Unsupervised Disentanglement Literature Review and Exploration with Scale

Berkan Ottlik, Jay Ram, Aaron Liss
Department of Computer Science
Columbia University
New York City, NY 10027
{ berkan.ottlik, jcr2211, ajl2277 }@columbia.edu

Abstract

We explore the field of unsupervised disentanglement, which aims to disentangle underlying factors of variations in representations in hopes that this creates more useful representations in models. Disentanglement has many different interpretations, so we start by explaining a probabilistic and a symmetry-based definition of disentanglement. We then give an overview of two influential methods in the field, namely β-VAE and GeoManCEr. Additionally, we describe two quantitative disentanglement metrics, SAP and MIG. We finish with some experiments analyzing how the scale of models impacts disentanglement and discuss future directions in the field.

1 Introduction to Unsupervised Disentanglement

Learning good representation of data is essential for success in machine learning. Bengio et al hypothesized that one disentangling underlying factor of variation in representations is important for good representations [1]. There are different factors of variation that cause certain changes in the data, and oftentimes in the real world, only a few of them occur at a time. For instance, if we have a picture of a cat, there are many factors of variation that could effect the way the image looks. The lighting could change, the lens could change, the cat hair color could change, the cat positioning could change, etc. We want our models to be able to disentangle that the entity that is the cat, from the lighting conditions it is in, or the position that it is in.

Disentanglement of factors of variation is not the same as learning invariant features though. Learning invariant features means creating representations that don't necessarily preserve information that is not directly required for the task. In the example with a picture of a cat, if we are learning an invariant feature of the cat, our representation can discard all information about the lighting, the cat color, and more. On the other hand, a disentangled representation, would aim to disentangle as many factors of variation as possible while discarding as little information about the data as is practical.

The dream is that we can do this disentanglement in an unsupervised manner, meaning that the underlying factors of variation or data labels are not given to us. This will likely require very large amounts of data, which could be a reason why many current approaches to unsupervised disentanglement don't work very well.

Disentanglement isn't an original idea to deep learning, algorithms such as independent component analysis (ICA) [11] have been attempting to do similar things for multiple decades.

1.1 Definitions of Disentanglement

Unfortunately, notions of disentanglement are mostly not rigorously defined. It is hard to define exactly what it means for a representation to be disentangled, it is hard to quantify how disentangled a representation is, and there are disagreements about how factors of variation should be represented (if they should for example be linearly separable, axis aligned, etc).

1.1.1 Probabilistic Disentanglement

Given some dataset of observations $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, we assume that there exists some number of generative processes g_i that produce the observations from a small set of corresponding K_i independent generative factors $\mathbf{c}_i$. For each i we have $g : \mathbf{c}_n \mapsto \mathbf{x}_n$, where $p(\mathbf{c}_n) = \prod_{j=1}^{K} p(c_n^j)$. A model has learned a disentangled representation if it learns to invert a generative process g_i and recover a latent representation $\mathbf{z} \in \mathbb{R}^L$ so that it best explains the observed data $p(\mathbf{z}, \mathbf{x}) \approx p(\mathbf{c}_i, \mathbf{x})$, and factorizes the same way as the corresponding data generative factors $\mathbf{c}_i$ [3].

1.1.2 Symmetry-Based Disentanglement

However, there is one paper that gives a more rigorous, symmetry-based definition of disentanglement [5].

The idea is that there are many so called "symmetry transformations" in the world, that change certain aspects of the world state, while keeping others invariant. At a high level, a representation is defined as disentangled if it can be decomposed into a number of subspaces, each of which is compatible with and can be transformed independently by a unique symmetry transformation.

Going back to our example of the picture of the cat, an example of a symmetry transformation would be translating the cat across the image. This would change the location of the cat, but leave other factors such as the identity of color of the cat invariant.

Definition 1.1 (Symmetry-Based Disentangled Representation)**.** *Let W be the set of world states, G be a group*[1] *that acts on those world states which factorizes as $G = G_1 \times \ldots \times G_m$ and $f : W \mapsto Z$ be a mapping to a latent representation space Z. The representation Z is said to be disentangled with respect to the group factorization $G = G_1 \times \ldots G_m$ if:*

(i) There exists an action of G on Z.

(ii) The map $f : W \mapsto Z$ is equivariant between the actions of G on W and Z, i.e. $\forall g \in G : \forall w \in W : g \cdot f(w) = f(g \cdot w)$.

(iii) There is a fixed decomposition $Z = Z_1 \times \ldots \times Z_m$ such that each Z_i is invariant to the action of G_j for all j except $j = i$.

One can also define linearly symmetry based disentangled representations where the group actions transform their corresponding disentangled subspace linearly. We won't go into the details of this here.

1.2 Challenging Current Progress in Unsupervised Disentanglement

Unfortunately, progress has been rather slow in unsupervised disentanglement and there are a few core problems as well.

For one, it has been theoretically proven that the unsupervised learning of disentangled representations is fundamentally impossible without inductive biases both on the considered learning approaches and the data sets. The argument behind this is similar to arguments in causality and ICA literature, which say that once we observe some data $\mathbf{x}$, we can construct infinitely many generative models with the same marginal distribution of $\mathbf{x}$. It is now impossible to know which is the true causal generative model for this given data. This doesn't mean unsupervised disentanglement is hopeless though, because in practice, if we choose the right inductive biases, we can perhaps find useful disentanglements for downstream tasks.

Furthermore, experiments reveal that disentanglement performance seems to depend more on random seeds and hyperparameters than it does on model choice and choice of objective function. These results are kind of sad, but we were curious if scale is the answer here! Perhaps these experiments

[1] If you are not familiar with group theory (or any of the other math in this report), I advise you to look at our project math notes, which give an overview of the relevant mathematical concepts.

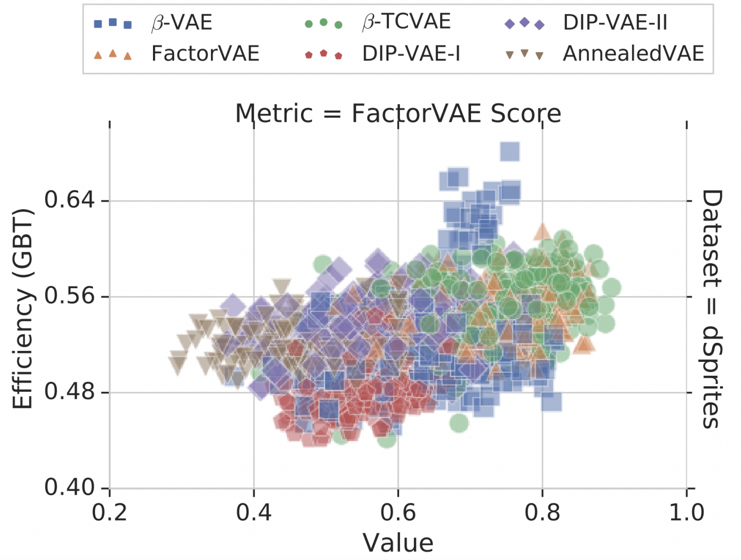


Figure 1: Statistical efficiency of the FactorVAE Score for learning a GBT downstream task on dSprites. Higher disentanglement scores don't reliably lead to higher sample efficiency [10].

are just on such small datasets with such small models that we can't make great sense with these disentanglement metrics. Later in this report, we will run experiments evaluating how disentangled common models are.

Additionally, it seems that models that have more disentanglement itself, don't seem to reliably decrease the sample complexity for learning downstream tasks. That being said, these empirical results are only on a few datasets with a small sample of models, the authors even say they should be treated with caution. Furthermore, disentanglement might have different benefits such as increased interpretability and fairness.

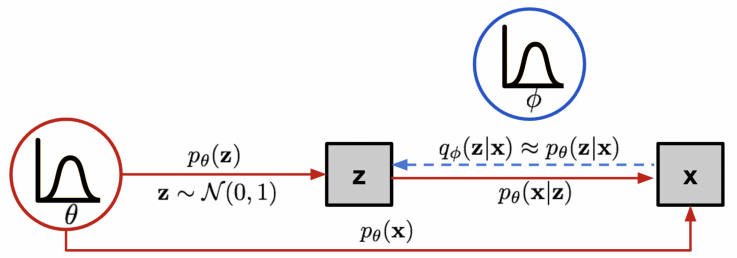
2 β-VAE

β-VAE is one of the most popular disentanglement methods around today, so we will give a brief explanation of it, as it will be used in our experiments [4].

2.1 Background on Autoencoders and Variational Autoencoders

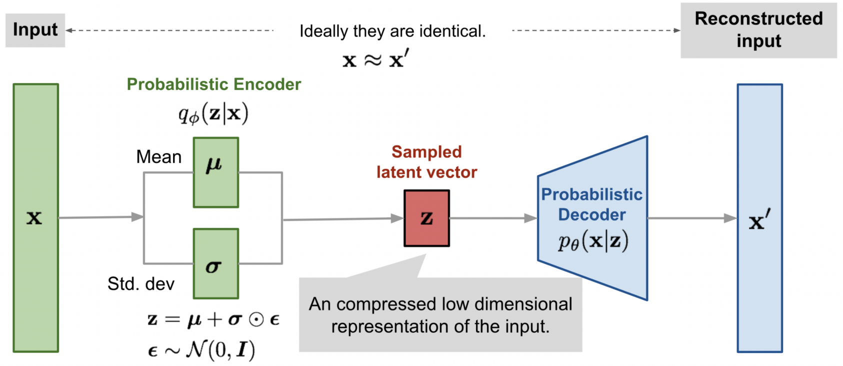
We will only offer a brief explanation of these architectures, for more details please visit this site (https://lilianweng.github.io/posts/2018-08-12-vae/). Autoencoders are neural networks that essentially aim to learn an identity function in an unsupervised manner [6]. They try to reconstruct an input by first compressing the input. We can denote our encoder $g_\phi : \mathbb{R}^d \mapsto \mathbb{R}^b$ and our decoder as $f_\theta : \mathbb{R}^b \mapsto \mathbb{R}^d$, where our input data is $\mathbf{x} \in \mathbb{R}^d$ and our bottleneck compressed representation is $\mathbf{z} \in \mathbb{R}^b$. Our model tries to minimize some loss (such as mean squared error) between the input image and the reconstruction as follows, $\min \mathcal{L}(\mathbf{x}, \mathbf{x}')$ where $\mathbf{x}' = f_\theta(g_\phi(\mathbf{x}))$.

Variational autoencoders are more rooted in variational bayesian methods and graphical models [8]. Instead of mapping to some fixed latent vector $\mathbf{z}$, we map to a latent distribution p_θ. We can now describe the relationship between the input $\mathbf{x}$ and our latent vector $\mathbf{z}$ by the prior $p_\theta(\mathbf{z})$, the likelihood $p_\theta(\mathbf{x}|\mathbf{z})$, and the posterior $p_\theta(\mathbf{z}|\mathbf{x})$. If we have the ideal parameters θ^*, we can generate a $\mathbf{x}^{(i)}$ by first sampling a $\mathbf{z}^{(i)}$ from a prior distribution $p_{\theta^*}(\mathbf{x})$, and then sampling $\mathbf{x}^{(i)}$ from the conditional distribution $p_{\theta^*}(\mathbf{x}|\mathbf{z} = \mathbf{z}^{(i)})$. The optimal θ^* maximizes the probability of generating real data samples as follows $\theta^* = \arg\max_\theta \prod_{i=1}^n p_\theta(\mathbf{x}^{(i)})$. Now we can try to calculate $p_\theta(\mathbf{x}^{(i)})$ as follows, $p_\theta(\mathbf{x}^{(i)}) = \int p_\theta(\mathbf{x}^{(i)}|\mathbf{z}) p_\theta(\mathbf{z}) d\mathbf{z}$, but this is very expensive. Instead, we will try to directly approximate the posterior as $p_\theta(\mathbf{z}|\mathbf{x}) \approx q_\phi(\mathbf{z}|\mathbf{x})$. We can use the following graphical model to represent the process.



ϕ
θ
$p_\theta(\mathbf{z})$
$\mathbf{z} \sim \mathcal{N}(0,1)$
z
$q_\phi(\mathbf{z}|\mathbf{x}) \approx p_\theta(\mathbf{z}|\mathbf{x})$
$p_\theta(\mathbf{x}|\mathbf{z})$
x
$p_\theta(\mathbf{x})$

Figure 2: The graphical model involved in Variational Autoencoder. Solid lines denote the generative distribution $p_\theta(.\ )$ and dashed lines denote the distribution $q_\phi(\mathbf{z}|\mathbf{x})$ to approximate the intractable posterior $p_\theta(\mathbf{z}|\mathbf{x})$ (Credits to Lilian Weng).



Input
Ideally they are identical.
$\mathbf{x} \approx \mathbf{x}'$
Reconstructed input
Probabilistic Encoder
$q_\phi(\mathbf{z}|\mathbf{x})$
Mean
$\boldsymbol{\mu}$
x
$\boldsymbol{\sigma}$
Std. dev
$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$
$\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$
Sampled latent vector
z
An compressed low dimensional representation of the input.
Probabilistic Decoder
$p_\theta(\mathbf{x}|\mathbf{z})$
$\mathbf{x}'$

Figure 3: Illustration of variational autoencoder model with the multivariate Gaussian assumption (Credits to Lilian Weng).

We can think of this as an autoencoder by viewing $q_\phi(\mathbf{z}|\mathbf{x})$ as the probabilistic encoder and analog to the encoder $g_\phi(\mathbf{x})$, and $f_\theta(\mathbf{x}|\mathbf{z})$ and the probabilistic decoder and analog to the decoder $f_\theta(\mathbf{z})$.

For the loss, we will try to minimize the following loss.

$$\mathcal{L}_{VAE}(\theta, \phi) = -\log p_\theta(\mathbf{x}) + D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x}))$$

Through some algebra, we can reformulate this as follows.

$$= -\mathbb{E}_{z \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) + D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}))$$

Unfortunately, in the current state, we can't actually backpropagate and train our model. This is because we can't sample $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$ during backpropagation, so we need to use the reparameterization trick, which expresses the random variable $\mathbf{z}$ as a deterministic variable. It is common to model $q_\phi(\mathbf{z}|\mathbf{x})$ as a multivariate Gaussian with a diagonal covariance structure as follows.

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}, \text{ where } \boldsymbol{\epsilon} \sim N(0, I)$$

2.2 Understanding β-VAE

β-VAE is just a slight variation on VAE. We first reformulate the objective in terms of a constrained optimization problem as a constrained optimization problem.

$$\max_{\phi,\theta} \mathbb{E}_{x \sim \mathcal{D}}[\mathbb{E}_{z \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z})]$$

$$\text{subject to } D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})) < \delta$$

We can rewrite this as a Lagrangian using the Lagrangian multiplier β. Now, with the power of algebra, we can rewrite the loss as follows.

$$\mathcal{L}_{\beta-VAE}(\theta, \phi) = -\mathbb{E}_{z \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) + \beta D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}))$$

Clearly, when $\beta = 1$, this is the same as the regular VAE loss. When $\beta > 1$, the model is more incentivised to minimize the $D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}))$ term. The authors hypothesized this helps to learn disentangled representations of the conditionally independent data generative factors $\mathbf{v}$. This is because the constraints limit the capacity of the latent variable $\mathbf{z}$, and therefore should encourage the model to learn more efficient representations of the data. These efficient representations will hopefully also be disentangled because the underlying conditionally independent data generative factors $\mathbf{v}$ generate $\mathbf{x}$ and with a higher β term the objective function encourages conditional independence of $q_\phi(\mathbf{z}|\mathbf{x})$, so therefore this should also encourage a disentangled representation (but it might also reduce reconstruction quality). β-VAE achieves disentanglement in terms of the probabilistic interpretation of disentanglement.

2.3 Results

The authors ran both qualitative and quantitative experiments with β-VAE. For qualitative results, the disentanglements learned by β-VAE look better than those learned by other generative models at the time such as InfoGAN, DC-IGN, and VAE. In terms of quantitative results, they create their own metric in the β-VAE paper, and β-VAE outperforms other models on that metric (as it turns out, this metric is also correlated with other disentanglement metrics we describe later [10]).

3 GeoManCEr

GeoManCEr is a novel technique in disentanglement and was the main motivation for this project [12]. This section will give a brief explanation, but we encourage everyone to read the full paper. GeoManCEr is a nonparametric algorithm that specifically focuses on symmetry-based disentanglement of data manifolds. The authors show that fully unsupervised factorization of a data manifold is possible if the true metric of the manifold is known and every factor has a nontrivial holonomy. GeoManCEr gives an approximation to the de Rham decomposition by estimating subspaces that are invariant under random walk diffusion.

GeoManCEr builds off of failings of the parallelogram model of analogical reasoning [12]. More specifically, it can complete the analogy $a : b :: c : d$ where d is unknown, by calculating $d = b+c-a$. It has been shown that word embeddings often work with this parallelogram model [12]. But unfortunately, many natural transformations, such as rotations, do not follow this nice model. Instead, these natural transformations can be represented as coming from the orbit of a group (a part of the motivation behind the symmetry-based model of disentanglement). If we have some $g, h \in G$, where G is a product of subgroups, both g and h leave all factors invariant except one, and each varies a different factor, then they commute and the analogy can be uniquely completed.

GeoManCEr only looks at Lie groups, groups that are also manifolds, and uses the failures of the parallelogram model as a learning signal. Directions on disentangled submanifolds comply with the parallelogram model . The de Rham decomposition showcases these intuitions. GeoManCEr learns a set of subspaces to assign to each point in the dataset, where each subspace is the tangent space of one disentangled submanifold.

3.1 Theory

The holonomy group of a manifold can be denoted as $\mathrm{Hol}_x(M)$, which consists of the holonomies H_γ (these can be written as linear transformations) for all loops on a given manifold M that start and end at x. The holonomy group of a manifold is very informative about the global structure of the manifold. By applying the the de Rham Decomposition Theorem, recursively, we can conclude that if the holonomy group at a point leaves multiple pairwise orthogonal subspaces invariant, our manifold M is the product of multiple Riemannian manifolds. That's the key to GeoManCEr! The goal is to discover a decomposition of a data manifold by investigating its holonomy group. Unfortunately, the holonomy group can't be computed directly because it is a property of all possible loops, so GeoManCEr finds an approximation. Each Riemannian manifold in this decomposition of the data manifold should correspond to the datapoints invariant to a certain symmetry transform (is this true?).

GeoManCEr considers the average properties of a random walk diffusion on a manifold. This can be modeled as a diffusion equation which given a probability density $p(x, 0)$, the probability of finding a particle at x at time t is given by the following differential equation.

$$\frac{\partial p(x,t)}{\partial t} = \tau \Delta^0[p](x,t)$$

Where Δ^0 is the Laplace-Beltrami operator and is defined as the trace of the second covariant derivative (derivative on tangent vectors of a manifold) $\Delta^0[f] = \mathrm{Tr}\nabla^2 f$. This operator can be generalized to the connection Laplacian fro rank-(p,q) tensor-valued functions. As it turns out, properties of the holonomy group can be inferred from the second-order connection Laplacian, denoted as Δ^2. Namely, for a product manifold, the eigenfunctions of the second-order connection Laplacian contain information about invariant subspaces of the holonomy group. If we have a Riemannian product manifold $M = M_1 \times \ldots \times M_m$ and let $T_x^{(1)}(M), \ldots, T_x^{(m)}(M)$ denote the tangent spaces to each submanifold. Then the tensor fields $\prod^{(i)} : M \mapsto T_x(M) \otimes T_x(M)$ for $i \in [m]$ where $\prod_x^{(i)}$ is the linear projection operator from $T_x(M) \mapsto T_x^{(i)}(M)$, go to 0 under the action of the connection Laplacian. For the second-order Laplacian, the zero eigenvalues correspond to factors of a product manifold, with the matrix-valued eigenfunction being the identity in the subspace tangent to one manifold and zero everywhere else. There are also spurious eigenfunctions of Δ^2 with zero eigenvalue.

3.2 Algorithm

The goal of the Geometric Manifold Component Estimator (GeoManCEr) is to approximate the second-order connection Laplacian from finite samples of points on the manifold, and then find the eigenvectors with nearly zero eigenvalue that correspond to the disentangled submanifolds of the data, that let us define local coordinates around every data point that are aligned with the disentangled manifolds.

We start with some given set of points $\mathbf{x}_1, \ldots, \mathbf{x}_t \in \mathbb{R}^{\Bbbk}$ sampled from some manifold embedded in $\mathbb{R}^n$. To construct our second-order connection Laplacian, we need to approximate properties of the manifold. We will assume the data is embedded a Euclidean space where we use the Euclidean metric on the manifold. We first construct a nearest neighbors graph, and then we construct a set of tangent spaces per data point by applying PCA to the difference between $\mathbf{x}_i$ and its neighbors in the nearest neighbor graph $\mathbf{x}_j$.

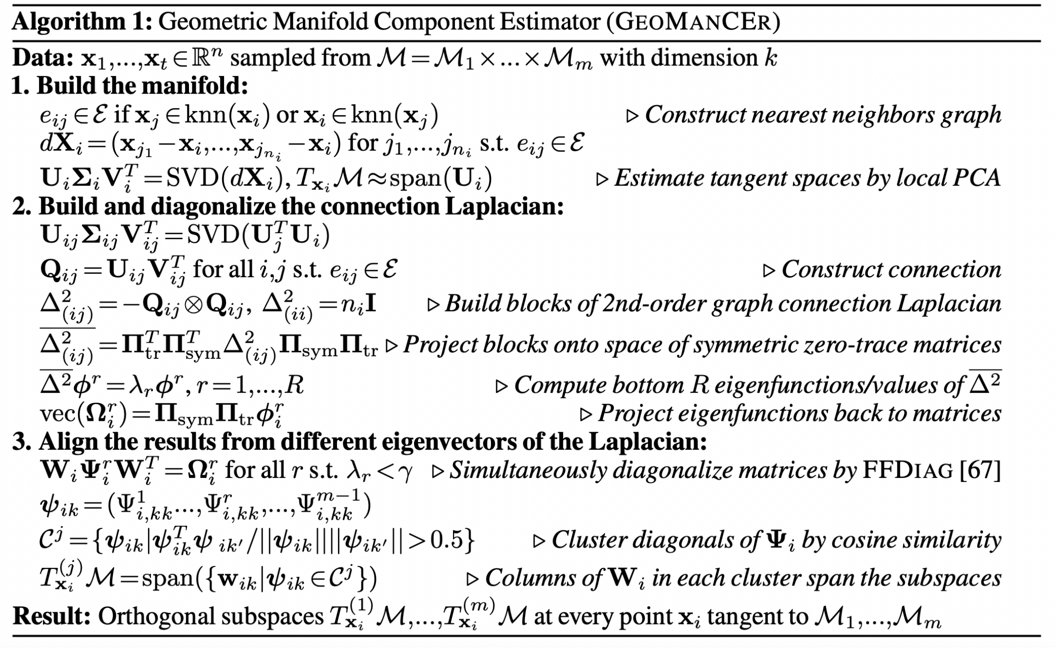
Now that we have our manifold, we want to get our second-order connection Laplacian. We can do this by generalizing the graph Laplacian to higher order tensors. With this, and connection matrices of the manifold that we have what we need to construct the second order connection-Laplacian, but we need to eliminate spurious eigenfunctions first. We can do this by a series of projections onto the space of operators on symmetric zero-trace matrices. Now we can compute the smallest R eigenvalues and R eigenvectors for each point and project back to full square matrices of the dimensions of the manifold k, denoted Ω_i^r.

Now that we have our eigenvalues and vectors, we need to align the results and output our orthogonal subspaces $T_x^{(1)}(M), \ldots, T_x^{(m)}(M)$ for every point tangent to the submanifolds $M_1 \times \ldots \times M_m$. We can use the orthogonal FFDiag Algorithm to get a decomposition of our Ω_i^r matrices that expresses an orthonormal basis for each point. Now we can cluster these orthonormal bases to create one cluster for each of the m disentangled subspaces.

3.3 Results

The authors tested GeoManCEr on synthetic manifolds and rendered 3D objects. On synthetic datasets where manifolds consisted of 5 or fewer submanifolds, GeoManCEr was able to successfully disentangle them, but on more complex manifolds, GeoManCEr failed.

On the 3D objects viewed at different orientations, when GeoManCEr was directly applied to the true latent state vectors, GeoManCEr performed very well, and was able to keep the angle between the true disentangled subspace and the GeoManCEr recovered subspaces small. However, when fed the raw images, GeoManCEr performed no better than random chance. GeoManCEr also performs poorly when applied to the latent vectors learned by β-VAE.



```
Algorithm 1: Geometric Manifold Component Estimator (GEOMANCER)
Data: $\mathbf{x}_1,\ldots,\mathbf{x}_t \in \mathbb{R}^n$ sampled from $\mathcal{M} = \mathcal{M}_1 \times \ldots \times \mathcal{M}_m$ with dimension $k$
1. Build the manifold:
    $e_{ij} \in \mathcal{E}$ if $\mathbf{x}_j \in \mathrm{knn}(\mathbf{x}_i)$ or $\mathbf{x}_i \in \mathrm{knn}(\mathbf{x}_j)$    ▷ Construct nearest neighbors graph
    $d\mathbf{X}_i = (\mathbf{x}_{j_1} - \mathbf{x}_i,\ldots,\mathbf{x}_{j_{n_i}} - \mathbf{x}_i)$ for $j_1,\ldots,j_{n_i}$ s.t. $e_{ij} \in \mathcal{E}$
    $\mathbf{U}_i \mathbf{\Sigma}_i \mathbf{V}_i^T = \mathrm{SVD}(d\mathbf{X}_i), T_{\mathbf{x}_i}\mathcal{M} \approx \mathrm{span}(\mathbf{U}_i)$    ▷ Estimate tangent spaces by local PCA
2. Build and diagonalize the connection Laplacian:
    $\mathbf{U}_{ij} \mathbf{\Sigma}_{ij} \mathbf{V}_{ij}^T = \mathrm{SVD}(\mathbf{U}_j^T \mathbf{U}_i)$
    $\mathbf{Q}_{ij} = \mathbf{U}_{ij} \mathbf{V}_{ij}^T$ for all $i,j$ s.t. $e_{ij} \in \mathcal{E}$    ▷ Construct connection
    $\Delta^2_{(ij)} = -\mathbf{Q}_{ij} \otimes \mathbf{Q}_{ij},\ \Delta^2_{(ii)} = n_i \mathbf{I}$    ▷ Build blocks of 2nd-order graph connection Laplacian
    $\overline{\Delta^2_{(ij)}} = \mathbf{\Pi}_{\mathrm{tr}}^T \mathbf{\Pi}_{\mathrm{sym}}^T \Delta^2_{(ij)} \mathbf{\Pi}_{\mathrm{sym}} \mathbf{\Pi}_{\mathrm{tr}}$ ▷ Project blocks onto space of symmetric zero-trace matrices
    $\overline{\Delta^2} \boldsymbol{\phi}^r = \lambda_r \boldsymbol{\phi}^r, r = 1,\ldots,R$    ▷ Compute bottom $R$ eigenfunctions/values of $\overline{\Delta^2}$
    $\mathrm{vec}(\mathbf{\Omega}_i^r) = \mathbf{\Pi}_{\mathrm{sym}} \mathbf{\Pi}_{\mathrm{tr}} \boldsymbol{\phi}_i^r$    ▷ Project eigenfunctions back to matrices
3. Align the results from different eigenvectors of the Laplacian:
    $\mathbf{W}_i \mathbf{\Psi}_i^r \mathbf{W}_i^T = \mathbf{\Omega}_i^r$ for all $r$ s.t. $\lambda_r < \gamma$  ▷ Simultaneously diagonalize matrices by FFDIAG [67]
    $\boldsymbol{\psi}_{ik} = (\Psi^1_{i,kk} \ldots, \Psi^r_{i,kk}, \ldots, \Psi^{m-1}_{i,kk})$
    $\mathcal{C}^j = \{\boldsymbol{\psi}_{ik} | \boldsymbol{\psi}_{ik}^T \boldsymbol{\psi}_{ik'} / ||\boldsymbol{\psi}_{ik}|| ||\boldsymbol{\psi}_{ik'}|| > 0.5\}$    ▷ Cluster diagonals of $\mathbf{\Psi}_i$ by cosine similarity
    $T^{(j)}_{\mathbf{x}_i}\mathcal{M} = \mathrm{span}(\{\mathbf{w}_{ik} | \boldsymbol{\psi}_{ik} \in \mathcal{C}^j\})$    ▷ Columns of $\mathbf{W}_i$ in each cluster span the subspaces
Result: Orthogonal subspaces $T^{(1)}_{\mathbf{x}_i}\mathcal{M},\ldots,T^{(m)}_{\mathbf{x}_i}\mathcal{M}$ at every point $\mathbf{x}_i$ tangent to $\mathcal{M}_1,\ldots,\mathcal{M}_m$
```

Figure 4: The GeoManCEr Algorithm

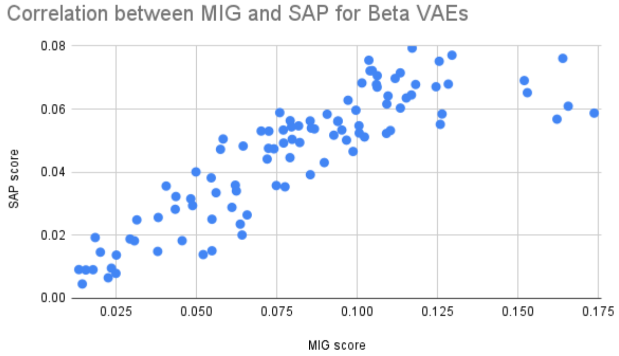


Figure 5: Many disentanglement metrics are highly correlated. Higher is better for both SAP and MIG

Unfortunately, GeoManCEr is also very expensive to run, since the number of nonzero elements in $\overline{\Delta^2}$ increases with the dimensionality of the manifold by $\mathcal{O}(k^4)$. Luckily there are only 3 hyperparameters to tune, the dimension of the data manifold, the number of nearest neighbors, and the gap γ in the spectrum of $\overline{\Delta^2}$ at which to stop splitting tangent spaces that is used to infer the appropriate number of submanifolds (equivalent to the number of underlying symmetry groups).

4 Disentanglement Metrics

Quantify how disentangled a representation is difficult, some papers [12] reject all current approaches, and they are very correlated [10] (and see graphs). Despite the difficulties, it can be useful for getting quantitative data that accelerated research. We describe two popular disentanglement metrics in the literature.

4.1 Mutual Information Gap (MIG)

MIG is a popular disentanglement metric because it is axis aligned, unbiased, and general to any factorized latent distribution, whether categorical, multimodal vectors, or otherwise. We will start by defining mutual information, which is simply,

$$I_n\left(z_j; v_k\right)=\mathbb{E}_{q\left(z_j, v_k\right)}\left[\log \sum_{n \in \mathcal{X}_{v_k}} q\left(z_j \mid n\right) p\left(n \mid v_k\right)\right]+H\left(z_j\right)$$

The MIG is defined as the gap between the largest two normalized mutual information scores between two representations from the same class (ie two pictures of cats) [2].

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

MIG is bounded between 0 and 1, and ideally we would like to maximize this value.

4.2 Separated Attribute Predictability score (SAP)

The Separated Attribute Predictability (SAP) score is the average difference of the prediction error of the two most predictive latent dimensions for each factor [9].

We compute SAP by first constructing a score matrix S which is $d \times k$. The ij'th entry is the R^2 score j'th factor, using the i'th latent. For each column of S, we get the top two dimensions and find the difference between them. Let's call this value λ_j. The mean of each columns λ_j is the SAP score.

This metric is well aligned with qualitative disentanglement observations and easy to compute.

5 Experiments

We investigate how scale is related to disentanglement. Current disentanglement algorithms have shown limited success with small models on carefully crafted datasets and we are curious how these approaches will scale to SOTA models. Previous experiments have shown that more of the variation in disentanglement performance can be accounted for by hyperparameter choice and random initialization than the actual disentanglement objective [10]. It is important that our disentangling methods scale to state of the art model sizes and datasets. This motivates the following questions.[2]

5.1 How much can disentanglement performance be accounted for by scale and not the algorithm?

Scale is comprised of compute, dataset size, and parameters [7], and we varied the amount of compute. β-VAE (with $\beta = 4$) shows MIG increase with scale. Results are still inconclusive, as we only tested on the dSprites dataset with very limited scaling, yet it suggests that scale and algorithm choice matter.

5.2 How does disentanglement scale on large models not explicitly trained to be disentangled?

We ran experiments using various self-supervised trained Contrastive Language-Image Pre-Training (CLIP) vision encoders. CLIP Scale seems to increase disentanglement [13]. Vision transformers (ViTs) also seem to have better disentanglement with scale than convolutional neural networks (CNNs), yet results are still limited. These results are potentially in line with previous comparisons between ViTs and CNNs, which indicate that ViTs might have better generalization abilities [14]. Given our hypothesis that disentangled representations might be useful representations that help generalization, our results align.

[2] Our code can be found at https://github.com/berkott/disentanglementAndScale

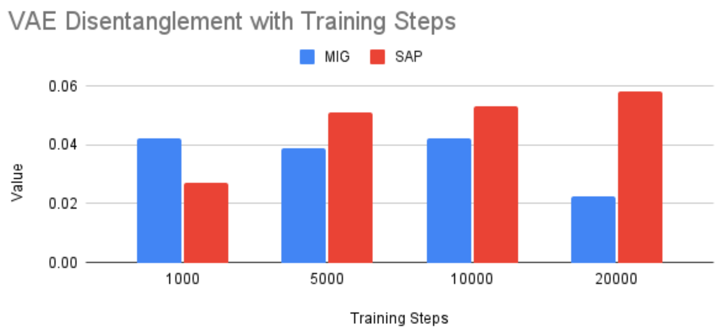


Figure 6: VAE Disentanglement

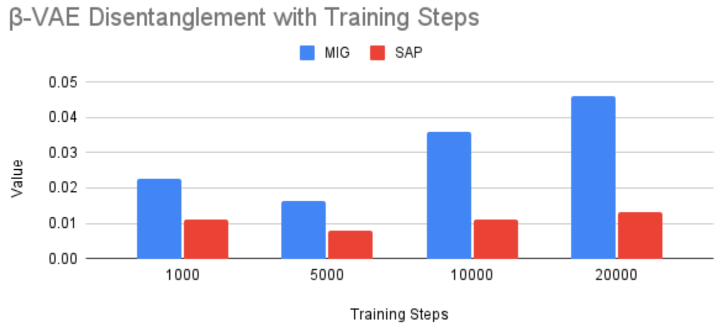


Figure 7: β-VAE Disentanglement with $\beta = 4$

6 Conclusions

Unsupervised disentanglement is a promising direction for machine learning that could yield insights that improve the generalization ability and other capabilities of models. Currently, much of the work is theoretical and with small model on toy datasets. We are excited to see these approaches scale and we hope our experiments motivate others.

Acknowledgments and Disclosure of Funding

A big thank you to professor Nakul Verma for teaching Unsupervised Machine Learning and our friend Zachary Lihn for explaining some of the underlying mathematics of GeoManCEr. Thank you to our friends and family for their support during this project. We didn't receive any funding for this project.

Table 1: Disentanglement of CLIP Encoder on dSprites

Model	**Parameters**	**MIG (higher better)**
RN50	102 M	0.00441
RN50x4	178 M	0.01763
RN50x16	291 M	0.02616
ViT-B/16	150 M	0.02350
ViT-L/14	428 M	0.02550

References

[1] Bengio, Yoshua, et al. Representation Learning: A Review and New Perspectives. arXiv, 23 Apr. 2014. arXiv.org, http://arxiv.org/abs/1206.5538.

[2] Chen, Ricky T. Q., et al. Isolating Sources of Disentanglement in Variational Autoencoders. arXiv, 23 Apr. 2019. arXiv.org, http://arxiv.org/abs/1802.04942.

[3] Duan, Sunny, et al. Unsupervised Model Selection for Variational Disentangled Representation Learning. arXiv, 14 Feb. 2020. arXiv.org, http://arxiv.org/abs/1905.12614.

[4] Higgins, Irina, Loic Matthey, et al. "Beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework." International Conference on Learning Representations, 2017, https://openreview.net/forum?id=Sy2fzU9gl.

[5] Higgins, Irina, David Amos, et al. Towards a Definition of Disentangled Representations. arXiv, 5 Dec. 2018. arXiv.org, http://arxiv.org/abs/1812.02230.

[6] Hinton, G. E., and R. R. Salakhutdinov. "Reducing the Dimensionality of Data with Neural Networks." Science, vol. 313, no. 5786, July 2006, pp. 504–07. DOI.org (Crossref), https://doi.org/10.1126/science.1127647.

[7] Kaplan, Jared, et al. Scaling Laws for Neural Language Models. arXiv, 22 Jan. 2020. arXiv.org, http://arxiv.org/abs/2001.08361.

[8] Kingma, Diederik P., and Max Welling. Auto-Encoding Variational Bayes. arXiv, 1 May 2014. arXiv.org, http://arxiv.org/abs/1312.6114.

[9] Kumar, Abhishek, et al. Variational Inference of Disentangled Latent Concepts from Unlabeled Observations. arXiv, 27 Dec. 2018. arXiv.org, http://arxiv.org/abs/1711.00848.

[10] Locatello, Francesco, et al. Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations. arXiv, 18 June 2019. arXiv.org, http://arxiv.org/abs/1811.12359.

[11] P. Common, Independent component analysis, a new concept?, Sig. Process. 36 (3) (1994) 287-314.

[12] Pfau, David, et al. Disentangling by Subspace Diffusion. arXiv, 18 Nov. 2020. arXiv.org, http://arxiv.org/abs/2006.12982.

[13] Radford, Alec, et al. Learning Transferable Visual Models From Natural Language Supervision. arXiv, 26 Feb. 2021. arXiv.org, http://arxiv.org/abs/2103.00020.

[14] Bai, Yutong, et al. Are Transformers More Robust Than CNNs? arXiv, 9 Nov. 2021. arXiv.org, http://arxiv.org/abs/2111.05464.