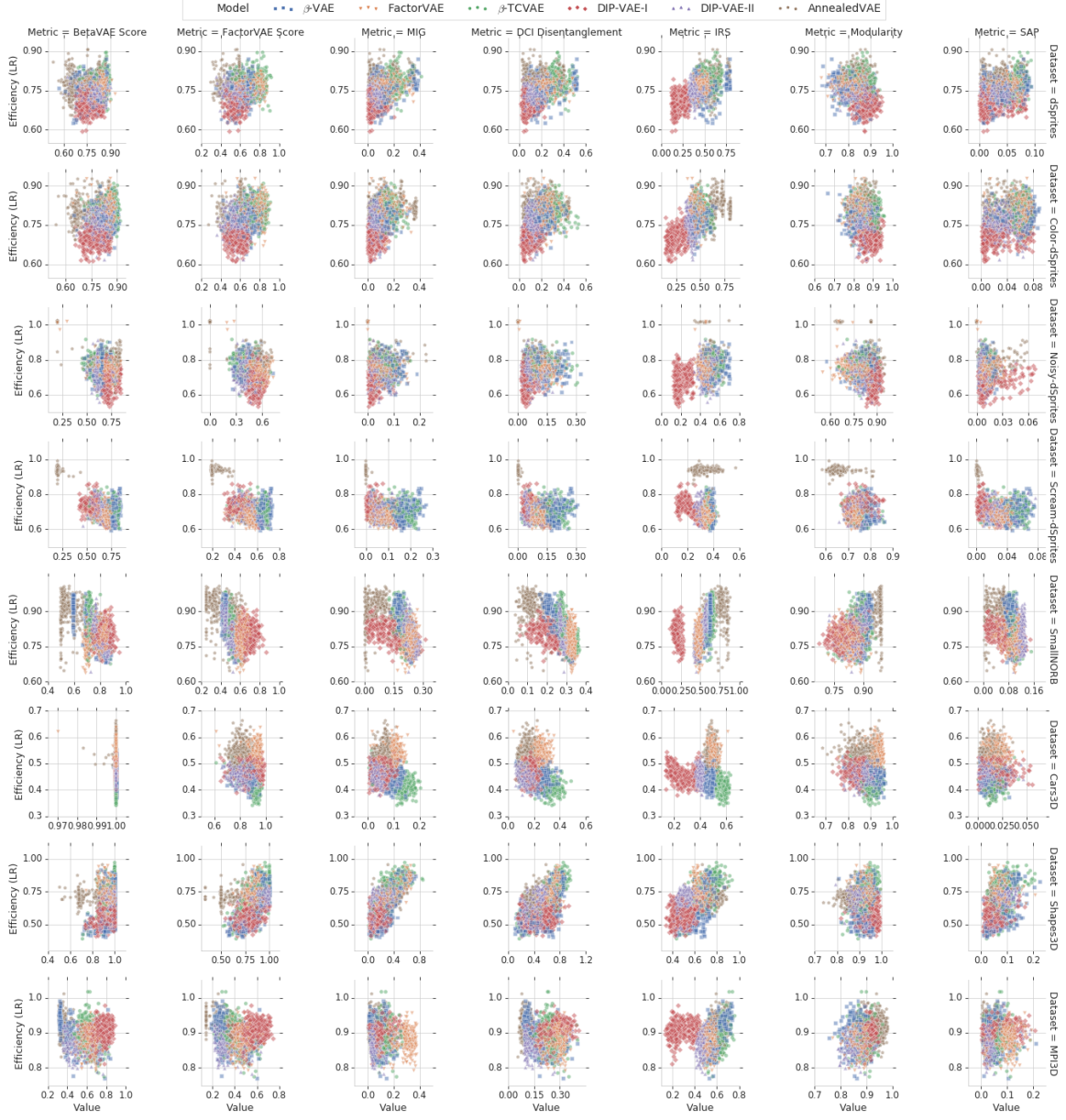


Figure 24: Statistical efficiency (accuracy with 100 samples  $\div$  accuracy with 10 000 samples) based on a logistic regression versus disentanglement metrics for different models and data sets. We do not observe that higher disentanglement scores lead to higher statistical efficiency.

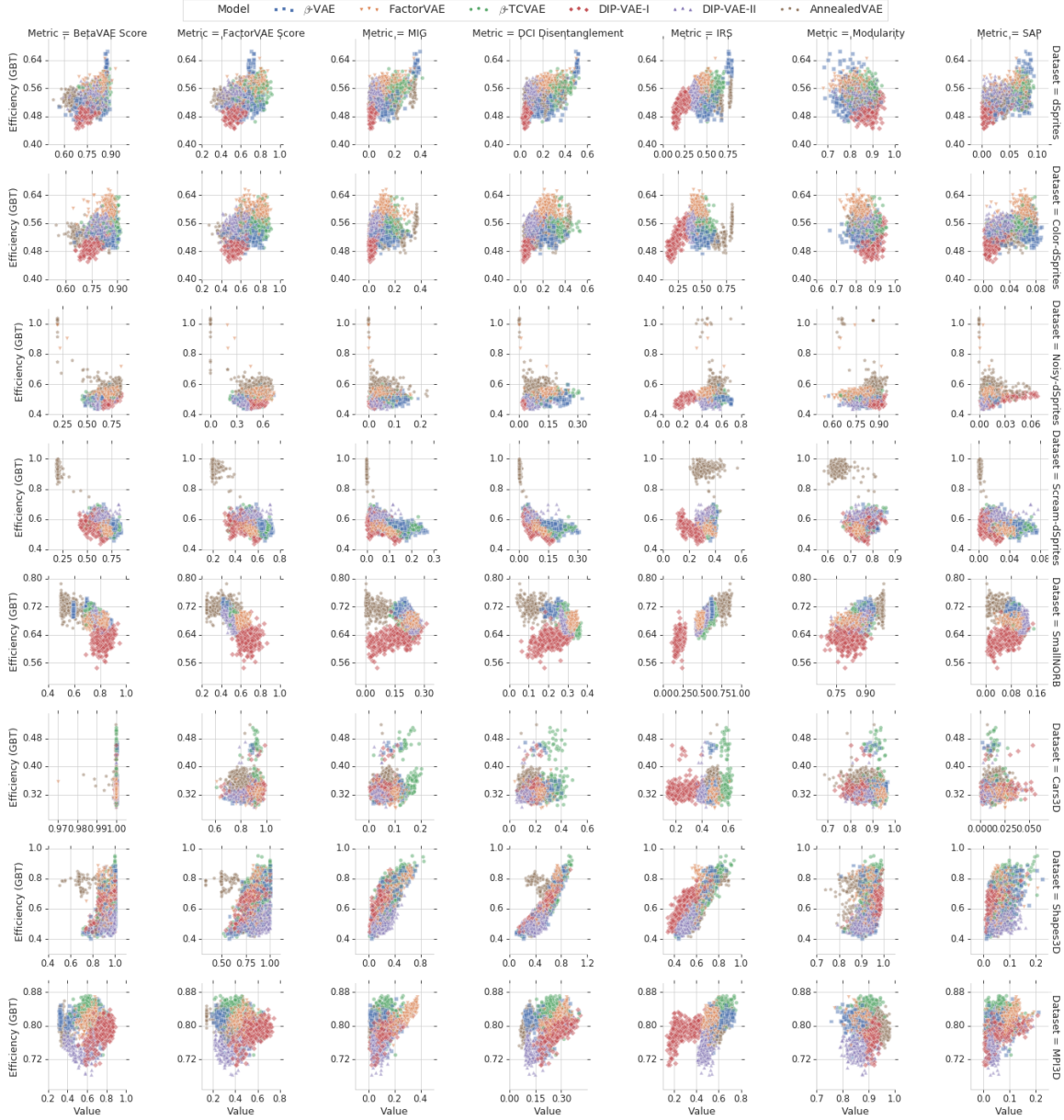


Figure 25: Statistical efficiency (accuracy with 100 samples  $\div$  accuracy with 10 000 samples) based on gradient boosted trees versus disentanglement metrics for different models and data sets. We do not observe that higher disentanglement scores lead to higher statistical efficiency (except for DCI Disentanglement and Mutual Information Gap on Shapes3D and to some extent in Cars3D).

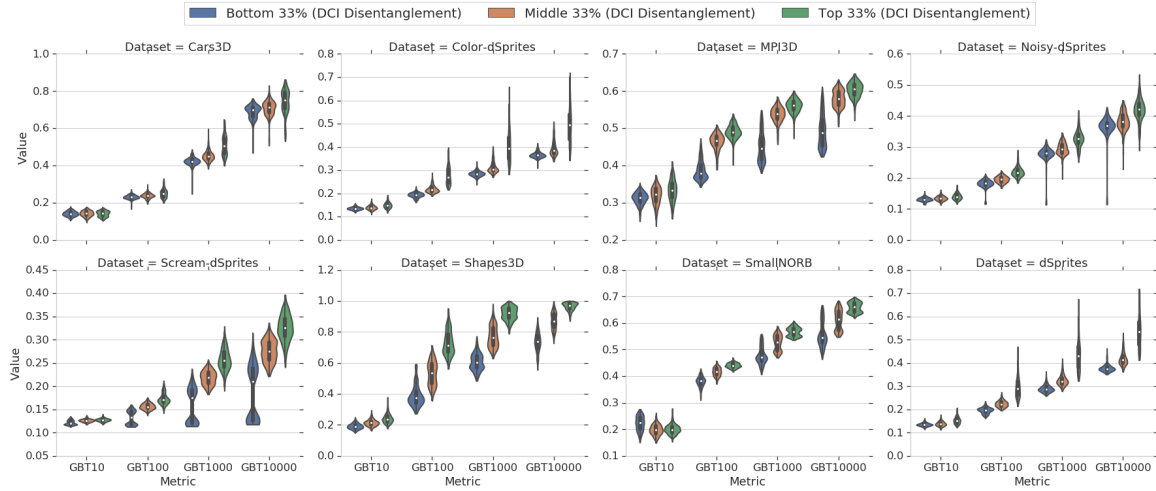


Figure 26: Downstream performance for three groups with increasing DCI Disentanglement scores.

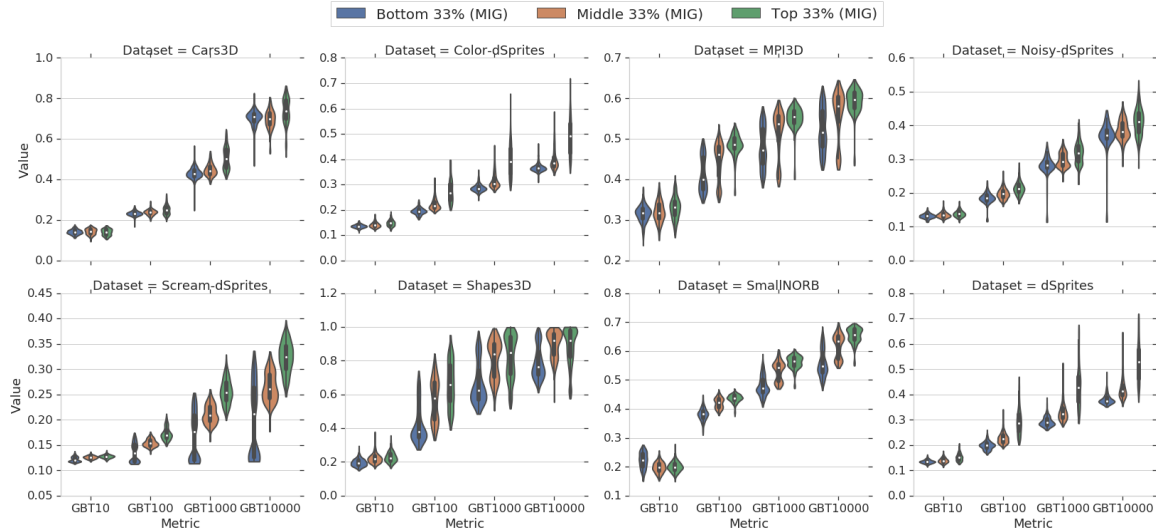


Figure 27: Downstream performance for three groups with increasing MIG scores.

Figure 23 shows the rank correlations between the disentanglement metrics and the downstream performance for all considered data sets. We observe that all metrics except Modularity seem to be correlated with increased downstream performance on the different variations of dSprites and to some degree on Shapes3D. However, it is not clear whether this is due to the fact that disentangled representations perform better or whether some of these scores actually also (partially) capture the informativeness of the evaluated representation. Furthermore, the correlation is weaker or inexistent on other data sets (for example, Cars3D). Finally, we report in Figure 28 the rank correlation between unsupervised scores computed after training on the mean and sampled representation and downstream performance. Depending on the data set, the rank correlation ranges from mildly negative, to mildly positive. In particular, we do not observe enough evidence supporting the claim that decreased total correlation of the aggregate posterior proves beneficial for downstream task performance.



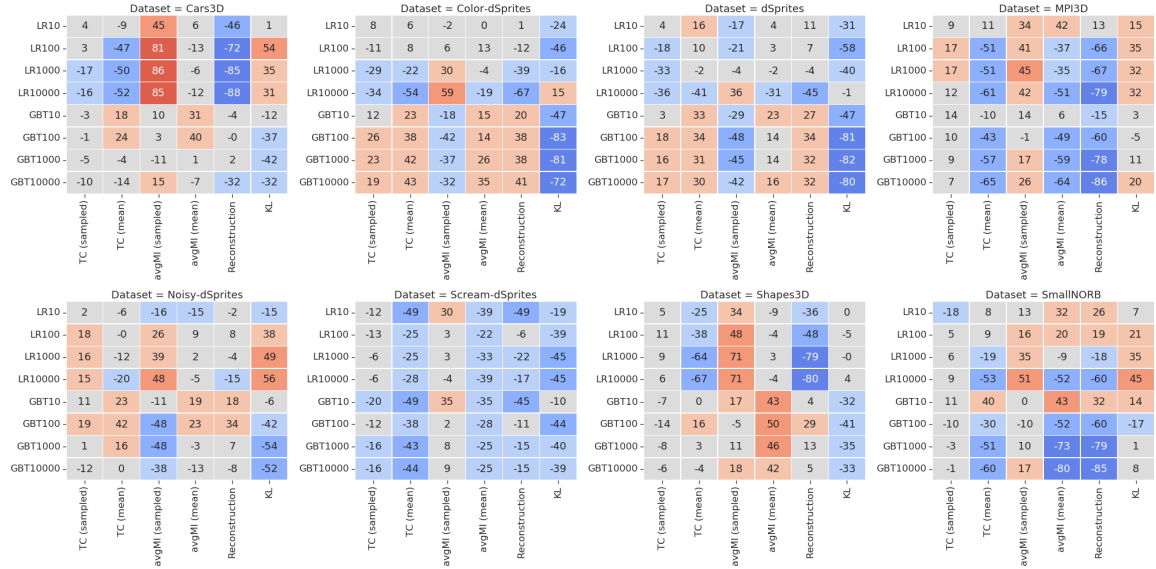


Figure 28: Rank correlation between unsupervised scores and downstream performance.

To assess the sample complexity argument we compute for each trained model a statistical efficiency score which we define as the average accuracy based on 100 samples divided by the average accuracy based on 10 000 samples for either the logistic regression or the gradient boosted trees. The key idea is that if disentangled representations lead to sample efficiency, then they should also exhibit a higher statistical efficiency score. We remark that this score differs from the definition of sample complexity commonly used in statistical learning theory. The corresponding results are shown in Figures 24 and 25 where we plot the statistical efficiency versus different disentanglement metrics for different data sets and models and in Figure 23 where we show rank correlations. Overall, we do not observe conclusive evidence that models with higher disentanglement scores also lead to higher statistical efficiency. We note that some AnnealedVAE models seem to exhibit a high statistical efficiency on Scream-dSprites and to some degree on Noisy-dSprites. This can be explained by the fact that these models have low downstream performance and that hence the accuracy with 100 samples is similar to the accuracy with 10 000 samples. We further observe that DCI Disentanglement and MIG seem to be lead to a better statistical efficiency on the the data set Shapes3D for gradient boosted trees. Figures 26 and 27 show the downstream performance for three groups with increasing levels of disentanglement (measured in DCI Disentanglement and MIG respectively). We observe that indeed models with higher disentanglement scores seem to exhibit better performance for gradient boosted trees with 100 samples. However, considering all data sets, it appears that overall increased disentanglement is rather correlated with better downstream performance (on some data sets) and not statistical efficiency. We do not observe that higher disentanglement scores reliably lead to a higher sample efficiency.

## 7.1 Implications

While the empirical results in this section are negative, they should also be interpreted with care. After all, we have seen in previous sections that the models considered in this study fail to reliably produce



disentangled representations. Hence, the results in this section might change if one were to consider a different set of models, for example semi-supervised or fully supervised one. Furthermore, there are many more potential notions of usefulness such as interpretability and fairness that we have not considered in our experimental evaluation. While prior work (Steenbrugge et al., 2018; Laversanne-Finot et al., 2018; Nair et al., 2018; Higgins et al., 2017b, 2018b) successfully applied disentanglement methods such as  $\beta$ -VAE on a variety of downstream tasks, it is not clear to us that these approaches and trained models performed well *because of disentanglement*. Finally, we remark that disentanglement is mostly about *how* the information is stored in the representation. Tasks that explicitly rely on this structure are likely to benefit more from disentanglement rather than the ones considered in this paper. Notable examples are applications in fairness (Locatello et al., 2019) and abstract visual reasoning (van Steenkiste et al., 2019). In the former, the authors show that disentanglement can be used to isolate the effect of unobserved sensitive variables to limit their negative impact to the downstream prediction. In the latter, the authors show compelling evidence that disentanglement is useful for abstract visual reasoning tasks in terms of sample complexity. We remark that the benefits Locatello et al. (2019) and van Steenkiste et al. (2019) observed are specific to some of the notions of disentanglement considered in this paper, such as DCI Disentanglement and FactorVAE.

## 8. Conclusions

In this work we first theoretically show that the unsupervised learning of disentangled representations is fundamentally impossible without inductive biases. We then performed a large-scale empirical study with six state-of-the-art disentanglement methods, seven disentanglement metrics on eight data sets and conclude the following: (i) A factorizing aggregated posterior (which is sampled) does not seem to necessarily imply that the dimensions in the representation (which is taken to be the mean) are uncorrelated. (ii) Random seeds and hyperparameters seem to matter more than the model but tuning seem to require supervision. (iii) The different evaluation metrics do measure the same notion of disentanglement and have different biases in their estimation. (iv) We did not observe that increased disentanglement necessarily implies a decreased sample complexity of learning downstream tasks. Based on these findings, we suggest three main directions for future research:

### 8.1 Inductive Biases and Implicit and Explicit Supervision

Our theoretical impossibility result in Section 3 highlights the need of inductive biases while our experimental results indicate that the role of supervision is crucial. As currently there does not seem to exist a reliable strategy to choose hyperparameters in the unsupervised learning of disentangled representations, we argue that future work should make the role of inductive biases and implicit and explicit supervision more explicit. Recent work (Duan et al., 2019) proposed a stability based heuristic for unsupervised model selection while (Locatello et al., 2020b) explored the few-labels regime. Further exploring these techniques may help us understand the practical role of inductive biases and implicit/explicit supervision.

On the other hand, we would encourage and motivate future work on disentangled representation learning that deviates from the static, purely unsupervised setting considered in this work. Promising settings (that have been explored to some degree) seem to be for example (i) disentanglement learning with interactions (Thomas et al., 2017), (ii) when weak forms of supervision like grouping information are available (Bouchacourt et al., 2018; Shu et al., 2020; Hosoya, 2019; Locatello et al., 2020a), or (iii) when temporal structure is available for the learning problem (Locatello et al.,

2020a). The last setting seems to be particularly interesting given recent identifiability results in non-linear ICA (Hyvarinen and Morioka, 2016) that enable semi-supervised Sorrenson et al. (2020); Khemakhem et al. (2020) and weakly-supervised approaches Bouchacourt et al. (2018); Hosoya (2019); Shu et al. (2020); Locatello et al. (2020a).

## 8.2 Concrete Practical Benefits of Disentangled Representations

In our experiments we investigated whether higher disentanglement scores lead to increased sample efficiency for downstream tasks and did not find evidence that this is the case. Note that these results only apply to the setting and downstream task used in our study. However, recent work (Locatello et al., 2019; van Steenkiste et al., 2019) shows compelling evidence supporting the usefulness of some notions of disentangled representations. On some tasks, the structure of the representation may indeed play an important role. A clear example is (van Steenkiste et al., 2019), where the task involves reasoning about the factors of variation in a sequence of images. Interpretability and fairness (Locatello et al., 2019) as well as interactive settings seem to be particularly promising candidates. One potential approach to include inductive biases, offer interpretability, and generalization is the concept of independent causal mechanisms and the framework of causal inference (Pearl, 2009; Peters et al., 2017). However, as the different scores considered in this paper measure different notions of disentanglement, it appears to be important to understand which benefits each specific notion may bring.

## 8.3 Experimental Setup and Diversity of Data Sets.

Our study also highlights the need for a sound, robust, and reproducible experimental setup on a diverse set of data sets in order to draw valid conclusions. We have observed that it is easy to draw spurious conclusions from experimental results if one only considers a subset of methods, metrics and data sets. Hence, we argue that it is crucial for future work to perform experiments on a wide variety of data sets to see whether conclusions and insights are generally applicable. This is particularly important in the setting of disentanglement learning as experiments are largely performed on toy-like data sets. Furthermore, as the considered metrics are measuring different notions of disentanglement, it is important for future work to be explicit about the properties of the learned representation and how these properties are being evaluated. For this reason, we released `disentanglement_lib`, the library we created to train and evaluate the different disentanglement methods and metrics on multiple data sets. We also released more than 10 000 trained models to provide a solid baseline for future research.

## Acknowledgments

The authors thank Irina Higgins, Ilya Tolstikhin, Paul Rubenstein and Josip Djolonga for helpful discussions and comments. This research was partially supported by the Max Planck ETH Center for Learning Systems, by an ETH core grant (to Gunnar Rätsch) and a Google Ph.D. Fellowship to FL. This work was partially done while FL was at Google Research Zurich and at the Max Planck Institute for Intelligent Systems.

## Appendix A. Proof of Theorem 1

**Proof** To show the claim, we explicitly construct a family of functions  $f$  using a sequence of bijective functions. Let  $d > 1$  be the dimensionality of the latent variable  $\mathbf{z}$  and consider the function  $g : \text{supp}(\mathbf{z}) \rightarrow [0, 1]^d$  defined by

$$g_i(\mathbf{v}) = P(\mathbf{z}_i \leq v_i) \quad \forall i = 1, 2, \dots, d.$$

Since  $P$  admits a density  $p(\mathbf{z}) = \prod_i p(\mathbf{z}_i)$ , the function  $g$  is bijective and, for almost every  $\mathbf{v} \in \text{supp}(\mathbf{z})$ , it holds that  $\frac{\partial g_i(\mathbf{v})}{\partial v_i} \neq 0$  for all  $i$  and  $\frac{\partial g_i(\mathbf{v})}{\partial v_j} = 0$  for all  $i \neq j$ . Furthermore, it is easy to see that, by construction,  $g(\mathbf{z})$  is a independent  $d$ -dimensional uniform distribution. Similarly, consider the function  $h : (0, 1]^d \rightarrow \mathbb{R}^d$  defined by

$$h_i(\mathbf{v}) = \psi^{-1}(v_i) \quad \forall i = 1, 2, \dots, d,$$

where  $\psi(\cdot)$  denotes the cumulative density function of a standard normal distribution. Again, by definition,  $h$  is bijective with  $\frac{\partial h_i(\mathbf{v})}{\partial v_i} \neq 0$  for all  $i$  and  $\frac{\partial h_i(\mathbf{v})}{\partial v_j} = 0$  for all  $i \neq j$ . Furthermore, the random variable  $h(g(\mathbf{z}))$  is a  $d$ -dimensional standard normal distribution.

Let  $\mathbf{A} \in \mathbb{R}^{d \times d}$  be an arbitrary orthogonal matrix with  $A_{ij} \neq 0$  for all  $i = 1, 2, \dots, d$  and  $j = 1, 2, \dots, d$ . An infinite family of such matrices can be constructed using a Householder transformation: Choose an arbitrary  $\alpha \in (0, 0.5)$  and consider the vector  $\mathbf{v}$  with  $v_1 = \sqrt{\alpha}$  and  $v_i = \sqrt{\frac{1-\alpha}{d-1}}$  for  $i = 2, 3, \dots, d$ . By construction, we have  $\mathbf{v}^T \mathbf{v} = 1$  and both  $v_i \neq 0$  and  $v_i \neq \sqrt{\frac{1}{2}}$  for all  $i = 1, 2, \dots, d$ . Define the matrix  $\mathbf{A} = \mathbf{I}_d - 2\mathbf{v}\mathbf{v}^T$  and note that  $A_{ii} = 1 - 2v_i^2 \neq 0$  for all  $i = 1, 2, \dots, d$  as well as  $A_{ij} = -v_i v_j \neq 0$  for all  $i \neq j$ . Furthermore,  $\mathbf{A}$  is orthogonal since

$$\mathbf{A}^T \mathbf{A} = (\mathbf{I}_d - 2\mathbf{v}\mathbf{v}^T)^T (\mathbf{I}_d - 2\mathbf{v}\mathbf{v}^T) = \mathbf{I}_d - 4\mathbf{v}\mathbf{v}^T + 4\mathbf{v}(\mathbf{v}^T \mathbf{v})\mathbf{v}^T = \mathbf{I}_d.$$

Since  $\mathbf{A}$  is orthogonal, it is invertible and thus defines a bijective linear operator. The random variable  $\mathbf{A}h(g(\mathbf{z})) \in \mathbb{R}^d$  is hence an independent, multivariate standard normal distribution since the covariance matrix  $\mathbf{A}^T \mathbf{A}$  is equal to  $\mathbf{I}_d$ .

Since  $h$  is bijective, it follows that  $h^{-1}(\mathbf{A}h(g(\mathbf{z})))$  is an independent  $d$ -dimensional uniform distribution. Define the function  $f : \text{supp}(\mathbf{z}) \rightarrow \text{supp}(\mathbf{z})$

$$f(\mathbf{u}) = g^{-1}(h^{-1}(\mathbf{A}h(g(\mathbf{u}))))$$

and note that by definition  $f(\mathbf{z})$  has the same marginal distribution as  $\mathbf{z}$  under  $P$ , i.e.,  $P(\mathbf{z} \leq \mathbf{u}) = P(f(\mathbf{z}) \leq \mathbf{u})$  for all  $\mathbf{u}$ . Finally, for almost every  $\mathbf{u} \in \text{supp}(\mathbf{z})$ , it holds that

$$\frac{\partial f_i(\mathbf{u})}{\partial u_j} = \frac{A_{ij} \cdot \frac{\partial h_j(g(\mathbf{u}))}{\partial v_j} \cdot \frac{\partial g_j(\mathbf{u})}{\partial u_j}}{\frac{\partial h_i(h_i^{-1}(\mathbf{A}h(g(\mathbf{u}))))}{\partial v_i} \cdot \frac{\partial g_i(g^{-1}(h^{-1}(\mathbf{A}h(g(\mathbf{u}))))}{\partial v_i}} \neq 0,$$

as claimed. Since the choice of the matrix  $\mathbf{A}$  was arbitrary, there exists an infinite family of such functions  $f$ . ■

## Appendix B. Experimental Conditions and Guiding Principles.

In our study, we seek controlled, fair and reproducible experimental conditions. We consider the case in which we can sample from a well defined and known ground-truth generative model by first sampling the factors of variations from a distribution  $P(\mathbf{z})$  and then sampling an observation from  $P(\mathbf{x}|\mathbf{z})$ . Our experimental protocol works as follows: During training, we only observe the samples of  $\mathbf{x}$  obtained by marginalizing  $P(\mathbf{x}|\mathbf{z})$  over  $P(\mathbf{z})$ . After training, we obtain a representation  $r(\mathbf{x})$  by either taking a sample from the probabilistic encoder  $Q(\mathbf{z}|\mathbf{x})$  or by taking its mean. Typically, disentanglement metrics consider the latter as the representation  $r(\mathbf{x})$ . During the evaluation, we assume to have access to the whole generative model: we can draw samples from both  $P(\mathbf{z})$  and  $P(\mathbf{x}|\mathbf{z})$ . In this way, we can perform interventions on the latent factors as required by certain evaluation metrics. We explicitly note that we effectively consider the statistical learning problem where we optimize the loss and the metrics on the known data generating distribution. As a result, we do not use separate train and test sets but always take i.i.d. samples from the known ground-truth distribution. This is justified as the statistical problem is well defined and it allows us to remove the additional complexity of dealing with overfitting and empirical risk minimization.

## Appendix C. Limitations of Our Study.

While we aim to provide a useful and fair experimental study, there are clear limitations to the conclusions that can be drawn from it due to design choices that we have taken. In all these choices, we have aimed to capture what is considered the state-of-the-art inductive bias in the community.

On the data set side, we only consider images with a heavy focus on synthetic images. We do not explore other modalities and we only consider the toy scenario in which we have access to a data generative process with uniformly distributed factors of variations. Furthermore, all our data sets have a small number of independent discrete factors of variations without any confounding variables.

For the methods, we only consider the inductive bias of convolutional architectures. We do not test fully connected architectures or additional techniques such as skip connections. Furthermore, we do not explore different activation functions, reconstruction losses or different number of layers. We also do not vary any other hyperparameters other than the regularization weight. In particular, we do not evaluate the role of different latent space sizes, optimizers and batch sizes. We do not test the sample efficiency of the metrics but simply set the size of the train and test set to large values.

Implementing the different disentanglement methods and metrics has proven to be a difficult endeavour. Few “official” open source implementations are available and there are many small details to consider. We take a best-effort approach to these implementations and implemented all the methods and metrics from scratch as any sound machine learning practitioner might do based on the original papers. When taking different implementation choices than the original papers, we explicitly state and motivate them.

## Appendix D. Differences with Previous Implementations.

As described above, we use a single choice of architecture, batch size and optimizer for all the methods which might deviate from the settings considered in the original papers. However, we argue that unification of these choices is the only way to guarantee a fair comparison among the different methods such that valid conclusions may be drawn in between methods. The largest change is that for DIP-VAE and for  $\beta$ -TCVAE we used a batch size of 64 instead of 400 and 2048 respectively.

Table 2: Encoder and Decoder architecture for the main experiment.

Encoder	Decoder
Input: $64 \times 64 \times$ number of channels	Input: $\mathbb{R}^{10}$
$4 \times 4$ conv, 32 ReLU, stride 2	FC, 256 ReLU
$4 \times 4$ conv, 32 ReLU, stride 2	FC, $4 \times 4 \times 64$ ReLU
$4 \times 4$ conv, 64 ReLU, stride 2	$4 \times 4$ upconv, 64 ReLU, stride 2
$4 \times 4$ conv, 64 ReLU, stride 2	$4 \times 4$ upconv, 32 ReLU, stride 2
FC 256, F2 $2 \times 10$	$4 \times 4$ upconv, 32 ReLU, stride 2
	$4 \times 4$ upconv, number of channels, stride 2

However, [Chen et al. \(2018\)](#) shows in Section H.2 of the Appendix that the bias in the mini-batch estimation of the total correlation does not significantly affect the performances of their model even with small batch sizes. For DIP-VAE-II, we did not implement the additional regularizer on the third order central moments since no implementation details are provided and since this regularizer is only used on specific data sets.

Our implementations of the disentanglement metrics deviate from the implementations in the original papers as follows: First, we strictly enforce that all factors of variations are treated as discrete variables as this corresponds to the assumed ground-truth model in all our data sets. Hence, we used classification instead of regression for the SAP score and the disentanglement score of ([Eastwood and Williams, 2018](#)). This is important as it does not make sense to use regression on true factors of variations that are discrete (for example on shape on dSprites). Second, wherever possible, we resorted to using the default, well-tested Scikit-learn ([Pedregosa et al., 2011](#)) implementations instead of using custom implementations with potentially hard to set hyperparameters. Third, for the Mutual Information Gap ([Chen et al., 2018](#)), we estimate the *discrete* mutual information (as opposed to continuous) on the *mean* representation (as opposed to sampled) on a *subset* of the samples (as opposed to the whole data set). We argue that this is the correct choice as the mean is usually taken to be the representation. Hence, it would be wrong to consider the full Gaussian encoder or samples thereof as that would correspond to a different representation. Finally, we fix the number of sampled train and test points across all metrics to a large value to ensure robustness.

## Appendix E. Main Experiment Hyperparameters

In our study, we fix all hyperparameters except one per each model. Model specific hyperparameters can be found in Table 3. The common architecture is depicted in Table 2 along with the other fixed hyperparameters in Table 4a. For the discriminator in FactorVAE we use the architecture in Table 4b with hyperparameters in Table 4c. All the hyperparameters for which we report single values were not varied and are selected based on the literature.

## Appendix F. Data Sets and Preprocessing

All the data sets contains images with pixels between 0 and 1. **Color-dSprites:** Every time we sample a point, we also sample a random scaling for each channel uniformly between 0.5 and 1. **Noisy-dSprites:** Every time we sample a point, we fill the background with uniform noise. **Scream-**

Table 3: Model’s hyperparameters. We allow a sweep over a single hyperparameter for each model.

Model	Parameter	Values
$\beta$ -VAE	$\beta$	[1, 2, 4, 6, 8, 16]
AnnealedVAE	$c_{max}$	[5, 10, 25, 50, 75, 100]
	iteration threshold	100000
	$\gamma$	1000
FactorVAE	$\gamma$	[10, 20, 30, 40, 50, 100]
DIP-VAE-I	$\lambda_{od}$	[1, 2, 5, 10, 20, 50]
	$\lambda_d$	$10\lambda_{od}$
DIP-VAE-II	$\lambda_{od}$	[1, 2, 5, 10, 20, 50]
	$\lambda_d$	$\lambda_{od}$
$\beta$ -TCVAE	$\beta$	[1, 2, 4, 6, 8, 10]

Table 4: Other fixed hyperparameters.

Parameter	Values	Discriminator
Batch size	64	FC, 1000 leaky ReLU
Latent space dimension	10	FC, 1000 leaky ReLU
Optimizer	Adam	FC, 1000 leaky ReLU
Adam: beta1	0.9	FC, 1000 leaky ReLU
Adam: beta2	0.999	FC, 1000 leaky ReLU
Adam: epsilon	1e-8	FC, 1000 leaky ReLU
Adam: learning rate	0.0001	FC, 1000 leaky ReLU
Decoder type	Bernoulli	FC, 2
Training steps	300000	

(a) Hyperparameters common to each of the considered methods.

(b) Architecture for the discriminator in FactorVAE.

Parameter	Values
Batch size	64
Optimizer	Adam
Adam: beta1	0.5
Adam: beta2	0.9
Adam: epsilon	1e-8
Adam: learning rate	0.0001

(c) Parameters for the discriminator in FactorVAE.

**dSprites:** Every time we sample a point, we sample a random  $64 \times 64$  patch of *The Scream* painting. We then change the color distribution by adding a random uniform number to each channel and divide the result by two. Then, we embed the dSprites shape by inverting the colors of each of its pixels.



## Appendix G. Additional Figures

In this section, we report additional figures complementing the experiments in the main text. In Figures 29 and 30, we report the same plot of Figures 2 and 3 including the AnnealedVAE method.

In Figures 31 and 32 we observed a trend similar to Figures 2 and 3 if we consider the distance from diagonal of the matrix encoding the pairwise mutual information between factors of variation and codes instead of the total correlation.

In Table 5, we report the variance per data set explained by the objective only (a) and both objective and hyperparameters (b).

In Figure 33, we plot the distribution of the total correlation of the mean representation of each method for different regularization strengths on the different data sets. Overall, we note that the different hyperparameters settings produce representations whose total correlation significantly overlaps. This trend is comparable to what we observed in Figure 6 for the disentanglement scores on Cars3D.

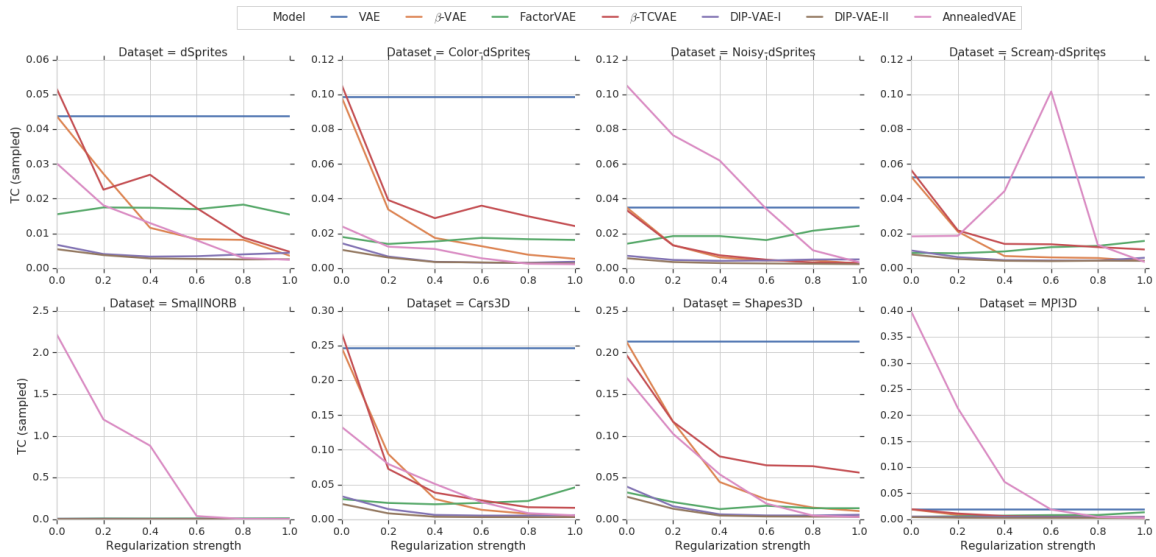


Figure 29: Total correlation of sampled representation plotted against regularization strength for different data sets and approaches (including AnnealedVAE).

	A	B	C	D	E	F	G
Cars3D	1%	38%	26%	78%	34%	35%	8%
Color-dSprites	30%	39%	50%	75%	25%	24%	28%
MPI3D	61%	59%	50%	78%	45%	45%	21%
Noisy-dSprites	17%	21%	18%	78%	10%	43%	9%
Scream-dSprites	90%	50%	78%	54%	45%	61%	55%
Shapes3D	33%	21%	14%	43%	21%	27%	10%
SmallNORB	68%	73%	60%	87%	72%	62%	57%
dSprites	31%	43%	49%	71%	27%	30%	33%

(a) Percentage of variance explained regressing the disentanglement scores on the different data sets from the objective function only.

	A	B	C	D	E	F	G
Cars3D	5%	67%	41%	97%	60%	49%	13%
Color-dSprites	68%	80%	59%	92%	75%	40%	56%
MPI3D	91%	81%	80%	94%	56%	62%	44%
Noisy-dSprites	27%	42%	25%	87%	29%	53%	22%
Scream-dSprites	93%	74%	84%	83%	66%	68%	75%
Shapes3D	61%	79%	53%	82%	57%	48%	33%
SmallNORB	87%	90%	82%	95%	89%	73%	78%
dSprites	64%	77%	55%	90%	71%	38%	57%

(b) Percentage of variance explained regressing the disentanglement scores on the different data sets from the Cartesian product of objective function and regularization strength.

Table 5: Variance of the disentanglement scores explained by the objective function or its cartesian product with the hyperparameters. The variance explained is computed regressing using ordinary least squares. Legend: A = BetaVAE Score, B = DCI Disentanglement, C = FactorVAE Score, D = IRS, E = MIG, F = Modularity, G = SAP.

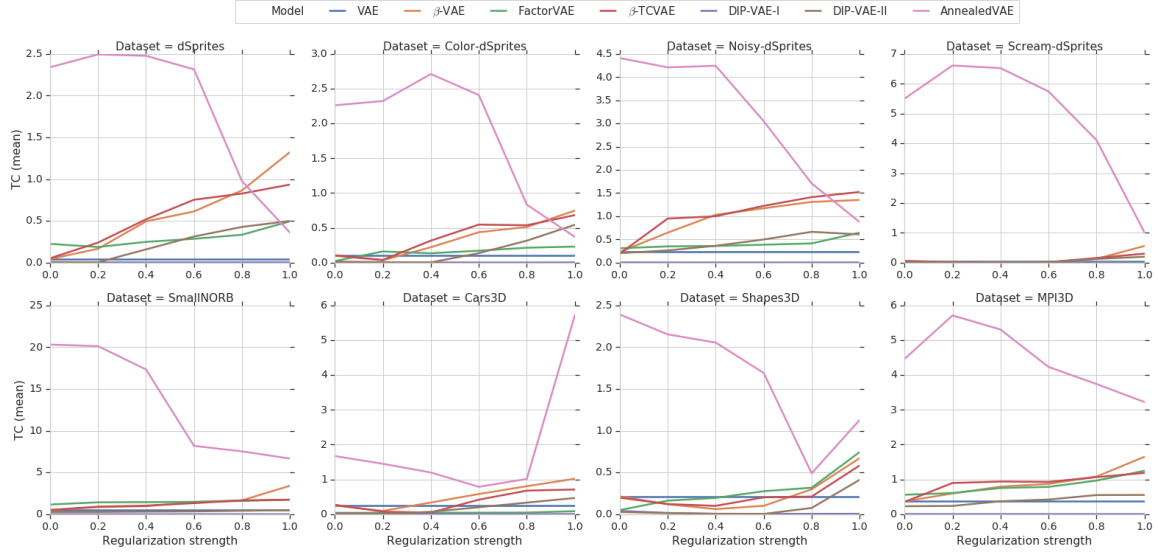


Figure 30: Total correlation of mean representation plotted against regularization strength for different data sets and approaches (including AnnealedVAE).

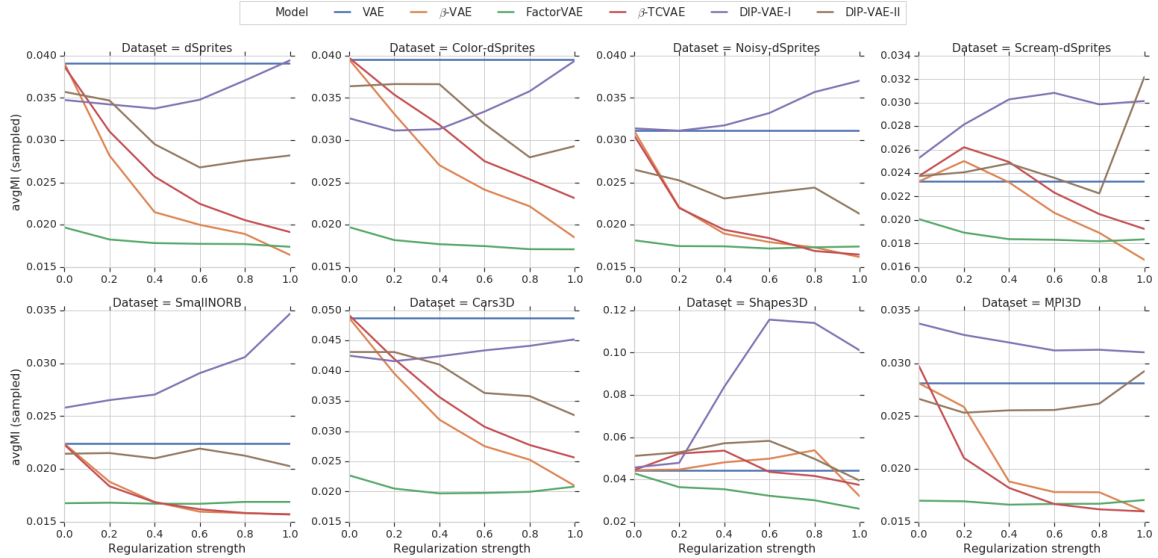


Figure 31: The average mutual information of the dimensions of the sampled representation generally decrease except for DIP-VAE-I.

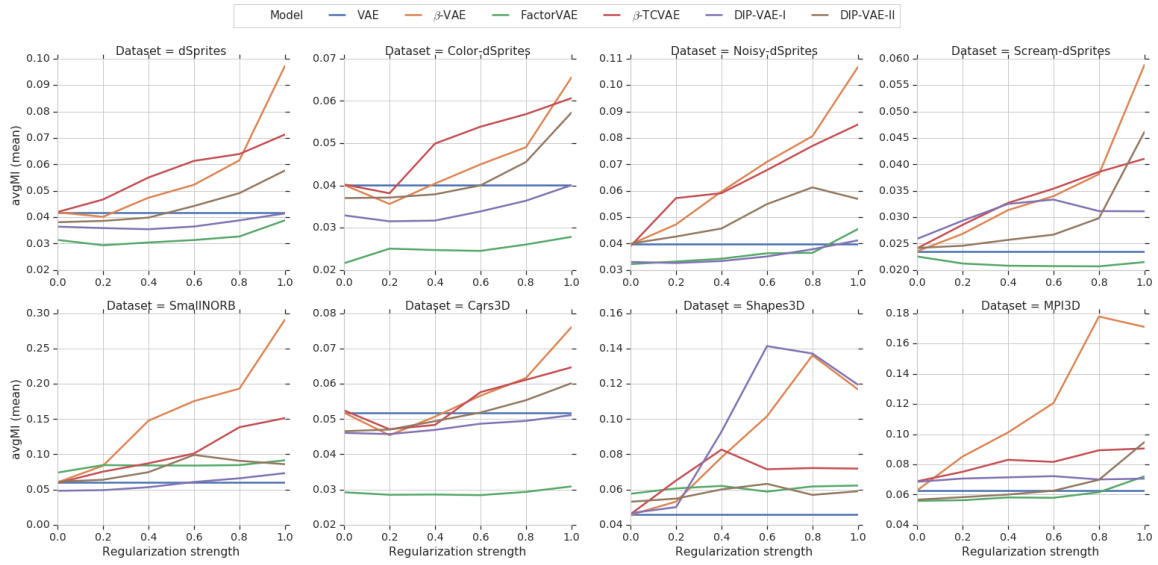


Figure 32: The average mutual information of the dimensions of the mean representation generally increase.

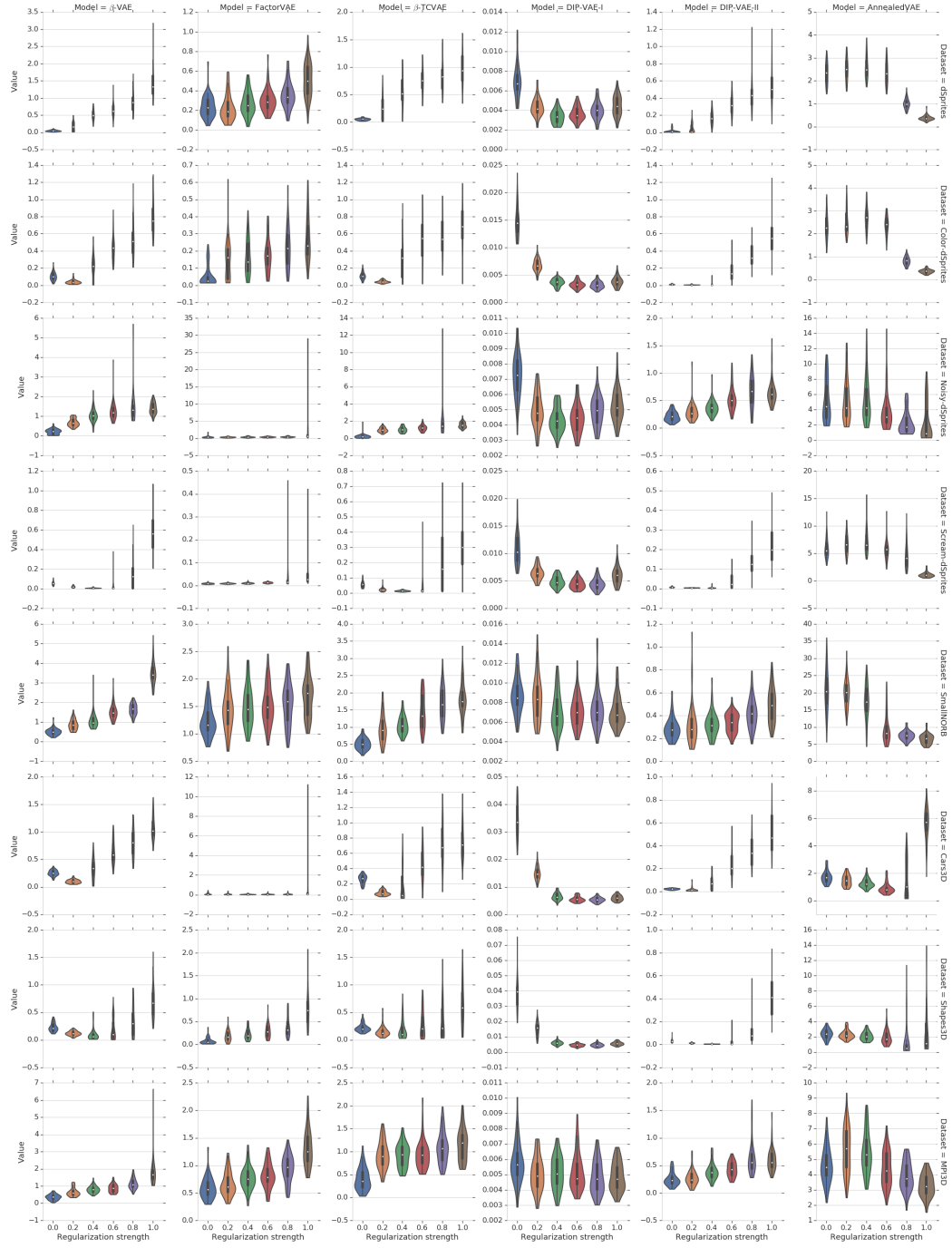


Figure 33: The effect of randomness on the total correlation of the mean representation for each method. We observe an overlap between the different hyperparameters settings similar to what we observed in Figure 6 for the disentanglement metrics on Cars3D.

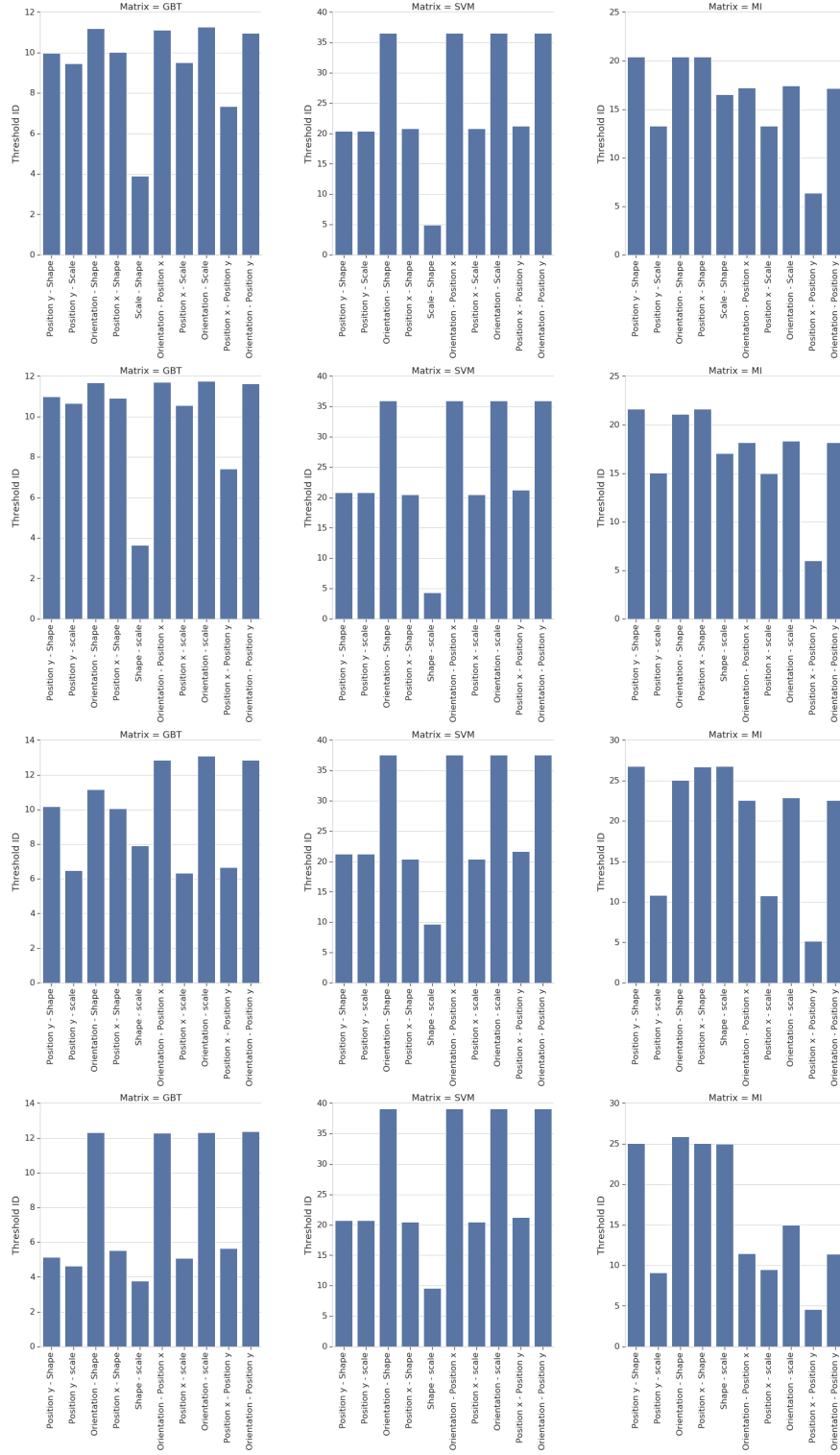


Figure 34: Threshold ID of confused factors for dSprites, Color-dSprites, Noisy-dSprites and Scream-dSprites. Lower threshold means that the two factors are found more entangled.



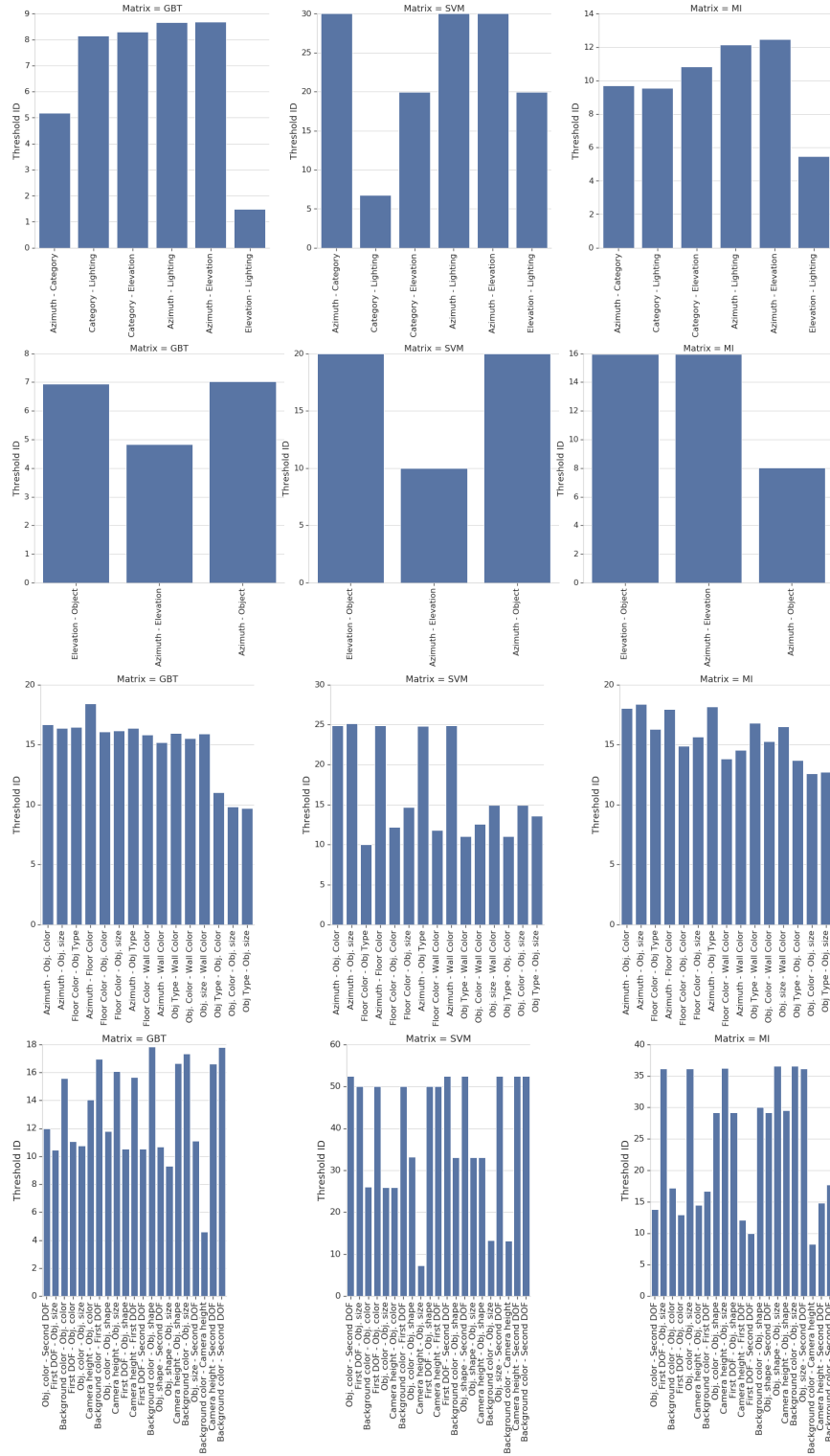


Figure 35: Threshold ID of confused factors for SmallNORB, Cars3D, Shapes3D and MPI3D. Lower threshold means that the two factors are found more entangled.

## References

- Miguel A Arcones and Evarist Gine. On the bootstrap of  $u$  and  $v$  statistics. *The Annals of Statistics*, pages 655–674, 1992.
- Francis Bach and Michael Jordan. Kernel independent component analysis. *Journal of Machine Learning Research*, 3(7):1–48, 2002.
- Yoshua Bengio, Yann LeCun, et al. Scaling learning algorithms towards AI. *Large-scale Kernel Machines*, 34(5):1–41, 2007.
- Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.
- Diane Bouchacourt, Ryota Tomioka, and Sebastian Nowozin. Multi-level variational autoencoder: Learning disentangled representations from grouped observations. In *AAAI Conference on Artificial Intelligence*, 2018.
- Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in beta-VAE. *arXiv preprint arXiv:1804.03599*, 2018.
- Tian Qi Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, 2018.
- Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2016.
- Brian Cheung, Jesse A Livezey, Arjun K Bansal, and Bruno A Olshausen. Discovering hidden factors of variation in deep networks. *arXiv preprint arXiv:1412.6583*, 2014.
- Taco Cohen and Max Welling. Learning the irreducible representations of commutative lie groups. In *International Conference on Machine Learning*, 2014a.
- Taco S Cohen and Max Welling. Transformation properties of learned visual representations. *arXiv preprint arXiv:1412.7659*, 2014b.
- Pierre Comon. Independent component analysis, a new concept? *Signal Processing*, 36(3):287–314, 1994.
- Zhiwei Deng, Rajitha Navarathna, Peter Carr, Stephan Mandt, Yisong Yue, Iain Matthews, and Greg Mori. Factorized variational autoencoders for modeling audience reactions to movies. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
- Emily L Denton and Vighnesh Birodkar. Unsupervised learning of disentangled representations from video. In *Advances in Neural Information Processing Systems*, 2017.
- Guillaume Desjardins, Aaron Courville, and Yoshua Bengio. Disentangling factors of variation via generative entangling. *arXiv preprint arXiv:1210.5474*, 2012.

- Sunny Duan, Nicholas Watters, Loic Matthey, Christopher P Burgess, Alexander Lerchner, and Irina Higgins. A heuristic for unsupervised model selection for variational disentangled representation learning. *arXiv preprint arXiv:1905.12614*, 2019.
- Cian Eastwood and Christopher KI Williams. A framework for the quantitative evaluation of disentangled representations. In *International Conference on Learning Representations*, 2018.
- Vincent Fortuin, Matthias Hüser, Francesco Locatello, Heiko Strathmann, and Gunnar Rätsch. Deep self-organization: Interpretable discrete representation learning on time series. In *International Conference on Learning Representations*, 2019.
- Marco Fraccaro, Simon Kamronn, Ulrich Paquet, and Ole Winther. A disentangled recognition and nonlinear dynamics model for unsupervised learning. In *Advances in Neural Information Processing Systems*, 2017.
- Muhammad Waleed Gondal, Manuel Wüthrich, Djordje Miladinović, Francesco Locatello, Martin Breidt, Valentin Volchkov, Joel Akpo, Olivier Bachem, Bernhard Schölkopf, and Stefan Bauer. On the transfer of inductive bias from simulation to the real world: a new disentanglement dataset. In *Advances in Neural Information Processing Systems*, 2019.
- Ian Goodfellow, Honglak Lee, Quoc V Le, Andrew Saxe, and Andrew Y Ng. Measuring invariances in deep networks. In *Advances in Neural Information Processing Systems*, 2009.
- Ross Goroshin, Michael F Mathieu, and Yann LeCun. Learning to linearize under uncertainty. In *Advances in Neural Information Processing Systems*, 2015.
- Luigi Gresele, Paul K. Rubenstein, Arash Mehrjou, Francesco Locatello, and Bernhard Schölkopf. The incomplete rosetta stone problem: Identifiability results for multi-view nonlinear ica. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2019.
- Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017a.
- Irina Higgins, Arka Pal, Andrei Rusu, Loic Matthey, Christopher Burgess, Alexander Pritzel, Matthew Botvinick, Charles Blundell, and Alexander Lerchner. Darla: Improving zero-shot transfer in reinforcement learning. In *International Conference on Machine Learning*, 2017b.
- Irina Higgins, David Amos, David Pfau, Sebastien Racaniere, Loic Matthey, Danilo Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. *arXiv preprint arXiv:1812.02230*, 2018a.
- Irina Higgins, Nicolas Sonnerat, Loic Matthey, Arka Pal, Christopher P Burgess, Matko Bošnjak, Murray Shanahan, Matthew Botvinick, Demis Hassabis, and Alexander Lerchner. Scan: Learning hierarchical compositional visual concepts. In *International Conference on Learning Representations*, 2018b.
- Geoffrey E Hinton, Alex Krizhevsky, and Sida D Wang. Transforming auto-encoders. In *International Conference on Artificial Neural Networks*, 2011.

- Haruo Hosoya. Group-based learning of disentangled representations with generalizability for novel contents. In *International Joint Conference on Artificial Intelligence*, pages 2506–2513, 2019.
- Jun-Ting Hsieh, Bingbin Liu, De-An Huang, Li F Fei-Fei, and Juan Carlos Niebles. Learning to decompose and disentangle representations for video prediction. In *Advances in Neural Information Processing Systems*, 2018.
- Wei-Ning Hsu, Yu Zhang, and James Glass. Unsupervised learning of disentangled and interpretable representations from sequential data. In *Advances in Neural Information Processing Systems*, 2017.
- Aapo Hyvarinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. In *Advances in Neural Information Processing Systems*, 2016.
- Aapo Hyvärinen and Petteri Pajunen. Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 1999.
- Aapo Hyvarinen, Hiroaki Sasaki, and Richard E Turner. Nonlinear ica using auxiliary variables and generalized contrastive learning. In *International Conference on Artificial Intelligence and Statistics*, 2019.
- Christian Jutten and Juha Karhunen. Advances in nonlinear blind source separation. In *International Symposium on Independent Component Analysis and Blind Signal Separation*, pages 245–256, 2003.
- Theofanis Karaletsos, Serge Belongie, and Gunnar Rätsch. Bayesian representation learning with oracle constraints. *arXiv preprint arXiv:1506.05011*, 2015.
- Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217, 2020.
- Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International Conference on Machine Learning*, 2018.
- Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.
- Tejas D Kulkarni, William F Whitney, Pushmeet Kohli, and Josh Tenenbaum. Deep convolutional inverse graphics network. In *Advances in Neural Information Processing Systems*, 2015.
- Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations. In *International Conference on Learning Representations*, 2018.
- Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40, 2017.
- Adrien Laversanne-Finot, Alexandre Pere, and Pierre-Yves Oudeyer. Curiosity driven exploration of learned disentangled goal spaces. In *Conference on Robot Learning*, 2018.

- Yann LeCun, Fu Jie Huang, and Leon Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2004.
- Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436, 2015.
- Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.
- Francesco Locatello, Damien Vincent, Ilya Tolstikhin, Gunnar Rätsch, Sylvain Gelly, and Bernhard Schölkopf. Competitive training of mixtures of independent deep generative models. In *Workshop at the 6th International Conference on Learning Representations (ICLR)*, 2018.
- Francesco Locatello, Gabriele Abbati, Tom Rainforth, Stefan Bauer, Bernhard Schölkopf, and Olivier Bachem. On the fairness of disentangled representations. In *Advances in Neural Information Processing Systems*, 2019.
- Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *International Conference on Machine Learning*, 2020a.
- Francesco Locatello, Michael Tschannen, Stefan Bauer, Gunnar Rätsch, Bernhard Schölkopf, and Olivier Bachem. Disentangling factors of variation using few labels. *International Conference on Learning Representations*, 2020b.
- Michael F Mathieu, Junbo J Zhao, Aditya Ramesh, Pablo Sprechmann, and Yann LeCun. Disentangling factors of variation in deep representation using adversarial training. In *Advances in Neural Information Processing Systems*, 2016.
- Edvard Munch. The scream, 1893.
- Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. In *Advances in Neural Information Processing Systems*, 2018.
- Siddharth Narayanaswamy, T Brooks Paige, Jan-Willem Van de Meent, Alban Desmaison, Noah Goodman, Pushmeet Kohli, Frank Wood, and Philip Torr. Learning disentangled representations with semi-supervised deep generative models. In *Advances in Neural Information Processing Systems*, 2017.
- XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.
- Judea Pearl. *Causality*. Cambridge University Press, 2009.
- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

- Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. MIT press, 2017.
- Scott Reed, Kihyuk Sohn, Yuting Zhang, and Honglak Lee. Learning to disentangle factors of variation with manifold interaction. In *International Conference on Machine Learning*, 2014.
- Scott Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. Deep visual analogy-making. In *Advances in Neural Information Processing Systems*, 2015.
- Karl Ridgeway and Michael C Mozer. Learning deep disentangled embeddings with the f-statistic loss. In *Advances in Neural Information Processing Systems*, 2018.
- Michal Rolínek, Dominik Zietlow, and Georg Martius. Variational autoencoders recover pca directions (by accident). In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition*, 2019.
- Jürgen Schmidhuber. Learning factorial codes by predictability minimization. *Neural Computation*, 4(6):863–879, 1992.
- Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning. In *International Conference on Machine Learning*, 2012.
- Rajen D Shah and Jonas Peters. The hardness of conditional independence testing and the generalised covariance measure. *arXiv preprint arXiv:1804.07203*, 2018.
- Rui Shu, Yining Chen, Abhishek Kumar, Stefano Ermon, and Ben Poole. Weakly supervised disentanglement with guarantees. In *International Conference on Learning Representations*, 2020.
- Peter Sorrenson, Carsten Rother, and Ullrich Köthe. Disentanglement by nonlinear ica with general incompressible-flow networks (gin). In *International Conference on Learning Representations*, 2020.
- P. Spirtes, C. Glymour, and R. Scheines. *Causation, prediction, and search*. MIT Press, 2000.
- Xander Steenbrugge, Sam Leroux, Tim Verbelen, and Bart Dhoedt. Improving generalization for abstract reasoning tasks using disentangled feature representations. In *Workshop on Relational Representation Learning at NeurIPS*, 2018.
- Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. Density-ratio matching under the bregman divergence: a unified framework of density-ratio estimation. *Annals of the Institute of Statistical Mathematics*, 64(5):1009–1044, 2012.
- Raphael Suter, Djordje Miladinović, Stefan Bauer, and Bernhard Schölkopf. Interventional robustness of deep latent variable models. In *International Conference on Machine Learning*, 2019.
- Valentin Thomas, Emmanuel Bengio, William Fedus, Jules Pondard, Philippe Beaudoin, Hugo Larochelle, Joelle Pineau, Doina Precup, and Yoshua Bengio. Disentangling the independently controllable factors of variation by interacting with the world. *Learning Disentangled Representations Workshop at NeurIPS*, 2017.



- Michael Tschannen, Olivier Bachem, and Mario Lucic. Recent advances in autoencoder-based representation learning. *arXiv preprint arXiv:1812.05069*, 2018.
- Sjoerd van Steenkiste, Francesco Locatello, Jürgen Schmidhuber, and Olivier Bachem. Are disentangled representations helpful for abstract visual reasoning? In *Advances in Neural Information Processing Systems*, 2019.
- Satosi Watanabe. Information theoretical analysis of multivariate correlation. *IBM Journal of research and development*, 4(1):66–82, 1960.
- William F Whitney, Michael Chang, Tejas Kulkarni, and Joshua B Tenenbaum. Understanding visual concepts with continuation learning. *arXiv preprint arXiv:1602.06822*, 2016.
- Jimei Yang, Scott E Reed, Ming-Hsuan Yang, and Honglak Lee. Weakly-supervised disentangling with recurrent transformations for 3D view synthesis. In *Advances in Neural Information Processing Systems*, 2015.
- Li Yingzhen and Stephan Mandt. Disentangled sequential autoencoder. In *International Conference on Machine Learning*, 2018a.
- Li Yingzhen and Stephan Mandt. Disentangled sequential autoencoder. In *International Conference on Machine Learning*, pages 5656–5665, 2018b.
- Zhenyao Zhu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Multi-view perceptron: a deep model for learning face identity and view representations. In *Advances in Neural Information Processing Systems*, 2014.