DCI-ES: An Extended Disentanglement Framework with Connections to Identifiability

Anonymous authors
Paper under double-blind review

Abstract

In representation learning, a common approach is to seek representations which disentangle the underlying factors of variation. Eastwood & Williams (2018) proposed three metrics for quantifying the quality of such disentangled representations: disentanglement (D), completeness (C) and informativeness (I). In this work, we first connect this DCI framework to two common notions of linear and nonlinear identifiability, thus establishing a formal link between disentanglement and the closely-related field of independent component analysis. We then propose an extended DCI-ES framework with two new measures of representation quality—*explicitness* (E) and *size* (S)—and point out how D and C can be computed for black-box predictors. Our main idea is that the *functional capacity required to use a representation* is an important but thus-far neglected aspect of representation quality, which we quantify using explicitness or *ease-of-use* (E). We illustrate the relevance of our extensions on the `MPI3D` and `Cars3D` datasets.

1 Introduction

A primary goal of representation learning is to learn representations $r(\boldsymbol{x})$ of complex data $\boldsymbol{x}$ that "make it easier to extract useful information when building classifiers or other predictors" (Bengio et al., 2013). *Disentangled* representations, which aim to recover and separate (or, more formally, *identify*) the underlying factors of variation $\boldsymbol{z}$ that generate the data as $\boldsymbol{x} = g(\boldsymbol{z})$, are a promising step in this direction. In particular, it has been argued that such representations are not only interpretable (Kulkarni et al., 2015; Chen et al., 2016) but also make it easier to extract useful information for downstream tasks by recombining previously-learnt factors in novel ways (Lake et al., 2017).

While there is no single, widely-accepted definition, many evaluation protocols have been proposed to capture different notions of disentanglement based on the relationship between the learnt representation or *code* $\boldsymbol{c} = r(\boldsymbol{x})$ and the ground-truth data-generative factors $\boldsymbol{z}$ (Higgins et al., 2017; Eastwood & Williams, 2018; Ridgeway & Mozer, 2018; Kim & Mnih, 2018; Chen et al., 2018; Suter et al., 2019). In particular, the metrics of Eastwood & Williams (2018)—*disentanglement* (D), *completeness* (C) and *informativeness* (I)—estimate this relationship by learning a *probe* f to predict $\boldsymbol{z}$ from $\boldsymbol{c}$ and can be used to relate many other notions of disentanglement (see Locatello et al. 2020, § 6).

In this work, we extend this DCI framework in several ways. Our main idea is that *the functional capacity required to recover $\boldsymbol{z}$ from $\boldsymbol{c}$ is an important but thus-far neglected aspect of representation quality.* For example, consider the case of recovering $\boldsymbol{z}$ from: (i) a noisy version thereof; (ii) raw, high-dimensional data (e.g. images); and (iii) a uniformly-mixed version thereof (each c_i containing the same amount of information about each z_j). The noisy version (i) will do quite well with just linear capacity, but is fundamentally limited by the noise corruption; the raw data (ii) will likely do quite poorly with linear capacity, but eventually outperform (i) given sufficient capacity; and the uniformly-mixed version (iii) will perfectly recover $\boldsymbol{z}$ with just linear capacity, yet achieve the worst-possible disentanglement score of $D = 0$. Motivated by this observation, we introduce a measure of *explicitness* or *ease-of-use* based a representation's *loss-capacity curve* (see Fig. 1).

Structure and contributions. First, we connect the DCI metrics to two common notions of linear and nonlinear identifiability (§ 3), thereby establishing a formal link to the related field of independent component analysis. We then propose an extended DCI-ES framework (§ 4) in which we: (i) introduce two new complementary measures of representation quality that permit a more fine-grained evaluation—*explicitness* (E), derived from a representation's *loss-capacity curve* (see Fig. 2), and *size* (S); and (ii) elucidate a means to compute the D and C scores for arbitrary black-box probes f (e.g., MLPs). Finally, in our experiments (§ 6), we illustrate the relevance of our extensions by comparing different representations on the `MPI3D-Real` (Gondal et al., 2019) and `Cars3D` (Reed et al., 2015) datasets.

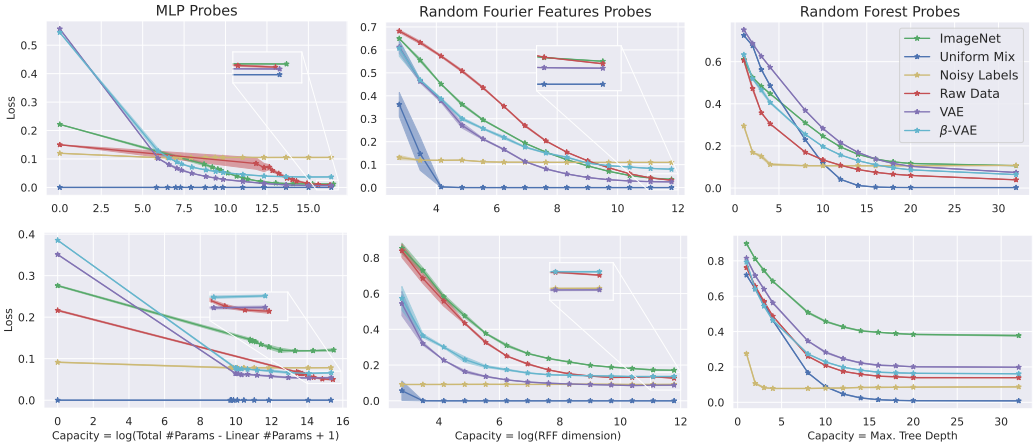

Figure 1: **Loss-capacity curves.** Empirical loss-capacity curves (see § 4.1) for various representations (see legend), datasets (top: `MPI3D-Real`, bottom: `Cars3D`), and probe types: multi-layer perceptrons (MLPs; left), Random Fourier Features (RFFs; middle), and Random Forests (RFs; right). The loss was first averaged over factors z_j, then means and 95% confidence intervals were computed over 3 random seeds. Details in § 6.

2 Background

Given a synthetic dataset of observations $\boldsymbol{x} = g(\boldsymbol{z})$ along with the corresponding K-dimensional data-generating factors $\boldsymbol{z} \in \mathbb{R}^K$, the DCI framework quantitatively evaluates an L-dimensional data representation or *code* $\boldsymbol{c} = r(\boldsymbol{x}) \in \mathbb{R}^L$ by: (i) training a probe f to predict $\boldsymbol{z}$ from $\boldsymbol{c}$, i.e., $\hat{\boldsymbol{z}} = f(\boldsymbol{c}) = f(r(\boldsymbol{x})) = f(r(g(\boldsymbol{z})))$; and (ii) quantifying f's prediction error and its deviation from the ideal one-to-one mapping, which is a permutation matrix (with extra "dead" units in $\boldsymbol{c}$ whenever $L > K$).[1] For step (i), Eastwood & Williams (2018) use Lasso (Tibshirani, 1996) or Random Forests (RFs, Breiman 2001) as linear or nonlinear predictors, respectively, for which it is straightforward to read-off suitable "relative feature importances".

Definition 2.1. *$R \in \mathbb{R}^{L \times K}$ is called a* matrix of relative importances *for predicting $\boldsymbol{z}$ from $\boldsymbol{c}$ via $\hat{\boldsymbol{z}} = f(\boldsymbol{c})$ if R_{ij} captures some notion of the contribution of c_i to predicting z_j such that for all i, j: $R_{ij} \geq 0$ and $\sum_{i=1}^{L} R_{ij} = 1$.*

For step (ii), Eastwood & Williams use R and the prediction error to define and quantify three desiderata of disentangled representations: *disentanglement* (D), *completeness* (C), and *informativeness* (I).

Disentanglement. Disentanglement (D) measures the average number of data-generating factors z_j that are captured by any single code c_i. The score D_i is given by $D_i = 1 - H_K(P_{i.})$, where $H_K(P_{i.}) = -\sum_{k=1}^{K} P_{ik} \log_K P_{ik}$ denotes the entropy of the distribution $P_{i.}$ over *row* i of R, with $P_{ij} = R_{ij} / \sum_{k=1}^{K} R_{ik}$. If c_i is only important for predicting a single z_j, we get a perfect score of $D_i = 1$. If c_i is equally important for predicting all z_j (for $j = 1, \ldots, K$), we get the worst score of $D_i = 0$. The disentanglement score D is then given by the weighted average $D = \sum_{i=1}^{L} \rho_i D_i$, with $\rho_i = \frac{1}{K} \sum_{k=1}^{K} R_{ik}$.

Completeness. Completeness (C) measures the average number of code variables c_i required to capture any single z_j; it has also been called *compactness* (Ridgeway & Mozer, 2018). The score C_j in capturing z_j is given by $C_j = (1 - H_L(\tilde{P}_{.j}))$, where $H_L(\tilde{P}_{.j}) = -\sum_{\ell=1}^{L} \tilde{P}_{\ell j} \log_L \tilde{P}_{\ell j}$ denotes the entropy of the distribution $\tilde{P}_{.j}$ over *column* j of R, with $\tilde{P}_{ij} = R_{ij}$. If a single c_i contributes to z_j's prediction, we get a perfect score of $C_j = 1$. If all c_i equally contribute to z_j's prediction (for $i = 1, \ldots, L$), we get the worst score of $C_j = 0$. The overall completeness score is given by $C = \frac{1}{K} \sum_{j=1}^{K} C_j$.

Remark 2.2. *Together*, D and C quantify the degree of "mixing" between $\boldsymbol{c}$ and $\boldsymbol{z}$, i.e., the deviation from a one-to-one mapping. They are reported separately as they capture distinct criteria.

[1] W.l.o.g., it can be assumed that z_i and c_j are normalised to have mean zero and variance one for all i, j, for otherwise such normalisation can be "absorbed" into $g(\cdot)$ and $r(\cdot)$.

Informativeness. The informativeness (I) of representation $\boldsymbol{c}$ about data-generative factor z_j is quantified by the prediction error, i.e., $I_j = 1 - \mathbb{E}[\ell(z_j, f_j(\boldsymbol{c}))]$, where ℓ is an appropriate loss function.[2] Note that I_j depends on the capacity of f_j, as depicted in Fig. 1. Thus, for I_j to accurately capture the informativeness of $\boldsymbol{c}$ about z_j, f_j must have sufficient capacity to extract *all* of the information in $\boldsymbol{c}$ about z_j. This capacity-informativeness dependency motivates a separate measure of representation *explictness* in § 4.1. The overall informativeness score is given by: $I = \frac{1}{K}\sum_{j=1}^{K} I_j$.

3 Connection to identifiability

The goal of learning a data representation which recovers the underlying data-generating factors is closely related to blind source separation and independent component analysis (ICA) (Comon, 1994; Hyvärinen & Pajunen, 1999; Hyvarinen et al., 2019). Whether a given learning algorithm provably achieves this goal up to acceptable ambiguities, subject to certain assumptions on the data-generating process, is typically formalised using the notion of *identifiability*. Two common types of identifiability for linear and nonlinear settings, respectively, are the following.

Definition 3.1. *We say that $\boldsymbol{c} = r(\boldsymbol{x}) = r(g(\boldsymbol{z}))$ identifies $\boldsymbol{z}$ up to sign and permutation if $\boldsymbol{c} = P\boldsymbol{z}$ for some matrix P such that $|P|$ is a permutation matrix.*

Definition 3.2. *We say that $\boldsymbol{c}$ identifies $\boldsymbol{z}$ up to permutation and element-wise reparametrisation if there exists a permutation π of $\{1, ..., K\}$ and invertible scalar-functions $\{h_k\}_{k=1}^{K}$ such that, for all j, $c_j = h_j(z_{\pi(j)})$.*

We now establish theoretical connections between the DCI framework and these identifiability types.

Proposition 3.3. *If $D = C = 1$ and $K = L$ (i.e., $dim(\mathbf{c}) = dim(\mathbf{z})$), then R is a permutation matrix.*

Proof. First, by Defn. 2.1, we have $0 \leq R_{ij}$ and $\sum_{i=1}^{L} R_{ij} = 1\ \forall i, j$, so $0 \leq R_{ij} \leq 1$. It follows that $\forall i, j:\ P_{i\cdot}, \tilde{P}_{\cdot j} \in \Delta_{K-1}$, where Δ_{K-1} denotes the K-dim. probability simplex, i.e., $P_{i\cdot}$ and $\tilde{P}_{\cdot j}$ are valid probability vectors. Hence, the Shannon entropies $H_K(P_{i\cdot}), H_K(\tilde{P}_{\cdot j})$ are well-defined $\forall i, j$, and, due to using $\log_K$ in the definition of H_K (see § 2), are bounded in $[0, 1]$. It follows that $\forall i, j:\ 0 \leq D_i \leq 1$ and $0 \leq C_j \leq 1$. Since D and C are convex combinations of the D_i and C_j, we have

$$D = 1 \iff \forall i:\ D_i = 1 \iff \forall i:\ H_K(P_{i\cdot}) = 0\,,$$
$$C = 1 \iff \forall j:\ C_j = 1 \iff \forall j:\ H_K(\tilde{P}_{\cdot j}) = 0\,.$$

Now for any $\boldsymbol{p} = (p_1, ..., p_K) \in \Delta_{K-1}$, we have that

$$H_K(\boldsymbol{p}) = -\sum_{k=1}^{K} p_k \log_K p_k = 0 \iff \forall k:\ p_k \log_K p_k = 0 \iff \forall k:\ p_k \in \{0, 1\}$$

where $p_k \log p_k := 0$ for $p_k = 0$, consistent with $\lim_{x \to 0^+} x \log x = 0$. Together with the simplex constraint, this implies that $\boldsymbol{p}$ must be a standard basis vector $\boldsymbol{p} = \boldsymbol{e}_l$ for some l, i.e., $p_l = 1$ and $p_k = 0$ for $k \neq l$. Hence, $P_{i\cdot}, \tilde{P}_{\cdot j}$ must be standard basis vectors for all i, j, and so each row and column of R contains exactly one non-zero element. Since columns of R sum to one, these non-zero elements must all be one. □

Using Prop. 3.3, we can establish links to identifiability, provided that the inferred representation $\boldsymbol{c}$ perfectly predicts the true data-generating factors $\boldsymbol{z}$, i.e., $I = 1$.

Corollary 3.4. *Under the same conditions as Prop. 3.3, if $\boldsymbol{z} = W^\top \boldsymbol{c}$ (so that $I = 1$) and $R = |W|$, then $\boldsymbol{c}$ identifies $\boldsymbol{z}$ up to permutation and sign (Defn. 3.1).*

Proof. By Prop. 3.3, R is a permutation matrix, so from $R = |W|$ it follows that W and thus $(W^\top)^{-1}$ must be signed permutation matrices. □

For nonlinear f, we give a more general statement for suitably-chosen feature-importance matrices R.

[2] Here we deviate from Eastwood & Williams (who had $I_j = \mathbb{E}[\ell(z_j, f_j(\boldsymbol{c}))]$) such that 1 is now the best score.

Corollary 3.5. *Under the same conditions as Prop. 3.3, let $\boldsymbol{z} = f(\boldsymbol{c})$ (so that $I = 1$) with f an invertible nonlinear function, and let R be a matrix of relative feature importances for f (Defn. 2.1) with the property that $R_{ij} = 0$ if and only if f_j does not depend on c_i, i.e., $\left|\left|\partial_i f_j\right|\right|_2 = 0$. Then $\boldsymbol{c}$ identifies $\boldsymbol{z}$ up to permutation and element-wise reparametrisation (Defn. 3.2).*

Proof. For any j consider $z_j = f_j(\boldsymbol{c})$. By Prop. 3.3, R is a permutation matrix, so column j of R contains exactly one non-zero entry in row $\pi(j)$ for some permutation π of $\{1, ..., K\}$. Hence, by the assumed property of R, $f_j(\boldsymbol{c})$ does not depend on c_i for all $i \neq \pi(j)$, and thus $z_j = f_j(c_{\pi(j)})\ \forall j$. By invertibility of f, we obtain $c_j = h_j(z_{j'})$ with $h_j = f_{j'}^{-1}$ and $j' = \pi^{-1}(j)$. □

Remark 3.6. While the *if* part of Corollary 3.5 holds for most feature importance measures, the *only if* part, in general, does not: not using a feature c_i is typically a *sufficient* condition for $R_{ij} = 0$, but it need not be a *necessary* condition (as required for Corollary 3.5). E.g., measures based on *average* performance may not satisfy this since a feature may not contribute on average, but still be used—sometimes helping and sometimes hurting performance, see § 7 for further discussion. In contrast, Gini importances, as used in random forests, *do* satisfy the necessary condition. While the non-invertibility of random forests prevents an explicit link to identifiability (typically studied for continuous features), they can still be a principled choice in practice (where features are often categorical).

Summary. We have established that the learnt representation $\boldsymbol{c}$ identifies the ground-truth $\boldsymbol{z}$ up to:
- sign and permutation if $D = C = I = 1$ and f is linear;
- permutation and element-wise reparametrisation if $D = C = I = 1$ and $R_{ij} = 0 \Leftrightarrow \left|\left|\partial_i f_j\right|\right|_2 = 0$.

4 EXTENDED DCI-ES FRAMEWORK

Motivated by our theoretical insights from § 3—considering different probe function classes provides links to different types of identifiability—and the empirically-observed performance differences between representations trained with different-capacity probes shown in Fig. 1, we now propose several extensions of the DCI framework.

4.1 EXPLICITNESS (E)

We first introduce a new complementary notion of disentanglement based on the functional capacity required to recover or predict $\boldsymbol{z}$ from $\boldsymbol{c}$. The key idea is to measure the *explicitness* or *ease-of-use* (E) of a representation using its *loss-capacity curve*.

Notation. Let $\mathcal{F}$ be a probe function class (e.g., random forests or MLPs), let $f_j^* \in \arg\min_{f \in \mathcal{F}} \mathbb{E}[\ell(z_j, f(\boldsymbol{c}))]$ be a minimum-loss probe for factor z_j on a held-out data split[3], and let $\text{Cap}(\cdot)$ be a suitable capacity measure on $\mathcal{F}$—e.g., for random forests, $\text{Cap}(f)$ could correspond to the maximum tree-depth of f.

Loss-capacity curves. A loss-capacity curve for representation $\boldsymbol{c}$, factor z_j, and probe class $\mathcal{F}$ displays test-set loss against probe capacity for increasing-capacity probes $f \in \mathcal{F}$ (see Fig. 1). To plot such a curve, we must train T predictors with capacities $\kappa_1, \ldots, \kappa_T$ to predict z_j, with

$$f_j^t \in \arg\min_{f \in \mathcal{F}} \mathbb{E}\left[\ell(z_j, f(\boldsymbol{c}))\right] \quad \text{s.t.} \quad \text{Cap}(f) = \kappa_t. \tag{4.1}$$

Here $\kappa_1, \ldots, \kappa_T$ is a list of T increasing probe *capacities*, ideally[4] shared by all representations, with suitable choices for κ_1 and κ_T depending on both $\mathcal{F}$ and the dataset. For example, we may choose κ_T to be large enough for all representations to achieve their lowest loss and, for random forest fs, we may choose $\kappa_1 = 1$ and then $T - 2$ tree depths between 1 and κ_T.

AULCC. We next define the *Area Under the Loss-Capacity Curve* (AULCC) for representation $\boldsymbol{c}$, factor z_j, and probe class $\mathcal{F}$ as the (approximate) area between the corresponding loss-capacity

[3] In practice, all expectations are taken w.r.t. the corresponding empirical (train/validation/test) distributions.
[4] True for RFs but not input-size dependent MLPs (see § 6).

curve and the loss-line of our best predictor $\ell_j^{*,\boldsymbol{c}} = \mathbb{E}[\ell(z_j, f_j^*(\boldsymbol{c}))]$. To compute this area, depicted in Fig. 2, we use the trapezoidal rule

$$\text{AULCC}(z_j, \boldsymbol{c}; \mathcal{F}) = \sum_{t=2}^{t^{*,\boldsymbol{c}}} \left(\frac{1}{2} \left(\ell_j^{t-1,\boldsymbol{c}} + \ell_j^{t,\boldsymbol{c}} \right) - \ell_j^{*,\boldsymbol{c}} \right) \cdot \Delta\kappa_t, \tag{4.2}$$

where $t^{*,\boldsymbol{c}}$ denotes the index of $\boldsymbol{c}$'s lowest-loss capacity $\kappa_{*,\boldsymbol{c}}$; $\ell_j^{t,\boldsymbol{c}} = \mathbb{E}[\ell(z_j, f_j^t(\boldsymbol{c}))]$ the test-set loss with predictor f_j^t, see Eq. (4.1); and $\Delta\kappa_t = \kappa_t - \kappa_{t-1}$ the size of the capacity interval at step t. If the lowest loss is achieved with the lowest capacity, resulting in $t^{*,\boldsymbol{c}} = 1$, we set AULCC $= 0$.

Explictness. We define the **explicitness** (E) of representation $\boldsymbol{c}$ for predicting factor z_j with predictor class $\mathcal{F}$ as

$$E(z_j, \boldsymbol{c}; \mathcal{F}) = 1 - \frac{\text{AULCC}(z_j, \boldsymbol{c}; \mathcal{F})}{\frac{1}{2}(\kappa_T - \kappa_1)(\ell_j^b - \ell_j^*)},$$

where ℓ_j^b is a suitable baseline loss (e.g., that of $\mathbb{E}[z_j]$) and ℓ_j^* a suitable lowest loss (e.g., 0) for $\mathcal{F}$. Here, the denominator represents the area of the light-blue triangle in Fig. 2, *normalizing* the AULCC such that $E \in [-1, 1]$ so long as $\ell_j^* < \ell_j^b$. The best score $E = 1$ means that the best loss was achieved with the lowest-capacity probe f_j^1, i.e., $\ell_j^{*,\boldsymbol{c}} = \ell_j^{1,\boldsymbol{c}}$ and $\kappa_{*,\boldsymbol{c}} = \kappa_1$, and thus our representation $\boldsymbol{c}$ was explicit or easy-to-use for predicting z_j with $f \in \mathcal{F}$ since there was *no surplus capacity required (beyond κ_1) to achieve our lowest loss*. In contrast, $E = 0$ means that the loss reduced *linearly* from ℓ_j^b to ℓ_j^* with increased probe capacity, i.e., AULCC = Normalizer in Fig. 2. More generally, if $\ell^{*,\boldsymbol{c}} = \ell^*$, i.e. the lowest loss for $\mathcal{F}$ can be reached with $\boldsymbol{c}$, then $E < 0$ implies that the loss decreased *sub-linearly* while $E > 0$ implies it decreased *super-linearly*. The overall explicitness score is simply the arithmetic mean over the K factors, i.e., the z_js.

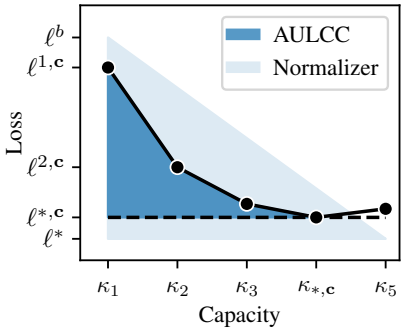

Figure 2: **Explicitness via the area under the loss-capacity curve (AULCC).** Here, $\kappa_1, ..., \kappa_T$ (x-axis) are a sequence of increasing function capacities and $\ell^{1,\mathbf{c}}, ..., \ell^{T,\mathbf{c}}$ (y-axis) are the losses achieved by the corresponding optimal predictors for $\mathbf{c}$. The lowest loss $\ell^{*,\mathbf{c}}$ is achieved at capacity $\kappa_{*,\mathbf{c}}$, while ℓ^b and ℓ^* are suitable baseline and best-possible losses for the probe class.

E vs. I. While the informativeness score I_j captures the (total) amount of information in $\boldsymbol{c}$ about z_j, the explicitness score E_j captures the *ease-of-use* of this information. In particular, while I_j is quantified by the *lowest prediction error with any (utilised) capacity* $\ell^{*,\boldsymbol{c}}$, corresponding to a single point on $\boldsymbol{c}$'s loss-capacity curve, E_j is quantified by the *area under this curve*.

A fine-grained picture of identifiability. Compared to the commonly-used mean correlation coefficient (MCC) or Amari distance (Amari et al., 1996; Yang & Amari, 1997), the D, C, I, E scores represent empirical measures which: (i) easily extend to mismatches in dimensionalities, i.e., $L > K$; and (ii) provide a more fine-grained picture of identifiability (violations), for if the initial probe capacity κ_1 is linear and R satisfies Corollary 3.5, we have that:

- $D = C = I = E = 1 \implies$ identified up to sign and permutation (Defn. 3.1);
- $D = C = I = 1 \implies$ identified up to permutation and element-wise reparametrisation (Defn. 3.2);
- $I = E = 1 \implies$ identified up to invertible linear transformation (cf. Khemakhem et al., 2020).

Thus, if $D = C = I = E = 1$ does not hold exactly, which score deviates the most from one may provide valuable insight into the type of identifiability violation.

Probe classes. As emphasized above, whether or not a representation $\boldsymbol{c}$ is explicit or easy-to-use for predicting factor z_j depends on the class of probe $\mathcal{F}$ used, e.g., random forests (RFs) or multi-layer perceptrons (MLPs). More generally, the explicitness of a representation depends on the way in which it is used in downstream applications, with different downstream uses or probes resulting in different definitions of explicit or easy-to-use information. We thus conduct experiments with different probes in § 6.

Representation	I	E	S
AE-5	0.75	0.74	1.4
AE-7	0.92	0.71	1.0
AE-10	0.99	0.72	0.7
AE-100	1.0	0.90	0.07
AE-500	1.0	0.93	0.01

Table 1: I, E and S scores for auto-encoders of various sizes on `MPI3D-Real` with MLP probes.

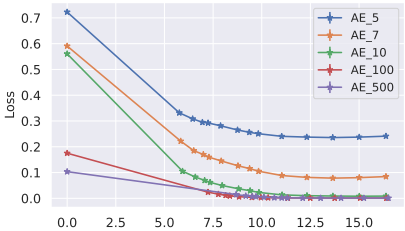

Figure 3: Loss-capacity curves for auto-encoders of various sizes on `MPI3D-Real` with MLP probes.

4.2 Size (S)

We next introduce a measure of representation size (S), motivated by the observation that larger representations tend to be both more informative and more explicit (see Tab. 1, more details below). Reporting S thus allows size-informativeness and size-explicitness trade-offs to be analysed.

A measure of size. We measure representation *size* (S) relative to the ground-truth as:

$$S = \frac{K}{L} = \frac{\dim(\boldsymbol{z})}{\dim(\boldsymbol{c})}.$$

When $L \geq K$, as often the case, we have $S \in (0, 1]$ with the perfect score being $S = 1$. However, if we also consider the $L < K$ case, which would likely sacrifice some informativeness, we have $S \in (1, K]$.

Larger representations are often more *informative*. When $\dim(\boldsymbol{c}) < \dim(\boldsymbol{z})$, it is intuitive that larger representations are more informative—they can simply preserve more information about $\boldsymbol{z}$. When $\dim(\boldsymbol{c}) > \dim(\boldsymbol{z})$, however, it is also common for larger representations to be more informative, perhaps due to an easier optimization landscape (Frankle & Carbin, 2019; Golubeva et al., 2021). Tab. 1 illustrates this point, where AE-5 denotes an autoencoder with $\dim(\boldsymbol{c}) = 5$. Note that $\dim(\boldsymbol{z}) = 7$ for `MPI3D-Real` (see § 6).

Larger representations are often more *explicit*. The explicitness of a representation also depends on its size: larger representations tend to be more explicit, as is apparent from the second column of Tab. 1. To explain this, we plot the corresponding loss-capacity curves in Fig. 3. Here we see that the increased explicitness (i.e., smaller AULLC) of larger representations stems from a substantially lower initial loss when using a linear-capacity MLP probe. The fact that larger representations perform better with linear capacity is unsurprising as they have more parameters, and is often exploited in self-supervised learning to improve ImageNet performance with linear capacity (Chen et al., 2020).

4.3 Probe-agnostic feature importances

Finally, to meaningfully discuss more flexible probe-function choices within the DCI-ES framework, we point out that the D and C scores can be computed for arbitrary black-box probes f by using *probe-agnostic* feature-importance measures. In particular, in our experiments (§ 6), we use SAGE (Covert et al., 2020) which summarises each feature's importance based on its contribution to predictive performance, making use of Shapley values (Shapley, 1953) to account for complex feature interactions. Such probe-agnostic measures allow D and C to be computed for probes with no inherent, built-in notion of feature importance (e.g., for MLPs), thereby generalising the Lasso and Random Forest examples of Eastwood & Williams (2018, § 4.3). While SAGE has several practical advantages over other probe-agnostic methods (see, e.g., Covert et al., 2020, Table 1), it may not satisfy the required conditions to link the D and C scores to different identifiability equivalence classes (see Remark 3.6). Future work may explore some of the alternative methods which do, e.g., by looking at a feature's mean *absolute* attribution value (Lundberg & Lee, 2017).

5 Related work

Explicit representations. Eastwood & Williams (2018, § 2) noted that the informativeness score with a linear probe quantifies the amount of information in $\boldsymbol{c}$ about $\boldsymbol{z}$ that is "explicitly represented", while

Ridgeway & Mozer (2018, § 3) proposed a measure of "explicitness" which simply reports the informativeness score with a linear probe. In contrast, our DCI-ES framework differentiates between the amount of information in $\boldsymbol{c}$ about $\boldsymbol{z}$ (*informativeness*) and the ease-of-use of this information (*explicitness*). This allows a more fine-grained analysis of the relationship between $\boldsymbol{c}$ and $\boldsymbol{z}$, both theoretically (distinguishing between more identifiability equivalence classes; § 3) and empirically (§ 6).

Loss-capacity curves. Plotting loss as a function of model complexity or capacity has long been used in statistical learning theory, e.g., for studying the bias-variance trade-off (Hastie et al., 2009, Fig. 7.1). More recently, such loss-capacity curves have been used to study the double-descent phenomenon of neural networks (Belkin et al., 2019; Nakkiran et al., 2021) as well as the scaling laws of language models (Kaplan et al., 2020). However, they have yet to be used for assessing the quality or explicitness of representations.

Loss-data curves. Whitney et al. (2020) use loss-data curves, which plot loss against dataset size, to assess representations. They measure the quality of a representation by the *sample complexity* of learning probes that achieve low loss on a task of interest. Loss-data curves are also studied under the term *learning curves* in standard/purely supervised-learning settings (see, e.g., Viering & Loog, 2021, for a recent review). In contrast, we focus on *functional complexity* and the task of predicting the data-generative factors $\boldsymbol{z}$, and then discuss the functional complexity/capacity for other tasks $\boldsymbol{y}$ in § 7.

6 Experiments

6.1 Setup

Data. We perform our analysis of loss-capacity curves on the `MPI3D-Real` (Gondal et al., 2019) and `Cars3D` (Reed et al., 2015) datasets. `MPI3D-Real` contains $\approx$ 1M real-world images of a robotic arm holding different objects with seven annotated ground-truth factors: object colour (6), object shape (6), object size (2), camera height (3), background colour (3) and two degrees of rotations of the arm (40×40); numbers in brackets indicate the number of possible values for each factor. `Cars3D` contains $\approx 17.5k$ rendered images of cars with three annotated ground-truth factors: camera elevation (4), azimuth (24) and car type (183).

Representations. We use the following synthetic baselines and standard models as representations:

- *Noisy labels:* $\boldsymbol{c} = \boldsymbol{z} + \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, 0.01 \cdot \boldsymbol{I}_K)$.
- *Uniform mix:* $\boldsymbol{c} = W\boldsymbol{z}$, where $W_{ij} = \frac{1}{LK} + \epsilon_{ij}$ with $\epsilon_{ij} \sim \mathcal{N}(0, 0.0016)$ (to ensure invertibility of W a.s.).
- *Raw data (pixels):* $\boldsymbol{c} = \boldsymbol{x}$.
- *Others:* We also use VAEs (Kingma & Welling, 2014) with 10 latents (L=10), β-VAEs (Higgins et al. 2017, β=16, L=10); and an ImageNet-pretrained ResNet18 (He et al. 2016, L=512).

Probes. We use *multi-layer perceptrons* (MLPs), *random forests* (RFs, Breiman 2001) and *random Fourier features* (RFFs, Rahimi & Recht 2007) as probes to predict $\boldsymbol{z}$ from $\boldsymbol{c}$, with RFFs having a linear classifier on top. For MLPs, we start with linear probes (no hidden layers) then increase capacity by adding two hidden layers and varying their widths from $2 \times K$ to $512 \times K$. We then measure capacity based on the number of "extra" parameters beyond that of the linear probe, and compute feature importances using SAGE with permutation-sampling estimators and marginal sampling of masked values (https://github.com/iancovert/sage). For RFs, we use ensembles of 100 trees, control capacity by varying the maximum allowed depth between 1 and 32, and compute feature importances using Gini importance. For RFFs, we control capacity by exponentially increasing the number of random features from 2^4 to 2^{17} , and compute feature importances using SAGE.

Implementation details. We split the data into training, validation and testing subsets of size 295k, 16k, and 726k respectively for `MPI3D-Real` and 12.6k, 1.4k, 3.4k for `Cars3d`. We use the validation split for hyperparameter selection and report results on the test split. We train the MLP probes using the Adam (Kingma & Ba, 2015) optimizer for 100 epochs. We use mean-square error and cross-entropy losses for continuous and discrete factors z_j, respectively. To compute E, we use $\mathbb{E}[z_j]$ or a random classifier as baseline losses for continuous and discrete z_j, respectively.

Table 2: **DCI-ES scores for different probes and representations.** Empirical scores using MLP, Random Fourier Features (RFF), and Random Forest (RF) probes trained on the `MPI3D-Real` and `Cars3d` datasets, as well as theoretical/oracle for MLPs (MLP*) for some simple representations. We show averages over 3 random seeds; standard deviations were all < 0.05.

Representation	Probe	MPI3D					CARS3D				
		D	**C**	**I**	**E**	**S**	**D**	**C**	**I**	**E**	**S**
GT Labels $\boldsymbol{z}$	MLP*	1	1	1	1	1	1	1	1	1	1
Noisy labels	MLP*	1	1	0.9	1	1.0	1	1	0.9	1	1.0
	MLP	0.97	0.97	0.89	0.99	1.0	0.99	0.99	0.92	0.99	1.0
	RFF	0.97	0.97	0.88	0.99	1.0	1.0	1.0	0.91	1.0	1.0
	RF	0.93	0.93	0.89	0.98	1.0	0.95	0.95	0.92	0.99	1.0
Uniform mix	MLP*	0	0	1	1	1.0	0	0	1	1	1.0
	MLP	0.13	0.22	1.0	1.0	1.0	0.21	0.22	1.0	1.0	1.0
	RFF	0.11	0.21	1.0	0.94	1.0	0.19	0.19	1.0	1.0	1.0
	RF	0.17	0.21	1.0	0.72	1.0	0.08	0.12	0.99	0.78	1.0
VAE	MLP	0.15	0.14	0.99	0.71	0.7	0.18	0.11	0.95	0.80	0.3
	RFF	0.13	0.14	0.97	0.69	1.0	0.16	0.11	0.91	0.87	0.3
	RF	0.10	0.10	0.93	0.65	0.7	0.14	0.09	0.80	0.81	0.3
β-VAE	MLP	0.46	0.41	0.96	0.74	0.7	0.27	0.23	0.94	0.78	0.3
	RFF	0.41	0.38	0.92	0.71	0.7	0.31	0.23	0.86	0.86	0.3
	RF	0.39	0.35	0.94	0.76	0.7	0.20	0.17	0.84	0.83	0.3
ImgNet-pretr	MLP	0.16	0.10	0.99	0.82	0.01	0.22	0.07	0.88	0.86	0.006
	RFF	0.15	0.13	0.96	0.58	0.01	0.24	0.10	0.83	0.65	0.006
	RF	0.35	0.20	0.89	0.78	0.01	0.20	0.09	0.62	0.83	0.006
Raw data	MLP	0.22	0.16	0.99	0.82	0.001	0.39	0.27	0.95	0.84	0.0002
	RFF	0.37	0.14	0.97	0.44	0.001	0.32	0.24	0.87	0.64	0.0002
	RF	0.84	0.41	0.96	0.80	0.001	0.53	0.31	0.86	0.82	0.0002

6.2 RESULTS

Loss-capacity curves. Fig. 1 depicts loss-capacity curves for the three probes and two datasets, averaged over factors z_j. In all six curves, the noisy-labels baseline performs well with low-capacity and then is surpassed by other representations given sufficient capacity, as expected. Note that the uniform-mix baseline immediately achieves ≈ 0 loss with MLP probes but not with RFF or RF probes, supporting the idea that the explicitness or ease-of-use of a representation also depends on the way in which it is used. Also note that, with MLP probes and $\log(\text{excess \#params})$ as the capacity measure, larger input representations are afforded more parameters with a linear probe and thus are more expressive. This further explains why larger representations are often more explicit, and highlights the difficulty of measuring the capacity of MLPs—an active area of research in its own right, which we discuss in § 7. Finally, in Appendix A, we investigate the effect of dataset size by plotting loss-capacity curves for different-sized datasets, observing that having more data reduces the performance gap between: (i) synthetic and learned representations; and (ii) small and large representations (see Fig. 5).

DCI-ES scores. Tab. 2 reports the corresponding DCI-ES scores, along with some oracle scores for MLPs. Note that: (i) the GT labels $\boldsymbol{z}$ get perfect scores of 1 for all metrics; (ii) by attaining very low mixing scores but near-perfect explicitness scores, the uniform-mix representation exposes the key difference between *mixing-based* (D,C) and functional-capacity-based (E) measures of the *simplicity of the* $\boldsymbol{c}$*-*$\boldsymbol{z}$ *relationship*; (iii) larger representations (ImgNet-pretr, raw data) tend to be more explicit than smaller ones (VAE, β-VAE), with S and E together capturing this size-explicitness trade-off; and (iv) β-VAE achieves better mixing-based scores (D,C) but similar E scores compared to VAE, illustrating that these two "disentanglement" notions are indeed orthogonal and complementary.

7 DISCUSSION

Why connect disentanglement and identifiability? Connecting prediction-based evaluation in the disentanglement literature to the more theoretical notion of identifiability has several benefits.

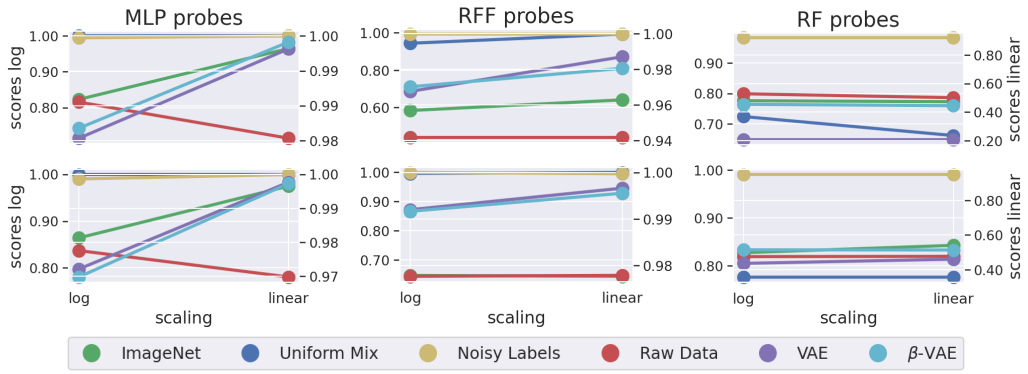

Figure 4: Explicitness scores for multiple representations (see legend) on `MPI3D-Real` and `Cars3D`. Each pair of points represents the explicitness score with logarithmic (left) and linear (right) scaling.

Firstly, it provides a concrete link between two often-separate communities. Secondly, it endows the often empirically-driven or practice-focused disentanglement metrics with a solid and well-studied theoretical foundation. Thirdly, compared to the commonly-used MCC or Amari distance, it provides the ICA or identifiability community with more fine-grained empirical measures, as discussed in § 4.1.

Measuring probe capacity. Our measure of explicitness E depends strongly on the choice of capacity measure for a probe or function class. For some probes, e.g., RFs or RFFs, there exist natural measures of capacity. However, for other probes like MLPs, coming up with a good capacity measure is itself an important and active area of research (Jiang et al., 2020; Dziugaite et al., 2021). While we used the number of parameters to measure the *available* (or upper-bound) probe capacity, future work may improve the soundness of E by leveraging recent (and future) advances in MLP capacity measures, e.g., those which measure the *used* or *effective* capacity of a trained MLP (Hanin & Rolnick, 2019; Maddox et al., 2020). Another difficulty arises from choosing a capacity *scale*, with different scales (e.g., log, linear, etc.) leading to different loss-capacity-curve shapes, and thus different explicitness scores. To investigate the extent of this issue, i.e., the sensitivity of our explicitness measure to the choice of scale, Fig. 4 compares the explicitness scores when using log and linear scaling. Here we see that the *ranking* essentially remains the same except for the raw-data representation with MLP probes.

Measuring feature importance. Similarly, the choice of feature-importance measure has a strong influence on the D and C scores, with some probes having natural or in-built measures (e.g., random forests) and others not (e.g., MLPs). For the latter, we proposed the use of probe-agnostic feature-importance measures like SAGE, and specified the conditions (Corollary 3.5) that importance measures must satisfy if the resulting D and C scores are to be connected to identifiability. As with probe capacity, coming up with good measures of feature importance is its own orthogonal field of study (e.g., model explainability), with future advances likely to improve the DCI-ES framework.

What about explicitness for other tasks $\boldsymbol{y}$? While we focused on the explicitness of a representation for predicting or recovering $\boldsymbol{z}$, one may also be interested in its explicitness for other tasks or labels $\boldsymbol{y}$. While it is often implicitly assumed that ease of predicting $\boldsymbol{z}$ correlates with ease-of-use for other common tasks of interest (e.g., object classification, segmentation, etc.), future work could directly evaluate the explicitness of a representation for particular tasks $\boldsymbol{y}$. For example, one could consider the entire loss-capacity curve when benchmarking self-supervised representations on ImageNet, rather than just linear-probe performance (a single slice). In addition, one could explore the trade-off between *explicit but task-specific* and *implicit but task-agnostic* representations.

8 Conclusion

We have presented DCI-ES—an extended disentanglement framework with two new complementary measures of representation quality—and proven its connection to identifiability. In particular, we have advocated for additionally measuring the explicitness (E) of a representation by the functional capacity required to use it, and proposed to quantify this explicitness using a representation's loss-capacity curve. Together with the size (S) of a representation, we believe that the extended DCI-ES framework allows for a more fine-grained and nuanced benchmarking of representation quality.

REFERENCES

Shun-ichi Amari, Andrzej Cichocki, Howard Hua Yang, et al. A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing Systems*, pp. 757–763, 1996.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Ricky TQ Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in vaes. In *Advances in Neural Information Processing Systems*, volume 31, pp. 2615–2625, 2018.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pp. 1597–1607. PMLR, 2020.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in Neural Information Processing Systems*, 29, 2016.

Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

Ian Covert, Scott M Lundberg, and Su-In Lee. Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, 33:17212–17223, 2020.

Gintare Karolina Dziugaite, Alexandre Drouin, Brady Neal, Nitarshan Rajkumar, Ethan Caballero, Linbo Wang, Ioannis Mitliagkas, and Daniel M. Roy. In search of robust measures of generalization, 2021.

Cian Eastwood and Christopher K I Williams. A framework for the quantitative evaluation of disentangled representations. In *International Conference on Learning Representations*, 2018.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.

Anna Golubeva, Guy Gur-Ari, and Behnam Neyshabur. Are wider nets better given the same number of parameters? In *International Conference on Learning Representations*, 2021.

Muhammad Waleed Gondal, Manuel Wuthrich, Djordje Miladinovic, Francesco Locatello, Martin Breidt, Valentin Volchkov, Joel Akpo, Olivier Bachem, Bernhard Schölkopf, and Stefan Bauer. On the transfer of inductive bias from simulation to the real world: a new disentanglement dataset. *Advances in Neural Information Processing Systems*, 32, 2019.

Boris Hanin and David Rolnick. Deep relu networks have surprisingly few activation patterns. *Advances in Neural Information Processing Systems*, 32:361–370, 2019.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

Aapo Hyvärinen and Petteri Pajunen. Nonlinear independent component analysis: Existence and uniqueness results. *Neural networks*, 12(3):429–439, 1999.

Aapo Hyvarinen, Hiroaki Sasaki, and Richard Turner. Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 859–868. PMLR, 2019.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *International Conference on Learning Representations*, 2020.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pp. 2207–2217. PMLR, 2020.

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International Conference on Machine Learning*, pp. 2649–2658, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

Tejas D Kulkarni, William F Whitney, Pushmeet Kohli, and Josh Tenenbaum. Deep convolutional inverse graphics network. *Advances in Neural Information Processing Systems*, 28, 2015.

Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40, 2017.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. A sober look at the unsupervised learning of disentangled representations and their evaluation. *Journal of Machine Learning Research*, 21(209):1–62, 2020.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 2017.

Wesley J. Maddox, Gregory Benton, and Andrew Gordon Wilson. Rethinking parameter counting in deep models: Effective dimensionality revisited, 2020.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 1(12), 2021.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, pp. 1177–1184, 2007.

Scott E Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. Deep visual analogy-making. *Advances in neural information processing systems*, 28, 2015.

Karl Ridgeway and Michael C Mozer. Learning deep disentangled embeddings with the f-statistic loss. In *Advances in Neural Information Processing Systems*, volume 31, pp. 185–194, 2018.

Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.

Raphael Suter, Djordje Miladinovic, Bernhard Schölkopf, and Stefan Bauer. Robustly disentangled causal mechanisms: Validating deep representations for interventional robustness. In *International Conference on Machine Learning*, pp. 6056–6065, 2019.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

Tom Viering and Marco Loog. The shape of learning curves: a review. *arXiv preprint arXiv:2103.10948*, 2021.

William F Whitney, Min Jae Song, David Brandfonbrener, Jaan Altosaar, and Kyunghyun Cho. Evaluating representations by the complexity of learning low-loss predictors. *arXiv preprint arXiv:2009.07368*, 2020.

Howard Hua Yang and Shun-ichi Amari. Adaptive online learning algorithms for blind separation: maximum entropy and minimum mutual information. *Neural Computation*, 9(7):1457–1482, 1997.

A DIFFERENT AMOUNTS OF DATA

In Fig. 5 we present loss-capacity curves obtained when using different amounts of data to train the MLP probes. As shown, larger datasets have smaller performance gaps between (i) synthetic and learned representations; and (ii) small and large representations.

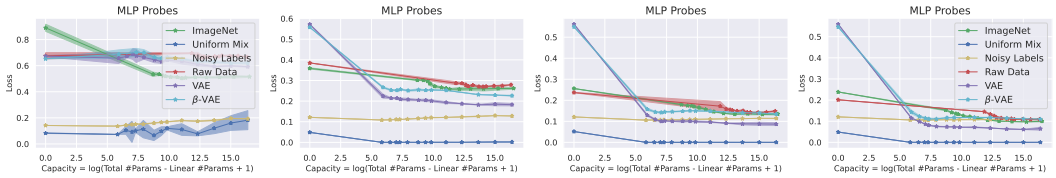

Figure 5: Loss-capacity curves for `MPI3D-Real` subsets of size 100, 1000, 5000 and 10000 respectively.