

Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations

Francesco Locatello^{2,3}, Stefan Bauer³, Mario Lucic¹, Sylvain Gelly¹, Bernhard Schölkopf³, and Olivier Bachem¹

¹Google AI, Brain team

²ETH Zurich, Dept. for Computer Science

³Max-Planck Institute for Intelligent Systems

Abstract

In recent years, the interest in *unsupervised* learning of *disentangled* representations has significantly increased. The key assumption is that real-world data is generated by a few explanatory factors of variation and that these factors can be recovered by unsupervised learning algorithms. A large number of unsupervised learning approaches based on *auto-encoding* and quantitative evaluation metrics of disentanglement have been proposed; yet, the efficacy of the proposed approaches and utility of proposed notions of disentanglement has not been challenged in prior work. In this paper, we provide a sober look on 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 more than 12 000 models covering the six most prominent methods, and evaluate them across six disentanglement metrics in a reproducible large-scale experimental study on seven different data sets. On the positive side, we observe that different methods successfully enforce properties “encouraged” by the corresponding losses. On the negative side, we observe that in our study (1) “good” hyperparameters seemingly cannot be identified without access to ground-truth labels, (2) good hyperparameters neither transfer across data sets nor across disentanglement metrics, and (3) that increased disentanglement does not seem to lead to a decreased sample complexity of learning for downstream tasks.

These 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.

1 Introduction

In representation learning it is often assumed that real-world observations \mathbf{x} (e.g. 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 (e.g. content + 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” [4].

A recent line of work has argued that representations that are *disentangled* are an important step towards a better representation learning. 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 [4]. A change in a single underlying factor of variation z_i should lead to a change in a single factor in the

This is a preliminary preprint based on our initial experimental results. Please send any comments to francesco.locatello@tuebingen.mpg.de and bachem@google.com.

learned representation $r(\mathbf{x})$. This assumption can be extended to groups of factors as e.g. in Bouchacourt et al. [5] or Suter et al. [49]. 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 [19, 27, 15, 30, 7, 44].

State-of-the-art approaches for unsupervised disentanglement learning are based on *Variational Autoencoders (VAEs)* [28]: 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 defined as the mean of the approximate posterior distribution $Q(\mathbf{z}|\mathbf{x})$. The usual choice for the prior and variational distribution are multivariate Gaussians. Several variations of VAEs were proposed with the motivation that they lead to better disentanglement [19, 6, 27, 7, 30, 45]. 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.

Our contributions. The original motivation of this work was to provide a neutral large-scale study that benchmarks different unsupervised disentanglement methods and metrics on a wide set of data sets in a fair, reproducible experimental set up. However, the empirical evidence led us to instead challenge many commonly held assumptions in this field. 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 reproducible¹ large-scale experimental study with a sound experimental protocol for unsupervised disentanglement learning. We implement from scratch six recent unsupervised disentanglement learning methods as well as six disentanglement measures and train more than 12 000 models on seven data sets.
- We evaluate our experimental results and challenge many common assumptions 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, only one method also consistently ensures that the individual dimensions of the representation (which is taken to be the mean) are not correlated. (ii) We do not find any evidence that they can be used to reliably learn disentangled representations in an *unsupervised* manner as hyper parameters seem to matter more than the model and “good” hyperparameters seemingly cannot be identified without access to ground-truth labels. Similarly, we observe that good hyperparameters neither transfer across data sets nor across disentanglement metrics. (iii) 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 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.

Roadmap. We begin by briefly reviewing the motivations for disentanglement with the current state-of-the-art methods and metrics in Section 2. In Section 3 we present and discuss the theoretical impossibility of unsupervised learning of disentangled representations. In Section 4 we discuss our experimental design with the research questions, our guiding principles, a detailed discussion of the limitation of this study and the differences with existing implementations. In Section 5, we present our the empirical results. Finally, in Section 6 we study the implications of this work.

¹Reproducing these experiments require approximately 2.54 GPU years (NVIDIA p100)

2 A review of (unsupervised) disentanglement learning

We first provide an overview of the key concepts and definitions, followed by quantitative evaluation metrics, and briefly review the proposed models based on VAEs and other related work.

2.1 What is disentanglement and why is it a desirable property?

Disentanglement is an abstract notion that has been brought back to the spotlight in the seminal paper of Bengio et al. [4]. It has been argued that an artificial intelligence able to understand and reason about the world should be able to identify and disentangle the explanatory factors of variations hidden in the data [4, 41, 33, 3, 46, 31]. A common modeling assumption is that we observe data from a high dimensional random variable \mathbf{x} generated by an underlying generative process which can be described by a low dimensional set of factors of variations. These factors can be fully observable, partially observable or not observable at all. We focus our study on this last setting as this one has recently gathered increased interest in the community.

There are many suggestions in the literature why disentangled representations could be useful: First of all, we could use them directly to build a predictive model instead of relying on the high dimensional \mathbf{x} . Indeed, they should contain all the information present in \mathbf{x} in a compact and interpretable structure [4, 29, 8]. Furthermore, they are independent from the task at hand [17, 34]. Therefore, they would be useful for (semi-)supervised learning of downstream tasks, transfer and few shot learning [4, 47, 41]. Moreover, one can use them to integrate out nuisance factors [30]. Last but not least, they enable interventions (i.e. to conditionally sample) and answer counterfactual questions (i.e. how would this sample look like had a factor been different?) [41, 39].

2.2 What are proposed definitions for disentanglement?

While there is a comprehensive list of requirements for a disentangled representation, there does not yet exist a single, commonly accepted definition. Instead, a variety of different concrete metrics have been proposed by the community where each of these metrics aims to formalize and measure the notion in a slightly different way. In general, all these methods assume that they have access to the ground-truth factors of variations if not the full ground-truth generative model. In the following, we briefly review the metrics that we consider in our study.

1. **BetaVAE metric.** Higgins et al. [19] suggest to fix a random factor of variation in the underlying generative model constant 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 difference between the representation vectors in the two mini batches.
2. **FactorVAE metric.** Kim and Mnih [27] 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.
3. **Mutual Information Gap.** Chen et al. [7] argue that the BetaVAE metric and the FactorVAE metric are neither general nor unbiased as they depend on some hyperparameters. They compare pairwise mutual information for each ground truth factor and each factor in the computed representation $r(\mathbf{x})$. For each ground-truth factor \mathbf{z}_i , they then consider the two factors in $r(\mathbf{x})$ that have the highest mutual information. The *Mutual Information Gap (MIG)* is then defined as the average, normalized difference between the representation factor with the highest and the second-highest mutual information.
4. **Modularity and Explicitness.** Ridgeway and Mozer [44] argue that two different properties of representations should be considered, i.e., *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 (i.e. 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.

5. **Disentanglement, Completeness and Informativeness.** Ridgeway and Mozer [44] consider three properties of representations, i.e., *Disentanglement*, *Completeness* and *Informativeness*. First, Eastwood and Williams [15] 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. In this paper, we mainly consider Disentanglement which we subsequently call “DCI Disentanglement” for clarity.
6. **SAP score.** Kumar et al. [30] 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.

2.3 Unsupervised learning of disentangled representations with VAEs

Variants of variational autoencoders [28] 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,

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

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.

Bottleneck capacity. Higgins et al. [19] propose the β -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. [6] argue that when the bottleneck has limited capacity, the network will be forced to specialize on the factor of variation that most contribute 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.

Penalizing the total correlation. Let $I(\mathbf{x}; \mathbf{z})$ denote the mutual information between \mathbf{x} and \mathbf{z} and note that 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 $\beta > 1$, β -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 [27] argues that penalizing $I(\mathbf{x}; \mathbf{z})$ is neither necessary nor desirable for disentanglement. Therefore, they propose the FactorVAE which augments the VAE objective with an additional regularizer that specifically penalizes dependencies between the dimensions of the representation:

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

This last term is also known as *total correlation* [51]. While this term is intractable and vanilla Monte Carlo approximations require marginalization over the training set, it can be optimized using the density ratio trick [38, 48]. Samples from $\prod_{j=1}^d q(\mathbf{z}_j)$ can be obtained shuffling samples from $q(\mathbf{z})$ [1]. Concurrently, Chen et al. [7] propose the β -TCVAE. As opposed to FactorVAE, they propose a tractable biased Monte-Carlo estimate for the total correlation.

Disentangled priors. Kumar et al. [30] 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, to 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

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

or the DIP-VAE-II objective

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

2.4 Other related work

In a similar spirit to disentanglement, (non-)linear independent component analysis [12, 2, 25, 22] 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 [12] 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 [23]. The lack of almost any identifiability result in non-linear ICA has been a main bottleneck for the utility of the approach [24] and partially motivated alternative machine learning approaches [14, 46, 11]. 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 [42, 53, 29, 26, 9, 35, 37, 55, 49]. [20, 10] 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 [18, 52, 16, 13, 21, 54]. This is not a limiting assumption especially in sequential data, i.e., for videos.

3 Impossibility of unsupervised disentanglement learning

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 $(\mathbf{z}, \mathbf{x}) \sim P$ denote any generative model which admits a density $p(\mathbf{z}) = \prod_{i=1}^d p(\mathbf{z}_i)$ and where \mathbf{z} denotes the independent latent variables and \mathbf{x} the data observations. Then, there exists an infinite family of bijective functions $f : \text{supp}(\mathbf{z}) \rightarrow \text{supp}(\mathbf{z})$ such that $P(\mathbf{z} \leq \mathbf{u}) = P(f(\mathbf{z}) \leq \mathbf{u})$ for all $\mathbf{u} \in \text{supp}(\mathbf{z})$ and $\frac{\partial f_i(\mathbf{u})}{\partial u_j} \neq 0$ almost everywhere for all i and j .*

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 prior² in the following sense: Consider any unsupervised disentanglement method and assume that it finds a representation $r(\mathbf{x})$ that is perfectly disentangled with respect to \mathbf{x} in a generative model $(\mathbf{z}, \mathbf{x}) \sim P$ with a factorizing prior on \mathbf{z} . 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 $r(\mathbf{x})$. Furthermore, since f is deterministic, both generative models have the same marginal distribution of the observations \mathbf{x} by construction. 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.

The intuition behind this result may be seen from a generative model where \mathbf{z} consists of two independent standard Gaussian random variables. Consider an alternative latent variable $\hat{\mathbf{z}}$ obtained by rotating \mathbf{z} by 45° . By definition, $\hat{\mathbf{z}}$ corresponds to two independent standard Gaussian random variables that are fully entangled with the original latent variable \mathbf{z} . Any disentanglement method can only be disentangled with respect to one of them and has to be entangled with respect to the other. We note that this result is obvious for multivariate Gaussians; however, perhaps surprisingly, Theorem 1 also holds for distributions which are not invariant to rotation, for example multivariate uniform distributions.

Theorem 1 may not be surprising to readers familiar with the causality 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 by Peters et al. [41] the right model cannot be identified only observing the distribution of \mathbf{x} . Furthermore, similar results have been obtained in the context of non-linear ICA[23].

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 implicit and investigated further as done in the following large-scale experimental study.

4 Experimental design

Research questions. Theorem 1 opens several questions on the role of inductive biases in the empirical performance of state-of-the-art models.

1. Are current methods effective at enforcing a factorizing aggregated posterior and representation?
2. How much do existing disentanglement metrics agree?
3. How important are different models and hyperparameters for disentanglement?
4. Are there reliable recipes for hyperparameter selection?
5. Are these disentangled representations useful for downstream tasks in terms of the sample complexity of learning?

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, i.e. we can draw

²Theorem 1 only applies to factorized priors; however, we expect that a similar result can be extended to non-factorizing priors.

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 scores. 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.

We consider four data sets in which \mathbf{x} is obtained as a deterministic function of \mathbf{z} : *dSprites* [19], *Cars3d* [43], *SmallNORB* [32], *Shapes3D* [27]. We also introduce three data sets in which \mathbf{x} is a stochastic function of \mathbf{z} as there are additional continuous noise variables: *Color-dSprites*, *Noisy-dSprites* and *Scream-dSprites*. This means that for a fixed realization of the factors of variations z , two samples from $P(\mathbf{x}|\mathbf{z} = z)$ can be different. The key idea behind these three additional data sets is that they have exactly the same latent factors as dSprites but that they correspond to increasingly harder problems with more complex observations. 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 [36]. The dSprites shape is embedded into the image by inverting the color of its pixels.

To fairly evaluate the different approaches, we separate the effect of the regularization from the other inductive biases (e.g., 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 net, a Bernoulli decoder and latent dimension fixed to 10. We note that these are all standard choices in prior work [19, 27].

We choose six different regularization strength, i.e., 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 suggested hyperparameters suggested by the authors).

We run each model for each regularization strength for 50 different random seeds on each data set. We fix our experimental setup in advance and we run all the methods described in Section 2.3 and evaluate them on every metric of Section 2.2 on each data set. The full details on the experimental setup are provided in the Appendix and we plan to release code to train the considered models, the trained models and the evaluation pipeline for the sake of reproducibility and to establish a strong and fair baseline.

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.

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 β -TCVAE

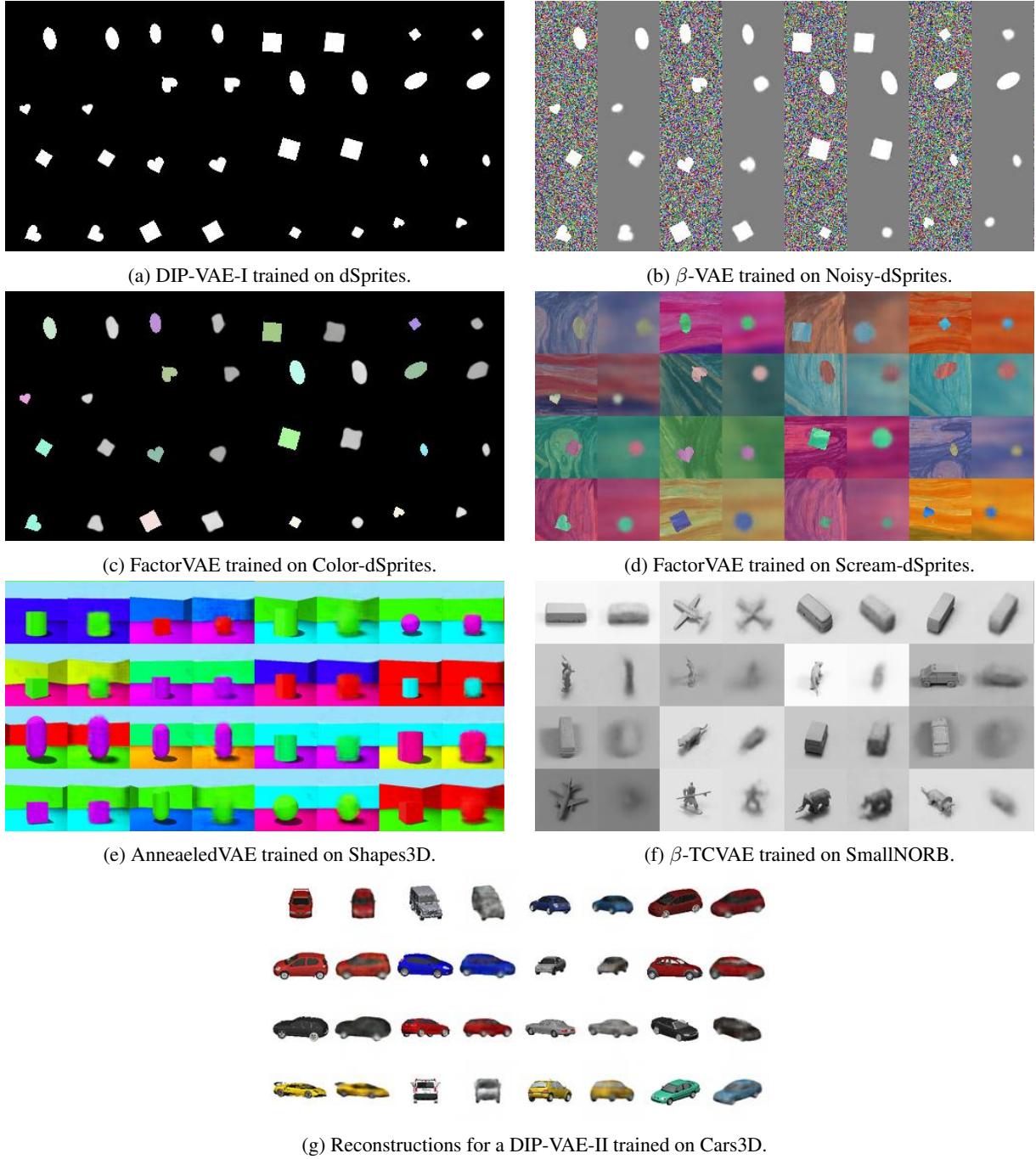


Figure 1: Reconstructions for different data sets and methods. Odd columns show real samples and even columns their reconstruction. As expected, the additional variants of dSprites with continuous noise variables are harder than the original data set. On Noisy-dSprites and Color-dSprites the models produce 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.

we used a batch size of 64 instead of 400 and 2048 respectively. However, Chen et al. [7] 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 an SVM with L_1 regularization as opposed to Lasso for the disentanglement score of [15]. This is important as it does not make sense to use regression on true factors of variations that are discrete (e.g., shape on dSprites). Second, wherever possible, we resorted to using the default, well-tested Scikit-learn [40] implementations instead of using custom implementations with potentially hard to set hyperparameters. Third, for the Mutual Information Gap [7], 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.

5 Experimental results

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. SmallNorb seems to be the most challenging data set.

5.2 Can current methods enforce a factorizing aggregated posterior and representation?

We investigate whether the considered unsupervised disentanglement approaches are effective at enforcing a factorizing aggregated posterior. For each trained model, we 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 matrix³. 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 (i.e. the β -VAE model with $\beta = 1$) enjoy the highest total correlation. For β -VAE and β -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 β) 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 β -VAE and β -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. This may seem surprising given that the FactorVAE objective aims to penalize the total correlation of the aggregated posterior. However, we expect this is due to the fact that the total correlation of the aggregated posterior is estimated using adversarial density estimation, which was shown to consistently underestimate the true TC, see Figure 7 [27]. We further observe that AnnealedVAE (shown in Figure 15 in the Appendix) is much more sensitive to the

³For numerical stability, we add 10^{-5} to the diagonal of the covariance matrix.

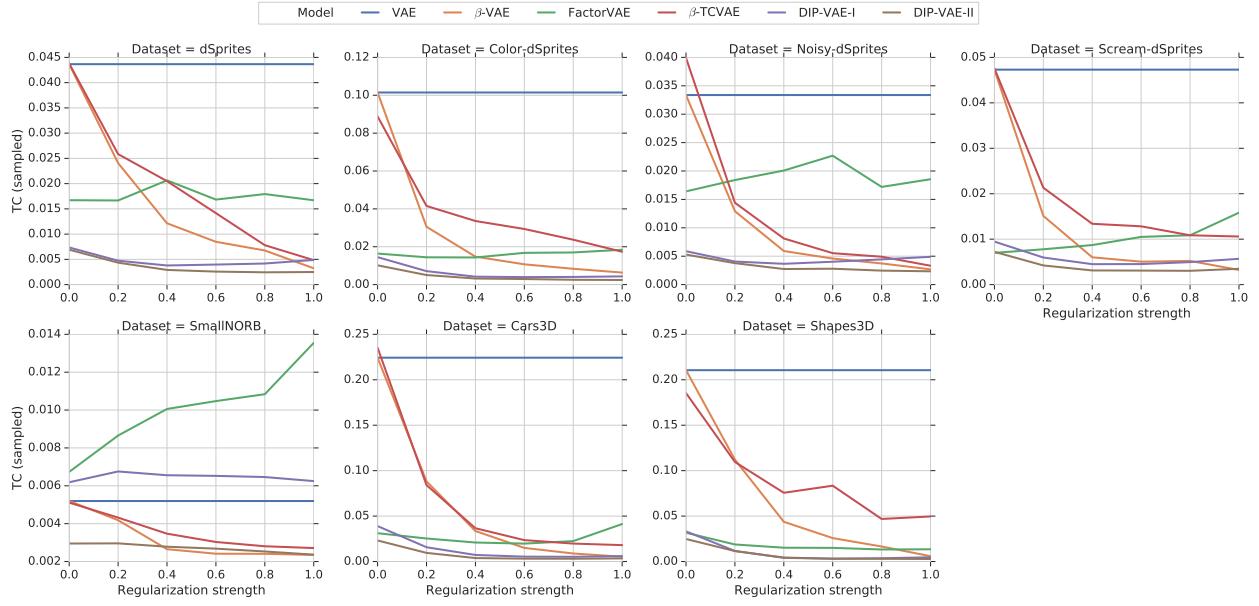


Figure 2: Total correlation of sampled representation plotted against regularization strength for different data sets and approaches (except AnnealedVAE). The total correlation of the sampled representation decreases as the regularization strength is increased.

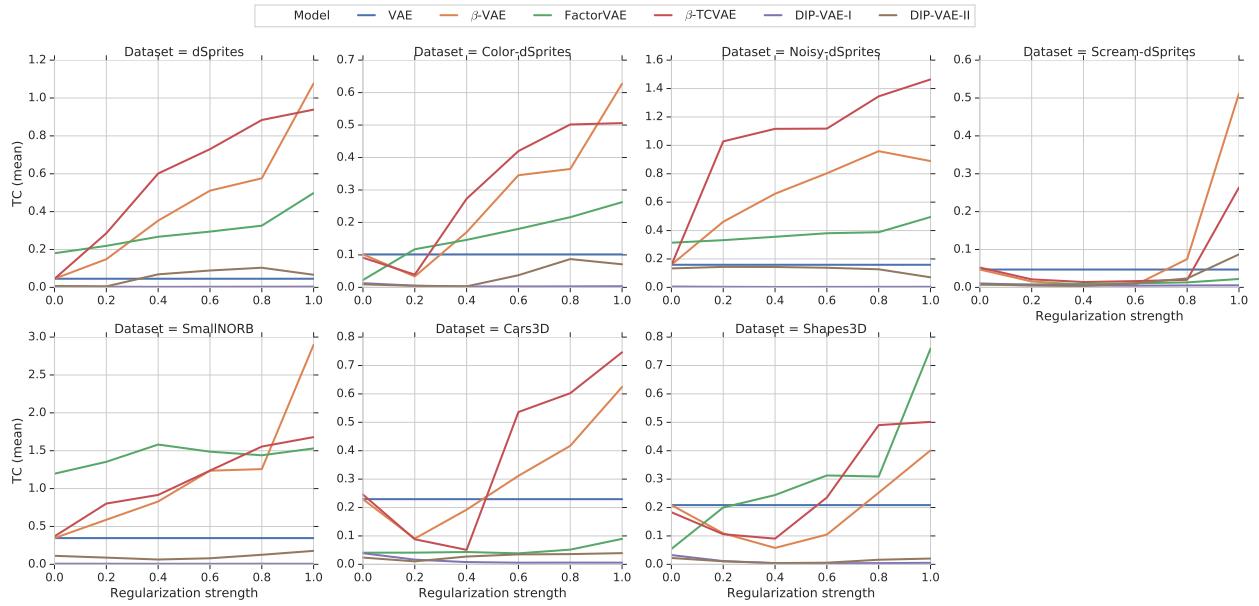


Figure 3: Total correlation of mean representation plotted against regularization strength for different data sets and approaches (except AnnealedVAE). The total correlation of the mean representation does not necessarily decrease as the regularization strength is increased.

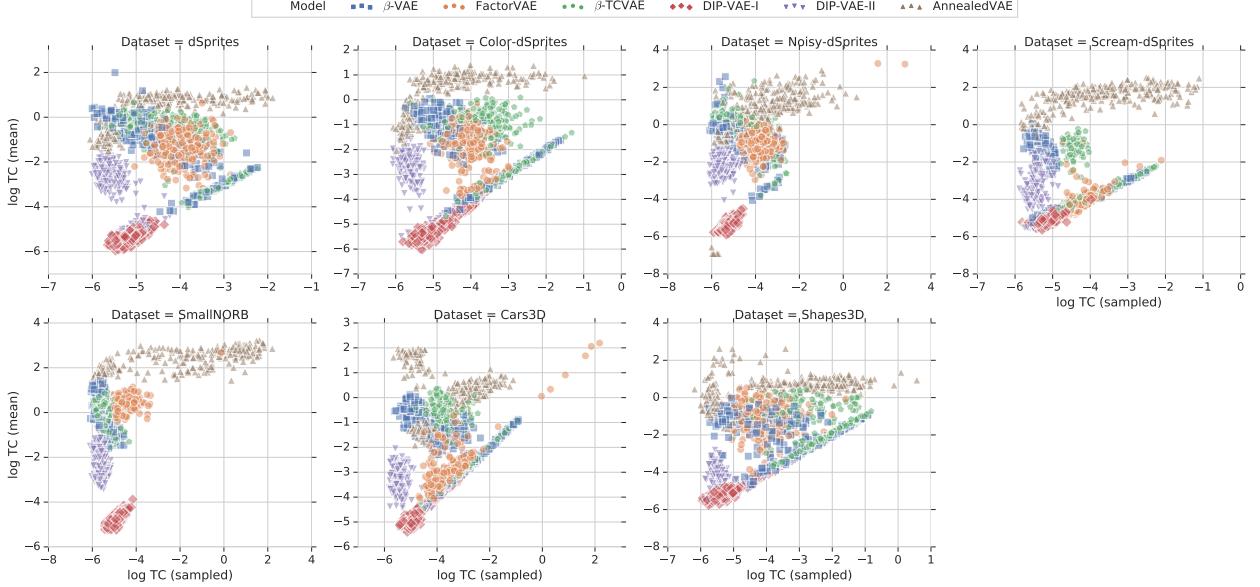


Figure 4: Log total correlation of mean vs sampled representations. For a large number of models, the total correlation of the mean representation is higher than that of the sampled representation.

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 a factorizing representation. To test whether this is true, we also 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 β -VAE and β -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 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 and Cars3d), but also seems to fail on others (dSprites). For AnnealedVAE (shown in Figure 16), 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 only method that achieves a consistently low total correlation both for the mean and the sampled representation is as expected DIP-VAE-I.

Implications. Overall, these results lead us to conclude that, with minor exceptions, the considered methods are effective at enforcing an aggregated posterior whose individual dimensions are not correlated. However, except for DIP-VAE-I, this does not seem to imply that the mean representation (usually used for representation) also factorizes.

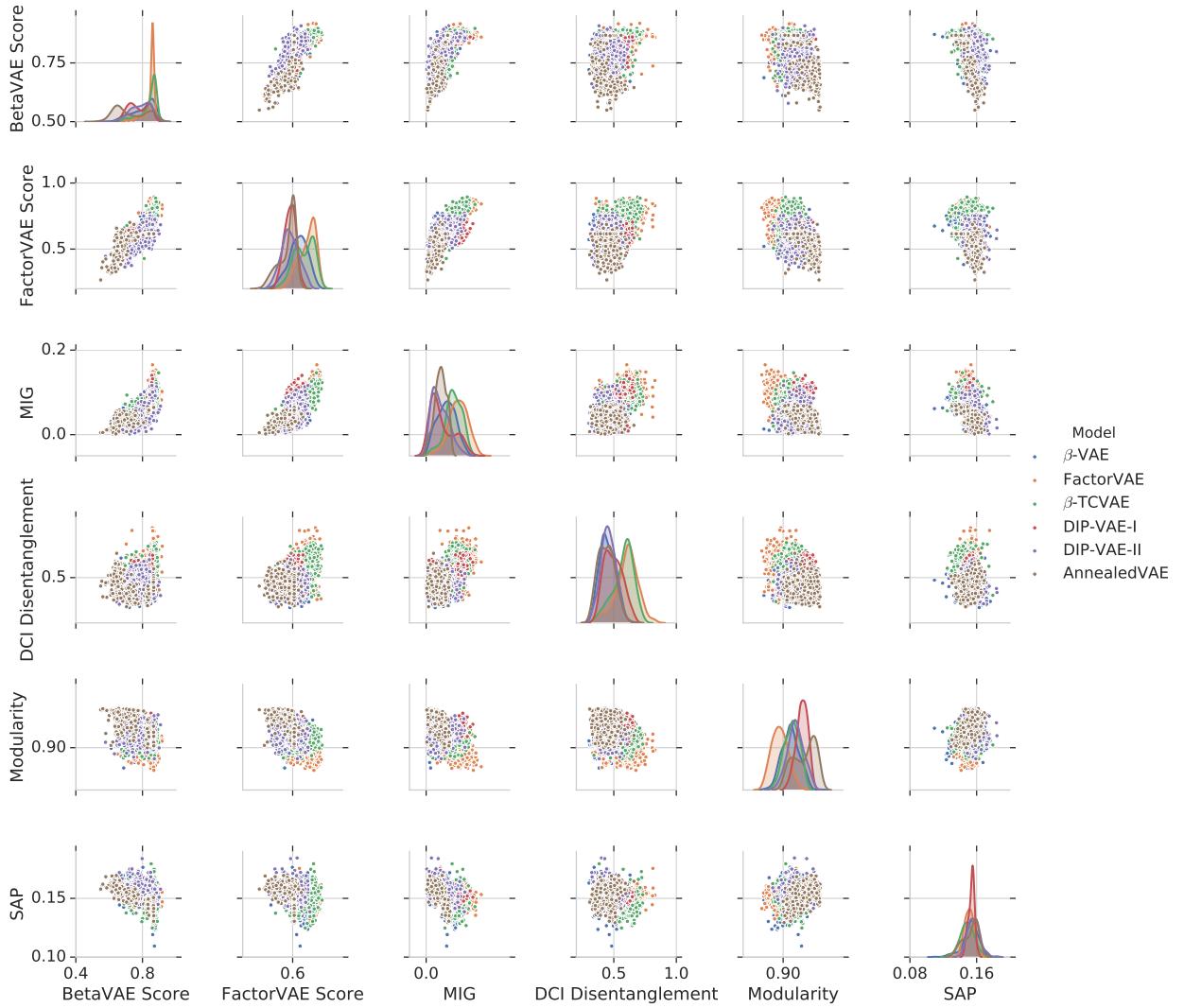


Figure 5: Pairwise scatter plots of different disentanglement metrics on dSprites. Except for the BetaVAE score, the FactorVAE score, and the Mutual Information Gap, different metrics seem to capture different aspects of the learned representation.

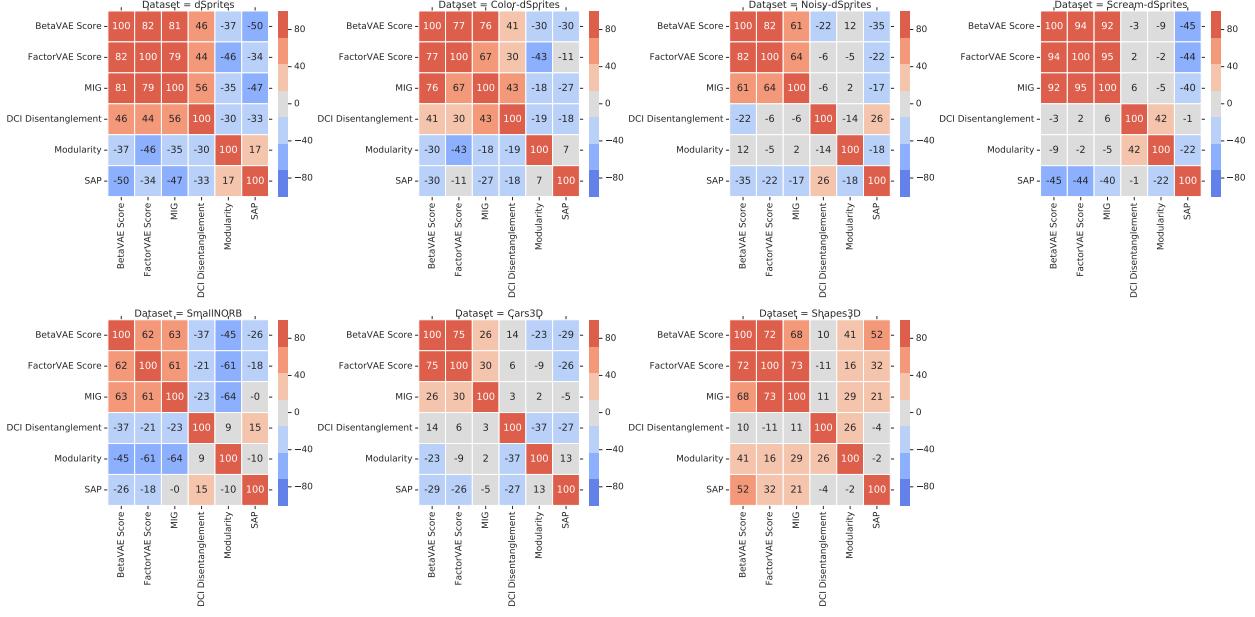


Figure 6: Rank correlation of different metrics on different data sets. The correlation between the BetaVAE score, the FactorVAE score, and the Mutual Information Gap appears consistent across data sets. The other metrics seem to capture different aspects of the learned representation.

5.3 How much do existing disentanglement metrics agree?

As there exists no single, common definition of disentanglement, an interesting question is to see how much different proposed metrics agree. Figure 5 shows pairwise scatter plots of the different considered metrics on dSprites where each point corresponds to a trained model. We observe that overall the different metrics seem to capture substantially different properties of the learned representation. Only the BetaVAE score, the FactorVAE score and the Mutual Information Gap seem to be substantially correlated. For the BetaVAE score and the FactorVAE score, this is expected as these scores are different variations of a similar idea. This view is confirmed by Figure 6 which shows the Spearman rank correlation between different disentanglement metrics on different data sets. Again, we see that that the BetaVAE score, the FactorVAE score and the Mutual Information Gap seem to be correlated while we do not observe significant correlation between the other metrics.

Implication. It seems that different metrics evaluate different properties of the learned representation. Therefore, it is challenging to compare different methods because it is not clear which metrics should be computed.

5.4 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. For each trained model, we compute all the considered disentanglement metrics. For the purpose of this section, we further take the median result across all random seeds to obtain a robust value for each data set, model, hyper parameter value, and disentanglement metric. In Figure 8, we show the range of attainable disentanglement scores for different hyperparameter values for each method on each data set. We clearly see that these ranges are heavily overlapping leading us to conclude that the choice of hyperparameters 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. It is worthwhile to note that in our study we have fixed the range of hyper parameters a priori to six different values. Hence, specific models may perform better than in Figure 8 if different hyperparameter settings are explored.

Implication. Tuning hyperparameters seems to be more important for disentanglement than the model choice.

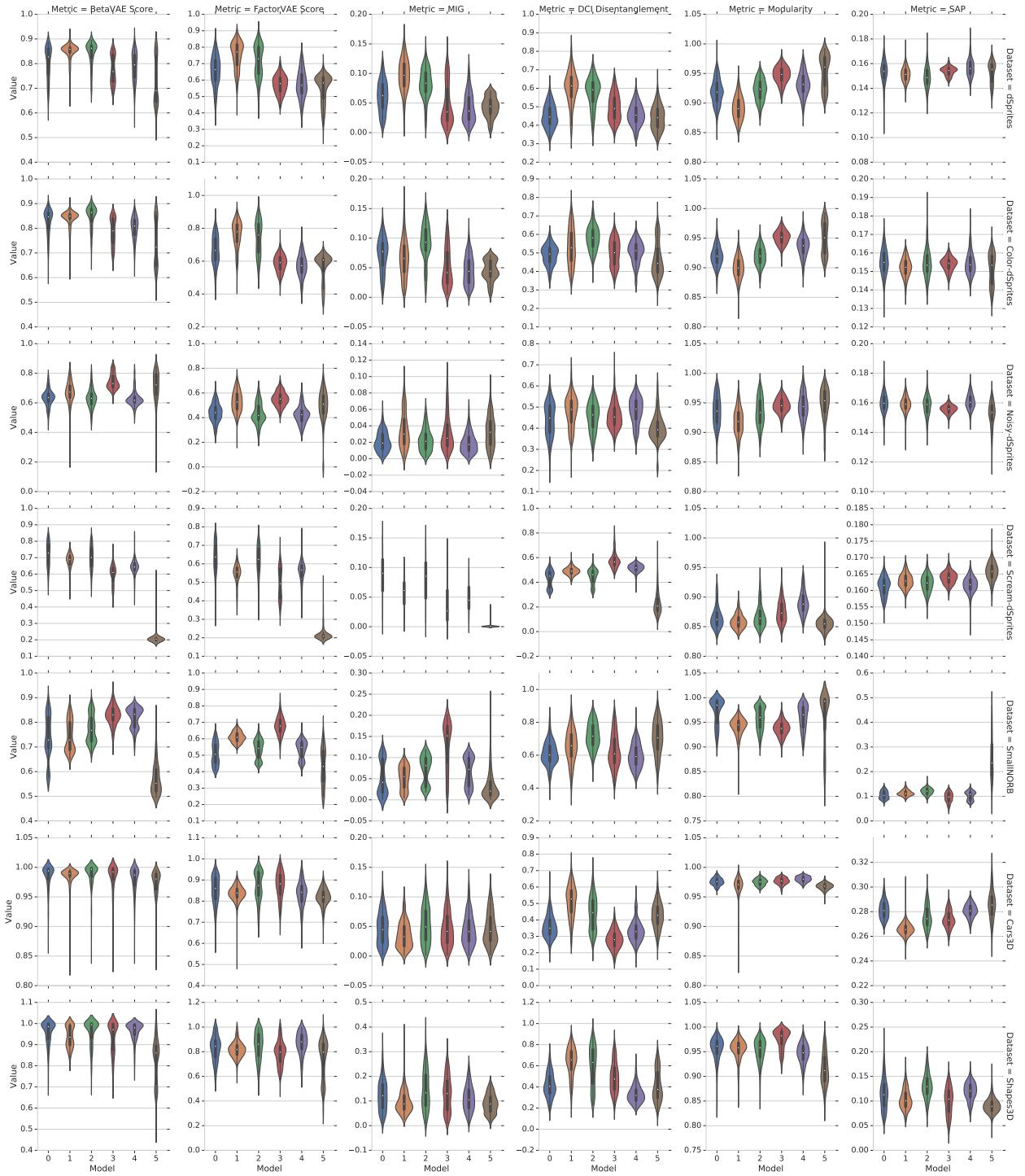


Figure 7: Score for each method for each score (column) and data set (row). Models are abbreviated (0= β -VAE, 1=FactorVAE, 2= β -TCVAE, 3=DIP-VAE-I, 4=DIP-VAE-II, 5=AnnealedVAE). The scores are heavily overlapping and we do not observe a consistent pattern. We conclude that tuning hyperparameters matters more than the choice of the objective function.

5.5 Are there reliable recipes for hyperparameter selection?

Given that the impact of the hyperparameters seems to be more critical than the loss function, this raises the natural question how hyperparameters should be chosen. In this paper, we advocate that a key principle that hyperparameter 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 achieves a good disentanglement score as it amounts to showing the existence of a good setting, but does not guide the practitioner. 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 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.

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 8 different disentanglement metrics against different regularization strengths for each model and each data set. As in prior sections, the values correspond to the median obtained values across 50 random seeds. Again, 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.

Hyperparameter 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. 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 9. While we do observe some correlations (in particular on SmallNORB), no clear pattern emerges which leads us to conclude that this approach is unlikely to be successful in practice.

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 11 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 are clearly correlated with the scores obtained on the regular version of dSprites. Yet, this correlation already vanishes if we consider the data sets Noisy-dSprites and

	Random different data set	Same data set
Random different metric	51.9%	51.8%
Same metric	55.0%	77.3%

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.

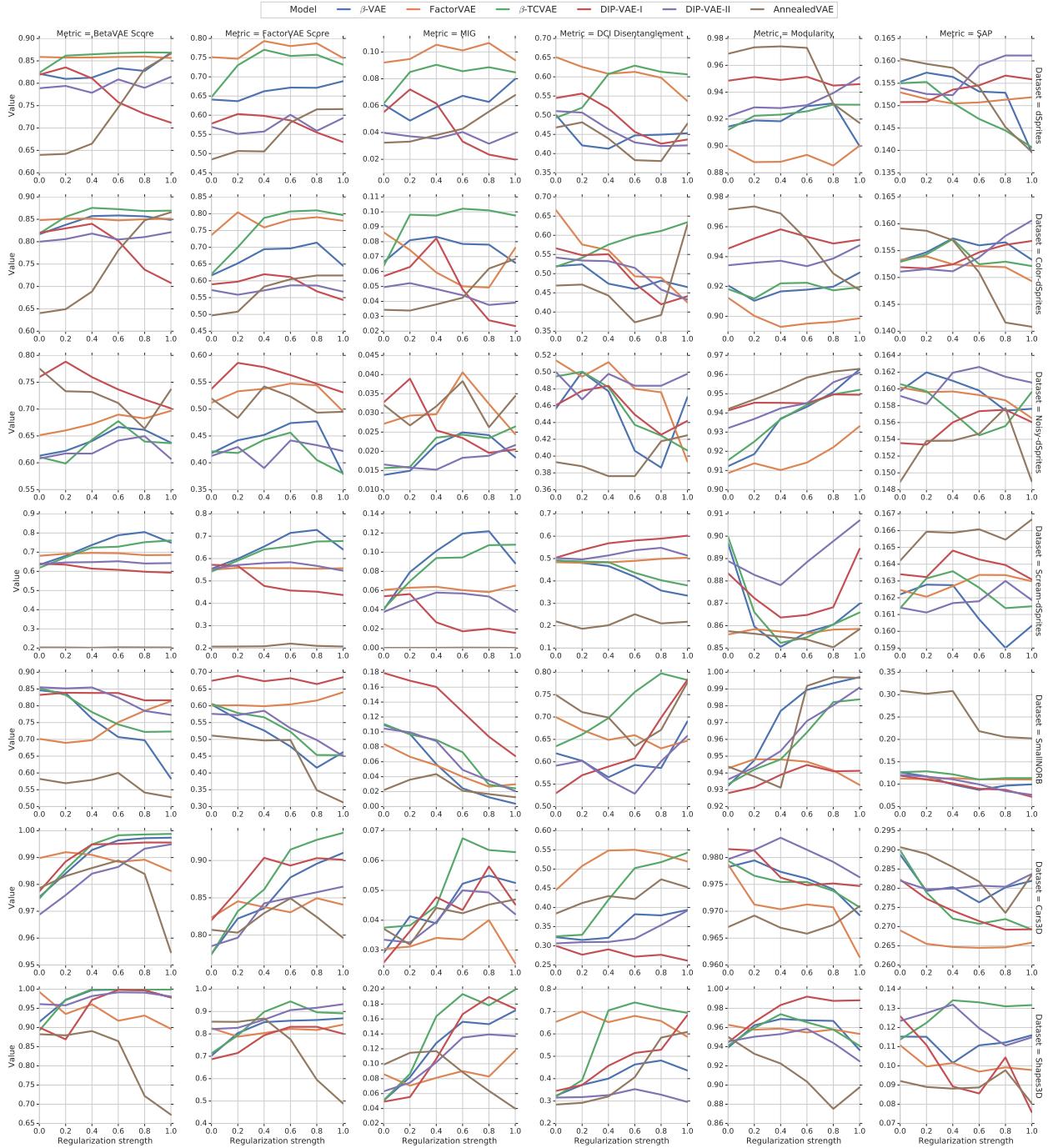


Figure 8: Score vs hyperparameters for each score (column) and data set (row). There seems to be no model dominating all the others and for each model there does not seem to be a consistent strategy in choosing the regularization strength.

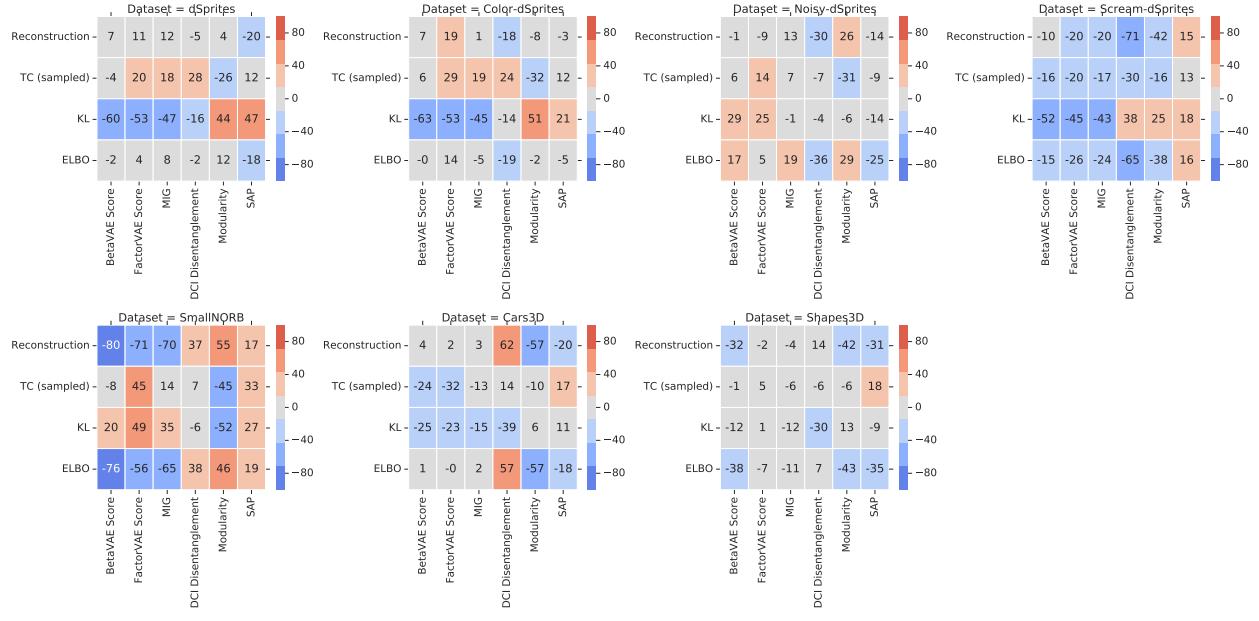


Figure 9: Score for each method for each score (column) and data set (row). The unsupervised scores we consider do not seem to be useful to tune hyperparameters.

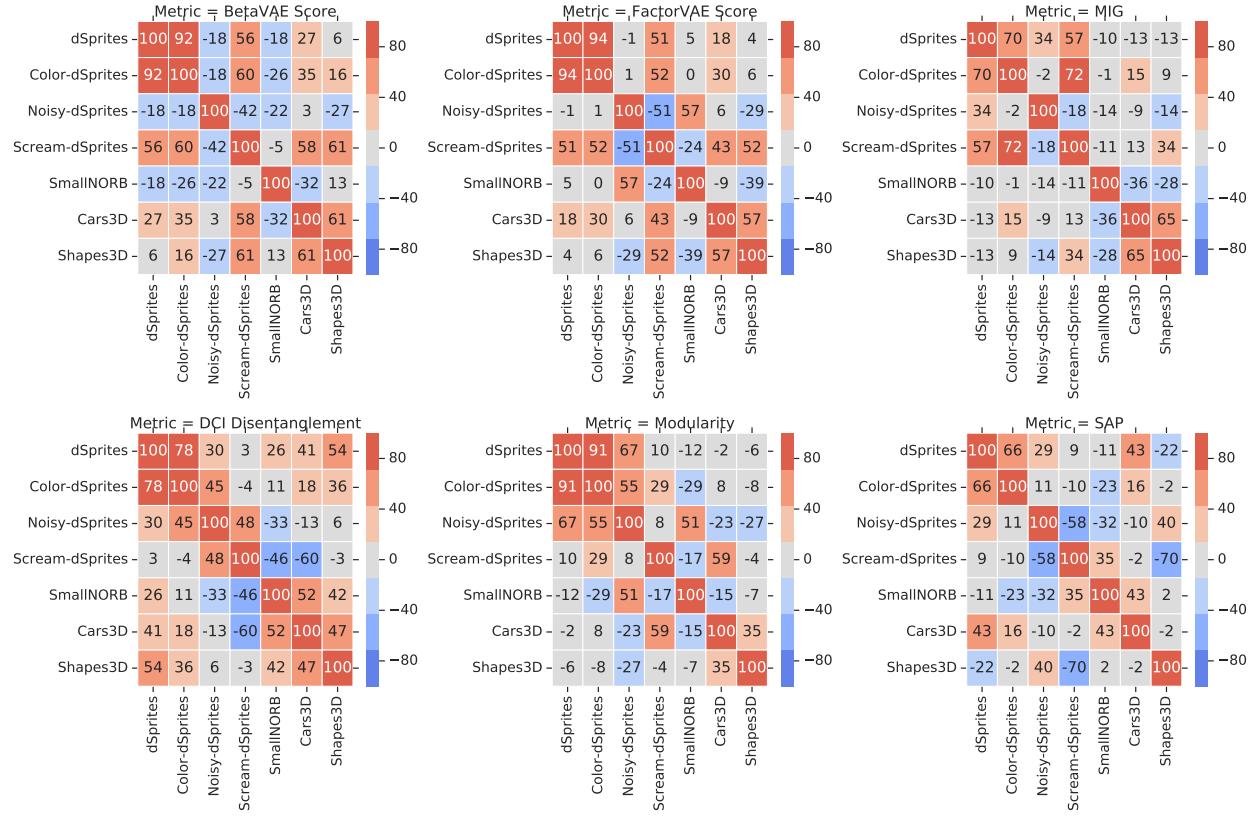


Figure 10: Rank-correlation of different disentanglement metrics across different data sets. Good hyperparameters only seem to transfer between dSprites and Color-dSprites but not in between the other data sets.

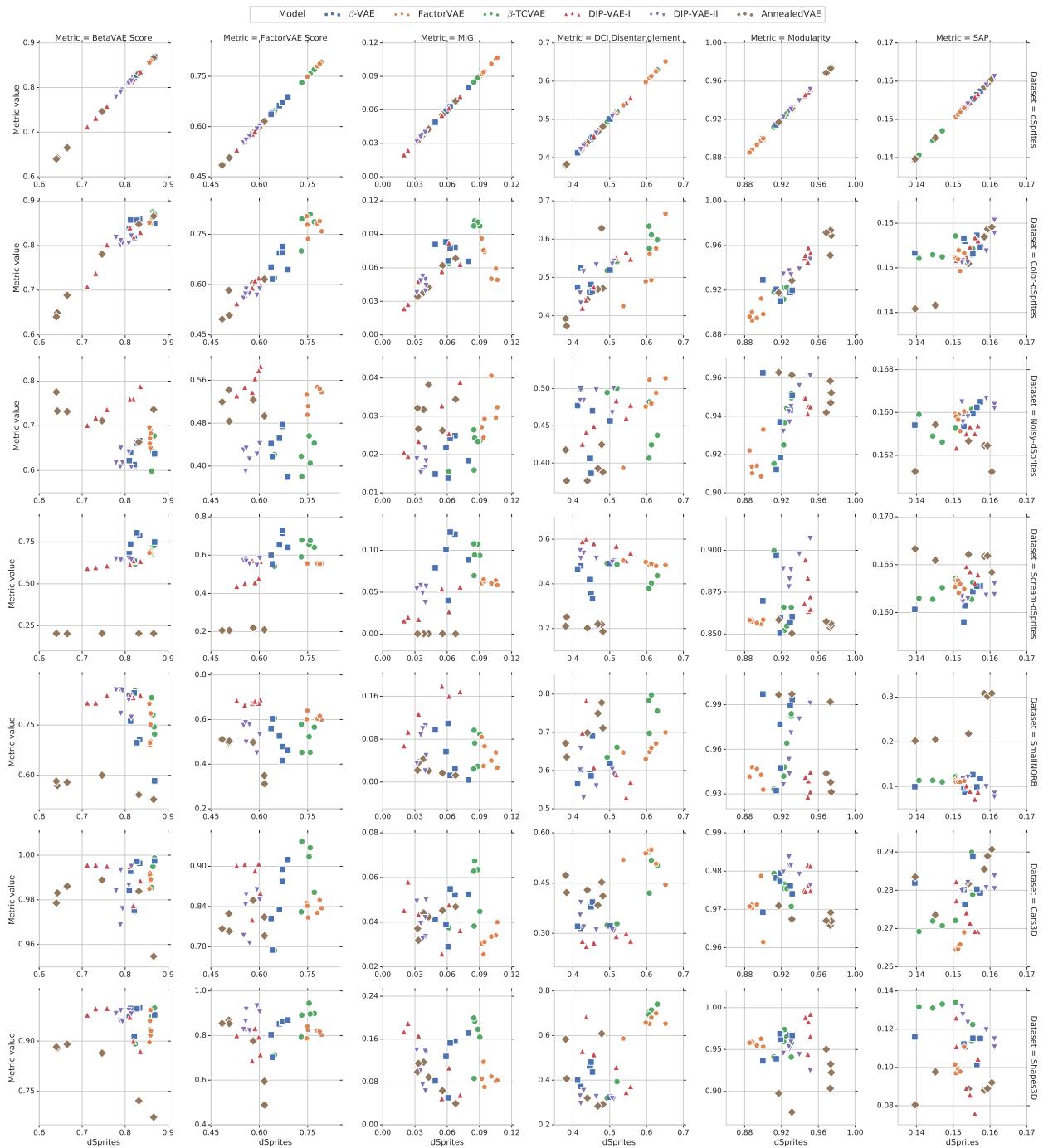


Figure 11: Disentanglement scores on dSprites vs other data sets. Good hyperparameters only seem to transfer from dSprites to Color-dSprites but not to the other data sets.

Scream-dSprites which is surprising as these data sets have exactly the same ground-truth generative factors as the regular version of dSprites. If we consider our other data sets, SmallNORB, Cars3D and Shapes3D, we also do not find any evidence that good hyperparameter values may be transferred to these data sets from dSprites. Figure 10 further shows the rank correlations obtained between different data sets for each disentanglement scores. Again, we reach a similar conclusion: there seems to be correlation across some versions of dSprites but not in between the other data sets.

Finally, 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 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 77.3%. If, we aim to either transfer for the same metric across data sets, we only achieve approximately 55%. Finally, if we transfer across metrics, we perform barely better than chance.

Implications. Unsupervised hyperparamrter selection remains an unsolved problem. Transfer of good hyperparameters in between metrics and data sets also does not seem to work.

5.6 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 [4, 47, 41]. In this section, we consider the simplest downstream task where the goal is to recover the true factors of variations from the observations. As all our ground-truth models have independent, discrete factors of variations this corresponds to a set of classification tasks. Our goal is to investigate the relationship between disentanglement and the average classification accuracy on this 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 12 shows the rank correlations between the disentanglement metrics and the downstream performance for all considered data sets. We observe that the BetaVAE score, the FactorVAE score and the Mutual Information GAP 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 capture the informativeness of the evaluated representation. The latter option seems likely as the scores that try to explicitly “isolate” the notion of disentanglement from the informativeness such as the DCI Disentanglement score and Modularity do not seem to be correlated with increased 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. The corresponding results are shown in Figures 13 and 14 where we plot the statistical efficiency versus different disentanglement metrics for different data sets and models. Overall, we do not observe any 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

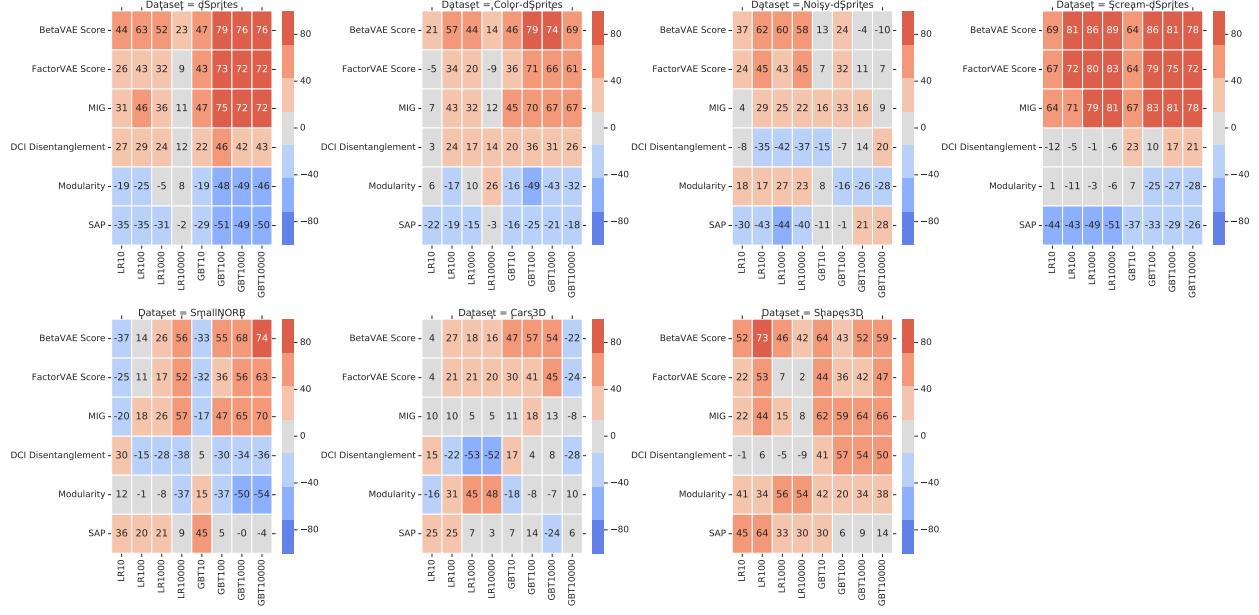


Figure 12: Rank-correlation between the metrics and the performance on downstream task on different data sets. We observe that the BetaVAE score, the FactorVAE score and the Mutual Information GAP seem to be correlated with increased downstream performance on the different variations of dSprites and 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 capture the informativeness of the evaluated representation. The latter option seems likely as the scores that try to explicitly “isolate” the notion of disentanglement from the informativeness such as the DCI Disentanglement score and Modularity do not seem to be correlated with increased downstream performance.

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.

Implications. While the empirical results in this section seem daunting, they should also be interpreted with care. After all, we have seen in previous sections that the considered models 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. Nevertheless, we argue that the lack of concrete examples of useful disentangled representations necessitates that future work on disentanglement methods should make this point more explicit.

6 Conclusions

In this work we show, perhaps unsurprisingly, 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, six disentanglement metrics on seven data sets and conclude the following: (i) While all considered methods prove effective at ensuring that the aggregated posterior (which is sampled) factorizes, only one method also consistently ensures that the representation (which is taken to be the mean) factorizes. (ii) We do not find any evidence that they can be used to reliably learn disentangled representations in an *unsupervised* manner as hyperparameters seem to matter more than the model and “good” hyperparameters seemingly cannot be identified without access to ground-truth labels. Similarly, we observe that good hyperparameters neither transfer across data sets nor across disentanglement metrics. (iii) For the considered models and data sets, we cannot validate the assumption

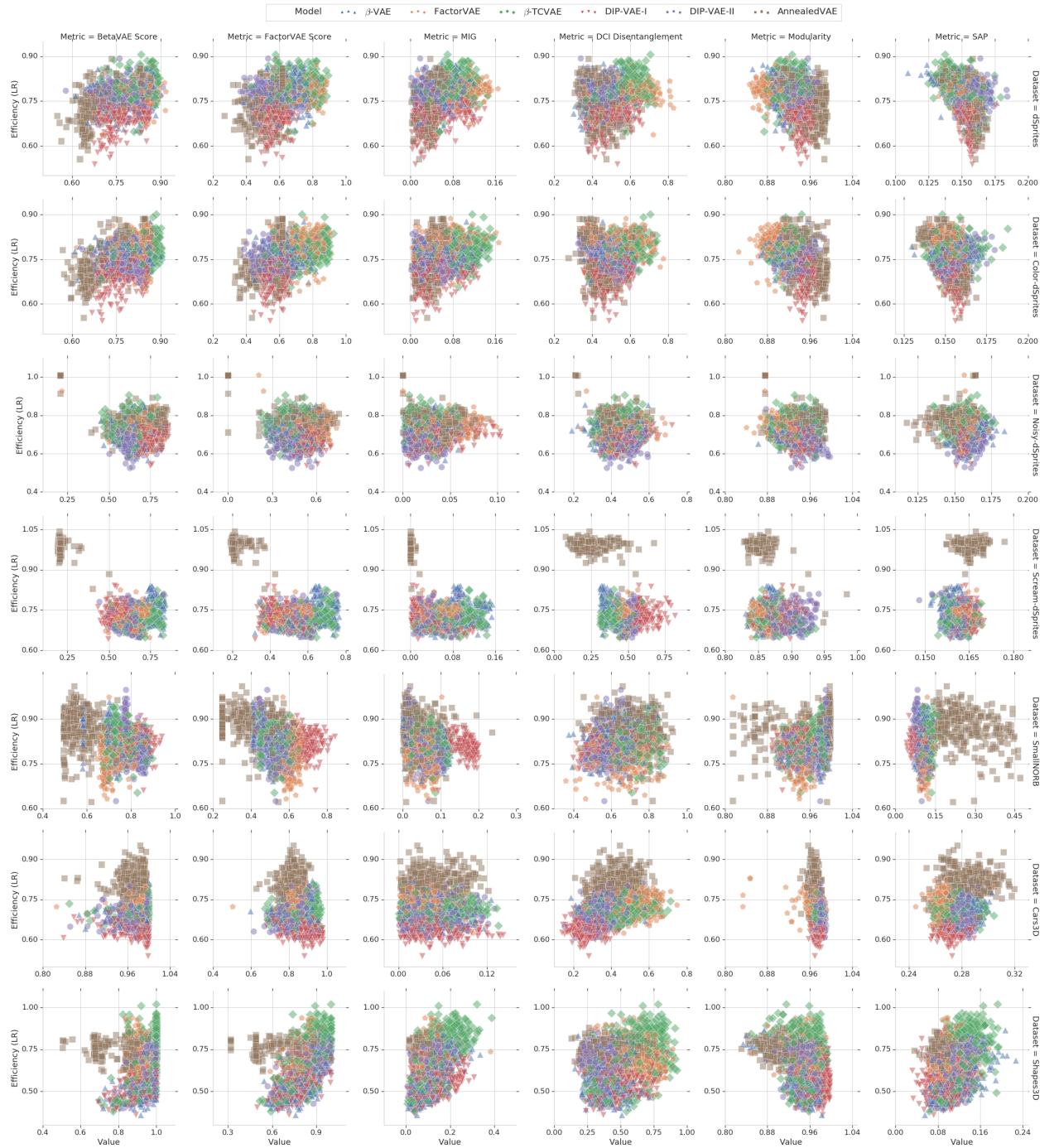


Figure 13: Statistical efficiency (accuracy with 100 samples \div accuracy with 10 000 samples) based on a logistic regression versus disentanglement metrics for different models and data sets. We do not observe that higher disentanglement scores lead to higher statistical efficiency.

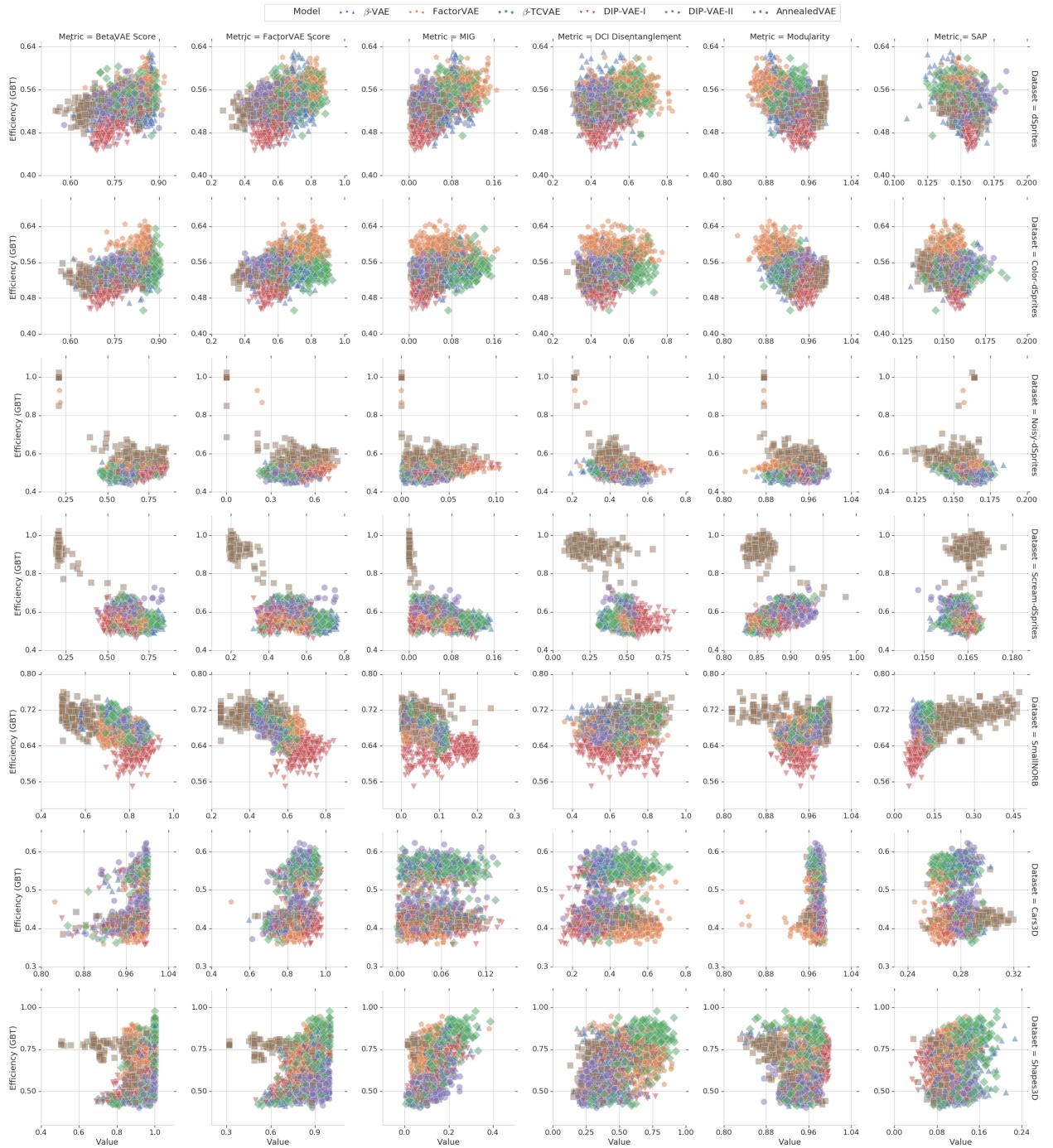


Figure 14: Statistical efficiency (accuracy with 100 samples \div accuracy with 10 000 samples) based on gradient boosted trees versus disentanglement metrics for different models and data sets. We do not observe that higher disentanglement scores lead to higher statistical efficiency.

that disentanglement is useful for downstream tasks through a decreased sample complexity of learning. Based on these findings, we suggest three main directions for future research:

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. Given the seemingly fundamental impossibility of purely unsupervised disentanglement learning, 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 [50], (ii) when weak forms of supervision e.g. through grouping information are available [5], or (iii) when temporal structure is available for the learning problem. The last setting seems to be particularly interesting given recent identifiability results in non-linear ICA [22] which could indicate that significant improvements for auto-encoding based approaches could be possible if the sequential structure of data can be exploited.
2. **Concrete practical benefits of disentangled representations.** In our experiments we investigated whether disentanglement leads to increased sample efficiency for downstream tasks and did not find evidence that this is the case. While these results only apply to the setting and downstream task used in our study, we are also not aware of other prior work that compellingly shows the usefulness of disentangled representations. Hence, we argue that future work should aim to show concrete benefits of disentangled representations. Interpretability and fairness as well as interactive settings seem to be particularly promising candidates. In such settings, feedback could potentially be incorporated and might alleviate some of the difficulties of the purely unsupervised setting. One potential approach to include inductive biases, offer interpretability, and generalization is the concept of independent causal mechanisms and the framework of causal inference [39, 41].
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. First, one has to be careful with the experimental design as for example hyperparameter selection has a substantial impact on the obtained results. We have tried to keep our study fair by choosing a wide set of hyperparameters up front and not modifying them while performing the study. We note that, as our hyperparameter selection was partially based on suggestions from prior work that considered some of the same data sets, one might validly argue that we have implicitly incorporated inductive biases on these data sets. We have also chosen the exact same training and evaluation protocol for all the different methods and disentanglement metrics. Second, 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. Seemingly innocent changes to data sets such as dSprites can lead to substantially different conclusions reached. We hence 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. We are hence interested in insights that generalize across multiple data sets rather than the absolute performance on specific data sets.

Acknowledgements

The authors thank Gunnar Rätsch, 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 and by an ETH core grant (to Gunnar Rätsch). This work was partially done while Francesco Locatello was at Google AI.

References

- [1] Miguel A Arcones and Evarist Gine. On the bootstrap of u and v statistics. *The Annals of Statistics*, pages 655–674, 1992.
- [2] Francis R Bach and Michael I Jordan. Kernel independent component analysis. *Journal of machine learning research*, 3(Jul):1–48, 2002.
- [3] Yoshua Bengio, Yann LeCun, et al. Scaling learning algorithms towards ai. *Large-scale kernel machines*, 34(5):1–41, 2007.
- [4] 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.
- [5] Diane Bouchacourt, Ryota Tomioka, and Sebastian Nowozin. Multi-level variational autoencoder: Learning disentangled representations from grouped observations. *arXiv preprint arXiv:1705.08841*, 2017.
- [6] 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.
- [7] Tian Qi Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *To Appear In Neural Information Processing Systems*, 2018.
- [8] 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*, pages 2172–2180, 2016.
- [9] 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.
- [10] Taco Cohen and Max Welling. Learning the irreducible representations of commutative lie groups. In *International Conference on Machine Learning*, pages 1755–1763, 2014.
- [11] Taco S Cohen and Max Welling. Transformation properties of learned visual representations. *arXiv preprint arXiv:1412.7659*, 2014.
- [12] Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.
- [13] Emily L Denton et al. Unsupervised learning of disentangled representations from video. In *Advances in Neural Information Processing Systems*, pages 4414–4423, 2017.
- [14] Guillaume Desjardins, Aaron Courville, and Yoshua Bengio. Disentangling factors of variation via generative entangling. *arXiv preprint arXiv:1210.5474*, 2012.
- [15] Cian Eastwood and Christopher KI Williams. A framework for the quantitative evaluation of disentangled representations. 2018.
- [16] 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*, pages 3601–3610, 2017.
- [17] 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*, pages 646–654, 2009.
- [18] Ross Goroshin, Michael F Mathieu, and Yann LeCun. Learning to linearize under uncertainty. In *Advances in Neural Information Processing Systems*, pages 1234–1242, 2015.

- [19] 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. 2016.
- [20] Geoffrey E Hinton, Alex Krizhevsky, and Sida D Wang. Transforming auto-encoders. In *International Conference on Artificial Neural Networks*, pages 44–51. Springer, 2011.
- [21] 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*, pages 1878–1889, 2017.
- [22] Aapo Hyvärinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. In *Advances in Neural Information Processing Systems*, pages 3765–3773, 2016.
- [23] Aapo Hyvärinen and Petteri Pajunen. Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 12(3):429–439, 1999.
- [24] Aapo Hyvärinen, Hiroaki Sasaki, and Richard E Turner. Nonlinear ica using auxiliary variables and generalized contrastive learning. *arXiv preprint arXiv:1805.08651*, 2018.
- [25] Christian Jutten and Juha Karhunen. Advances in nonlinear blind source separation. In *Proc. of the 4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 245–256, 2003.
- [26] Theofanis Karaletsos, Serge Belongie, and Gunnar Rätsch. Bayesian representation learning with oracle constraints. *arXiv preprint arXiv:1506.05011*, 2015.
- [27] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. *arXiv preprint arXiv:1802.05983*, 2018.
- [28] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- [29] Tejas D Kulkarni, William F Whitney, Pushmeet Kohli, and Josh Tenenbaum. Deep convolutional inverse graphics network. In *Advances in neural information processing systems*, pages 2539–2547, 2015.
- [30] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations. In *International Conference on Learning Representations*, 2017.
- [31] 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.
- [32] Yann LeCun, Fu Jie Huang, and Leon Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–104. IEEE, 2004.
- [33] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.
- [34] Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 991–999, 2015.
- [35] Michael F Mathieu, Junbo Jake Zhao, Junbo 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*, pages 5040–5048, 2016.
- [36] Edvard Munch. The scream, 1893.
- [37] 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*, pages 5925–5935, 2017.

- [38] 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.
- [39] Judea Pearl. *Causality*. Cambridge university press, 2009.
- [40] 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.
- [41] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. MIT press, 2017.
- [42] Scott Reed, Kihyuk Sohn, Yuting Zhang, and Honglak Lee. Learning to disentangle factors of variation with manifold interaction. In *International Conference on Machine Learning*, pages 1431–1439, 2014.
- [43] Scott E Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. Deep visual analogy-making. In *Advances in neural information processing systems*, pages 1252–1260, 2015.
- [44] Karl Ridgeway and Michael C Mozer. Learning deep disentangled embeddings with the f-statistic loss. In *To Appear In Neural Information Processing Systems*, 2018.
- [45] Paul K Rubenstein, Bernhard Schoelkopf, and Ilya Tolstikhin. Learning disentangled representations with wasserstein auto-encoders. 2018.
- [46] Jürgen Schmidhuber. Learning factorial codes by predictability minimization. *Neural Computation*, 4(6):863–879, 1992.
- [47] 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.
- [48] 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.
- [49] Raphael Suter, Đorđe Miladinović, Stefan Bauer, and Bernhard Schölkopf. Interventional robustness of deep latent variable models. *arXiv preprint arXiv:1811.00007*, 2018.
- [50] 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. *arXiv preprint arXiv:1802.09484*, 2018.
- [51] Satosi Watanabe. Information theoretical analysis of multivariate correlation. *IBM Journal of research and development*, 4(1):66–82, 1960.
- [52] William F Whitney, Michael Chang, Tejas Kulkarni, and Joshua B Tenenbaum. Understanding visual concepts with continuation learning. *arXiv preprint arXiv:1602.06822*, 2016.
- [53] 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*, pages 1099–1107, 2015.
- [54] Li Yingzhen and Stephan Mandt. Disentangled sequential autoencoder. In *International Conference on Machine Learning*, pages 5656–5665, 2018.
- [55] 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*, pages 217–225, 2014.

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 as well as $\frac{\partial g_i(\mathbf{v})}{\partial v_j} \neq 0$ 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 as well as $\frac{\partial h_i(\mathbf{v})}{\partial v_j} \neq 0$ 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 $1, 2, \dots, d$ 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

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

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

B Implementation of metrics

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).

BetaVAE metric. We sample two batches of 64 points with one factor fixed to some random value and the others varying randomly. We encode these points and take the absolute difference between pairs. 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.

FactorVAE metric 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. Then, 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. Then, 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 smallest normalized variance. We train on 10 000 points and evaluate on 5000 points.

Mutual Information Gap. 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:

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

Where v_k is a factor of variation, z_j is a latent representation and $j_k = \arg \max_j I(z_j, v_k)$.

Modularity and Explicitness 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. To compute the explicitness score, we first remove mean and standard deviation from the representations. Then, we fit a one-versus-rest logistic regression from Scikit-learn with default hyperparameters on each factor. We compute the AUC-ROC using Scikit-learn. The explicitness score is then the average of these values. We test on 5000 points.

Disentanglement, completeness and informativeness We sample 10 000 and 5000 training and test points respectively. We remove mean and standard deviation from their latent representation and compute the importance of each dimension to predict the values of each factor. For each factor, we compute a SVM from Scikit-learn with l_1 regularization and $C = 1$. From this model, we can extract some 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 (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 $\sum_i \rho_i (1 - H(R_i))$. The completeness is the average of $1 - H(R_j)$ which is the entropy computed across the dimensions of the representation rather than factors as before. The informativeness is the test accuracy if the classifier used in the other scores.

SAP score. We sample 10 000 points for training and 5000 for testing. We remove mean and standard deviation from their latent representation. 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.

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

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

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

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

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 ($C_s = 10$) and 5 folds. Finally, we train a gradient boosting classifier from Scikit-learn with default parameters.

Total correlation. 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 this distribution:

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

where j indexes the dimensions in the latent space.

C Main experiment hyperparameters

In the main experiment, 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.

Table 4: Other fixed hyperparameters.

(a) Hyperparameters common to each of the considered methods.		(b) Architecture for the discriminator in FactorVAE.		(c) Parameters for the discriminator in FactorVAE.	
Parameter	Values	Discriminator		Parameter	Values
Batch size	64	FC, 1000 leaky ReLU		batch size	64
Latent space dimension	10	FC, 1000 leaky ReLU		Optimizer	Adam
Optimizer	Adam	FC, 1000 leaky ReLU		Adam: beta1	0.5
Adam: beta1	0.9	FC, 1000 leaky ReLU		Adam: beta2	0.9
Adam: beta2	0.999	FC, 1000 leaky ReLU		Adam: epsilon	1e-8
Adam: epsilon	1e-8	FC, 1000 leaky ReLU		Adam: learning rate	0.0001
Adam: learning rate	0.0001	FC, 1000 leaky ReLU			
Decoder type	Bernoulli	FC, 2			

D 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-dSprites:** Every time we sample a point, we sample a random 64×64 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.

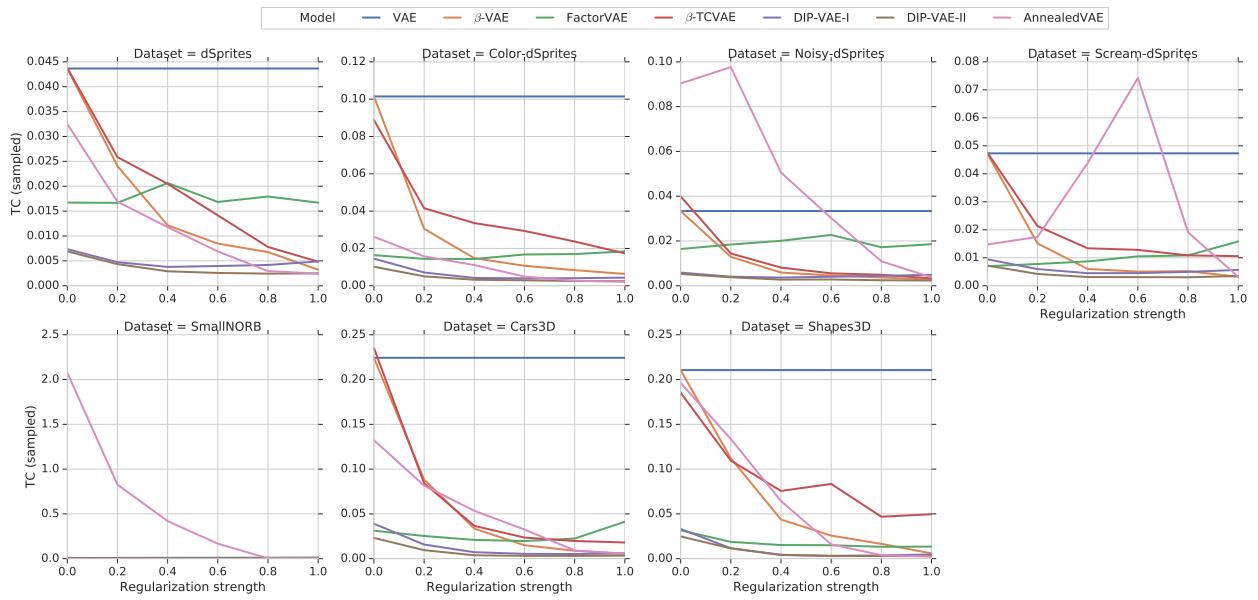


Figure 15: Total correlation of sampled representation plotted against regularization strength for different data sets and approaches (including AnnealedVAE).

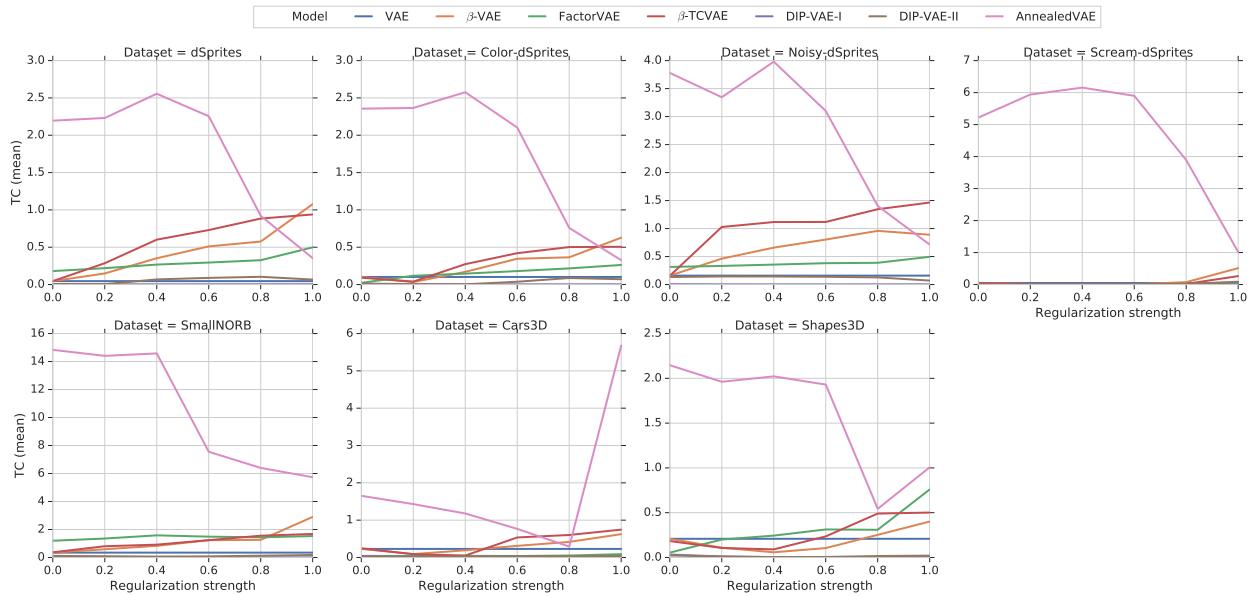


Figure 16: Total correlation of mean representation plotted against regularization strength for different data sets and approaches (including AnnealedVAE).