Challenging common assumptions in the unsupervised learning of disentangled representations

GNN-YYK study

Presenter: Yong-Min Shin

Locatello F, Bauer S, Lucic M, Raetsch G, Gelly S, Schölkopf B, Bachem O., ICML 2019 (Best paper)

Contents

Preliminary: VAE

Variational Autoencoders

Formulation

$$\log p(X) - \mathcal{D}_{KL}(Q(z|X)||p(z|X)) = \underbrace{\mathbb{E}_{z\sim Q}[\log p(X|z)]}_{\text{Encoder}\quad\text{Decoder}} - \underbrace{\mathcal{D}_{KL}(Q(z|X)||p(z))}_{\text{Encoder}}$$

$$\underbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}_{\text{VAE (ELBO)}}$$

Actual loss function

$$\arg\min_{\phi,\theta} -\sum_i \underbrace{[\mathbb{E}_{z\sim Q_\phi}[\log p(x^i|g_\theta(z))]}_{\text{Reconstruction loss: Encoder, Decoder}} + \underbrace{\mathcal{D}_{KL}(Q_\phi(z|x^i)||p(z))]}_{\text{Regularization: Encoder}}$$

- Reconstruction loss: compares the input and output data
- Regularization: Additional KL-divergence term to regularize latent space

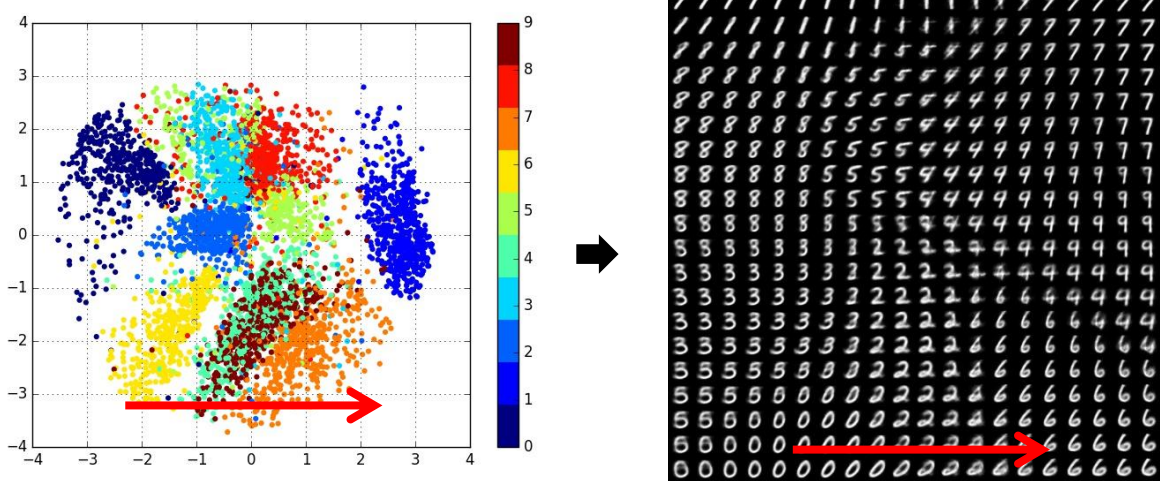
Encode various features into a latent space

1. What is disentanglement?

Intuitive description

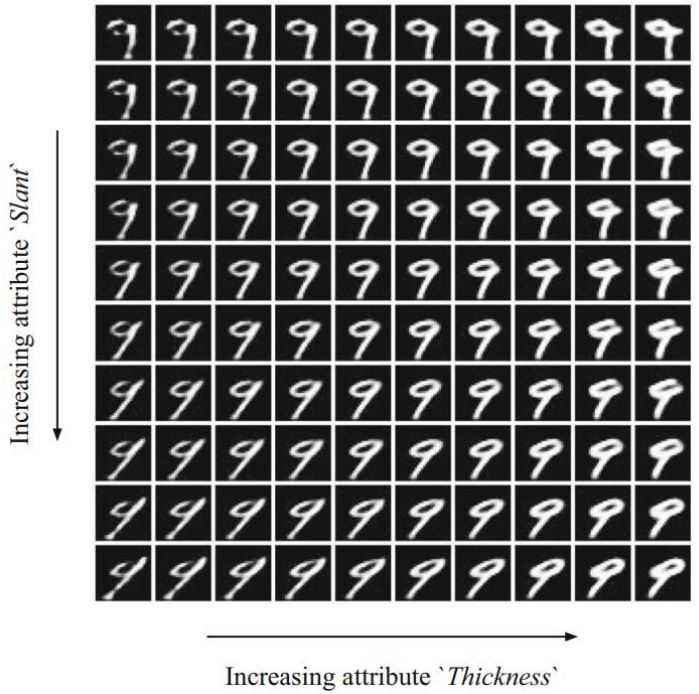
Design of the latent space such that it can **explicitly control certain features**.
Disentanglement can be thought of as:

What if we can explicitly set the axis of the latent space as the (human-recognizable) feature of our choice?



Here, changing the value of an axis results in changing **several** features in the decoder.

(Not disentangled)



Example of a disentangled latent space:
changing one value results in a single change in the decoder.

Left from: https://github.com/hwalsuklee/tensorflow-mnist-VAE

Assumptions and criteria of disentanglement

1. They should **contain all the information present in x in a compact and interpretable structure (i)** while being **independent from the task at hand (ii)**.

   (i) (Bengio, Y. et al., Representation learning: A review and new perspectives, IEEE TPAMI (2013))
   (ii) (Goodfellow et al., Measuring invariances in deep networks, NeurIPS 2009)

2. They should be useful for **(semi-)supervised learning of downstream tasks (i)**, **transfer (iii)** and **few shot learning (iv)**

   (iii) (Schölkopf et al., On the causal and anticausal learning, ICML 2012)
   (iv) (Peters et al, Elements of causal inference: foundations and learning algorithms, MIT Press, 2017)

3. They should enable to **integrate out nuisance factors (v)** to **perform interventions, and to answer counterfactual questions (iv)**.

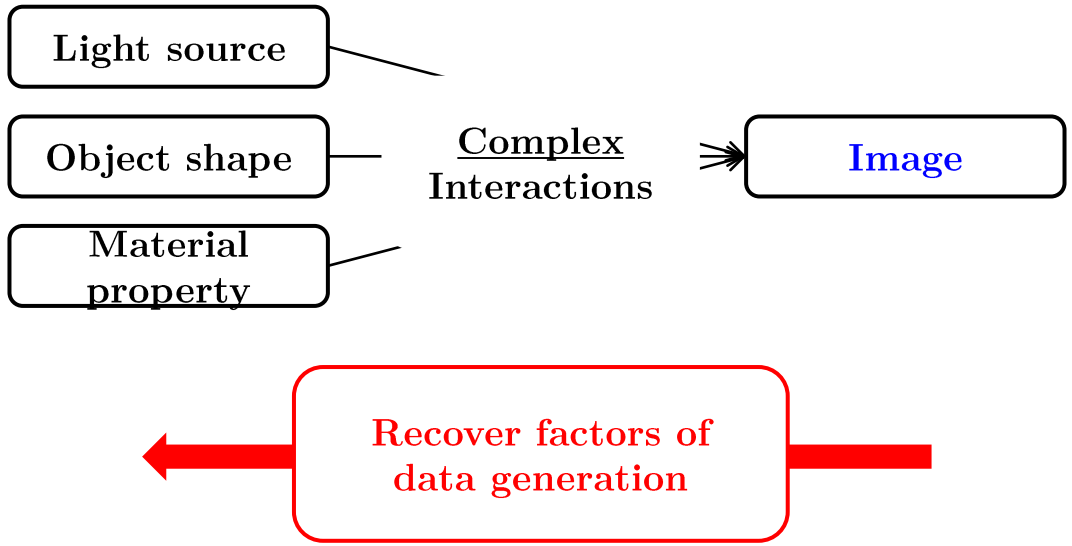
   (v) (Kumar et al., Variational inference of disentangled latent concepts from unlabeled observations. ICLR 2017)

They should **contain all the information present in x in a compact and interpretable structure (i)** while being independent from the task at hand (ii).

(i) (Bengio, Y. et al., Representation learning: A review and new perspectives, IEEE TPAMI (2013))

3 What makes a representation good?

• **Multiple explanatory factors**: the data generating distribution is generated by different underlying factors, and for the most part what one learns about one factor generalizes in many configurations of the other factors. The objective to recover or at least disentangle these underlying factors of variation is discussed in Section 3.5.



3.5 Disentangling Factors of Variation

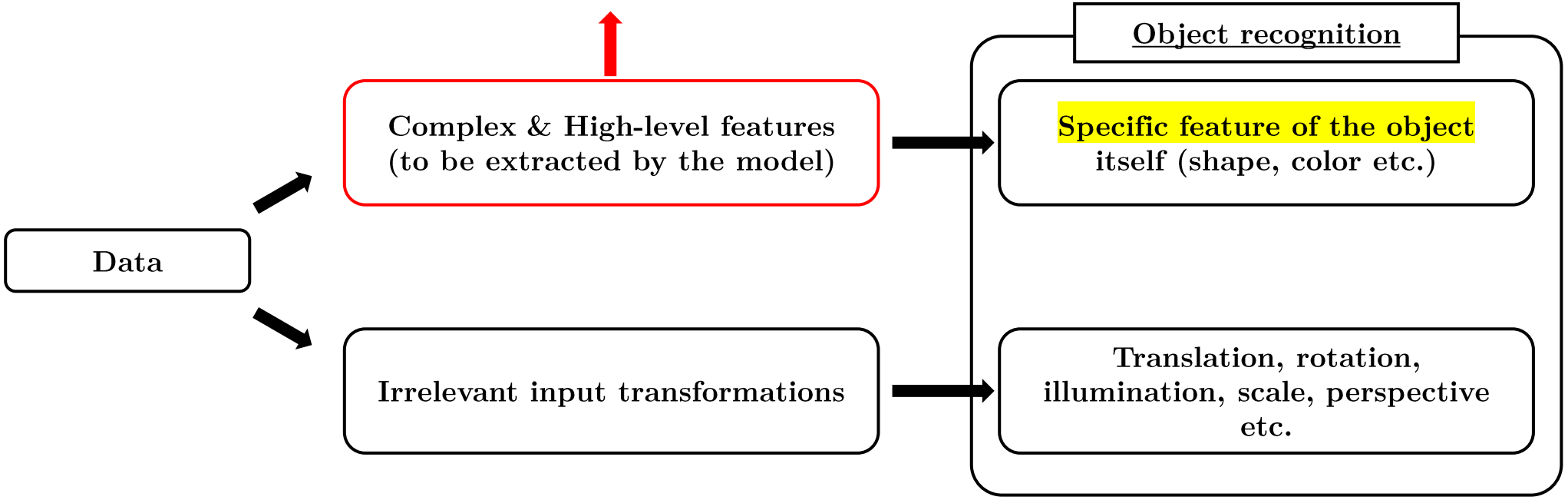
Complex data arise from the rich interaction of many sources. These factors interact in a complex web that can complicate AI-related tasks such as object classification. For example, an image is composed of the interaction between one or more light sources, the object shapes and the material properties of the various surfaces present in the image. Shadows from objects in the scene can fall on each other in complex patterns, creating the illusion of object boundaries where there are none and dramatically effect the perceived object shape. How can we cope with these complex interactions? How can we *disentangle* the objects and their shadows? Ultimately, we believe the approach we adopt for overcoming these challenges must leverage the data itself, using vast quantities of unlabeled examples, to learn representations that separate the various explanatory sources. Doing so should give rise to a representation significantly more robust to the complex and richly structured variations extant in natural data sources for AI-related tasks.

Considerations such as these lead us to the conclusion that the most robust approach to feature learning is *to disentangle as many factors as possible, discarding as little information about the data as is practical.*

They should contain all the information present in x in a compact and interpretable structure (i) while being **independent from the task at hand (ii)**.

(ii) (Goodfellow et al., Measuring invariances in deep networks, NeurIPS 2009)

Deep architectures are capable to extracting these features.



Summary

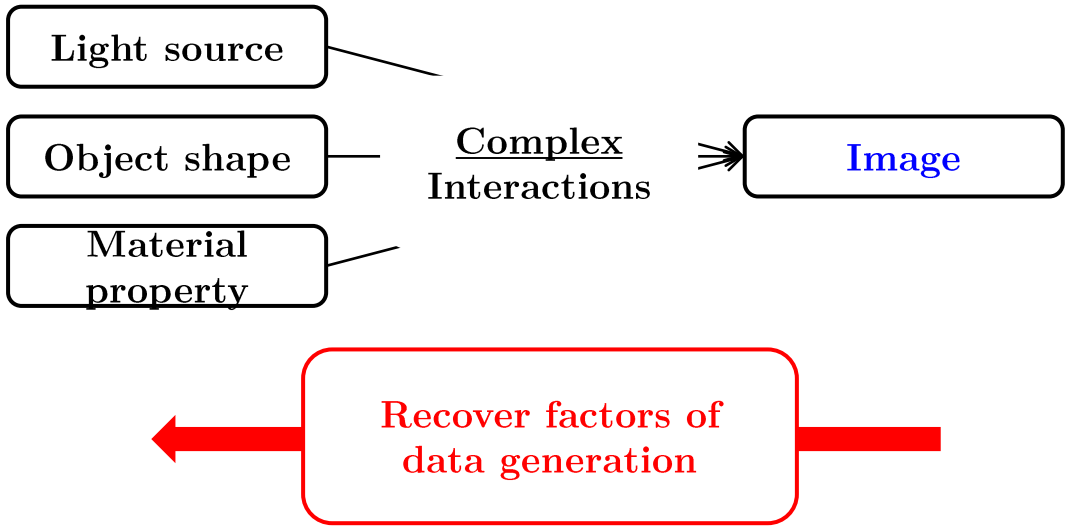
(Bengio, Y. et al., Representation learning: A review and new perspectives, IEEE TPAMI (2013))

A disentangled representation separates informative factors (align the axis) of variations in the data.

+ Better to find human-interpretable factors

One-to-one correspondence to axis

Probably the reason why it fits so well for generative models.



Left figure: Ashis Pati et al., "Attribute-based regularization of latent spaces for variational autoencoders", Neural Computing and Applications , 2020.

2. Related works (Baseline methods)

Before going in: VAEs and factorized aggregated posterior

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., 2017; Kim & Mnih, 2018; Chen et al., 2018; Kumar et al., 2017; Rubenstein 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.

Francsco Locatello et al., ICML'19

$$\log p(X) - \mathcal{D}_{KL}(Q(z|X)||p(z|X)) = \underbrace{\mathbb{E}_{z\sim Q}[\log p(X|z)]}_{\text{Decoder}} - \underbrace{\mathcal{D}_{KL}(Q(z|X)||p(z))}_{\text{Encoder}}$$

$$\text{VAE (ELBO)}$$

$$\downarrow$$

Recent works (Higgins et al., 2017; Kim & Mnih, 2018; Chen et al., 2018; Kumar et al., 2017; Ridgeway & Mozer, 2018) have introduced various regularizers to the objective function of the *Variational Autoencoder* (VAE) (Kingma & Welling, 2013; Bengio et al., 2007), *Evidence Lower Bound* (ELBO). They aim at factorizing *aggregated posterior*, $q(z) = \int q(z|x)p(x)dx$, which hopefully can encourage disentanglement.

Ze Cheng et al., Revisiting Factorizing Aggregated Posterior in Learning Disentangled Representations

$$\arg\min_{\phi,\theta} -\sum_i \underbrace{[\mathbb{E}_{z\sim Q_\phi}[\log p(x^i|g_\theta(z))]}_{\text{Reconstruction loss (Decoder)}} + \underbrace{\mathcal{D}_{KL}(Q_\phi(z|x^i)||p(z))]}_{\text{Regularization (Encoder)}}$$

Considered methods. All the considered methods augment the VAE loss with a regularizer: The β-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., 2017) 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 & Mnih, 2018) and the β-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., 2017) both penalize the mismatch between the aggregated posterior and a factorized prior. Implementation details and further discussion on the methods can be found in Appendix B and G.

1. β-VAEs
2. AnnealedVAE
3. FactorVAE
4. β-TCVAEs
5. ~~DIP-VAE-I/II~~

** I tried to use all notations that are frequently used in the literature, and some of them may not be consistent.*

β-VAEs

[1] Higgins, Irina, et al. "beta-VAE: Learning basic visual concepts with a constrained variational framework." ICLR'17. 1199 citations.

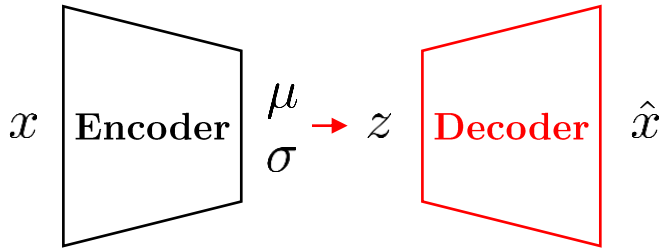
Under review as a conference paper at ICLR 2017

β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot,
Matthew Botvinick, Shakir Mohamed, Alexander Lerchner
Google DeepMind
{irinah,lmatthey,arkap,cpburgess,glorotx,
botvinick,shakir,lerchner}@google.com

2. Related works (Baseline methods)

β-VAEs



Previously, most of the studies in disentangled representation learning has focused on supervised or semi-supervised setting, and there were almost no models on **learning disentanglement in an unsupervised way**.

Recently, *__InfoGAN__ has proposed a scalable unsupervised approach using the framework of GAN.
However, it comes at the cost of

1. Training instability
2. Low sample diversity

which are common shortcomings of the framework.

The objective is to use a **VAE framework** to learn disentanglement in an **unsupervised** manner.

*Chen, Xi, et al. "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets." NIPS'16. 2202 citations.

β-VAEs

Assumption and proposed method

v: Conditionally independent factors (want to disentangle)

w: Conditionally dependent factors (not interested, remain entangled)

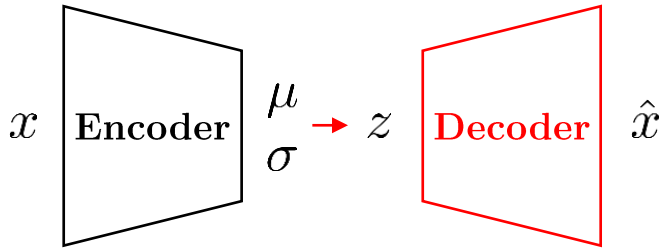
$\mathbf{AppleSimulator}(\mathbf{v}, \mathbf{w})$

$\mathbf{v} = [color, size]$

Small green apple *Big green apple* *Big red apple*

Assumption

$x \sim \mathbf{AppleSimulator}(\mathbf{v}, \mathbf{w})$



$$p(x|z) \approx p(x|v, w) = \mathbf{AppleSimulator}(\mathbf{v}, \mathbf{w})$$

To make the latent vector to act like the generative parameters v and w, the authors propose to **enforce a higher constraint on the regularization term.**

$$\arg\min_{\phi,\theta} -\sum_i [\mathbb{E}_{z\sim Q_\phi}[\log p(x^i|g_\theta(z))] + \beta \cdot \mathcal{D}_{KL}(Q_\phi(z|x^i)||p(z))]$$

$$(\beta > 1)$$

Apple images: https://www.applesfromny.com/varieties/

*Chen, Xi, et al. "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets." NIPS'16. 2202 citations.

β-VAEs

Lagrangian and KKT method, alignment ([1], [3])

$$p(x|z) \approx p(x|v,w) = \textbf{AppleSimulator}(\mathbf{v},\mathbf{w})$$

$$\downarrow$$

$$\arg\max_{\theta} \mathbb{E}_{p_\theta(z)}[p_\theta(x|z)]$$

$$\downarrow$$

$$\arg\max_{\phi,\theta} \mathbb{E}_{x\sim\textbf{Simul}}[\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]]$$

$$\text{subject to: } D_{KL}(q_\phi(z|x)||p(z)) < \epsilon$$

$$\downarrow$$

$$\arg\max_{\phi,\theta} \mathbb{E}_{x\sim\textbf{Simul}}[\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)]] - \beta(D_{KL}(q_\phi(z|x)||p(z)) - \epsilon)$$

$$\downarrow$$

$$\arg\min_{\phi,\theta} \sum_i -\mathbb{E}_{z\sim Q_\phi}[\log P(x^i|g_\theta(z))] + \beta \cdot \mathcal{D}_{KL}(Q_\phi(z|x^i)||P(z)) \quad (\beta > 1)$$

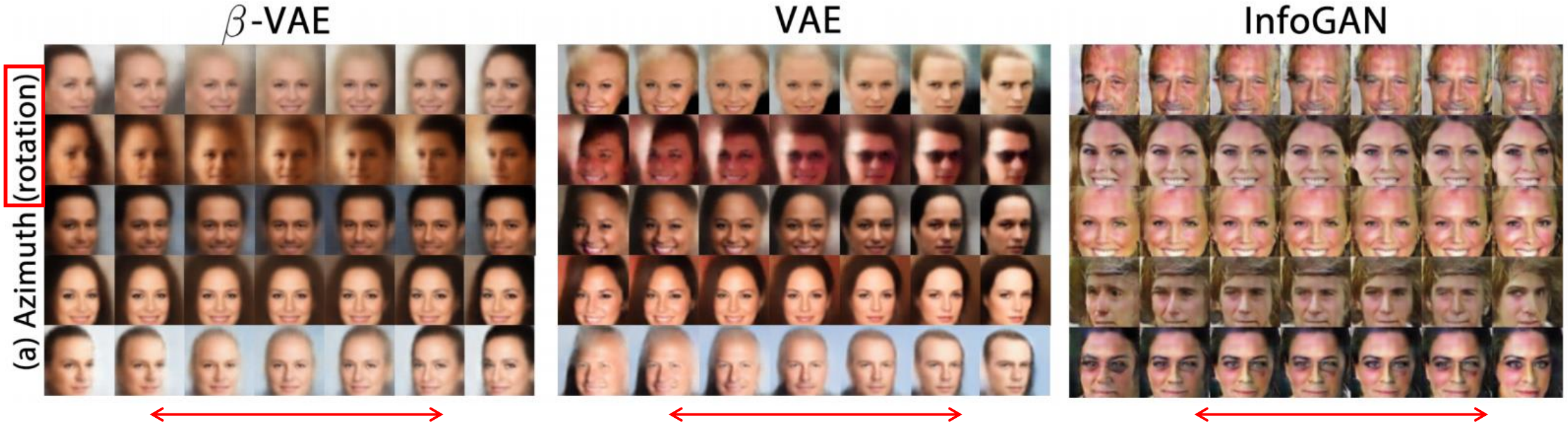
Stronger regularization term results in **lowering the capacity of the latent space**.

Which **forces** the encoder to **focus on the most critical aspects of the input**.

β-VAEs

Experimental results

Dataset: *celebA*



(Proposed, beta = 250)

Several features are simultaneously changed, resulting in a messier transition.

The samples are not clean, and the disentanglement is not clear.

Image: Higgins, Irina, et al. "beta-VAE: Learning basic visual concepts with a constrained variational framework." ICLR'17. 1199 citations.

** I tried to use all notations that are frequently used in the literature, and some of them may not be consistent.*

AnnealedVAEs

[1] Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., & Lerchner, A. (2018). "Understanding disentangling in β-VAE" NeurIPS 2017. 351 citations

[2] Alemi, Alexander A., et al. "Deep variational information bottleneck." ICLR'17. 411 citations.

Understanding disentangling in β-VAE

Christopher P. Burgess, Irina Higgins, Arka Pal,
Loic Matthey, Nick Watters, Guillaume Desjardins, Alexander Lerchner
DeepMind
London, UK
{cpburgess,irinah,arkap,lmatthey,nwatters,gdesjardins,lerchner}@google.com

2. Related works (Baseline methods)

AnnealedVAEs

Objective

Advantage of β-VAE

- Unsupervised disentanglement
(Does not need supervised knowledge
of the data generative factors)

Disadvantage of β-VAE

- What exactly causes the disentanglement?
- Trade-off between disentanglement and reconstruction fidelity

➡

1. **Why β-VAE can perform disentanglement?**
2. **Suggest practical improvements to the trade-off caused by the formulation of β-VAE.**

AnnealedVAEs

Informational bottleneck

Q. What is the best latent representation Z of a data X?

If we want the representation to **contain as much information** of the target Y, the best option is to just set

$$Z = X$$

which is undesirable. The information bottleneck (first proposed by *Tishby) applies an information constant I_c:

$$\max_{\theta} I(Z, Y; \theta) \text{ s.t. } I(X, Z; \theta) < I_c$$

that returns a familiar formulation.

$$\max_{\theta} I(Z, Y; \theta) - \beta I(X, Z; \theta)$$

Therefore the resolution learns an encoding that is

1. **Maximally expressive (informative) of Y**: *first term*
2. **Maximally compressive (to 'forget') of X**: *second term*

AnnealedVAEs

Informational bottleneck

$$\max_{\theta} I(Z, Y; \theta) - \beta I(X, Z; \theta)$$

$$\mathbb{E}_{z\sim Q}[\log p(X|z)] - \mathcal{D}_{KL}(Q(z|X)||p(z))$$

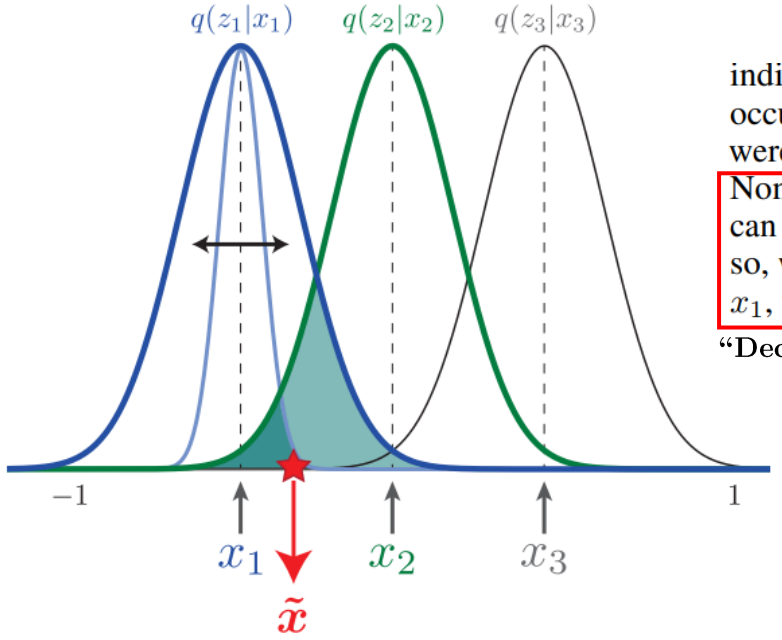
We can consider the posterior distribution as the information bottleneck for the reconstruction task.

In β-VAE, the posterior $q(\mathbf{z}|\mathbf{x})$ is encouraged to match the unit Gaussian prior $p(z_i) = \mathcal{N}(0, 1)$. Since the posterior and the prior are factorised (i.e. have diagonal covariance matrix) and posterior samples are obtained using the reparametrization (Eq. 4) of adding scaled independent Gaussian noise $\sigma_i \epsilon_i$ to a deterministic encoder mean μ_i for each latent unit z_i, we can take an information theoretic perspective and think of $q(\mathbf{z}|\mathbf{x})$ as a set of independent additive white Gaussian noise channels z_i, each noisily transmitting information about the data inputs x_n. In this perspective, the KL divergence term $D_{KL}\big(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z})\big)$ of the β-VAE objective (see Eq. 5) can be seen as an upper bound on the amount of information that can be transmitted through the latent channels per data sample (since it is taken in expectation across the data). The KL divergence is zero when $q(z_i|\mathbf{x}) = p(\mathbf{z})$, i.e μ_i is always zero, and σ_i always 1, meaning the latent channels z_i have zero capacity. The capacity of the latent channels can only be increased by dispersing the posterior means across the data points, or decreasing the posterior variances, which both increase the KL divergence term.

1. The posterior q and prior p is factorized.
2. Think of q as the bottleneck that noisily (Gaussian noise) transmitting information.
3. KL divergence is the upper bound of the information that can be transmitted through the latent channels
4. Zero KL divergence = mean of all latent space is zero = zero information capacity
5. Increasing KL divergence = Increasing capacity 1) Diversifying the posterior means across data points 2) Decreasing posterior variance

AnnealedVAEs

Informational bottleneck



For example, in Figure 1, the sample indicated by the red star might be drawn from the (green) posterior $q(z_2|x_2)$, even though it would occur more frequently under the overlapping (blue) posterior $q(z_1|x_1)$, and so (assuming x_1 and x_2 were equally probable), an optimal decoder would assign a higher log likelihood to x_1 for that sample. Nonetheless, under a constraint of maximising such overlap, the smallest cost in the log likelihood can be achieved by arranging nearby points in data space close together in the latent space. By doing so, when samples from a given posterior $q(z_2|x_2)$ are more likely under another data point such as x_1, the log likelihood $\mathbb{E}_{q(\mathbf{z}_2|\mathbf{x}_2)}[\log p(\mathbf{x}_2|\mathbf{z}_2)]$ cost will be smaller if x_1 is close to x_2 in data space.

"Decrease variance"

Increasing KL divergence = Increasing capacity
1) Diversifying the posterior means across data points 2) Decreasing posterior variance

Increasing capacity = Decreasing overlap between data distributions

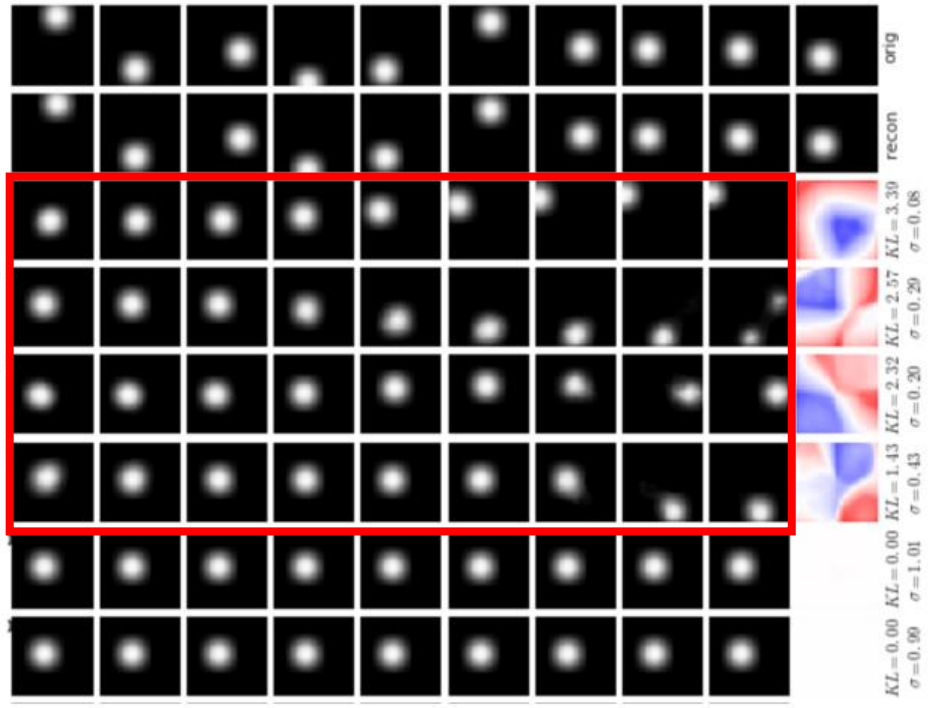
Figure 1: **Connecting posterior overlap with minimizing the KL divergence and reconstruction error.** Broadening the posterior distributions and/or bringing their means closer together will tend to reduce the KL divergence with the prior, which both increase the overlap between them. But, a datapoint $\tilde{x}$ sampled from the distribution $q(z_2|x_2)$ is more likely to be confused with a sample from $q(z_1|x_1)$ as the overlap between them increases. Hence, ensuring neighbouring points in data space are also represented close together in latent space will tend to reduce the log likelihood cost of this confusion.

AnnealedVAEs

Original

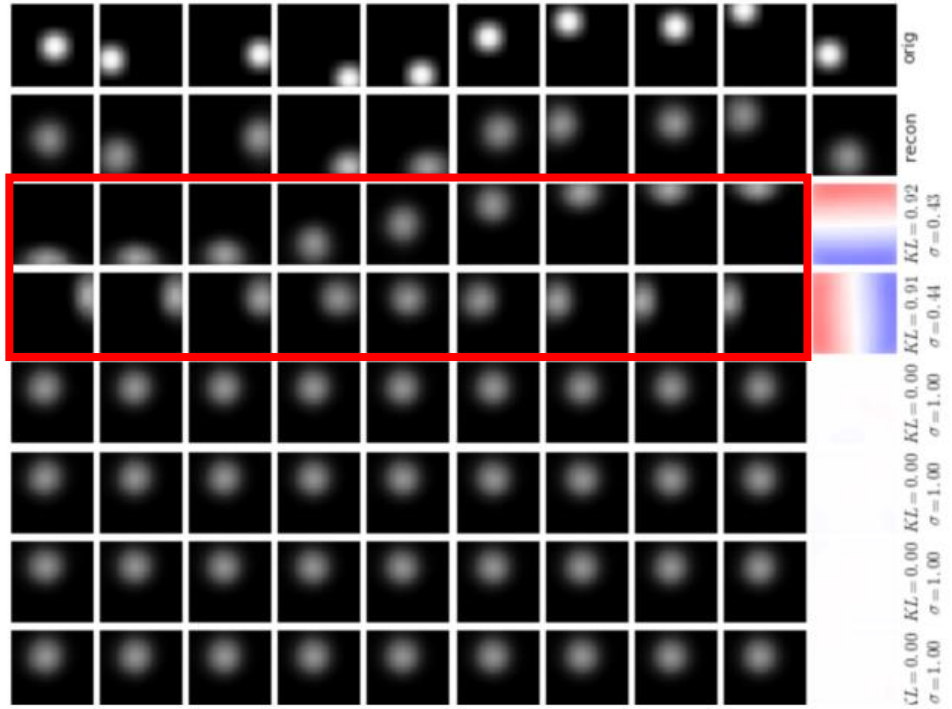
Reconstruction

β = 1



The variation is spread across 4 dimensions

β = 150



The variation is spread across 2 dimensions (with axis alignment)

More pressure of representation of the data (High pressure = stronger bottleneck)

⬇

Answer to "How does β-VAE perform disentanglement?"

AnnealedVAEs

Answer to "Why axis alignment?"

Pressure 1)
Optimal way of representation

Pressure 2)
Diagonal covariance of posterior distribution

At this point we can ask what pressures could encourage this new factor of variation to be encoded into a distinct latent dimension. We hypothesise that two properties of β-VAE encourage this. Firstly, embedding this new axis of variation of the data into a distinct latent dimension is a natural way to satisfy the data locality pressure described in Sec. 4.2. A smooth representation of the new factor will allow an optimal packing of the posteriors in the new latent dimension, without affecting the other latent dimensions. We note that this pressure alone would not discourage the representational axes from rotating relative to the factors. However, given the differing contributions each factor makes to the reconstruction log-likelihood, the model will try to allocate appropriately differing average capacities to the encoding axes of each factor (e.g. by optimising the posterior variances). But, the diagonal covariance of the posterior distribution restricts the model to doing this in different latent dimensions, giving us the second pressure, encouraging the latent dimensions to align with the factors.

AnnealedVAEs

Answer to "Why axis alignment?"

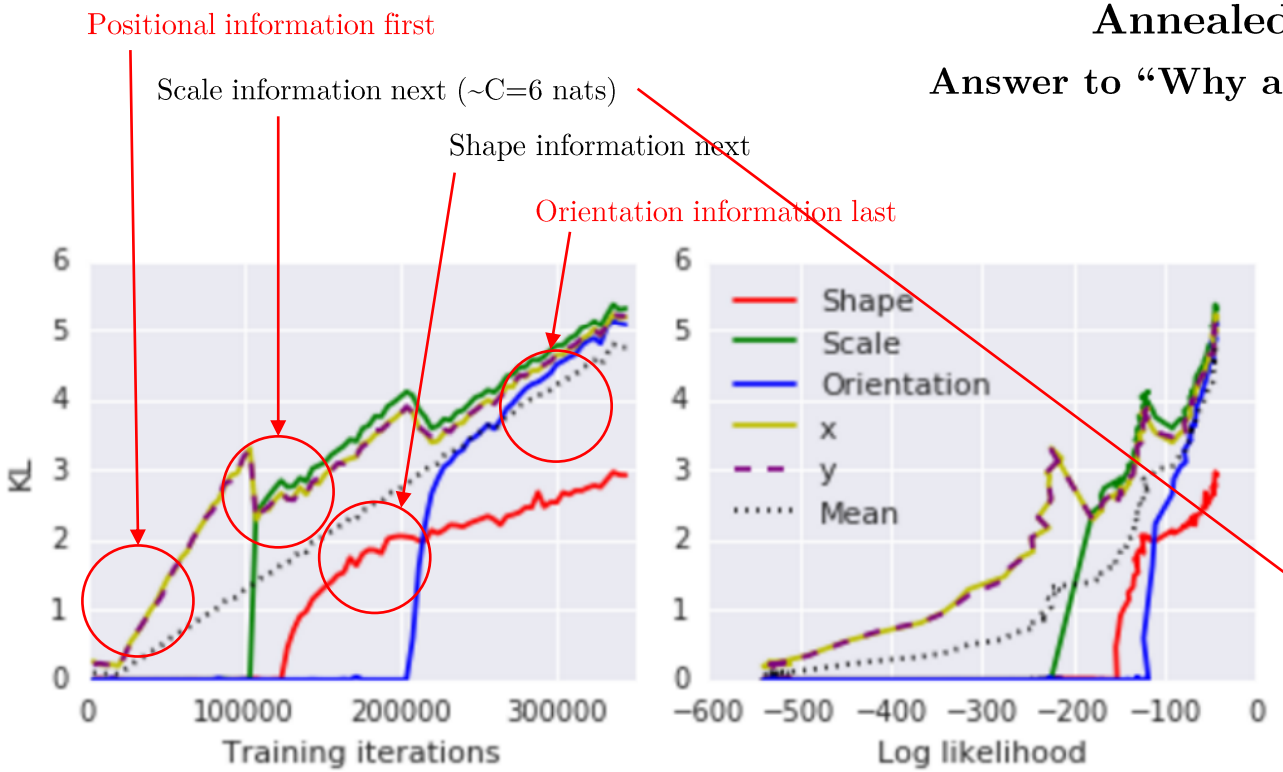
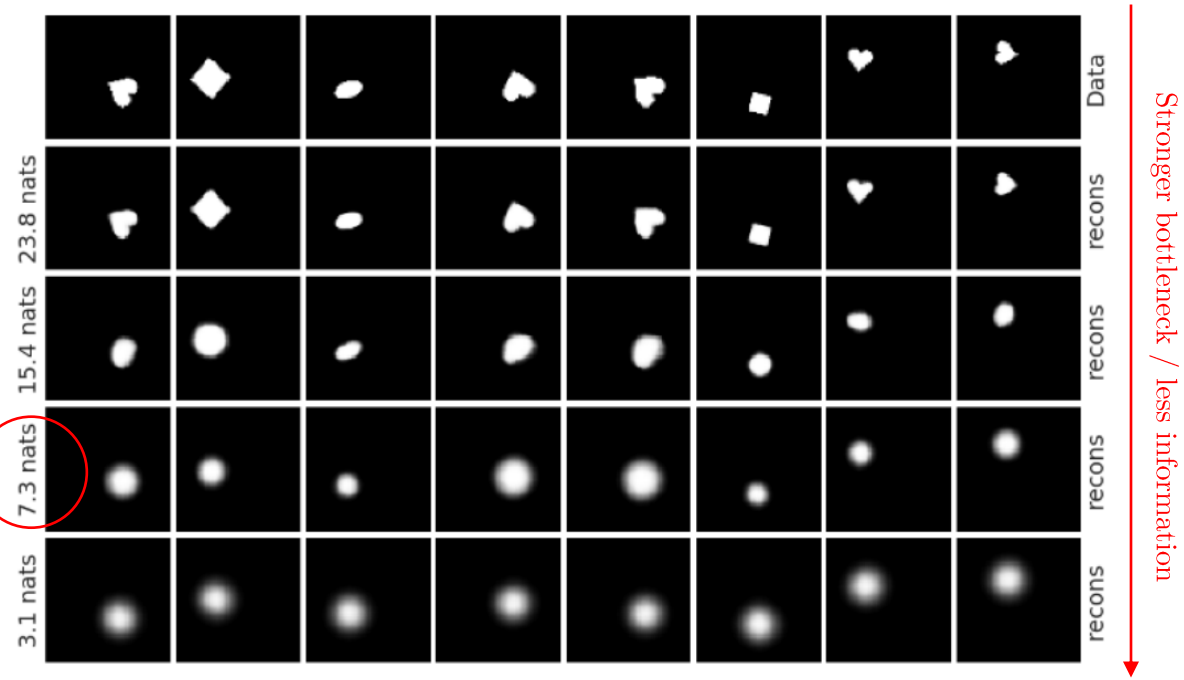


Figure 3: **Utilisation of data generative factors as a function of coding capacity.** Top left: the average KL (in nats) per factor f_i as the training progresses and the total information capacity C of the latent bottleneck $q(\mathbf{z}|\mathbf{f})$ is increased. It can be seen that the early capacity is allocated to positional latents only (x and y), followed by a scale latent, then shape and orientation latents. Top right: same but plotted with respect to the reconstruction accuracy.

*The log can be base-2 to give units in "bits,"
or the natural logarithm base-e with units in "nats."
(From: https://yongchaohuang.github.io/2020-07-08-kl-divergence/)



Bottom: image samples and their reconstructions throughout training as the total information capacity of $\mathbf{z}$ increases and the different latents z_i associated with their respective data generative factors become informative. It can be seen that at 3.1 nats only location of the sprite is reconstructed. At 7.3 nats the scale is also added reconstructed, then shape identity (15.4 nats) and finally rotation (23.8 nats), at which point reconstruction quality is high.

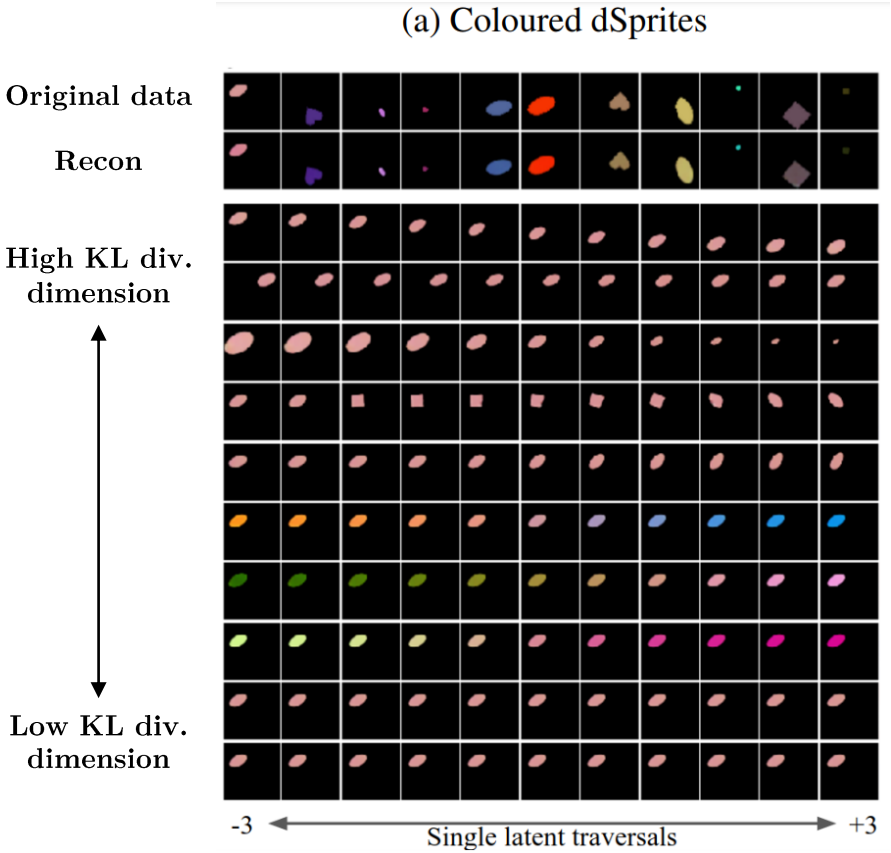
$$\mathcal{L}(\theta, \phi; \mathbf{x}(\mathbf{f}), \mathbf{z}, C) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{f})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \gamma \left| D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{f}) \parallel p(\mathbf{z})\right) - C \right|$$

**This formulation allows to control the "capacity" of the KL div.

AnnealedVAEs

Modified training objective

$$\mathcal{L}(\theta, \phi; \mathbf{x}(\mathbf{f}), \mathbf{z}, C) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{f})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \gamma \left| D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{f}) \parallel p(\mathbf{z})\right) - C \right|$$



A.2 Training Details

γ used was 1000, which was chosen to be large enough to ensure the actual KL was always close to the target KL, C. For dSprites, C was linearly increased from 0 to 25 nats over the course of 100,000 training iterations, for CelebA it was increased to 50 nats.

** I tried to use all notations that are frequently used in the literature, and some of them may not be consistent.*

FactorVAEs

[1] Kim, H., & Mnih, A. (2018, July). Disentangling by factorising. ICML 2018. 458 citations.

[2] Watanabe, S. Information theoretical analysis of multivariate correlation. IBM Journal of research and development, 4(1):66-82, 1960. 596 citations.

Disentangling by Factorising

Hyunjik Kim [1 2] **Andriy Mnih** [1]

[1]DeepMind, UK [2]Department of Statistics, University of Oxford. Correspondence to: Hyunjik Kim <hyunjikk@google.com>.

FactorVAEs

Benefits of Unsupervised disentanglement

1. Humans are able to learn factors of variation **unsupervised**.
2. **Labels are costly** as obtaining them requires a human in the loop.
3. Labels assigned by humans might be **inconsistent** or leave out the factors that are **difficult for humans to identify**.

Contributions

1. We introduce **FactorVAE**, a method for disentangling that gives higher disentanglement scores than β-VAE for the same reconstruction quality.
2. We identify the **weaknesses of the disentanglement metric** of Higgins et al. (2016) (β-VAE) and propose a more robust alternative.
3. ~~We give quantitative comparisons of FactorVAE and β-VAE against InfoGAN's WGAN-GP counterpart for disentanglement.~~

FactorVAEs

Revisit to Higgins et al. (β-VAE)

The distribution of representations for the entire data set is then given by

$$q(z) = \mathbb{E}_{p_{data}(x)}[q(z|x)] = \frac{1}{N}\sum_{i=1}^{N} q(z|x^{(i)}), \quad (1)$$

which is known as the marginal posterior or aggregate posterior, where p_{data} is the empirical data distribution.

Actual calculation

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., 2017; Kim & Mnih, 2018; Chen et al., 2018; Kumar et al., 2017; Rubenstein 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.

Francsco Locatello et al., ICML'19

Since we assume that these factors vary independently, we wish for a factorial distribution $q(z) = \prod_{j=1}^{d} q(z_j)$.

Directly target dimension-wise independence...!

We may further break down this KL term as (Hoffman & Johnson, 2016; Makhzani & Frey, 2017)

$$\mathbb{E}_{p_{data}(x)}[KL(q(z|x)||p(z))] = I(x;z) + KL(q(z)||p(z)),$$

Retain information

Penalizing this is not beneficial.

Remember the second pressure in the previous paper (diagonal covariance in the prior distribution)

Source of disentanglement!

Directly encourage independence

FactorVAEs

Vanilla VAE

$$\frac{1}{N}\sum_{i=1}^{N}\left[\mathbb{E}_{q(z|x^{(i)})}[\log p(x^{(i)}|z)] - KL(q(z|x^{(i)})||p(z))\right]$$

New term

"Total correlation (TC, [2])" $$-\gamma KL(q(z)||\bar{q}(z))$$ $$\bar{q}(z) := \prod_{j=1}^{d} q(z_j)$$

Force the latent representation to dimension-wise independence

⬇ Intractable...!

Algorithm 1 `permute_dims`

Input: $\{z^{(i)} \in \mathbb{R}^d : i = 1, \ldots, B\}$
for $j = 1$ **to** d **do**
  $\pi \leftarrow$ random permutation on $\{1, \ldots, B\}$
  $(z_j^{(i)})_{i=1}^{B} \leftarrow (z_j^{(\pi(i))})_{i=1}^{B}$
end for
Output: $\{z^{(i)} : i = 1, \ldots, B\}$

A more efficient alternative involves sampling a batch from $q(z)$ and then randomly permuting across the batch for each latent dimension (see Alg. 1). This is a standard trick used in the independence testing literature (Arcones & Gine, 1992) and as long as the batch is large enough, the distribution of these samples samples will closely approximate $\bar{q}(z)$.

⬇

$$TC(z) = KL(q(z)||\bar{q}(z)) = \mathbb{E}_{q(z)}\left[\log \frac{q(z)}{\bar{q}(z)}\right]$$

$$\approx \mathbb{E}_{q(z)}\left[\log \frac{D(z)}{1 - D(z)}\right].$$

Use a discriminator D(z) that distinguish

$$q(z) \quad \bar{q}(z)$$

** I tried to use all notations that are frequently used in the literature, and some of them may not be consistent.*

β-TCVAEs

[1] Chen, Ricky TQ, et al. "Isolating sources of disentanglement in variational autoencoders." NIPS'18, 427 citations.

"While Kim & Mnih [8] (FactorVAE) have independently proposed augmenting VAEs with an equivalent total correlation penalty to the β-TCVAE, their proposed training method differs from ours and requires an auxiliary discriminator network. (No need for additional discriminator or hyperparameter)"

Isolating Sources of Disentanglement in VAEs

Ricky T. Q. Chen, Xuechen Li, Roger Grosse, David Duvenaud
University of Toronto, Vector Institute
rtqichen, lxuechen, rgrosse, duvenaud@cs.toronto.edu

β-TCVAE

Decomposition of the regularization term

$$\mathbb{E}_{p(n)}\Big[\mathrm{KL}\big(q(z|n)||p(z)\big)\Big] = \underbrace{\mathrm{KL}\left(q(z,n)||q(z)p(n)\right)}_{\text{(i) Index-Code MI}} + \underbrace{\mathrm{KL}\big(q(z)||\prod_j q(z_j)\big)}_{\text{(ii) Total Correlation}} + \underbrace{\sum_j \mathrm{KL}\left(q(z_j)||p(z_j)\right)}_{\text{(iii) Dimension-wise KL}}$$

Mutual information between the data and latent
(Notice that if this term is zero, it becomes the independent criteria)

Measurement of the dependent between multiple variables
(Penalizing this term is the most impactful factor in beta-VAEs)

Dimension-wise KL divergence
(Penalization of each dimension w.r.t. prior distribution)

We would like to verify the claim that TC is the most important term in this decomposition for learning disentangled representations by penalizing only this term; however, it is difficult to estimate the three terms in the decomposition. In the following section, we propose a simple yet general framework for training with the TC-decomposition using minibatches of data.

β-TCVAE

Estimation of the individual term

$$\mathbb{E}_{p(n)}\Big[\mathrm{KL}\big(q(z|n)||p(z)\big)\Big] = \underbrace{\mathrm{KL}\left(q(z,n)||q(z)p(n)\right)}_{\text{i) Index-Code MI}} + \underbrace{\mathrm{KL}\big(q(z)||\prod_j q(z_j)\big)}_{\text{ii) Total Correlation}} + \underbrace{\sum_j \mathrm{KL}\left(q(z_j)||p(z_j)\right)}_{\text{iii) Dimension-wise KL}}$$

Footnote 3: similar for this also

1. Evaluation of this term depends on the entire dataset.

$$q(z) = \mathbb{E}_{p(n)}[q(z|n)]$$

2. Naïve Monte Carlo approximation from a minibatch from $p(n)$ is likely to underestimate $q(z)$.

With a randomly sampled component, $q(z|n)$ is close to 0, whereas $q(z|n)$ would be large if n is the component that z came from. So it is much better to sample this component and weight the probability appropriately.

To this end, we propose using a weighted version for estimating the function $\log q(z)$ during training, inspired by importance sampling. When provided with a minibatch of samples $\{n_1, ..., n_M\}$, we can use the estimator

$$\mathbb{E}_{q(z)}[\log q(z)] \approx \frac{1}{M}\sum_{i=1}^{M}\left[\log \frac{1}{NM}\sum_{j=1}^{M} q(z(n_i)|n_j)\right] \tag{3}$$

where $z(n_i)$ is a sample from $q(z|n_i)$ (see derivation in Appendix C). This minibatch estimator is biased, since its expectation is a lower bound[3]. However, computing it does not require any additional hyperparameters.

β-TCVAE

3.1.1 Special case: β-TCVAE

With minibatch-weighted sampling, it is easy to assign different weights (α, β, γ) to the terms in (2):

$$\mathcal{L}_{\beta-\text{TC}} := \mathbb{E}_{q(z|n)p(n)}[\log p(n|z)] - \alpha I_q(z;n) - \beta \, \text{KL}\big(q(z)||\prod_j q(z_j)\big) - \gamma \sum_j \text{KL}\left(q(z_j)||p(z_j)\right) \tag{4}$$

While we performed ablation experiments with different values for α and γ, we ultimately find that tuning β leads to the best results. Our proposed β-TCVAE uses $\alpha = \gamma = 1$ and only modifies the hyperparameter β. While Kim & Mnih [8] have proposed an equivalent objective, they estimate TC using an auxiliary discriminator network.

** I tried to use all notations that are frequently used in the literature, and some of them may not be consistent.*

DIP-VAE-I/II

[1] Kumar, A., Sattigeri, P., & Balakrishnan, A. Variational inference of disentangled latent concepts from unlabeled observations. ICLR 2018. 171 citations.

"Unlike β-VAE (Higgins et al., 2017), our approach does not introduce any extra conflict between disentanglement of the latents and the observed data likelihood, …"

Published as a conference paper at ICLR 2018

VARIATIONAL INFERENCE OF DISENTANGLED LATENT CONCEPTS FROM UNLABELED OBSERVATIONS

Abhishek Kumar, Prasanna Sattigeri, Avinash Balakrishnan
IBM Research AI
Yorktown Heights, NY
{abhishk,psattig,avinash.bala}@us.ibm.com

DIP-VAE-I/II

Expected variational posterior $q_\phi(\mathbf{z}) = \int q_\phi(\mathbf{z}|\mathbf{x})p(\mathbf{x})d\mathbf{x}$

Expected posterior $p_\theta(\mathbf{z}) = \int p_\theta(\mathbf{z}|\mathbf{x})p(\mathbf{x})d\mathbf{x}$

In general, the prior $p(\mathbf{z})$ and expected posterior $p_\theta(\mathbf{z})$ will be different, although they may be close (they will be same when $p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$ is equal to $p(\mathbf{x})$). Hence, variational posterior inference of latent variables with disentangled prior naturally encourages inferring factors that are *close* to being disentangled. We think this is the reason that the original VAE (Eq. (1)) has also been observed to exhibit some disentangling behavior on simple datasets such as MNIST (Kingma & Welling, 2013). However, this behavior does not carry over to more complex datasets (Aubry et al., 2014; Liu et al., 2015; Higgins et al., 2017), unless extra supervision on the generative factors is provided (Kulkarni et al., 2015; Karaletsos et al., 2015). This can be due to: (i) $p(\mathbf{x})$ and $p_\theta(\mathbf{x})$ being far apart which in turn causes $p(\mathbf{z})$ and $p_\theta(\mathbf{z})$ being far apart, and (ii) the non-convexity of the ELBO objective which prevents us from achieving the global minimum of $\mathbb{E}_\mathbf{x}\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x}))$ (which is 0 and implies $\mathrm{KL}(q_\phi(\mathbf{z})\|p_\theta(\mathbf{z})) = 0$). In other words, maximizing the ELBO (Eq. (1)) might also result in reducing the value of $\mathrm{KL}(q_\phi(\mathbf{z})\|p(\mathbf{z}))$, however, due to the aforementioned reasons, the gap between $\mathrm{KL}(q_\phi(\mathbf{z})\|p(\mathbf{z}))$ and $\mathbb{E}_\mathbf{x}\,\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x}))$ could be large at the stationary point of convergence. Hence, minimizing $\mathrm{KL}(q_\phi(\mathbf{z})\|p(\mathbf{z}))$ or any other suitable distance $D(q_\phi(\mathbf{z}), p(\mathbf{z}))$ explicitly will give us better control on the disentanglement. This motivates us to add $D(q_\phi(\mathbf{z})\|p(\mathbf{z}))$ as part of the objective to encourage disentanglement during inference, i.e.,

$$\max_{\theta,\phi} \mathbb{E}_\mathbf{x}\left[\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))\right] - \boxed{\lambda\, D(q_\phi(\mathbf{z})\|p(\mathbf{z}))}, \tag{4}$$

where λ controls its contribution to the overall objective. We refer to this as DIP-VAE (for Disentangled Inferred Prior) subsequently.

We have seen similar arguments several times now.

(i) The original data distribution and the reconstructed output being far apart

(ii) The laten code does not actually represent the true latent distribution

Although ELBO helps reducing (ii), it would be better to explicitly add a regularization term for this.

Impossibility of unsup. disentanglement

- 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.

Investigation of current approaches

- We investigate current approaches and their inductive biases in a reproducible large-scale experimental study[1] with a sound experimental protocol for unsupervised disentanglement learning. We implement six recent unsupervised disentanglement learning methods as well as six disentanglement measures from scratch and train more than 12 000 models on seven data sets.

Library that contains this work

- We release `disentanglement_lib`[2], 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.

Challenges in assumptions

- 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) 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.

Suggestions for future research

- 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.

How difficult is the problem of unsupervised disentanglement?

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.

What is inductive bias? [1]

An **inductive bias** allows a learning algorithm to prioritize one solution (or interpretation) over another, independent of the observed data. [...] Inductive biases can **express assumptions** about either the data-generating process or the space of solutions.

Component	Entities	Relations	Rel. inductive bias	Invariance
Fully connected	Units	All-to-all	Weak	-
Convolutional	Grid elements	Local	Locality	Spatial translation
Recurrent	Timesteps	Sequential	Sequentiality	Time translation
Graph network	Nodes	Edges	Arbitrary	Node, edge permutations

Table 1: Various relational inductive biases in standard deep learning components. See also Section 2.

[1] Battaglia, Peter W., et al. "Relational inductive biases, deep learning, and graph networks." (2018).

How difficult is the problem of unsupervised disentanglement?

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

- Assume disentanglement is possible
- *"The problem is very difficult"*
- Not disentangled anymore
- But the latent distribution did not essentially change

In other words, even if we succeed in disentanglement, there are an **infinite number of ways to generate the same latent space** which **we cannot distinguish** from a distribution standpoint.

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

The proof find an explicit function of such a bijective function f.

** Definition of support set*

$$\mathrm{supp}(f) = \{x \in X \mid f(x) \neq 0\}$$

What does this imply?

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.

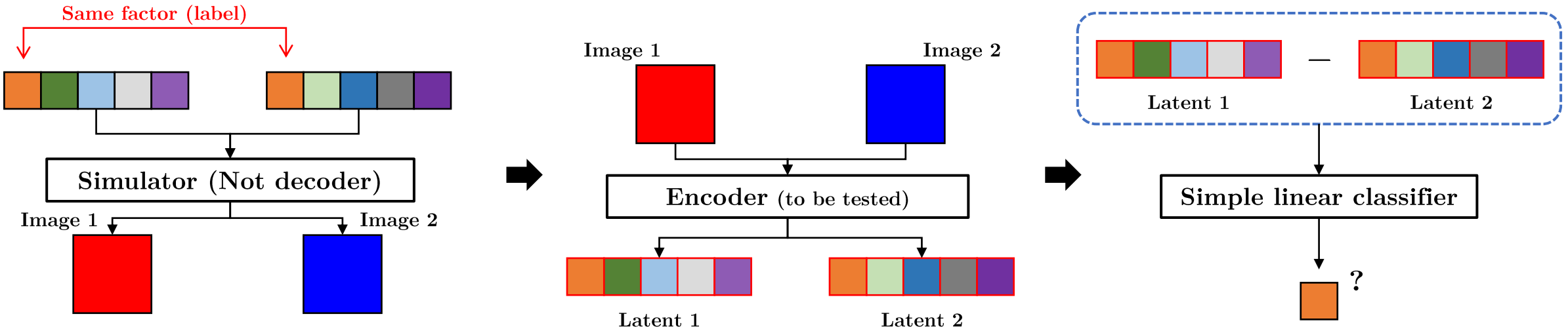
We need additional assumptions (model & data) to narrow down the solution space.

Metrics

BetaVAE metric

Factors of generation (want to disentangle)

Given a dataset $\mathcal{D} = \{X, C, W\}$ as described in Sec. 2, where data points are obtained using a ground truth simulator process $\mathbf{x} \sim \mathbf{Sim}(\mathbf{v}, \mathbf{w})$, and given labels $y \sim Unif[1...K]$ of a subset of the independent data generative factors $\mathbf{v} \in V$ for at least some instances, we train a linear classifier to predict $p(y|\mathbf{z}_{\text{diff}})$. We choose a classifier with a low VC-dimension in order to ensure that it has no capacity to perform nonlinear disentangling itself. Assuming that the dataset $\mathcal{D}$ contains a balanced distribution of ground truth factors $(\mathbf{v}, \mathbf{w})$, we calculate $\mathbf{z}_{\text{diff}}$ as the average pairwise difference between L vectors $\mathbf{z}_{li}$ and $\mathbf{z}_{lj}$, where $q(\mathbf{z}|\mathbf{x}) \sim N(\mu(\mathbf{x}), \sigma(\mathbf{x}))$, and $\mathbf{z}_{li} = \mu(\mathbf{x}_{li})$ and $\mathbf{z}_{lj} = \mu(\mathbf{x}_{lj})$. Images $\mathbf{x}_{li}$ and $\mathbf{x}_{lj}$ are generated using the ground truth factors $(\mathbf{v}_{li}, \mathbf{w}_{li})$ and $(\mathbf{v}_{lj}, \mathbf{w}_{lj})$ using the process $\mathbf{x}_l \sim \mathbf{Sim}(\mathbf{v}_l, \mathbf{w}_l)$. Ground truth factor vectors $(\mathbf{v}_{li}, \mathbf{w}_{li})$ and $(\mathbf{v}_{lj}, \mathbf{w}_{lj})$ are randomly sampled from their corresponding distributions $\mathbf{v}_l \sim p(\mathbf{v})$ $\mathbf{w}_l \sim p(\mathbf{w})$. In each pair $(\mathbf{v}_i, \mathbf{v}_j)$ for all $l \in L$, one of the factors v_k remains unchanged ($v_{ik} = v_{jk}$). The index of this stable generative factor is equal to the label y that the linear classifier is trying to predict.



Metrics

FactorVAE metric

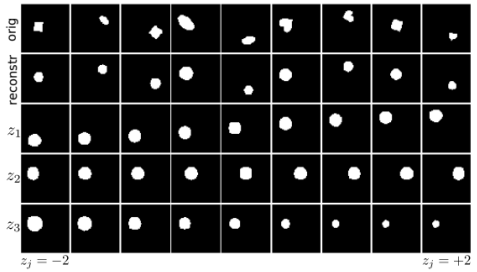


Figure 3. A β-VAE model trained on the 2D Shapes data that scores 100% on metric in Higgins et al. (2016) (ignoring the shape factor). First row: originals. Second row: reconstructions. Remaining rows: reconstructions of latent traversals. The model only uses three latent units to capture x-position, y-position, scale and ignores orientation, yet achieves a perfect score on the metric.

However this metric has several weaknesses. Firstly, it could be sensitive to hyperparameters of the linear classifier optimisation, such as the choice of the optimiser and its hyperparameters, weight initialisation, and the number of training iterations. Secondly, having a linear classifier is not so intuitive – we could get representations where each factor corresponds to a linear combination of dimensions instead of a single dimension. Finally and most importantly, the metric has a failure mode: it gives 100% accuracy even when only $K - 1$ factors out of K have been disentangled; to predict the remaining factor, the classifier simply learns to detect when all the values corresponding to the $K - 1$ factors are non-zero. An example of such a case is shown in Figure 3.

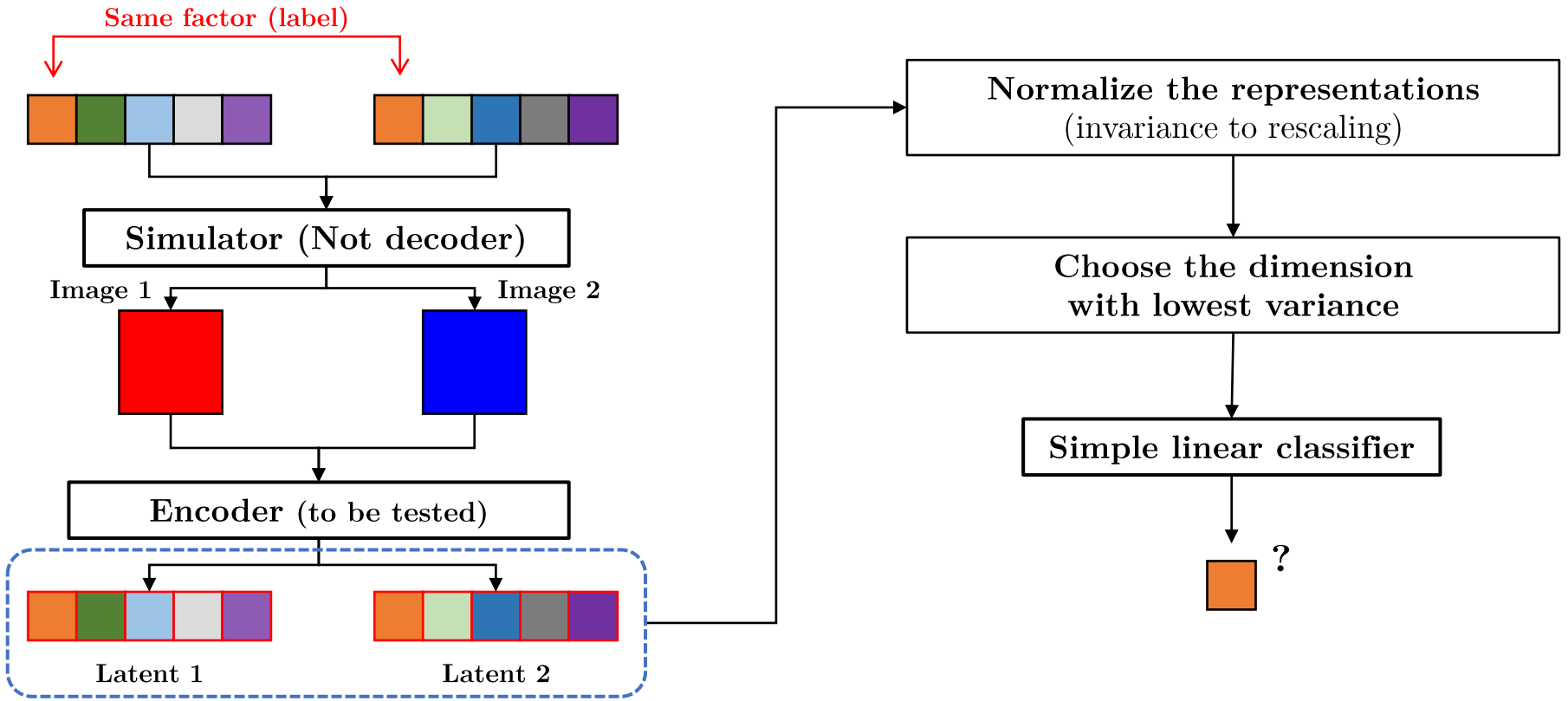
(hyperparameters)

(linear entanglement)

(under entanglement)

Metrics

FactorVAE metric



Metrics

Mutual Information Gap (β-TCVAEs)

Our key insight is that the *empirical mutual information* between a latent variable z_j and a ground truth factor v_k can be estimated using the joint distribution defined by $q(z_j, v_k) = \sum_{n=1}^{N} p(v_k)p(n|v_k)q(z_j|n)$.

Measure the *normalized mutual information* between a ground truth factor and the latent variable.

Note that a single factor can have high mutual information with multiple latent variables. We enforce axis-alignment by measuring the difference between the top two latent variables with highest mutual information. The full metric we call *mutual information gap* (MIG) is then

$$\frac{1}{K}\sum_{k=1}^{K}\frac{1}{H(v_k)}\left(I_n(z_{j^{(k)}};v_k) - \max_{j\neq j^{(k)}} I_n(z_j;v_k)\right) \tag{6}$$

where $j^{(k)} = \text{argmax}_j\, I_n(z_j; v_k)$ and K is the number of known factors. MIG is bounded by 0 and 1.

If the disentanglement is perfect, then one factor will have **high mutual information with only one latent variable**, and therefore the difference between the second largest mutual information will be high.

Metrics

[1] Modularity, [2] DCI Disentanglement, [3] SAP Score

Modularity (Ridgeway & Mozer, 2018) measures if each dimension of $r(\mathbf{x})$ depends on at most a factor of variation using their mutual information. The Disentanglement metric of Eastwood & Williams (2018) (which we call *DCI Disentanglement* for clarity) computes 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. The *SAP score* (Kumar et al., 2017) is the average difference of the prediction error of the two most predictive latent dimensions for each factor.

If the importance is similar for all dimension, low entropy will be observed.

[1] Ridgeway, K., & Mozer, M. C. (2018). Learning deep disentangled embeddings with the f-statistic loss. NeurIPS 2018
[2] Eastwood, C. and Williams, C. K. I. A framework for the quantitative evaluation of disentangled representations. ICLR 2018.
[3] Kumar, A., Sattigeri, P., and Balakrishnan, A. Variational inference of disentangled latent concepts from unlabeled observations. ICLR 2017.

Inductive bias

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 (e.g., the choice of the neural architecture). Each method uses the same convolutional architecture, optimizer, hyperparameters of the optimizer and batch size.

→ Directly consider the effect of latent space constraint

β-VAEs

$$\arg\min_{\phi,\theta} -\sum_i [\mathbb{E}_{z\sim Q_\phi}[\log p(x^i|g_\theta(z))] + \beta \cdot \mathcal{D}_{KL}(Q_\phi(z|x^i)||p(z))]$$

AnnealedVAEs

$$\mathcal{L}(\theta,\phi;\mathbf{x}(\mathbf{f}),\mathbf{z},C) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{f})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \gamma\,|D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{f})\parallel p(\mathbf{z})\right) - C|$$

FactorVAEs

$$\frac{1}{N}\sum_{i=1}^{N}\left[\mathbb{E}_{q(z|x^{(i)})}[\log p(x^{(i)}|z)] - KL(q(z|x^{(i)})||p(z))\right]$$
$$-\gamma KL(q(z)||\bar{q}(z)). \quad (2)$$

β-TCVAEs

$$\mathcal{L}_{\beta-\mathrm{TC}} := \mathbb{E}_{q(z|n)p(n)}[\log p(n|z)] - \alpha I_q(z;n) - \beta\,\mathrm{KL}(q(z)||\prod_j q(z_j)) - \gamma\sum_j \mathrm{KL}\,(q(z_j)||p(z_j))$$

DIP-VAE

$$\max_{\theta,\phi}\mathbb{E}_{\mathbf{x}}\left[\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))\right] - \lambda\, D(q_\phi(\mathbf{z})\|p(\mathbf{z}))$$

Key experimental results 1
Can current methods enforce an uncorrelated aggregated posterior and representation?

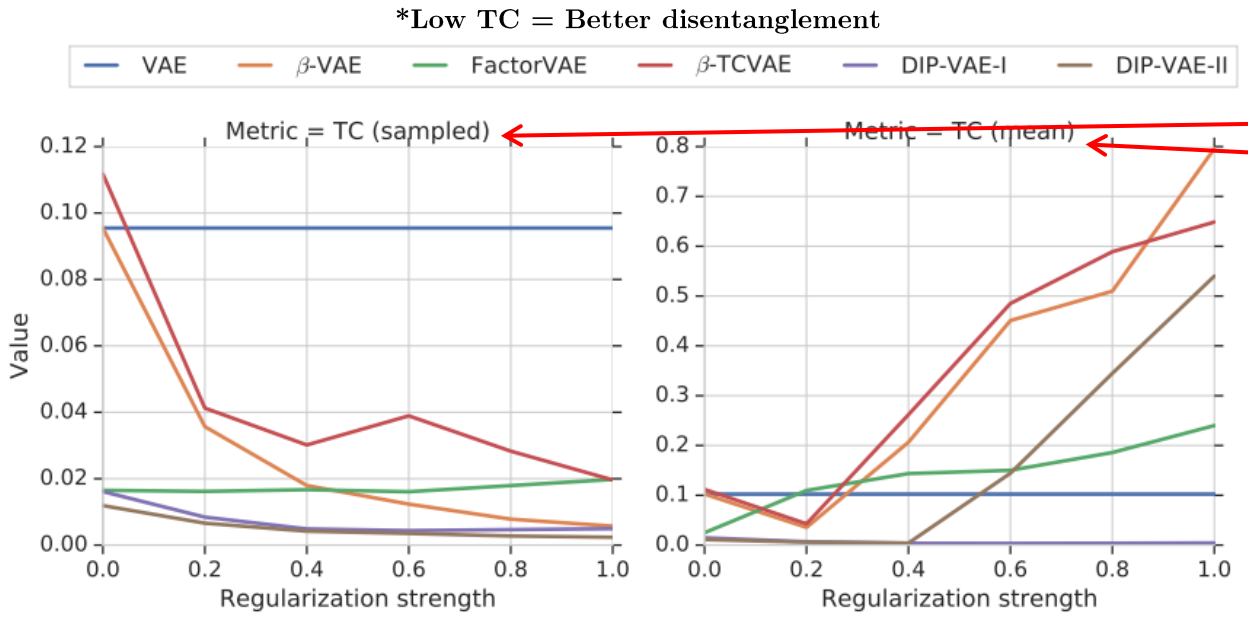
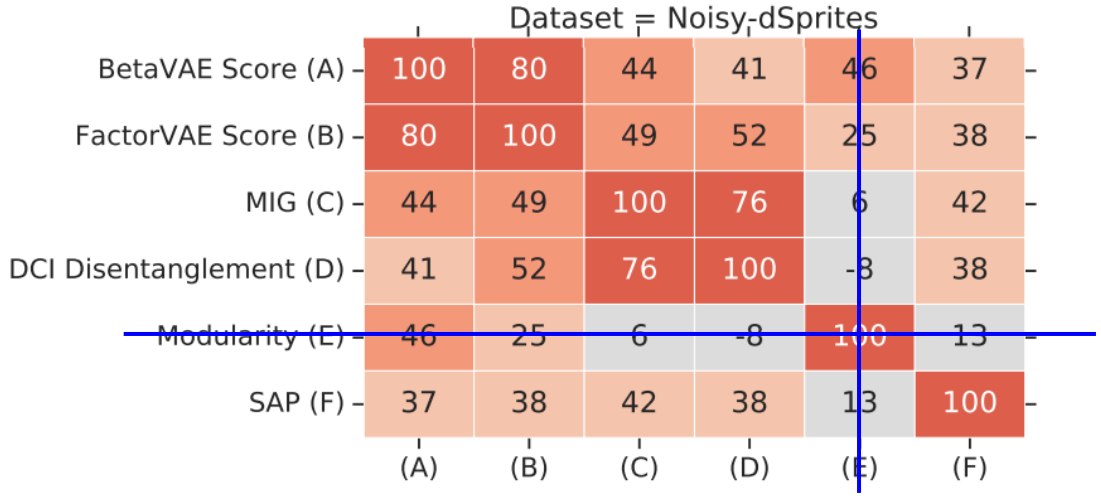


Figure 1. Total correlation based on a fitted Gaussian of the sampled (left) and the mean representation (right) plotted against regularization strength for Color-dSprites and approaches (except AnnealedVAE). The total correlation of the sampled representation decreases while the total correlation of the mean representation increases as the regularization strength is increased.

While many of the considered methods aim to enforce a factorizing and thus uncorrelated aggregated posterior (e.g., regularizing the total correlation of the sampled representation), 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 the impact of this, we compute the total correlation of both the mean and the sampled representation based on fitting Gaussian distributions for each data set, model and hyperparameter value (see Appendix C and I.2 for details).

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.

Key experimental results 2
How much do the disentanglement metrics agree?



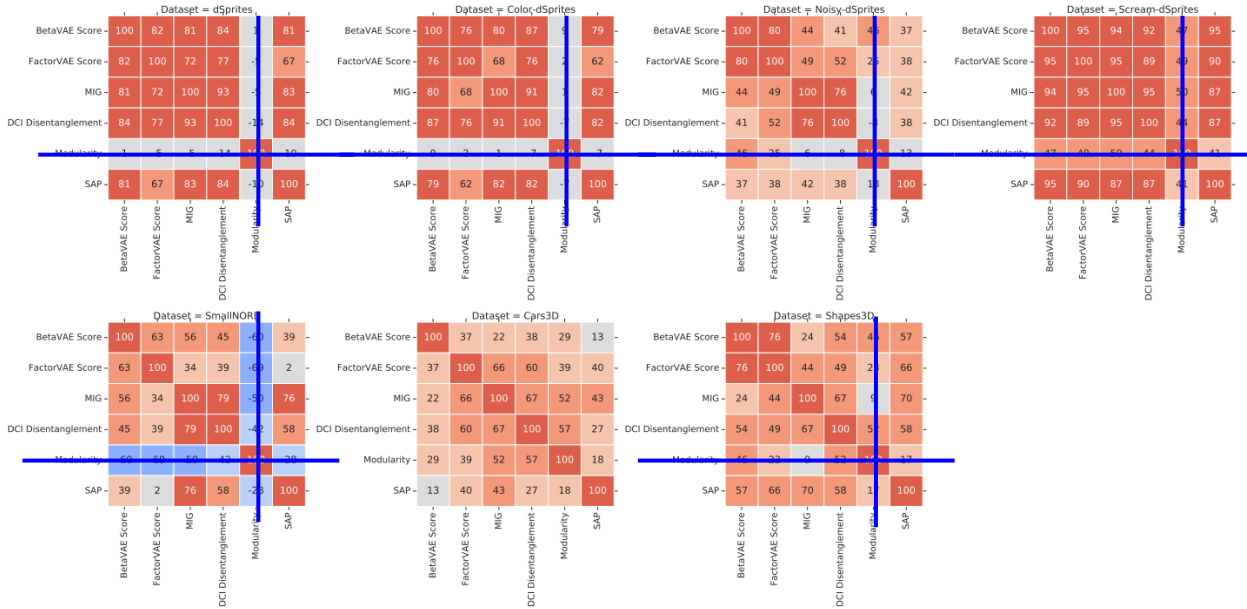


Figure 2. Rank correlation of different metrics on Noisy-dSprites Overall, we observe that all metrics except Modularity seem mildly correlated with the pairs BetaVAE and FactorVAE, and MIG and DCI Disentanglement strongly correlated with each other.

Key experimental results 3

How important are different models and hyperparameters for disentanglement?

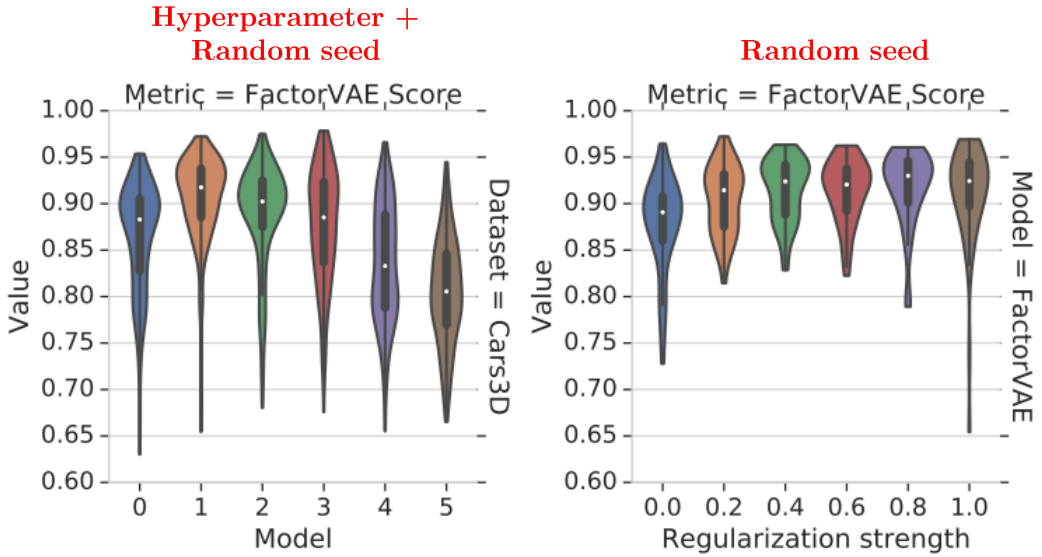


Figure 3. (left) FactorVAE score for each method on Cars3D. Models are abbreviated (0=β-VAE, 1=FactorVAE, 2=β-TCVAE, 3=DIP-VAE-I, 4=DIP-VAE-II, 5=AnnealedVAE). The variance is due to different hyperparameters and random seeds. The scores are heavily overlapping. (right) Distribution of FactorVAE scores for FactorVAE model for different regularization strengths on Cars3D. In this case, the variance is only due to the different random seeds. We observe 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.

Implication. 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.

Key experimental results 3
How important are different models and hyperparameters for disentanglement?

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

	BetaVAE Score	DCI Disentanglement	FactorVAE Score	MIG	Modularity	SAP
Cars3D	1%	36%	26%	34%	37%	13%
Color-dSprites	30%	39%	52%	26%	23%	29%
Noisy-dSprites	17%	21%	17%	11%	41%	6%
Scream-dSprites	89%	50%	76%	45%	60%	56%
Shapes3D	31%	21%	14%	20%	26%	10%
SmallNORB	68%	71%	58%	71%	62%	62%
dSprites	29%	41%	47%	26%	29%	31%

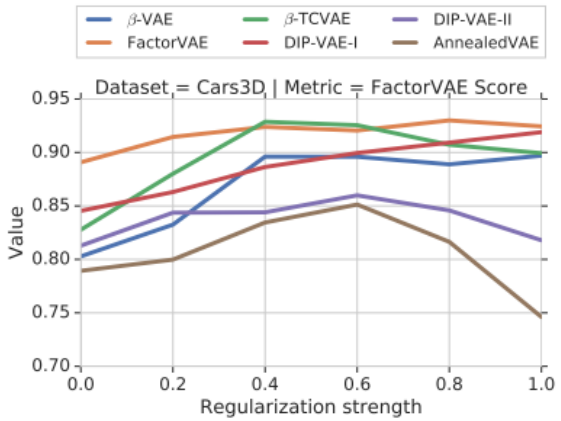
(b) Percentage of variance explained regressing the disentanglement scores on the different data sets from the Cartesian product of objective function and regularization strength.

	BetaVAE Score	DCI Disentanglement	FactorVAE Score	MIG	Modularity	SAP
Cars3D	4%	69%	42%	59%	51%	17%
Color-dSprites	69%	80%	61%	76%	40%	56%
Noisy-dSprites	26%	42%	25%	29%	50%	20%
Scream-dSprites	93%	74%	83%	66%	68%	75%
Shapes3D	61%	78%	53%	59%	49%	35%
SmallNORB	87%	89%	81%	88%	72%	82%
dSprites	59%	77%	54%	72%	39%	56%

Rest of the variance is due to random seed (open for debate)

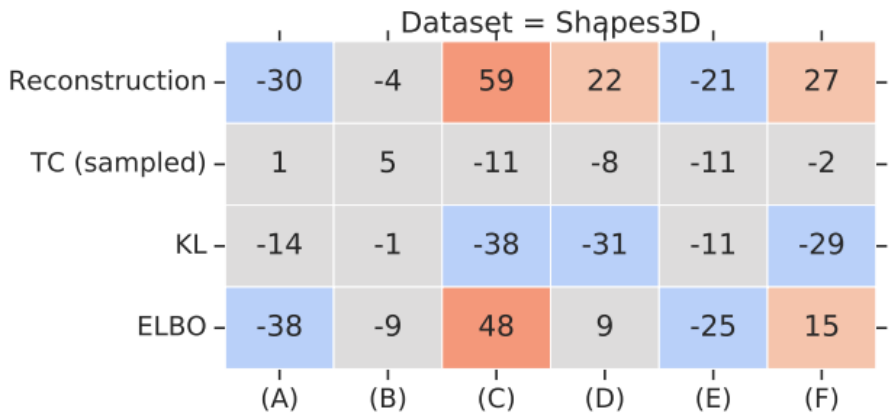
Key experimental results 4
Are there reliable recipes for model selection?

Model type & Reg. strength



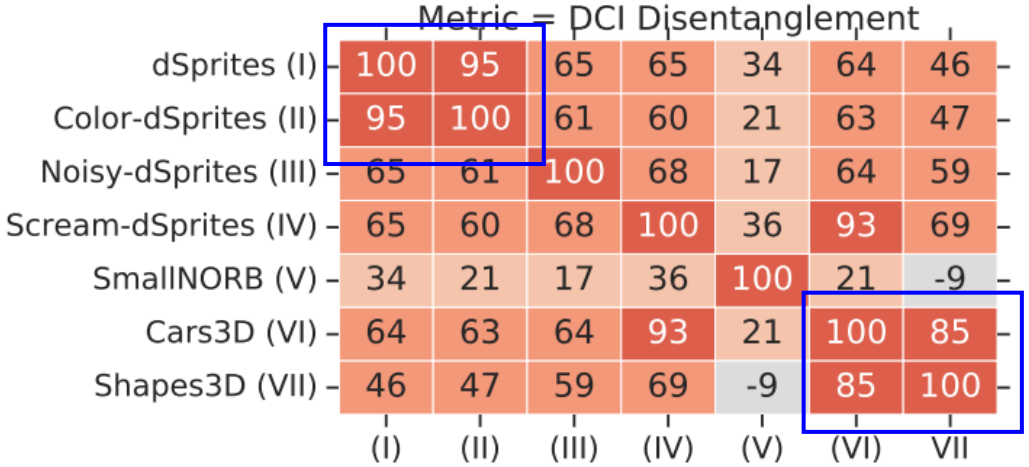
Overall, there seems to be no model consistently dominating all the others and for each model there does not seem to be a consistent strategy in choosing the regularization strength to maximize disentanglement scores. Furthermore, even if we could identify a good objective function and corresponding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

Unsup. score vs. Metrics



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.

Transferability of hyperparameters between datasets



We find a strong and consistent correlation between dSprites and Color-dSprites. While these results 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.

If we choose the same metric and the same data set (but a different random seed), we obtain a score of 80.7%. If we aim to transfer for the same metric across data sets, we achieve around 59.3%. Finally, if we transfer both across metrics and data sets, our performance drops to 54.9%.

Less than half success rate

Key experimental results 5
Are these disentangled representations useful for downstream tasks in terms of the sample complexity of learning?

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).

Seem like a variation on the disentanglement metrics

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.

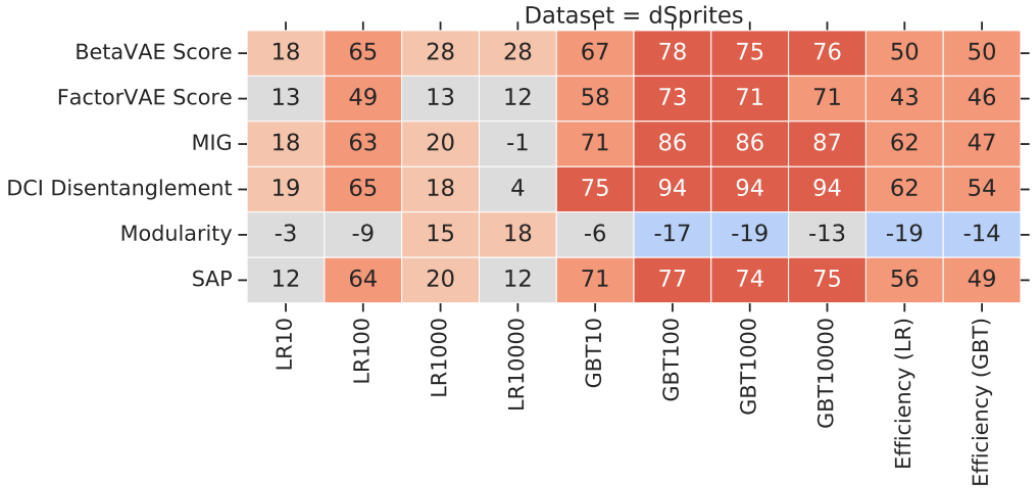
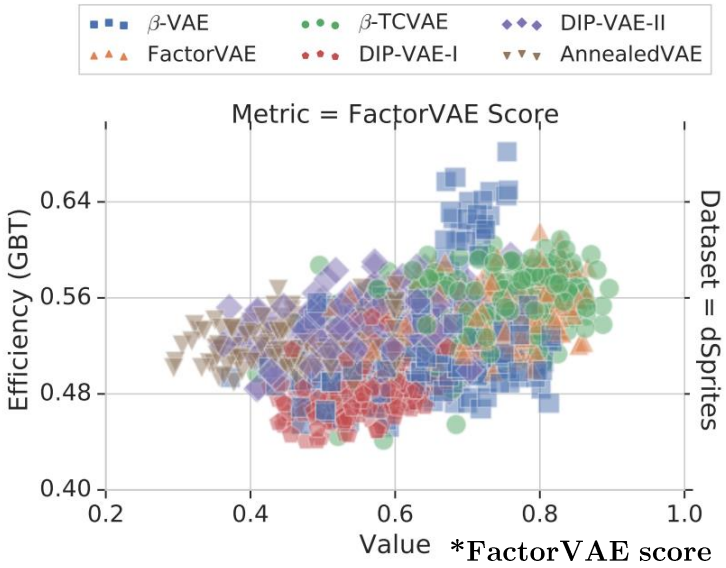


Figure 5. Rank correlations between disentanglement metrics and downstream performance (accuracy and efficiency) on dSprites.

Key experimental results 5
Are these disentangled representations useful for downstream tasks in terms of the sample complexity of learning?



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. Figure 6 show the sample efficiency of learning (based on GBT) versus the FactorVAE Score on dSprites. We do not observe that higher disentanglement scores reliably lead to a higher sample efficiency.

6. Conclusion

1. 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**.

2. **Random seeds and hyperparameters** seem to matter more than the model but tuning seem to require supervision.

3. We **did not observe** that increased disentanglement implies a **decreased sample complexity of learning downstream tasks**.

4. **Unsupervised disentanglement is hard to achieve unless accompanied by additional inductive bias.**

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 $g : \text{supp}(\mathbf{z}) \to [0, 1]^d$ defined by

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

Since P admits a density $p(\mathbf{z}) = \prod_i p(\mathrm{z}_i)$, the function g is bijective and, for almost every $\boldsymbol{v} \in \text{supp}(\mathbf{z})$, it holds that $\frac{\partial g_i(\boldsymbol{v})}{\partial v_i} \neq 0$ for all i and $\frac{\partial g_i(\boldsymbol{v})}{\partial v_j} = 0$ 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 $h : (0, 1]^d \to \mathbb{R}^d$ defined by

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

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

Let $\boldsymbol{A} \in \mathbb{R}^{d \times d}$ be an arbitrary orthogonal matrix with $A_{ij} \neq 0$ for all $i = 1, 2, \ldots, d$ and $j = 1, 2, \ldots, d$. 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 $\boldsymbol{v}$ with $v_1 = \sqrt{\alpha}$ and $v_i = \sqrt{\frac{1-\alpha}{d-1}}$ for $i = 2, 3, \ldots, d$. By construction, we have $\boldsymbol{v}^T \boldsymbol{v} = 1$ and both $v_i \neq 0$ and $v_i \neq \sqrt{\frac{1}{2}}$ for all $i = 1, 2, \ldots, d$. Define the matrix $\boldsymbol{A} = \boldsymbol{I}_d - 2\boldsymbol{v}\boldsymbol{v}^T$ and note that $A_{ii} = 1 - 2v_i^2 \neq 0$ for all $1, 2, \ldots, d$ as well as $A_{ij} = -v_i v_j \neq 0$ for all $i \neq j$. Furthermore, $\boldsymbol{A}$ is orthogonal since

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

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

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

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

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

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

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