InfoNCE: Identifying the Gap Between Theory and Practice

Evgenia Rusak[*,1,2]   Patrik Reizinger[*,2,4]   Attila Juhos[*,2,4]   Oliver Bringmann[1]
Roland Zimmermann[†,2,4]   Wieland Brendel[†,2,3,4]

[1]University of Tübingen   [2]Max Planck Institute for Intelligent Systems
[3]ELLIS Institute Tübingen   [4]Tübingen AI Center

Abstract

Prior theory work on Contrastive Learning via the InfoNCE loss showed that, under certain assumptions, the learned representations recover the ground-truth latent factors. We argue that these theories overlook crucial aspects of how CL is deployed in practice. Specifically, they either assume equal variance across all latents or that certain latents are kept invariant. However, in practice, positive pairs are often generated using augmentations such as strong cropping to just a few pixels. Hence, a more realistic assumption is that all latent factors change with a continuum of variability across all factors. We introduce AnInfoNCE, a generalization of InfoNCE that can provably uncover the latent factors in this anisotropic setting, broadly generalizing previous identifiability results in CL. We validate our identifiability results in controlled experiments and show that AnInfoNCE increases the recovery of previously collapsed information in CIFAR10 and ImageNet, albeit at the cost of downstream accuracy. Finally, we discuss the remaining mismatches between theoretical assumptions and practical implementations.

1 INTRODUCTION

Self-Supervised Learning (SSL) employs surrogate objectives to learn useful representations from large, un-labeled datasets. A particularly successful and simple sub-family, called *Contrastive Learning (CL)*, minimizes the representational distance between similar samples (e.g., augmentations of the same sample) while maximizing the distance between dissimilar samples (drawn i.i.d.) (Oord et al., 2018; Chen et al., 2020a,c; Caron et al., 2020; Chen et al., 2020b). Two similar samples form a *positive pair*, while two dissimilar samples form a *negative pair*. Most of today's successful SSL techniques build on this principle, including non-contrastive ones that only use positive pairs (Chen and He, 2021; Grill et al., 2020; Zbontar et al., 2021).

InfoNCE (Oord et al., 2018) is a commonly used objective for CL. Zimmermann et al. (2021) have shown that under certain assumptions put on the *Data Generating Process (DGP)*, a model can recover the ground-truth latent factors if trained with InfoNCE. Their identifiability result requires an isotropic change across all latent factors in the positive pair. Von Kügelgen et al. (2021) relaxed the isotropic assumption on the positive conditional distributions to two partitions: content (δ-distribution) and style (non-degenerate distribution), yielding a weaker block-identifiability result.

In practice, positive pairs for CL are generated using augmentations. For instance, strong cropping, which often extracts just a tiny area of the image, is argued to be crucial for high downstream performance (Chen et al., 2020a). Assuming an underlying latent variable model, these augmentations correspond to changes in latent factors, modeled by the positive conditional distribution. Cropping discards a significant amount of information and, hence, harshly changes some latent factors. Another augmentation, Gaussian blurring, causes moderate perturbations to the sharpness of objects. Lastly, latents alluding to class information are invariably left intact. We posit that augmentations imply anisotropic changes to different latent factors, a setting previously not covered theoretically. In fact, we believe that this gap between theory and practice can in part explain why empirical work shows that CL loses information (Chen et al., 2020a; Jing et al., 2022),

---

[*]Equal contribution. [†]Joint senior authors. Correspondence to evgenia.rusak@uni-tuebingen.de

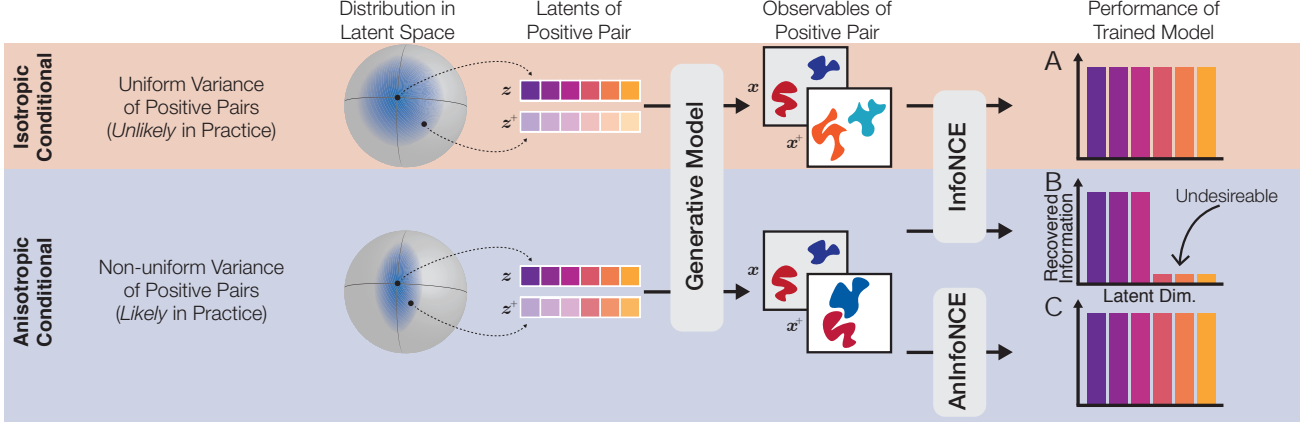Code at github.com/brendel-group/AnInfoNCE. Website at brendel-group.github.io/AnInfoNCE/.

Figure 1: **Illustration of the Mismatch Between the Standard Contrastive Learning (CL) Model and Practice. A:** CL with the commonly used InfoNCE objective is identifiable when all latents change to the same extent across the positive pair (Zimmermann et al., 2021), which is unlikely to happen in practice. **B:** The more likely scenario, when augmentations affect different latents to a different extent, leads to dimensional collapse and information loss. **C:** Our proposed objective, Anistropic InfoNCE (AnInfoNCE), models features that can vary to a different degree in the positive pair, avoiding collapse.

despite the identifiability guarantees of Zimmermann et al. (2021); von Kügelgen et al. (2021).

The information loss in CL hinders the development of task-agnostic models, which require an accurate and holistic low- and high-level representation of their inputs. The failure of InfoNCE to capture style latents is already apparent in popular vision-language models (VLMs) trained by adaptations of InfoNCE, such as CLIP (Radford et al., 2021). By focusing on a few semantic concepts and dismissing low-level style information, CLIP-based VLMs struggle with low-level vision (Rahmanzadehgervi et al., 2024), scene compositionality (Yuksekgonul et al., 2023a), and attribute binding tasks (Trusca et al., 2024). Identifiability considers how to reconstruct *all* latent factors underlying the inputs, aiming to capture general-purpose representations, suitable for a plethora of downstream tasks.

Our work serves two purposes. Firstly, we push the frontier of identifiability research by generalizing existing results to anisotropic latent distributions, which allow for more expressive generative processes. Equally importantly, we design a non-trivial extension of InfoNCE, coined AnInfoNCE, that is provably able to capture these inhomogeneous latents. We validate our results in controlled experiments and show that AnInfoNCE successfully increases latent recovery on CIFAR10 and ImageNet, although at the expense of downstream classification accuracy. We perform ablations to understand the remaining gap caused by the accuracy–identifiability trade-off. Our **contributions**:

- We introduce AnInfoNCE, a generalized contrastive loss, that provably recovers latents with augmentation-induced anisotropic variances (§ 3.2).

- We extend the identifiability result to hard negative mining (§ 3.3) and loss ensembles (§ 3.4).
- We verify our loss in synthetic and well-controlled image experiments, having full knowledge of the ground-truth generative process (§ 4.1 and § 4.2).
- We further demonstrate the efficacy of AnInfoNCE on CIFAR10 and ImageNet in recovering the latent factors, but observe a trade-off with downstream classification accuracy, done with linear probing (§ 4.3).
- We link the accuracy–identifiability trade-off on real-world data to augmentations and analyze remaining mismatches between CL theory and practice (§ 5).

2 BACKGROUND

Contrastive Learning. Contrastive Learning (CL) is an SSL paradigm where an encoder $\boldsymbol{f} : \mathcal{X} \to \mathcal{Z}$ learns to map observations $\boldsymbol{x}$ to latent vectors $\boldsymbol{z}$. CL uses positive pairs $(\boldsymbol{x}, \boldsymbol{x}^+)$ (similar data points, e.g., augmentations of the same sample), and negative pairs $(\boldsymbol{x}, \boldsymbol{x}^-)$ (dissimilar data points, e.g., uniformly drawn from the dataset). CL mostly employs the InfoNCE loss (Sohn, 2016; Gutmann and Hyvärinen, 2010; Oord et al., 2018):

$$\mathcal{L}_{\text{INCE}}(\boldsymbol{f}) = \tag{1}$$
$$= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}, \boldsymbol{x}^+ \\ \{\boldsymbol{x}_i^-\}}} \left[-\ln \frac{e^{\boldsymbol{f}^\top(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x}^+)/\boldsymbol{\tau}}}{e^{\boldsymbol{f}^\top(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x}^+)/\boldsymbol{\tau}} + \sum_{i=1}^{M} e^{\boldsymbol{f}^\top(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x}_i^-)/\boldsymbol{\tau}}} \right],$$

where $\boldsymbol{\tau}$ is a scalar temperature, and $\boldsymbol{x}^+ \sim p(\boldsymbol{x}^+|\boldsymbol{x})$ and $(\boldsymbol{x}, \boldsymbol{x}^-) \sim^{i.i.d.} p(\boldsymbol{x})$ are the positive and negative pairs observed.

In order to analyze the behavior of CL, we surmise that

the observations come from a Data Generating Process (DGP) $\boldsymbol{g} : \mathcal{Z} \to \mathcal{X}$. The samples follow a marginal distribution $p(\boldsymbol{z})$ over the latent space $\mathcal{Z}$, and the positive samples follow a conditional $p(\boldsymbol{z}^+|\boldsymbol{z})$. Under certain assumptions, CL inverts the DGP, i.e., the composition $\boldsymbol{h} = \boldsymbol{f} \circ \boldsymbol{g}$ is a trivial map (usually an affine linear map, depending on the assumptions) (Zimmermann et al., 2021). The identifiability of SSL is further analyzed by von Kügelgen et al. (2019); Lyu et al. (2021) and Cui et al. (2022), with insightful connections between contrastive and non-contrastive methods in Balestriero and LeCun (2022); Garrido et al. (2023); Assran et al. (2023).

Content-style Partitioning of $\mathcal{Z}$. Von Kügelgen et al. (2021) present an identifiability result for a partitioning of $\boldsymbol{z} \in \mathcal{Z}$ into invariant (*content*) $\boldsymbol{z}_c \in \mathcal{Z}_c$ and variant (*style*) $\boldsymbol{z}_s \in \mathcal{Z}_s$ dimensions with $\mathcal{Z} = \mathcal{Z}_c \times \mathcal{Z}_s$ (see Appx. A). The content latents $\boldsymbol{z}_c$ in the positive pairs are assumed to follow a δ-distribution, whereas the distribution of $\boldsymbol{z}_s$ is not degenerate, yielding $p(\boldsymbol{z}^+|\boldsymbol{z}) = \delta(\boldsymbol{z}_c^+ - \boldsymbol{z}_c) p(\boldsymbol{z}_s^+|\boldsymbol{z}_s)$. Note that Jing et al. (2022) implicitly use a similar distinction—defined as different augmentation amplitudes. Thus, their approach is akin to generalized content and style variables with variances $\boldsymbol{\sigma}_s^2 \gg \boldsymbol{\sigma}_c^2 > 0$; Dubois et al. (2022) also have a similar approach. Other means to partition the latent variables are tied to the "simplicity" of features (Chen et al., 2021) or sparsity arguments Wen and Li (2021). In concurrent work, Eastwood et al. (2023) propose a *relative* notion of content and style, assuming the presence of multiple environments, each with distinct content and style dimensions.

Practical Shortcomings of CL. Despite the identifiability guarantees of CL methods (Zimmermann et al., 2021; von Kügelgen et al., 2021; Cui et al., 2022), practitioners struggle to make models learn all underlying latent factors. Chen et al. (2020a) noted that the representations could become invariant to some input transformations. Several works connected data augmentations to failing to learn all latent factors (Jing et al., 2022; Wang et al., 2022; Bendidi et al., 2023; Wu et al., 2020; HaoChen et al., 2021; Cosentino et al., 2022; Wagner et al., 2022; Zhai et al., 2023).

A potential way to improve augmentation readout (a proxy for capturing the information in highly-varying, i.e., style, latents) is by introducing inductive biases. Lee et al. (2021a) add a regularizer to predict the augmentation parameters between two views, whereas Dangovski et al. (2021) use a regularizer to induce equivariance by predicting rotation parameters. Xiao et al. (2021) introduce Leave-one-out Contrastive Learning (LooC) with multiple latent spaces. They train multiple embeddings by leaving out one augmentation at a time, thus inducing different invariances, which re-

sults in better downstream and augmentation readout performance. Zhang and Ma (2022) propose a similar strategy with different augmentation partitions and calculate the contrastive loss at different hierarchical levels in the encoder, meant to induce invariances at different feature levels. Recently, Eastwood et al. (2023) proposed a similar method to LooC (Xiao et al., 2021), but with strong theoretical guarantees.

3 IDENTIFIABLE ANISOTROPIC CL

3.1 Setup and Intuition

We argue that the augmentations usually employed in applied Contrastive Learning (CL) do not change all latents evenly. For example, color distortions do not drastically affect latents encoding semantic information or edges. We use a Latent Variable Model (LVM) to model CL, where a positive conditional $p(\boldsymbol{z}^+|\boldsymbol{z})$ relates the positive pairs $\boldsymbol{z}^+$ to the anchor sample $\boldsymbol{z}$. The augmentations used in practice induce a specific form of the positive conditional $p(\boldsymbol{z}^+|\boldsymbol{z})$, e.g., by assigning larger variance to latents strongly affected by augmentations. As the current CL theory does not account for augmentations that affect latents to different levels, we propose an anisotropic model for CL.

Following the practice of CL, we assume that the latent space is a $(d-1)$-dimensional hypersphere: $\mathcal{Z} = \mathbb{S}^{d-1}$. As proposed by Zimmermann et al. (2021), the anchor latent $\boldsymbol{z}$, and all negative pairs $\{\boldsymbol{z}_i^-\}$ are assumed to be uniformly distributed on $\mathcal{Z}$. Akin to Kirchhof et al. (2022), we allow non-uniform variances for the individual latent factors in the positive pair $\boldsymbol{z}^+$. We model this in the positive conditional by scaling the latent factors with *unknown* positive *concentration parameters*, collected in the diagonal matrix $\boldsymbol{\Lambda}$:

$$p(\boldsymbol{z}) = \text{const}, \quad p(\boldsymbol{z}^+|\boldsymbol{z}) \propto e^{-(\boldsymbol{z}^+ - \boldsymbol{z})^\top \boldsymbol{\Lambda} (\boldsymbol{z}^+ - \boldsymbol{z})}, \quad (2)$$

where a higher value of $\boldsymbol{\Lambda}_{ii}$ means a higher concentration of $\boldsymbol{z}^+$ around the anchor $\boldsymbol{z}$ along dimension i. Introducing $\boldsymbol{\Lambda}$ in the conditional generalizes previous works: $\boldsymbol{\Lambda} = \mathbf{I}_d$ recovers the von Mises-Fisher (vMF) conditional from Zimmermann et al. (2021), whereas two latent partitions, namely content (high $\boldsymbol{\Lambda}_{ii}$) and style (low $\boldsymbol{\Lambda}_{jj}$), yields the model of von Kügelgen et al. (2021). Our new conditional incorporates an entire spectrum of distributions between the two edge cases.

We assume that observations $\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_i^-\}$ are the result of passing the respective latents through an invertible generator $\boldsymbol{g} : \mathcal{Z} \to \mathcal{X} \subseteq \mathbb{R}^D$. Our goal is to train an encoder $\boldsymbol{f} : \mathbb{R}^D \to \mathcal{Z}$ such that the reconstructed latents, $\hat{\boldsymbol{z}} = \boldsymbol{f}(\boldsymbol{x})$, recover $\boldsymbol{z}$ up to permissible transformations. To train $\boldsymbol{f}$, we introduce our new loss:

$$\mathcal{L}_{\text{AINCE}}(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}}) = \tag{3}$$

$$= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}, \boldsymbol{x}^+ \\ \{\boldsymbol{x}_i^-\}}} \left[-\ln \frac{e^{-\left\| \boldsymbol{f}(\boldsymbol{x}^+) - \boldsymbol{f}(\boldsymbol{x}) \right\|_{\hat{\boldsymbol{\Lambda}}}^2}}{e^{-\| f(\boldsymbol{x}^+) - \boldsymbol{f}(\boldsymbol{x}) \|_{\hat{\boldsymbol{\Lambda}}}^2} + \sum_{i=1}^{M} e^{-\left\| \boldsymbol{f}(\boldsymbol{x}_i^-) - \boldsymbol{f}(\boldsymbol{x}) \right\|_{\hat{\boldsymbol{\Lambda}}}^2}} \right],$$

where $\hat{\boldsymbol{\Lambda}}$ is a *trainable* diagonal scaling matrix, aimed to capture $\boldsymbol{\Lambda}$, and the weighted and squared ℓ_2-distance, $-\| \boldsymbol{f}(\boldsymbol{x}^+) - \boldsymbol{f}(\boldsymbol{x}) \|_{\hat{\boldsymbol{\Lambda}}}^2 = -(\boldsymbol{f}(\boldsymbol{x}^+) - \boldsymbol{f}(\boldsymbol{x}))^\top \hat{\boldsymbol{\Lambda}}(\boldsymbol{f}(\boldsymbol{x}^+) - \boldsymbol{f}(\boldsymbol{x}))$, is the *similarity function*. The idea behind AnInfoNCE is that the similarity function should be flexible enough to accommodate latents anisotropically perturbed by the augmentations. We defer the details to Appx. C and summarize our assumptions here:

Assumption 1 (Anisotropic CL on $\mathbb{S}^{d-1}$). *The **DGP** satisfies (cf. Defn. C.1 for details):*

i) $\mathcal{Z} = \mathbb{S}^{d-1}$, $\mathcal{X} \subseteq \mathbb{R}^D$ *are latent and observation spaces.*

ii) *Anchors are uniform on $\mathcal{Z}$, i.e., $p(\boldsymbol{z}) = \text{const}$.*

iii) *For $\boldsymbol{\Lambda} \in \mathbb{R}^{d \times d}$ PD diagonal matrix, the positive conditional $p(\boldsymbol{z}^+|\boldsymbol{z})$ has the form of (2).*

iv) *Negative pairs $\{\boldsymbol{z}_i^-\}$ are uniform on $\mathcal{Z}$.*

v) *A continuous, invertible generator function $\boldsymbol{g}$ maps $\boldsymbol{z}, \boldsymbol{z}^\pm \in \mathcal{Z} \xmapsto{\boldsymbol{g}} \boldsymbol{x}, \boldsymbol{x}^\pm \in \mathcal{X}$.*

*The **model** satisfies (cf. Defn. C.2 for details):*

vi) *The encoder $\boldsymbol{f} : \mathbb{R}^D \to \mathbb{S}^{d-1}$ is continuous.*

vii) $\hat{\boldsymbol{\Lambda}} \in \mathbb{R}^{d \times d}$ *is a PD diagonal matrix.*

viii) $\boldsymbol{f}$ *and $\hat{\boldsymbol{\Lambda}}$ are trained with $\mathcal{L}_{\text{AINCE}}$, i.e. (3).*

Collectively i)-viii) define an anisotropic CL problem.

3.2 Identifiability of Anisotropic CL

When studying identifiability, we aim to recover the ground-truth latent $\boldsymbol{z}$ from observations $\boldsymbol{x} = \boldsymbol{g}(\boldsymbol{z})$ with as little ambiguity as possible. Our main contribution is proving the identifiability of the DGP from Assum. 1, generalizing Zimmermann et al. (2021); von Kügelgen et al. (2021):

Theorem 3.1 (Identifiability of Anisotropic CL). *Under Assum. 1, if a pair $(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}})$ minimizes $\mathcal{L}_{\text{AINCE}}$ (3), then $\boldsymbol{f} \circ \boldsymbol{g}$ is a block-orthogonal transformation, where each block acts on latents with equal weight $\boldsymbol{\Lambda}_{ii}$. In other words, $\boldsymbol{z}$ is identified up to a block-orthogonal transformation.*

Thm. 3.1 means that if the encoder $\boldsymbol{f}$, usually parametrized by a neural network, is expressive enough to minimize AnInfoNCE, then the reconstructed latent $\hat{\boldsymbol{z}} = \boldsymbol{f}(\boldsymbol{x})$ is related to the ground-truth $\boldsymbol{z}$ via a simple orthogonal linear transformation. This simple ambiguity remains inconsequential as solving any downstream task involves at least one linear layer trained on top of the backbone $\boldsymbol{f}$.

Proof Sketch (full proof in Appx. C.1). We rewrite AnInfoNCE into a categorical cross-entropy of predicting the correct index of the positive pair from randomly shuffled data $\boldsymbol{x}^+, \{\boldsymbol{x}_i^-\}$. The minimum loss is attained iff the similarity function matches the log-density-ratio between the positive and negative conditionals (Thm. B.1). This provides us with a functional equation that we solve for $(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}})$. $\qquad\square$

3.3 Extending Identifiability to Hard Negative Mining

Our generalized result in § 3.2 accounts for anisotropically varying latent factors. Here, we build on Thm. 3.1 to theoretically model the practice of hard negative (HN) mining.

HN mining effectively uses a negative conditional distribution to select negative samples more similar to the anchor, instead of sampling from a uniform marginal. Put differently, HN mining makes the classification problem of positive/negative samples *more difficult*, which is reasoned to help avoid latent collapse (Wu et al., 2020; Chuang et al., 2020; Robinson et al., 2021b; Zheng et al., 2021; Wang and Liu, 2021; Robinson et al., 2021a; Khosla et al., 2020; Yuksekgonul et al., 2023b). Our proposed anisotropic loss still identifies the ground-truth latents, when the negative conditional has the same form as (2) (we denote its concentration parameter as $\boldsymbol{\Lambda}^-$, and the one of the positive conditional as $\boldsymbol{\Lambda}^+$). Thus, we further generalize the setting of § 3.2, which is subsumed as $\boldsymbol{\Lambda}^- = \boldsymbol{0}$ (i.e., a uniform marginal). Intuitively, with a negative conditional, the density ratios will be the same as with a uniform marginal and a positive conditional with $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^+ - \boldsymbol{\Lambda}^-$.

Corollary 3.1 (Identifiability with HN mining. Proof in Appx. C.2). *Under Assum. 1, but with a negative conditional in iv) as (2) with a concentration parameter $\boldsymbol{\Lambda}^-$, such that $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^+ - \boldsymbol{\Lambda}^-$ has full rank, the pair $(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}})$ minimizing $\mathcal{L}_{\text{AINCE}}$ identifies the latent factors up to a block-orthogonal transformation.*

We include experimental results for hard negative mining in Appx. D.3. We empirically show that our theory correctly predicts the loss optimum at $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^+ - \boldsymbol{\Lambda}^-$, and that fine-tuning an encoder on a concatenation of hard negative samples and regular negative samples improves the identifiability score.

3.4 Extending Identifiability to Ensemble AnInfoNCE

Recent works started exploring both theoretically (Eastwood et al., 2023; Kirchhof et al., 2023) and empirically (Xiao et al., 2021; Zhang and Ma, 2022) the advantages of ensemble CL losses, where each loss component is calculated with different (potentially partially overlapping) data augmentations. We show that an ensemble version of AnInfoNCE is also identifiable. For more details, see Appx. C.3.

Corollary 3.2 (Identifia. of ensemble AnInfoNCE)**.** *Assume k DGPs, such that each satisfy Assum. 1. Assume that they share $\boldsymbol{g}$, but have different concentration parameters $\{\boldsymbol{\Lambda}^i\}$, for $1 \leq i \leq k$. Assume a shared encoder $\boldsymbol{f}$ and k learnable $\hat{\boldsymbol{\Lambda}}^i$ parameters. Then, the tuple $(\boldsymbol{f}, \{\hat{\boldsymbol{\Lambda}}^i\})$ minimizing the ensemble loss $\sum_i^k \mathcal{L}_{\text{AINCE}}(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}}^i)$ identifies the latents up to a block-orthogonal transformation.*

Cor. 3.2 shows that using an ensemble with proper augmentations could lead to stronger identifiability guarantees (e.g., up to permutation, cf. Rem. C.4). Ideally, we could design such sets of augmentations, but it is not evident that this is possible in practice.

We include experimental results for ensemble AnInfoNCE in Appx. D.3. To verify Cor. 3.2, we define two DGPs that share a uniform anchor distribution and differ in their respective conditional distributions. We sample data from both DGPs, calculate the individual losses, and train the encoder on the sum of both losses. We see improved linear identifiability scores with ensemble AnInfoNCE, compared to training the encoder with InfoNCE or AnInfoNCE.

4 EXPERIMENTS

We now validate our theoretical results in experiments with increasing complexity. This section focuses on our main result and demonstrates the efficacy of AnInfoNCE in various settings. We include additional experimental ablations, results for hard negative mining and ensembling AnInfoNCE and detailed compute requirements for all our experiments in the Appendix.

4.1 Synthetic Experiments

Preliminaries. We follow Zimmermann et al. (2021) and consider a fully-controlled Data Generating Process (DGP), which implements Assum. 1. That is, the ground-truth latent space is a hypersphere ($\mathcal{Z} = \mathbb{S}^{d-1}$), and positive/negative pairs are defined through conditional sampling in the latent space. We model the generative process $\boldsymbol{g}$ as an invertible Multi-Layer Perceptron (MLP) with the same input, intermediate, and output dimensionality d and with Leaky ReLU non-linearities. We set $d = 10$. The encoder $\boldsymbol{f}$ is also parameterized by an MLP with the same input, intermediate, and output dimensionality, and with Leaky ReLU non-linearities. The encoder is trained only with observations, which are obtained by passing $\boldsymbol{z}$ through $\boldsymbol{g}$. As we have access to the ground-truth $\boldsymbol{z}$, we can post-hoc compute the information captured in the inferred latents $\hat{\boldsymbol{z}}$ about the ground-truth $\boldsymbol{z}$. We do this by fitting a linear map $\mathbf{A}$ between $\hat{\boldsymbol{z}}$ and $\boldsymbol{z}$ and compute the average coefficient of determination R^2 Wright (1921) between the relevant features

(e.g. all of them, content-only dimensions, style-only dimensions) of $\mathbf{A}\hat{\boldsymbol{z}}$ and $\boldsymbol{z}$. Unless noted otherwise, we use a uniform marginal and a projected Gaussian as the conditional distribution, modeling Eq. (2) and the assumptions in § 3.1.

Results. We study the effect of anisotropy in the positive conditional by varying the ground-truth $\boldsymbol{\Lambda}$: We keep the value of five latent dimensions fixed at $\Lambda_1 = 5$, while we vary the value of the other five dimensions (i.e., Λ_2), resulting in a ground-truth diagonal $\boldsymbol{\Lambda} = \text{diag}(\Lambda_1, \ldots, \Lambda_1, \Lambda_2, \ldots \Lambda_2)$. While von Kügelgen et al. (2021) coined the terms "content" and "style" to refer to invariant and variant latents, respectively, we here relax this notion and use these terms comparatively: In an anisotropic setting, where $\boldsymbol{\Lambda}$ is composed of different values, higher (lower) values reflect more content-like (style-like) dimensions. We train an encoder on the observed data generated by the process described above and summarize the results in Fig. 2**A**. We observe high R^2 scores over a wide range of $\boldsymbol{\Lambda}$-values for both content and style latents when training with our AnInfoNCE loss (markers in Fig. 2**A** indicate converged training runs). In contrast, regular InfoNCE loss cannot identify the style latents ($R^2 \approx 0$). We also analyze the setting where the conditionals of five latents are distributed according to $\Lambda_2 = 25$ in more detail in Fig. 2**B-D**. We observe that content dimensions are learned before style (Fig. 2**B**). Our loss converges to the global minimum value, which can be computed based on the ground-truth latents (Fig. 2**C**). When learning $\boldsymbol{\Lambda}$, we observe that the learned $\hat{\boldsymbol{\Lambda}}$ values almost match the ground-truth values (dashed black lines in Fig. 2**D**).

4.2 VAE Experiments on MNIST

Preliminaries. We now move closer to actual images by replacing the MLP in our generative model with a ten-dimensional Variational Autoencoder (VAE; Kingma and Welling, 2013) trained on MNIST (Deng, 2012). This setup is beneficial as we have valid images while fully controlling the DGP. Thus, we still sample ground-truth latents, which we then pass through the VAE decoder to generate observations to train on. Both the encoder and the decoder are fully connected three-layer MLPs with Leaky ReLU activations. Generated samples for different concentration parameters $\boldsymbol{\Lambda}$ are shown in Appx. D.5, Fig. 13.

Results. We start with the same anisotropic ground-truth $\boldsymbol{\Lambda}$ as in § 4.1. When training an encoder on MNIST samples with AnInfoNCE, we observe perfect linear identifiability scores (R^2) for all latents, whereas training with the standard InfoNCE loss yields zero R^2 scores for the style latents (Fig. 3**A**). The identifiability scores correlate with KNN classification accuracy,
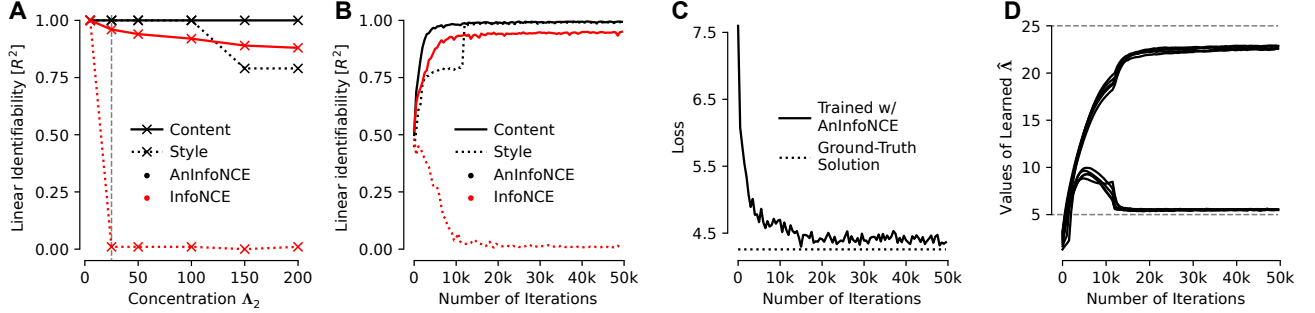
Figure 2: **Behavior of AnInfoNCE on Synthetic Data.** ($\Lambda_1 = 5$) **A:** We maintain high R^2-scores with AnInfoNCE for both content and style dimensions, while the style dimensions are lost when training with the regular InfoNCE loss. For $\Lambda_2 = 25$ (dotted black vertical line), we show: **B:** The evolution of the linear R^2 scores during training; **C:** AnInfoNCE reaches the global minimum, computed based on ground-truth latents; **D:** The evolution of the learned $\hat{\Lambda}$ values.

which is stable across a wide range of $\mathbf{\Lambda}$-values when training with AnInfoNCE, but degrades when training with InfoNCE. As our theoretical claims are only valid for the representations after the final layer of the model, we evaluate those. This means we deviate from previous literature and do not use a projection head.

Next, we model each latent with a distinct concentration parameter $\mathbf{\Lambda}_{ii}$ by linearly sampling values between 5 and a maximum of either 50 or 200. When training the encoder with AnInfoNCE, we recover all latent dimensions: The linear identifiability scores are 99.7% and 99.6%, respectively. However, training with the regular InfoNCE loss leads to substantially lower scores of 77.4% and 76.3%, respectively. Fig. 3**C & D** show how the learned $\hat{\mathbf{\Lambda}}$ values change during training. We observe that the learned $\hat{\mathbf{\Lambda}}$ approaches the ground truth, except for very high $\mathbf{\Lambda}$ values: This is visible in Fig. 3**D** when $\mathbf{\Lambda}_{ii} > 100$). We analyze this behavior further in Appx. D.

4.3 Real-world Experiments

Preliminaries. On CIFAR10 (Krizhevsky, 2009), we train ResNet18 (He et al., 2016) models for 1000 epochs with the code from Hua (2021), while on ImageNet (Russakovsky et al., 2015), we train ResNet50 models for 100 epochs using the code from Facebook Research (2022).

Results. On real-world datasets, we do not have access to true generative factors and the latent conditional distribution. Therefore, we need to create views using regular SimCLR augmentations. Another consequence is that we cannot directly measure how well models recover the ground-truth information. Thus, we use proxy evaluations and calculate the linear readout accuracy on the augmentations used during training to judge how well we recover the latent factors. Chance-level

performance is 50%. Our experiments on CIFAR10 and ImageNet show that our loss with a trainable $\hat{\mathbf{\Lambda}}$ leads to much higher readout accuracy on the used augmentations, i.e., we successfully recover more latent dimensions compared to the regular InfoNCE loss (Tab. 1). While we reduce the information loss, this surprisingly does not lead to better downstream accuracy (Tab. 1, *Classes* column). Removing ℓ_2-normalization results in an encoder with the highest augmentations readout accuracy and simultaneously the lowest classification accuracy. We analyze this discrepancy, the validity of our assumptions for real-world data, and the required steps for closing the remaining gap between CL theory and practice in § 5.

5 ANALYSIS

Our experiments (§ 4) show some scenarios in which recovering more (latent) information results in better downstream performance (e.g., readout accuracy on MNIST). While AnInfoNCE outperforms InfoNCE in these controlled scenarios, it fails in others: On CIFAR10 and ImageNet, we observe a trade-off between the augmentation readout and linear classification readout accuracies. While higher accuracy in augmentation readout indicates a better capture of style latents (i.e., more latent information is recovered), it does not coincide with higher classification accuracy. Scaling up synthetic experiments to ImageNet comes with changes to various training aspects: The encoder's architecture is switched to a ResNet from an invertible MLP, views are sampled using augmentations instead of the ground-truth DGP, and the encoder's output dimensionality is orders of magnitude higher. To control for these differences, we conduct a minimal-change experiment on MNIST, changing only how the views are generated while the encoder and the training procedure remain the same. We demonstrate that using augmentations
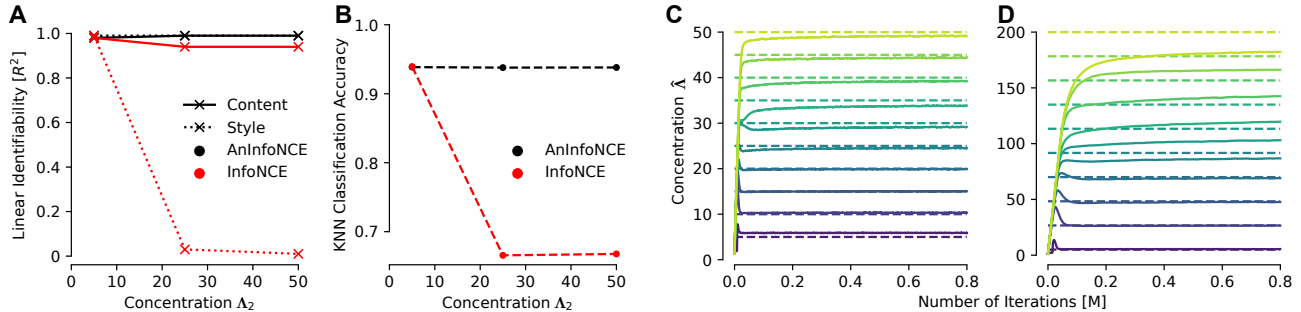
Rusak*, Reizinger*, Juhos*, Bringmann, Zimmermann[†], Brendel[†]

Figure 3: **Behavior of AnInfoNCE on MNIST. A:** Linear identifiability (R^2) scores for an encoder trained on VAE-generated MNIST samples, when varying Λ_2 and keeping $\Lambda_1 = 5$ for the positive conditional. Style dimensions collapse ($R^2 = 0$) for regular InfoNCE. **B:** KNN-accuracy evaluated on the regular MNIST dataset using the encoders trained as described in **A**. KNN accuracy degrades when style dimensions are lost. **C & D:** The evolution of learned $\hat{\Lambda}$ during training. We set the diagonal entries of Λ to ten different values by linearly interpolating between 5 and 50 (**C**) or 5 and 200 (**D**). The ground-truth Λ values are indicated by dashed lines. The learned $\hat{\Lambda}$ are shown in the corresponding colors as solid lines.

Table 1: **Better Augmentations Readout Does Not Imply Better Classification Performance.** Comparison of different training objectives for linear downstream classification (*Classes*) and augmentation readout (*Grayscale* to *Saturate*, *Avg.* denotes the average over those) performance. Learning Λ results in higher readout accuracy of the used augmentations, which can be interpreted as a better recovery of style latents. However, better augmentations readout does not result in better linear classification accuracy. We note that our theory and all synthetic and VAE experiments apply to the post-projector (backb.+proj.) features which are typically not used in practice because using backbone features results in better performance.

	Training Objective	Features	Classes	Grayscale	Brightness	Contrast	Hue	Saturate	Avg.
CIFAR10	InfoNCE	backbone	90.9	56.1	59.3	62.6	55.4	55.1	57.7
	AnInfoNCE	backbone	88.4	80.1	93.4	91.1	65.5	72.3	80.5
	AnInfoNCE, w/o ℓ_2-norm.	backbone	83.2	96.2	98.0	96.6	76.8	88.4	91.2
	InfoNCE	backb.+proj.	89.8	52.0	52.0	52.0	51.8	51.0	51.8
	AnInfoNCE	backb.+proj.	85.4	71.4	71.4	77.1	59.7	65.5	69.0
	AnInfoNCE, w/o ℓ_2-norm.	backb.+proj.	79.1	65.5	65.5	87.6	70.5	83.8	74.6
ImageNet	InfoNCE	backbone	68.2	98.7	71.6	72.5	68.0	61.2	74.4
	AnInfoNCE	backbone	59.0	98.8	81.5	82.3	71.6	62.5	79.3
	InfoNCE	backb.+proj.	57.7	58.6	51.9	51.9	51.5	51.1	53.0
	AnInfoNCE	backb.+proj.	26.7	71.0	55.2	54.1	52.6	52.7	57.1

instead of generating the views using the ground-truth DGP is the main reason for this trade-off.

Minimal-change experiment on MNIST exhibits the same identifiability-accuracy trade-off. To visualize how augmentations influence the positive conditional distribution in latent space, we train a VAE on augmented MNIST images and then project these augmented images onto the learned VAE's latent space. We observe that the latents of positive pairs do not follow a projected Gaussian distribution and can even follow a bimodal one (see Fig. 7 in Appx. D). Our theory does not cover this case; neither do other theories we are aware of, and so this mismatch presents a fruitful direction for future research on closing the gap between theory and practice.

Further, to better analyze the interplay between augmentations and disentanglement, we train a model either using InfoNCE or AnInfoNCE on MNIST when views are generated using SimCLR augmentations (except color augmentations as MNIST is grayscale):

Loss	Class	Lin. id. [R^2]	Crop	Gaussian Blur
InfoNCE	**0.97**	0.34	0.53	0.58
AnInfoNCE	0.94	**0.44**	**0.59**	**0.63**

We observe the same accuracy-identifiability trade-off on MNIST as before on ImageNet and CIFAR10: While the augmentations' prediction accuracy improves, the classification accuracy decreases. Using the VAE trained on MNIST, we can also calculate linear identifiability scores, which are higher for AnInfoNCE (but

Table 2: **We Observe the Accuracy-Identifiability Trade-Off on C-3DIdent.** When using augmentations to generate views (Aug.), InfoNCE achieves better accuracy, while AnInfoNCE has higher R^2 scores. When using the ground truth DGP to generate views (DGP), AnInfoNCE outperforms InfoNCE in terms of both accuracy and identifiability scores. We report averaged results over three random seeds and the standard deviation.

Loss	Class	Data Augmentation Lin. id. $[R^2]$	Nonlin. id. $[R^2]$	Class	DGP Lin. id. $[R^2]$	Nonlin. id. $[R^2]$
InfoNCE	$\mathbf{1.0}_{\pm 0.0}$	$0.31_{\pm 0.0}$	$0.51_{\pm 0.0}$	$1.0_{\pm 0.0}$	$0.2_{\pm 0.0}$	$0.24_{\pm 0.005}$
AnInfoNCE	$0.84_{\pm 0.012}$	$\mathbf{0.34}_{\pm 0.008}$	$\mathbf{0.65}_{\pm 0.0}$	$1.0_{\pm 0.0}$	$\mathbf{0.38}_{\pm 0.017}$	$\mathbf{0.94}_{\pm 0.0}$

still moderately low). Note that training with AnInfoNCE on MNIST when views are generated by passing ground-truth latents through the VAE leads to perfect linear identifiability scores ($R^2 = 1$). In that case, we do not observe a trade-off between linear identifiability scores and downstream accuracy.

We observe the accuracy-identifiability trade-off on the Causal 3DIdent dataset. The Causal 3DIdent dataset (C-3DIdent, von Kügelgen et al., 2021; Zimmermann et al., 2021) is a computer-generated image dataset with seven classes and ten ground-truth latent factors, rendered with Blender (Blender Online Community, 2018). This dataset is more complex than MNIST and thus constitutes an interesting intermediate step between MNIST and visually complex datasets such as ImageNet.

We repeat our analysis from above and train a ResNet18 encoder with InfoNCE and AnInfoNCE on C-3DIdent when views are generated using both the ground truth DGP and data augmentation. In addition to the linear identifiability scores, we also report nonlinear identifiability scores calculated using a Kernel Ridge Regression, following von Kügelgen et al. (2021). In Tab. 2, we observe the same accuracy-identifiability trade-off as on the other datasets when using augmentations: Training with AnInfoNCE leads to better identifiability scores as predicted by our theory. However, training with InfoNCE results in better downstream accuracy. When the ground-truth generative model (DGP) is used for generating the views, AnInfoNCE outperforms InfoNCE in both identifiability and accuracy. Training details can be found in Appx. D.6.

The accuracy-identifiability trade-off can be mitigated when using only one augmentation. To better understand the accuracy-identifiability trade-off, we train models with InfoNCE and AnInfoNCE on Causal 3DIdent using a single augmentation to generate the views. We generate views using small crops (scale parameter between 0.08 and 1.0), large crops (scale parameter between 0.8 and 1.0), by applying color jitter or rotations. We further repeat these experiments using the DGP to generate views by modifying

the corresponding latents. We report the results in Table 3.

We observe that modeling singular data augmentations, e.g., only applying crops or color jitter, works very well with AnInfoNCE, and we see no trade-off between classification accuracy and identifiability. For all augmentations we tested, AnInfoNCE is always better at recovering them than InfoNCE. However, the classification accuracy is generally not very high when using only one augmentation type, except for color jitter where the classification accuracy is 98% with AnInfoNCE. This observation does not hold on real-world data and demonstrates the limitations of the Causal 3D-Ident dataset: When using only color jitter for generating positive samples and training the encoder on CIFAR10, the downstream accuracy drops from 90.9% to 48.6%. While combining multiple augmentations results in high classification accuracy, it also leads to the accuracy-identifiability trade-off as shown in Table 2. We hypothesize that modeling two augmentations is more challenging and requires extending our simple diagonal matrix to a more complex function. This likely would make deriving theoretical guarantees more challenging. When using transformation in the latent codes, AnInfoNCE is always superior to InfoNCE, irrespective of which transformation has been used.

We conclude that AnInfoNCE performs better than InfoNCE on all singular data augmentations we tested and is not sensitive to any of them. However, it struggles when *multiple* augmentations are combined, which is necessary for good downstream accuracy on real-world datasets.

Further ablation studies. We analyze the learning dynamics of AnInfoNCE and conduct a thorough hyperparameter study in Appx. D. We observe that the performance of AnInfoNCE is strongly influenced by sample efficiency, i.e., the batch size and the latent dimensionality. Further, we observe that larger concentration parameters also require larger batch sizes. We link this observation to the role of the *augmentation overlap* (Wang et al., 2022): Higher $\mathbf{\Lambda}$ means a more concentrated conditional and requires larger batch sizes for sufficient augmentation overlap. Investigating the

Table 3: **There is no accuracy-identifiability trade-off, if a single augmentation is used for training.** On Causal 3DIdent, classification and identifiability results improve simultaneously when switching from InfoNCE to AnInfoNCE, if using a single data augmentation to generate views (**DA**). The same is observed when perturbing single components in the latent code (**DGP**). This relative advantage of AnInfoNCE compared to InfoNCE was unlocked by giving up sophisticated augmentation pipelines and, hence, absolute performance.

| Loss | DA: Small Crops | | | DA: Large Crops | | | DGP: Change Position | | |
	Class	Lin. id. $[R^2]$	Nonlin. id. $[R^2]$	Class	Lin. id. $[R^2]$	Nonlin. id. $[R^2]$	Class	Lin. id. $[R^2]$	Nonlin. id. $[R^2]$
InfoNCE	0.14	0.02	0.10	0.26	0.07	0.23	**1.0**	0.25	0.28
AnInfoNCE	**0.37**	**0.17**	**0.44**	**0.47**	**0.18**	**0.41**	**1.0**	**0.31**	**0.91**

| Loss | DA: Rotations | | | DGP: Rotations | | |
	Class	Lin. id. $[R^2]$	Nonlin. id. $[R^2]$	Class	Lin. id. $[R^2]$	Nonlin. id. $[R^2]$
InfoNCE	0.29	0.08	0.30	**1.0**	0.33	0.45
AnInfoNCE	**0.48**	**0.15**	**0.31**	**1.0**	**0.57**	**0.86**

| Loss | DA: Apply ColorJitter | | | DGP: Change Hues | | |
	Class	Lin. id. $[R^2]$	Nonlin. id. $[R^2]$	Class	Lin. id. $[R^2]$	Nonlin. id. $[R^2]$
InfoNCE	0.90	0.22	0.41	**1.0**	0.31	0.36
AnInfoNCE	**0.98**	**0.56**	**0.85**	**1.0**	**0.44**	**0.85**

evolution of the identifiability scores over time, we observe a step-like behavior in how the latents are learned, corroborating the results by Simon et al. (2023). We discuss connections between AnInfoNCE and TriCL (Zhang et al., 2023) and show how the losses compare experimentally in Appx. D.4.

Discussion. Given our analysis above, we identify the use of augmentations in real-world datasets for generating views as the main cause for the accuracy-identifiability trade-off: Assuming an underlying LVM, these augmentations correspond to changes in latent factors, modeled by the positive conditional distribution. This points to a potential cause for unexpectedly low performance: The conditional distribution implied by augmentations may not correspond to the one assumed by the loss. First, the form of the conditional implicitly defined by the augmentations may not be Gaussian or even uni-modal. Second, the corresponding concentration parameters may be high, leading to a flat loss landscape and optimization issues in practice.

6 CONCLUSION

Our work advances CL theory by generalizing InfoNCE. The proposed framework, AnInfoNCE-based Contrastive Learning (CL), is identifiable and relies on more realistic assumptions on the data, reducing the gap between CL theory and practice: It can account for anisotropic latent conditionals that more likely reflect how augmentations affect the latent factors in practice. In addition, we showed extensions of our results towards other empirical techniques such as hard negative mining and loss ensembling. We demonstrated that our

framework accounts for certain known limitations of CL, such as collapsing information, by better capturing style information. At the same time, we showed that there exists a trade-off between recovered style information and downstream classification. Further, even our relaxed assumptions can still be too strong in real-world problems: For example, the positive conditional distribution can be non-Gaussian.

Exploring the gap between theory and practice led to a new loss formulation. We hope that further investigations into overcoming the remaining gap between theory and practice will result in refinements and insights that can improve practical self-supervised learning techniques and our understanding of them. We discussed some directions in § 5 and Appx. D, suggesting that more theoretical work on the identifiability of large-scale DGPs is a fruitful endeavor for future work.

Author Contributions

WB conceived the project idea. ER led the project. ER, RSZ, and WB designed the experiments and ER executed them. ER conducted the analysis with support from RSZ and PR. ER and RSZ jointly created all figures. PR conceived how to incorporate anisotropy in SimCLR. This idea led to WB and RSZ developing the main theorem. AJ developed corollaries for hard negative mining and loss ensembling with help from PR and feedback from RSZ. RSZ and WB jointly supervised the project. AJ was responsible for presenting the theoretical results in a standardized way. ER, PR, AJ, RSZ, and WB contributed to writing the manuscript.

Acknowledgements

References

Assran, M., Balestriero, R., Duval, Q., Bordes, F., Misra, I., Bojanowski, P., Vincent, P., Rabbat, M. G., and Ballas, N. (2023). The hidden uniform cluster prior in self-supervised learning. In *ICLR*. OpenReview.net. Cited on page 3.

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv preprint*. Cited on page 30.

Balestriero, R. and LeCun, Y. (2022). Contrastive and non-contrastive self-supervised learning recover global and local spectral embedding methods. In *NeurIPS*. Cited on page 3.

Bendidi, I., Bardes, A., Cohen, E., Lamiable, A., Bollot, G., and Genovesio, A. (2023). No Free Lunch in Self Supervised Representation Learning. *arXiv preprint*. Cited on pages 3 and 30.

Blender Online Community (2018). *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam. Cited on page 8.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. (2020). Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In *NeurIPS*. Cited on page 1.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. E. (2020a). A Simple Framework for Contrastive

Learning of Visual Representations. In *ICML*. PMLR. Cited on pages 1, 3, and 24.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., and Hinton, G. E. (2020b). Big Self-Supervised Models are Strong Semi-Supervised Learners. *Advances in neural information processing systems*. Cited on page 1.

Chen, T., Luo, C., and Li, L. (2021). Intriguing properties of contrastive losses. In *NeurIPS*. Cited on pages 3 and 16.

Chen, X., Fan, H., Girshick, R., and He, K. (2020c). Improved Baselines with Momentum Contrastive Learning. *arXiv preprint*. Cited on page 1.

Chen, X. and He, K. (2021). Exploring simple siamese representation learning. In *CVPR*. Computer Vision Foundation / IEEE. Cited on pages 1 and 30.

Chuang, C., Hjelm, R. D., Wang, X., Vineet, V., Joshi, N., Torralba, A., Jegelka, S., and Song, Y. (2022). Robust contrastive learning against noisy views. In *CVPR*. IEEE. Cited on page 31.

Chuang, C., Robinson, J., Lin, Y., Torralba, A., and Jegelka, S. (2020). Debiased contrastive learning. In *NeurIPS*. Cited on pages 4 and 31.

Cosentino, R., Sengupta, A., Avestimehr, S., Soltanolkotabi, M., Ortega, A., Willke, T., and Tepper, M. (2022). Toward a Geometrical Understanding of Self-supervised Contrastive Learning. *arXiv preprint*. Cited on page 3.

Cui, J., Huang, W., Wang, Y., and Wang, Y. (2022). AggNCE: Asymptotically Identifiable Contrastive Learning. In *NeurIPS 2022 Workshop: Self-Supervised Learning - Theory and Practice*. Cited on pages 3 and 29.

Dangovski, R., Jing, L., Loh, C., Han, S., Srivastava, A., Cheung, B., Agrawal, P., and Soljačić, M. (2021). Equivariant Contrastive Learning. *arXiv preprint*. Cited on page 3.

Daunhawer, I., Bizeul, A., Palumbo, E., Marx, A., and Vogt, J. E. (2023). Identifiability results for multimodal contrastive learning. In *ICLR*. OpenReview.net. Cited on page 29.

Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, (6). Cited on page 5.

Dubois, Y., Ermon, S., Hashimoto, T. B., and Liang, P. (2022). Improving self-supervised learning by characterizing idealized representations. In *NeurIPS*. Cited on pages 3, 16, and 29.

Eastwood, C., von Kügelgen, J., Ericsson, L., Bouchacourt, D., Vincent, P., Schölkopf, B., and Ibrahim, M. (2023). Self-Supervised Disentanglement by Leveraging Structure in Data Augmentations. *arXiv preprint.* Cited on pages 3, 4, 23, and 29.

Ericsson, L., Gouk, H., and Hospedales, T. M. (2021). Why Do Self-Supervised Models Transfer? Investigating the Impact of Invariance on Downstream Tasks. *arXiv preprint.* Cited on page 30.

Ermolov, A., Siarohin, A., Sangineto, E., and Sebe, N. (2021). Whitening for self-supervised representation learning. In *ICML*. PMLR. Cited on page 31.

Facebook Research (2022). Vicreg: Variance-invariance-covariance regularization for self-supervised learning. https://github.com/facebookresearch/vicreg. Cited on page 6.

FOLDING.LAR.SYSTEMS (Accessed on May 14th, 2024). Gpu folding overview. https://folding.lar.systems/gpu_ppd/brands/nvidia/folding_profile/ga100_a100_pcie_40gb. Cited on page 29.

Garrido, Q., Chen, Y., Bardes, A., Najman, L., and LeCun, Y. (2023). On the duality between contrastive and non-contrastive self-supervised learning. In *ICLR*. OpenReview.net. Cited on page 3.

Gresele, L., Rubenstein, P. K., Mehrjou, A., Locatello, F., and Schölkopf, B. (2019). The incomplete rosetta stone problem: Identifiability results for multi-view nonlinear ICA. In *UAI*. AUAI Press. Cited on page 23.

Grill, J., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. Á., Guo, Z., Azar, M. G., Piot, B., Kavukcuoglu, K., Munos, R., and Valko, M. (2020). Bootstrap your own latent - A new approach to self-supervised learning. In *NeurIPS*. Cited on page 1.

Gutmann, M. and Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*. JMLR.org. Cited on pages 2 and 17.

Halvagal, M. S., Laborieux, A., and Zenke, F. (2022). An eigenspace view reveals how predictor networks and stop-grads provide implicit variance regularization. Cited on page 31.

HaoChen, J. Z., Wei, C., Gaidon, A., and Ma, T. (2021). Provable guarantees for self-supervised deep learning with spectral contrastive loss. In *NeurIPS*. Cited on page 3.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *CVPR*. IEEE Computer Society. Cited on page 6.

Hua, P. (2021). Simsiam: A pytorch implementation. https://github.com/PatrickHua/SimSiam. Cited on page 6.

Hyvärinen, A. and Morioka, H. (2016). Unsupervised feature extraction by time-contrastive learning and nonlinear ICA. In *NeurIPS*. Cited on page 23.

Jing, L., Vincent, P., LeCun, Y., and Tian, Y. (2022). Understanding dimensional collapse in contrastive self-supervised learning. In *ICLR*. OpenReview.net. Cited on pages 1, 3, and 16.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. (2020). Supervised contrastive learning. In *NeurIPS*. Cited on page 4.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114.* Cited on page 5.

Kirchhof, M., Kasneci, E., and Oh, S. J. (2023). Probabilistic contrastive learning recovers the correct aleatoric uncertainty of ambiguous inputs. In *ICML*. PMLR. Cited on page 4.

Kirchhof, M., Roth, K., Akata, Z., and Kasneci, E. (2022). A non-isotropic probabilistic take on proxy-based deep metric learning. In *ECCV (26)*. Springer. Cited on page 3.

Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. *Master's thesis, University of Toronto.* Cited on page 6.

Lachapelle, S., Deleu, T., Mahajan, D., Mitliagkas, I., Bengio, Y., Lacoste-Julien, S., and Bertrand, Q. (2022). Synergies Between Disentanglement and Sparsity: a Multi-Task Learning Perspective. *arXiv preprint.* Cited on page 29.

Lee, H., Lee, K., Lee, K., Lee, H., and Shin, J. (2021a). Improving transferability of representations via augmentation-aware self-supervision. In *NeurIPS*. Cited on page 3.

Lee, J. D., Lei, Q., Saunshi, N., and Zhuo, J. (2021b). Predicting what you already know helps: Provable self-supervised learning. In *NeurIPS*. Cited on page 29.

Liu, B., Rosenfeld, E., Ravikumar, P. K., and Risteski, A. (2022). Analyzing and improving the optimization landscape of noise-contrastive estimation. In *ICLR*. OpenReview.net. Cited on page 26.

Liu, B. and Wang, B. (2023). Bayesian Self-Supervised Contrastive Learning. *arXiv preprint*. Cited on page 31.

Lyu, Q., Fu, X., Wang, W., and Lu, S. (2021). Latent Correlation-Based Multiview Learning and Self-Supervision: A Unifying Perspective. *arXiv preprint*. Cited on pages 3 and 29.

Ma, Z. and Collins, M. (2018). Noise Contrastive Estimation and Negative Sampling for Conditional Models: Consistency and Statistical Efficiency. *arXiv preprint arXiv:1809.01812*. Cited on page 17.

Marcel, S. and Rodriguez, Y. (2010). Torchvision the machine-vision package of torch. In *ACM International Conference on Multimedia*. Cited on page 29.

Matthes, S., Han, Z., and Shen, H. (2023). Towards a Unified Framework of Contrastive Learning for Disentangled Representations. In *Thirty-seventh Conference on Neural Information Processing Systems*. Cited on page 17.

Merkel, D. (2014). Docker: Lightweight linux containers for consistent development and deployment. *Linux J.*, 2014(239). Cited on page 29.

Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation Learning with Contrastive Predictive Coding. *arXiv preprint*. Cited on pages 1 and 2.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*. Cited on page 29.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR. Cited on page 2.

Rahmanzadehgervi, P., Bolton, L., Taesiri, M. R., and Nguyen, A. T. (2024). Vision language models are blind. *arXiv preprint arXiv:2407.06581*. Cited on page 2.

Rajendran, G., Reizinger, P., Brendel, W., and Ravikumar, P. (2023). An interventional perspective on identifiability in gaussian lti systems with independent component analysis. *arXiv preprint*. Cited on page 23.

Rensmart (Accessed on May 14th, 2024). Carbon emissions calculator. `https://www.rensmart.com/Calculators/KWH-to-CO2`. Cited on page 29.

Robinson, J., Sun, L., Yu, K., Batmanghelich, K., Jegelka, S., and Sra, S. (2021a). Can contrastive learning avoid shortcut solutions? In *NeurIPS*. Cited on pages 4 and 31.

Robinson, J. D., Chuang, C., Sra, S., and Jegelka, S. (2021b). Contrastive learning with hard negative samples. In *ICLR*. OpenReview.net. Cited on pages 4 and 31.

Roeder, G., Metz, L., and Kingma, D. (2021). On linear identifiability of learned representations. In *ICML*. PMLR. Cited on page 27.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, (3). Cited on page 6.

Saunshi, N., Ash, J., Goel, S., Misra, D., Zhang, C., Arora, S., Kakade, S., and Krishnamurthy, A. (2022). Understanding contrastive learning requires incorporating inductive biases. In *ICML*. PMLR. Cited on page 26.

Saunshi, N., Plevrakis, O., Arora, S., Khodak, M., and Khandeparkar, H. (2019). A theoretical analysis of contrastive unsupervised representation learning. In *ICML*. PMLR. Cited on pages 29 and 31.

Saxe, A. M., McClelland, J. L., and Ganguli, S. (2014). Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *ICLR*. Cited on page 26.

Simon, J. B., Knutins, M., Ziyin, L., Geisz, D., Fetterman, A. J., and Albrecht, J. (2023). On the stepwise nature of self-supervised learning. In *ICML*. Cited on pages 9 and 26.

Sohn, K. (2016). Improved Deep Metric Learning with Multi-class N-pair Loss Objective. *Advances in neural information processing systems*, 29. Cited on pages 2 and 17.

Sylabs Inc (2018). Singularity 2.5.2 - linux application and environment containers for science. `https://doi.org/10.5281/zenodo.1308868`. Cited on page 29.

Tamkin, A. and He, X. (2022). Feature Dropout: Revisiting the Role of Augmentations in Contrastive Learning. In *NeurIPS 2022 Workshop: Self-Supervised Learning - Theory and Practice*. Cited on page 16.

Techpowerup (Accessed on May 14th, 2024). Gpu specs database. `https://www.techpowerup.com/gpu-specs`. Cited on page 29.

Tian, Y. (2023). Understanding the role of nonlinearity in training dynamics of contrastive learning. In *ICLR*. OpenReview.net. Cited on pages 16 and 30.

Travel Navigator (Accessed on May 14th, 2024). https://travelnav.com/emissions-from-new-york-ny-to-london-united-kingdom. Cited on page 29.

Trusca, M. M., Nuyts, W., Thomm, J., Honig, R., Hofmann, T., Tuytelaars, T., and Moens, M.-F. (2024). Object-attribute binding in text-to-image generation: Evaluation and control. *arXiv preprint arXiv:2404.13766*. Cited on page 2.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Jarrod Millman, K., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C., Polat, İ., Feng, Y., Moore, E. W., Vand erPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and Contributors, S. . . (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272. Cited on page 29.

von Kügelgen, J., Rubenstein, P. K., Schölkopf, B., and Weller, A. (2019). Optimal experimental design via bayesian optimization: active causal structure learning for gaussian process networks. *arXiv preprint*. Cited on page 3.

von Kügelgen, J., Sharma, Y., Gresele, L., Brendel, W., Schölkopf, B., Besserve, M., and Locatello, F. (2021). Self-supervised learning with data augmentations provably isolates content from style. In *NeurIPS*. Cited on pages 1, 2, 3, 4, 5, 8, 16, 27, 28, and 29.

Wagner, D., Ferreira, F., Stoll, D., Schirrmeister, R. T., Müller, S., and Hutter, F. (2022). On the Importance of Hyperparameters and Data Augmentation for Self-Supervised Learning. *arXiv preprint*. Cited on pages 3 and 30.

Wang, F. and Liu, H. (2021). Understanding the behaviour of contrastive loss. In *CVPR*. Computer Vision Foundation / IEEE. Cited on page 4.

Wang, Y., Zhang, Q., Wang, Y., Yang, J., and Lin, Z. (2022). Chaos is a ladder: A new theoretical understanding of contrastive learning via augmentation overlap. In *ICLR*. OpenReview.net. Cited on pages 3, 8, and 24.

Wen, Z. and Li, Y. (2021). Toward understanding the feature learning process of self-supervised contrastive learning. In *ICML*. PMLR. Cited on pages 3 and 16.

Wen, Z. and Li, Y. (2022). The mechanism of prediction head in non-contrastive self-supervised learning. In *NeurIPS*. Cited on page 30.

Wright, S. (1921). Correlation and causation. *Journal of Agricultural Research*, (7). Cited on page 5.

Wu, M., Zhuang, C., Mosse, M., Yamins, D., and Goodman, N. (2020). On Mutual Information in Contrastive Learning for Visual Representations. *arXiv preprint*. Cited on pages 3, 4, 30, and 31.

Xiao, T., Wang, X., Efros, A. A., and Darrell, T. (2021). What should not be contrastive in contrastive learning. In *ICLR*. OpenReview.net. Cited on pages 3, 4, 23, 29, and 30.

Yuksekgonul, M., Bianchi, F., Kalluri, P., Jurafsky, D., and Zou, J. (2023a). When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations*. Cited on page 2.

Yuksekgonul, M., Bianchi, F., Kalluri, P., Jurafsky, D., and Zou, J. (2023b). When and why vision-language models behave like bags-of-words, and what to do about it? In *International Conference on Learning Representations*. Cited on page 4.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. (2021). Barlow Twins: Self-Supervised Learning via Redundancy Reduction. In Meila, M. and Zhang, T., editors, *ICML*. PMLR. Cited on page 1.

Zhai, R., Liu, B., Risteski, A., Kolter, Z., and Ravikumar, P. (2023). Understanding Augmentation-based Self-Supervised Representation Learning via RKHS Approximation. *arXiv preprint*. Cited on pages 3 and 16.

Zhang, J. and Ma, K. (2022). Rethinking the augmentation module in contrastive learning: Learning hierarchical augmentation invariance with expanded views. In *CVPR*. Cited on pages 3, 4, 23, and 30.

Zhang, Q., Wang, Y., and Wang, Y. (2023). Identifiable contrastive learning with automatic feature importance discovery. In *NeurIPS*. Cited on pages 9 and 27.

Zheng, H., Chen, X., Yao, J., Yang, H., Li, C., Zhang, Y., Zhang, H., Tsang, I., Zhou, J., and Zhou, M. (2021). Contrastive Attraction and Contrastive Repulsion for Representation Learning. *arXiv preprint*. Cited on page 4.

Zimmermann, R. S., Sharma, Y., Schneider, S., Bethge, M., and Brendel, W. (2021). Contrastive Learning Inverts the Data Generating Process. In *ICML*. PMLR. Cited on pages 1, 2, 3, 4, 5, 8, 16, 27, and 29.

Ziyin, L., Lubana, E. S., Ueda, M., and Tanaka, H. (2022). Loss Landscape of Self-Supervised Learning. In *NeurIPS 2022 Workshop: Self-Supervised Learning - Theory and Practice*. Cited on page 16.

Rusak*, Reizinger*, Juhos*, Bringmann, Zimmermann†, Brendel†

Checklist

i) For all models and algorithms presented, check if you include:

   i) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes] Our theoretical assumptions are summarized in Assum. 1, the experimental settings in § 4.

   ii) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes] We provide a detailed analysis of the effect of batch size and computational requirements in Appxs. D and E.

   iii) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes] We include anonymized code in the supplement. In addition, we described our software stack in Appx. F.

ii) For any theoretical claim, check if you include:

   i) Statements of the full set of assumptions of all theoretical results. [Yes] Confer Assum. 1 and Appx. C.

   ii) Complete proofs of all theoretical results. [Yes] Confer Appx. C

   iii) Clear explanations of any assumptions. [Yes] Confer § 3.

iii) For all figures and tables that present empirical results, check if you include:

   i) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes] We describe our experimental details in § 4 and the software stack in Appx. F.

   ii) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes] We describe our experimental details in § 4

   iii) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes] Confer § 4

   iv) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes] We detail compute requirements in Appx. E.

iv) If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

   i) Citations of the creator If your work uses existing assets. [Yes] Confer § 4 and Appx. F.

   ii) The license information of the assets, if applicable. [Not Applicabble]

   iii) New assets either in the supplemental material or as a URL, if applicable. [No] We will release the code upon acceptance.

   iv) Information about consent from data providers/curators. [Not Applicable]

   v) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

v) If you used crowdsourcing or conducted research with human subjects, check if you include:

   i) The full text of instructions given to participants and screenshots. [Not Applicable]

   ii) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

   iii) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A Partitioning the Latent Space

Zimmermann et al. (2021) assumes that the latent variables are interchangeable for both their marginals and conditionals are the same. This gives rise to specific symmetries in the latent space $\mathcal{Z}$, which might become restrictive in real-world scenarios. von Kügelgen et al. (2021) introduces a two-fold partitioning of $\mathcal{Z}$, where the marginal is assumed to be the same for all components z_i of the latent vector $\boldsymbol{z}$. Namely, they assume that some latent factors (termed content variables $\boldsymbol{z}_c$) have a delta distribution as conditional, i.e., those variables do not change across the positive pairs. This model has been deemed useful in understanding why dimensionality collapse happens. However, such an assumption might be too simplistic in specific scenarios (at least for some of the latents), which depends on the augmentation strategy Dubois et al. (2022); Ziyin et al. (2022); Tian (2023); Chen et al. (2021); Tamkin and He (2022); Wen and Li (2021); Zhai et al. (2023). Thus, we can think of Jing et al. (2022) as a further generalization, though the authors do not make the connection explicit. They consider augmentations, but their reasoning concerns the observation space $\mathcal{X}$. They conclude—similar to Dubois et al. (2022); Tamkin and He (2022)—that the stronger the augmentation, the more prevalent the collapse. Both to generalize and to formalize these insights, we introduce the notion of partitioning of $\mathcal{Z}$:

Definition A.1 (Partitions of $\mathcal{Z}$). Given a latent space $\mathcal{Z}$ and a latent vector $\boldsymbol{z} \in \mathcal{Z}$, we say that subsets of latent components $z_i^k, \ldots, z_{i+n-1}^k$ belong to partition $\mathcal{Z}^k$ of dimension n, when in the ground truth DGP their conditional variances $\sigma_{\tilde{z}_i | \boldsymbol{z}}^2$ are the same, i.e.:

$$\forall j \neq l \in \{i, \ldots, i+n-1\} : \sigma_{z_j^+ | \boldsymbol{z}}^2 = \sigma_{z_l^+ | \boldsymbol{z}}^2. \tag{4}$$

When all conditional variances are equal, then we get back the setting of Zimmermann et al. (2021); when the number of partitions is two, with one of the conditional variances being zero, then we are in the regime of von Kügelgen et al. (2021). We formalize this scenario as follows:

Definition A.2 (Content–style partitioning of $\mathcal{Z}$). Following von Kügelgen et al. (2021), the latent space $\mathcal{Z}$ is said to be partitioned into content ($\boldsymbol{z}_c$, invariant) and style ($\boldsymbol{z}_s$, changing) subspaces if and only if

   i) $\mathcal{Z} = \mathcal{Z}_c \times \mathcal{Z}_s : \mathcal{Z}_c \cap \mathcal{Z}_s = \emptyset; \quad \dim \mathcal{Z}_c + \dim \mathcal{Z}_s = \dim \mathcal{Z};$
   ii) the conditional distribution of $\boldsymbol{z}_c$ latents is a δ-distribution (von Kügelgen et al., 2021, Assum. 3.1);
   iii) the conditional distribution of $\boldsymbol{z}_s$ latents is non-degenerate (von Kügelgen et al., 2021, Assum. 3.2);

yielding the following factorization of the latent conditional $p\left(\boldsymbol{z}^+ | \boldsymbol{z}\right) = \delta\left(\boldsymbol{z}^{+,c} - \boldsymbol{z}_c\right) p(\boldsymbol{z}^{+,s} | \boldsymbol{z}_s)$.

That is, our DGP does not cover the exact setting of von Kügelgen et al. (2021), as we cannot have zero variance. If having zero variance (or equivalently, infinite concentration in $\boldsymbol{\Lambda}$), then we might say that our results incorporate the content-style setting.

B Contrastive Learning: Bayes Optimum

For the sake of completeness, this section presents contrastive learning in a more general form, alongside a theorem regarding its unconditional (Bayes-) optimum. This theorem is referenced several times across the identifiability proofs in Appx. C. Although the content of this section is considered to be known by the research community, we still decided to include it to make the presentation of the theory more self-contained. Appx. C presents the novelties introduced by this paper.

Remark B.1. In the following, we will refer to vectors as $\boldsymbol{y}$ to provide a unified treatment both for $\boldsymbol{y} = \boldsymbol{z}$ and $\boldsymbol{y} = \boldsymbol{x}$.

Definition B.1 (General Contrastive Learning (CL) Problem). Let $\mathcal{Y}$ be a data domain. Let $\boldsymbol{Y} := (\boldsymbol{y}, \boldsymbol{y}^+, \boldsymbol{y}_1^-, \ldots, \boldsymbol{y}_M^-) \sim p_{\boldsymbol{Y}}$ be a random vector with all elements coming from $\mathcal{Y}$. We refer to $\boldsymbol{y}$ as the *anchor point*, to $\boldsymbol{y}^+$ as its *positive pair* and to $\boldsymbol{y}_i^-$ as one of its *negative pairs*. Let us assume that

   i) the pairs $\boldsymbol{y}^+, \boldsymbol{y}_1^-, \ldots, \boldsymbol{y}_M^-$ are conditionally completely independent when conditioned on $\boldsymbol{y}$ and
   ii) the conditional distribution of negative pairs w.r.t. the anchor coincide, i.e., for any i, j we have $p_{\boldsymbol{y}_i^- | \boldsymbol{y}} = p_{\boldsymbol{y}_j^- | \boldsymbol{y}}$.

Let $\mathcal{U}$ be a class of bivariate functions $u : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, hereinafter called *similarity functions* and let us optimize the following objective amongst all functions $u \in \mathcal{U}$:

$$\min_{u \in \mathcal{U}} \quad \mathcal{L}_g(u) := \mathbb{E}\left[-\ln \frac{e^{u(\boldsymbol{y}, \boldsymbol{y}^+)}}{e^{u(\boldsymbol{y}, \boldsymbol{y}^+)} + \sum_{j=1}^M e^{u(\boldsymbol{y}, \boldsymbol{y}_j^-)}} \right]. \tag{5}$$

The tuple $(\mathcal{Y}, p_{\boldsymbol{Y}}, \mathcal{U}, \mathcal{L}_g)$ is called a *general contrastive learning (CL) problem*.

Remark B.2. Denoting $p_{\text{pos}} := p_{\boldsymbol{y}^+|\boldsymbol{y}}$ and $p_{\text{neg}} := p_{\boldsymbol{y}_j^-|\boldsymbol{y}}$, the latter being independent of the exact choice of j per assumption *ii*), the joint data distribution is the following:

$$p_{\boldsymbol{Y}}(\boldsymbol{y}, \boldsymbol{y}^+, \boldsymbol{y}_1^-, \ldots, \boldsymbol{y}_M^-) = p_{\boldsymbol{y}}(\boldsymbol{y}) \, p_{\text{pos}}(\boldsymbol{y}^+|\boldsymbol{y}) \prod_{j=1}^{M} p_{\text{neg}}(\boldsymbol{y}_j^-|\boldsymbol{y}). \tag{6}$$

Theorem B.1 (Bayes-Optima of CL). *Let $(\mathcal{Y}, p_{\boldsymbol{Y}}, \mathcal{U}, \mathcal{L}_g)$ be a general CL problem (Defn. B.1) with $\boldsymbol{Y} := (\boldsymbol{y}, \boldsymbol{y}^+, \boldsymbol{y}_1^-, \ldots, \boldsymbol{y}_M^-) \sim p_{\boldsymbol{Y}}$ a continuous random vector, such that:*

   i) $\boldsymbol{y}$ is supported on $\mathcal{Y}$,
   ii) for any $\boldsymbol{y} \in \mathcal{Y}$, both conditional distributions $p_{\text{neg}}(\cdot|\boldsymbol{y}), p_{\text{pos}}(\cdot|\boldsymbol{y}) := p_{\boldsymbol{y}^+|\boldsymbol{y}}(\cdot|\boldsymbol{y})$ are supported on $\mathcal{Y}$.

Then an arbitrary measurable function $u : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is a global minimum of $\mathcal{L}_g$ (amongst all measurable u's) if and only if there exists a measurable function $c : \mathcal{Y} \to \mathbb{R}$ such that the following holds almost everywhere (w.r.t. any continuous measure of $\mathcal{Y}$):

$$u(\boldsymbol{y}, \tilde{\boldsymbol{y}}) = c(\boldsymbol{y}) + \ln \frac{p_{\text{pos}}(\tilde{\boldsymbol{y}}|\boldsymbol{y})}{p_{\text{neg}}(\tilde{\boldsymbol{y}}|\boldsymbol{y})}. \tag{7}$$

This statement has already been proven by Matthes et al. (2023), however we provide an alternative argument. The key ideas have also emerged in Sohn (2016); Gutmann and Hyvärinen (2010); Ma and Collins (2018).

To prove the theorem, we rewrite the optimization objective as the classification objective of discriminating the positive pair from all the negative pairs, given an anchor point. More specifically, all the pairs $\boldsymbol{y}^+, \boldsymbol{y}_1^-, \ldots, \boldsymbol{y}_M^-$ are randomly shuffled, and, then, the task is to estimate the index K of the positive pair from the new sequence, $\tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M$, given the unchanged anchor $\boldsymbol{y}$. The optimization objective is a categorical cross-entropy (CE) with a predictor of the form (being the output of a softmax function, the predictor's output is normalized to a finite probability distribution):

$$\left[\frac{e^{u(\boldsymbol{y}, \tilde{\boldsymbol{y}}_i)}}{\sum_j e^{u(\boldsymbol{y}, \tilde{\boldsymbol{y}}_j)}} \,\middle|\, i \in \{0, 1, \ldots, M\} \right]. \tag{8}$$

We formalize the instance discrimination task as follows:

Definition B.2 (Instance Discrimination Data). The *instance discrimination data* $(\boldsymbol{y}, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M; K)$ corresponding to a general CL problem $(\mathcal{Y}, p_{\boldsymbol{Y}}, \mathcal{U}, \mathcal{L}_g)$ is defined by the following process:

   Step 1) $\boldsymbol{y} \sim p_{\boldsymbol{y}}$
   Step 2) $K \sim Uni(\{0, 1, \ldots, M\})$
   Step 3) $\tilde{\boldsymbol{y}}_K \sim p_{\text{pos}}(\cdot|\boldsymbol{y})$
   Step 4) $\forall j \in \{0, 1, \ldots, K-1, K+1, \ldots, M\} : \tilde{\boldsymbol{y}}_j \sim p_{\text{neg}}(\cdot|\boldsymbol{y})$.

Remark B.3. The joint data distribution of the instance discrimination data is:

$$p(\boldsymbol{y}, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M, K = i) = p_{\boldsymbol{y}}(\boldsymbol{y}) \, \mathbb{P}(K = i) \, p_{\text{pos}}(\tilde{\boldsymbol{y}}_i|\boldsymbol{y}) \prod_{0 \leq j \leq M, \, j \neq i} p_{\text{neg}}(\tilde{\boldsymbol{y}}_j|\boldsymbol{y}). \tag{9}$$

Definition B.3 (Instance Discrimination Problem). Let the instance discrimination data $(\boldsymbol{y}, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M; K)$ correspond to a general CL problem $(\mathcal{Y}, p_{\boldsymbol{Y}}, \mathcal{U}, \mathcal{L}_g)$ (Defn. B.2) that also satisfies the assumptions of Thm. B.1. We define the *instance discrimination problem* as:

$$\min_{u \in \mathcal{U}} \quad \mathcal{L}^{id}(u) := \mathbb{E}\left[-\sum_{i=0}^{M} \delta_{K=i} \ln \frac{e^{u(\boldsymbol{y}, \tilde{\boldsymbol{y}}_i)}}{\sum_{j=0}^{M} e^{u(\boldsymbol{y}, \tilde{\boldsymbol{y}}_j)}} \right], \tag{10}$$

where δ_E is the indicator variable of event E.

Remark B.4. When conditioned on $K = k$, the data distribution Eq. (9) can be slightly simplified to:

$$p(\boldsymbol{y}, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M | K = k) = p_{\boldsymbol{y}}(\boldsymbol{y}) \, p_{\text{pos}}(\tilde{\boldsymbol{y}}_k|\boldsymbol{y}) \prod_{0 \leq j \leq M, \, j \neq k} p_{\text{neg}}(\tilde{\boldsymbol{y}}_j|\boldsymbol{y}). \tag{11}$$

Comparing Eq. (11) to Eq. (6) in Rem. B.2, this distribution acts as if $\tilde{\boldsymbol{y}}_k$ (or $\tilde{\boldsymbol{y}}_{j \neq k}$) was a positive pair (or negative pairs) in a contrastive learning problem w.r.t. anchor $\boldsymbol{y}$.

Lemma B.1. *Under the assumptions of Defn. B.3 we have $\mathcal{L}_g(u) = \mathcal{L}^{id}(u)$, for any $u \in \mathcal{U}$.*

Proof. First, using the law of total expectation $\mathbb{E}[A] = \mathbb{E}[\mathbb{E}[A|B]]$, we expand $\mathcal{L}^{id}$ w.r.t. K:

$$\mathcal{L}^{id}(u) = \mathbb{E}_K\left[\mathbb{E}\left[-\sum_{i=0}^{M}\delta_{K=i}\ln\frac{e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_i)}}{\sum_{j=0}^{M}e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_j)}}\bigg|K\right]\right] \tag{12}$$

$$= \frac{1}{M+1}\sum_{k=0}^{M}\mathbb{E}\left[-\sum_{i=0}^{M}\delta_{K=i}\ln\frac{e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_i)}}{\sum_{j=0}^{M}e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_j)}}\bigg|K=k\right]. \tag{13}$$

However, for the individual summands:

$$\mathbb{E}\left[-\sum_{i=0}^{M}\delta_{K=i}\ln\frac{e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_i)}}{\sum_{j=0}^{M}e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_j)}}\bigg|K=k\right] = \mathbb{E}\left[-\ln\frac{e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_k)}}{\sum_{j=0}^{M}e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_j)}}\bigg|K=k\right] \tag{14}$$

Considering Rem. B.4, the last term equals the contrastive objective $\mathcal{L}_g(u)$.
Therefore, substituting into Eq. (12) we get

$$\mathcal{L}^{id}(u) = \frac{1}{M+1}\sum_{k=0}^{M}\mathcal{L}_g(u) = \mathcal{L}_g(u). \tag{15}$$

$\square$

After having rewritten the general contrastive objective, we are ready to prove Thm. B.1.

Proof of Thm. B.1. By taking the corresponding instance discrimination problem $\mathcal{L}^{id}$, it remains to compute the optima of $\mathcal{L}^{id}$. For this, we again expand $\mathcal{L}^{id}$ with the law of total expectation, but now w.r.t. $(\boldsymbol{y}, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M)$:

$$\mathcal{L}^{id}(u) = \mathbb{E}\left[\mathbb{E}\left[-\sum_{i=0}^{M}\delta_{K=i}\ln\frac{e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_i)}}{\sum_{j=0}^{M}e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_j)}}\bigg|\boldsymbol{y},\tilde{\boldsymbol{y}}_0,\ldots,\tilde{\boldsymbol{y}}_M\right]\right] \tag{16}$$

$$= \mathbb{E}\left[-\sum_{i=0}^{M}\mathbb{E}\left[\delta_{K=i}\big|\boldsymbol{y},\tilde{\boldsymbol{y}}_0,\ldots,\tilde{\boldsymbol{y}}_M\right]\ln\frac{e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_i)}}{\sum_{j=0}^{M}e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_j)}}\right] \tag{17}$$

$$= \mathbb{E}\left[-\sum_{i=0}^{M}\mathbb{P}\left(K=i\big|\boldsymbol{y},\tilde{\boldsymbol{y}}_0,\ldots,\tilde{\boldsymbol{y}}_M\right)\ln\frac{e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_i)}}{\sum_{j=0}^{M}e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_j)}}\right], \tag{18}$$

where we used the properties that $\mathbb{E}\left[A\,f(B)\big|B\right] = f(B)\,\mathbb{E}\left[A\big|B\right]$ and that for any event E, $\mathbb{E}[\delta_E] = \mathbb{P}(E)$.
Let us denote $\tilde{\boldsymbol{Y}} = (\boldsymbol{y}, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M)$ and $\eta_i(\tilde{\boldsymbol{Y}}) := \mathbb{P}\left(K=i\big|\boldsymbol{y},\tilde{\boldsymbol{y}}_0,\ldots,\tilde{\boldsymbol{y}}_M\right)$, the quantity $\boldsymbol{\eta}(\tilde{\boldsymbol{Y}}) = (\eta_0, \ldots, \eta_M)$ is the ground-truth conditional distribution of index K. Denoting $F_i(\tilde{\boldsymbol{Y}}) = e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_i)}/(\sum_{j=0}^{M}e^{u(\boldsymbol{y},\tilde{\boldsymbol{y}}_j)})$, the quantity $\boldsymbol{F}(\tilde{\boldsymbol{Y}}) = (F_0, \ldots, F_M)$ is the inferred conditional distribution of index K. Then,

$$\mathcal{L}^{id}(u) = \mathbb{E}_{\tilde{\boldsymbol{Y}}}\left[-\sum_{i=0}^{M}\eta_i(\tilde{\boldsymbol{Y}})\ln F_i(\tilde{\boldsymbol{Y}})\right] \tag{19}$$

$$= \mathbb{E}_{\tilde{\boldsymbol{Y}}}\left[\sum_{i=0}^{M}\eta_i(\tilde{\boldsymbol{Y}})\ln\frac{\eta_i(\tilde{\boldsymbol{Y}})}{F_i(\tilde{\boldsymbol{Y}})}\right] + \mathbb{E}_{\tilde{\boldsymbol{Y}}}\left[-\sum_{i=0}^{M}\eta_i(\tilde{\boldsymbol{Y}})\ln\eta_i(\tilde{\boldsymbol{Y}})\right] \tag{20}$$

$$= \mathbb{E}_{\tilde{\boldsymbol{Y}}}\left[KL\left(\boldsymbol{\eta}(\tilde{\boldsymbol{Y}})\big\|\boldsymbol{F}(\tilde{\boldsymbol{Y}})\right)\right] + \mathbb{E}_{\tilde{\boldsymbol{Y}}}\left[H\left(\boldsymbol{\eta}(\tilde{\boldsymbol{Y}})\right)\right], \tag{21}$$

where KL and H denote the Kullback-Leibler Divergence (KL) and entropy of the finite distributions over indices $\{0, 1, \ldots, M\}$, provided as arguments. Due to the non-negativity of the KL, we can lower bound $\mathcal{L}_g(u)$:

$$\mathcal{L}_g(u) = \mathcal{L}^{id}(u) \geq \mathbb{E}_{\tilde{\boldsymbol{Y}}}\left[H\left(\boldsymbol{\eta}(\tilde{\boldsymbol{Y}})\right)\right], \tag{22}$$

which is independent of the choice of the function u. Equality is attained if and only if the KL vanishes, which occurs if and only if the true and inferred conditional distribution of index K coincide almost everywhere w.r.t. $\tilde{\boldsymbol{Y}} \sim p(\boldsymbol{y}, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M)$. The statement is in terms of the indices, since the instance discrimination task is about classifying which index belongs to the positive pair. Hence, for any i and almost everywhere w.r.t. $\tilde{\boldsymbol{Y}}$:

$$\frac{e^{u(\boldsymbol{y}, \tilde{\boldsymbol{y}}_i)}}{\sum_{j=0}^{M} e^{u(\boldsymbol{y}, \tilde{\boldsymbol{y}}_j)}} = F_i(\tilde{\boldsymbol{Y}}) = \eta_i(\tilde{\boldsymbol{Y}}) = \mathbb{P}\big(K = i \big| \boldsymbol{y}, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M\big). \tag{23}$$

To rewrite the last expression, we use assumptions $i)$ and $ii)$ and the uniformity of K (step *Step 2*)). Thus, the data distribution Eq. (9) of the instance discrimination problem can be rewritten as

$$p(\boldsymbol{y}, K = i, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M) = \frac{1}{M+1} \, p_{\boldsymbol{y}}(\boldsymbol{y}) \frac{p_{\mathrm{pos}}(\tilde{\boldsymbol{y}}_i | \boldsymbol{y})}{p_{\mathrm{neg}}(\tilde{\boldsymbol{y}}_i | \boldsymbol{y})} \prod_{j=0}^{M} p_{\mathrm{neg}}(\tilde{\boldsymbol{y}}_j | \boldsymbol{y}). \tag{24}$$

With this in mind, Eq. (23) becomes

$$\frac{e^{u(\boldsymbol{y}, \tilde{\boldsymbol{y}}_i)}}{\sum_{j=0}^{M} e^{u(\boldsymbol{y}, \tilde{\boldsymbol{y}}_j)}} = \mathbb{P}\big(K{=}i \big| \boldsymbol{y}, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M\big) = \frac{p(\boldsymbol{y}, K{=}i, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M)}{\sum_{j=0}^{M} p(\boldsymbol{y}, K{=}j, \tilde{\boldsymbol{y}}_0, \ldots, \tilde{\boldsymbol{y}}_M)} = \frac{\frac{p_{\mathrm{pos}}(\tilde{\boldsymbol{y}}_i | \boldsymbol{y})}{p_{\mathrm{neg}}(\tilde{\boldsymbol{y}}_i | \boldsymbol{y})}}{\sum_{j=0}^{M} \frac{p_{\mathrm{pos}}(\tilde{\boldsymbol{y}}_j | \boldsymbol{y})}{p_{\mathrm{neg}}(\tilde{\boldsymbol{y}}_j | \boldsymbol{y})}}. \tag{25}$$

From this it can be seen that there exists a measurable, positive function $\tilde{c} : \mathcal{Y} \to \mathbb{R}$ such that

$$e^{u(\boldsymbol{y}, \tilde{\boldsymbol{y}})} = \tilde{c}(\boldsymbol{y}) \cdot \frac{p_{\mathrm{pos}}(\tilde{\boldsymbol{y}} | \boldsymbol{y})}{p_{\mathrm{neg}}(\tilde{\boldsymbol{y}} | \boldsymbol{y})} \quad \text{holds almost everywhere (w.r.t. any continuous measure of } \mathcal{Y}). \tag{26}$$

By taking the logarithm we get the desired result. $\qquad\square$

C   Identifiability Theory

C.1   Identifiability of Anisotropic CL

Definition C.1 (Anisotropic data generating process, ADGP). Let $\mathcal{Z} := \mathbb{S}^{d-1}$ be the $(d-1)$-dimensional unit hypersphere and let $\boldsymbol{g} : \mathcal{Z} \to \mathcal{X} \subseteq \mathbb{R}^D$ be an invertible and continuous function. Let $\boldsymbol{\Lambda} \in \mathbb{R}^{d \times d}$ be a positive definite diagonal matrix.

The following process is called an *anisotropic data generating process (ADGP)*:

Step 1° $\boldsymbol{z} \sim Uni(\mathcal{Z})$

Step 2° $\boldsymbol{z}^+ \sim p_{\boldsymbol{z}^+|\boldsymbol{z}}(\cdot | \boldsymbol{z})$, where $p_{\boldsymbol{z}^+|\boldsymbol{z}}(\boldsymbol{z}^+ | \boldsymbol{z}) \propto e^{-\|\boldsymbol{z}^+ - \boldsymbol{z}\|_{\boldsymbol{\Lambda}}^2}$

Step 3° for all $j \in \{1, \ldots, M\}, \boldsymbol{z}_j^- \sim Uni(\mathcal{Z})$

Step 4° $\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_j^-\} \xleftarrow{\boldsymbol{g}} \boldsymbol{z}, \boldsymbol{z}^+, \{\boldsymbol{z}_j^-\}$

The pair $(\boldsymbol{g}, \boldsymbol{\Lambda})$ is called the *characterization* of the process. Sets $\mathcal{Z}, \mathcal{X}$ are called the *latent* and *observation spaces*, respectively.

Remark C.1. The random vector $(\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_j^-\})$ meets the assumptions of Defn. B.1, so a contrastive learning problem can be well-defined.

Definition C.2 (Anistropic InfoNCE loss, AnInfoNCE). Let an ADGP be characterized by $(\boldsymbol{g}, \boldsymbol{\Lambda})$ and let $(\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_j^-\})$ be the observed data (all coming from observation space $\mathcal{X} \subseteq \mathbb{R}^D$). Let us optimize the following objective amongst all positive definite (PD) diagonal matrices $\hat{\boldsymbol{\Lambda}}$ and all continuous encoders $\boldsymbol{f} : \mathbb{R}^D \to \mathcal{Z}$:

$$\min_{\substack{\hat{\boldsymbol{\Lambda}} \text{ PD diag} \\ \boldsymbol{f} \text{ continuous}}} \mathcal{L}_{\mathrm{AINCE}}(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}}) := \mathbb{E}\left[-\ln \frac{e^{-\|\boldsymbol{f}(\boldsymbol{x}^+) - \boldsymbol{f}(\boldsymbol{x})\|_{\hat{\boldsymbol{\Lambda}}}^2}}{e^{-\|\boldsymbol{f}(\boldsymbol{x}^+) - \boldsymbol{f}(\boldsymbol{x})\|_{\hat{\boldsymbol{\Lambda}}}^2} + \sum_{j=1}^{M} e^{-\|\boldsymbol{f}(\boldsymbol{x}_j^-) - \boldsymbol{f}(\boldsymbol{x})\|_{\hat{\boldsymbol{\Lambda}}}^2}} \right]. \tag{27}$$

We call the objective $\mathcal{L}_{\mathrm{AINCE}}$ the *Anistropic InfoNCE (AnInfoNCE) loss*.

Remark C.2. The tuple $(\mathcal{X}, p(\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_j^-\}), \mathcal{U}, \mathcal{L}_{\mathrm{AINCE}})$ is a special case of the general contrastive learning problem (Defn. B.1), where $\mathcal{U} = \{u \mid u(\boldsymbol{x}, \boldsymbol{x}^+) = -\|\boldsymbol{f}(\boldsymbol{x}^+) - \boldsymbol{f}(\boldsymbol{x})\|_{\hat{\boldsymbol{\Lambda}}}^2 \text{ for any pair } (\boldsymbol{f}, \hat{\boldsymbol{\Lambda}})\}$ is the hypothesis class of similarity functions.

Theorem 3.1b (Identifiability of Anisotropic CL)**.** *Let an ADGP (Defn. C.1) be characterized by $(\boldsymbol{g}, \boldsymbol{\Lambda})$ with latent and observed data being $(\boldsymbol{z}, \boldsymbol{z}^+, \{\boldsymbol{z}_j^-\})$ and $(\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_j^-\})$. Let $\mathcal{L}_{\mathrm{AINCE}}$ denote the AnInfoNCE loss (Defn. C.2).*

If a pair $(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}})$ (globally) minimizes the $\mathcal{L}_{\mathrm{AINCE}}$ loss, then:

i) $\hat{\boldsymbol{\Lambda}}$ is equal to $\boldsymbol{\Lambda}$ up to a permutation of elements and
ii) $\boldsymbol{f} \circ \boldsymbol{g}$ is a block-orthogonal transformation, where each block acts on latents of equal weight $\boldsymbol{\Lambda}_{ii}$.

In other words, latent $\boldsymbol{z}$ is identified up to a block-orthogonal transformation.

Proof of Thm. 3.1b. Our proof consists of two steps:

Step 1: Based on Appx. B, optimums of AnInfoNCE connect the density ratio to the similarity function u.
Step 2: Then, we solve the resulting functional equation.

Step 1: Connecting the density ratio to the similarity function. First, we are going rewrite the $\mathcal{L}_{\mathrm{AINCE}}$ loss function into a form that favors the analysis of $\boldsymbol{f} \circ \boldsymbol{g}$. Plugging in $\boldsymbol{x} = \boldsymbol{g}(\boldsymbol{z})$ (*Step 4°* in Defn. C.1) into the $\mathcal{L}_{\mathrm{AINCE}}$-loss:

$$\mathcal{L}_{\mathrm{AINCE}}(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}}) = \mathbb{E}\left[-\ln \frac{e^{-\|\boldsymbol{f} \circ \boldsymbol{g}(\boldsymbol{z}^+) - \boldsymbol{f} \circ \boldsymbol{g}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2}}{e^{-\|\boldsymbol{f} \circ \boldsymbol{g}(\boldsymbol{z}^+) - \boldsymbol{f} \circ \boldsymbol{g}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2} + \sum_{j=1}^M e^{-\|\boldsymbol{f} \circ \boldsymbol{g}(\boldsymbol{z}_j^-) - \boldsymbol{f} \circ \boldsymbol{g}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2}} \right] =: \tilde{\mathcal{L}}(\boldsymbol{f} \circ \boldsymbol{g}, \hat{\boldsymbol{\Lambda}}), \qquad (28)$$

with the optimization still over $\boldsymbol{f}$ (and $\hat{\boldsymbol{\Lambda}}$). However, as the generator $\boldsymbol{g}$ is continuously invertible on the *compact* set $\mathcal{Z}$, its inverse $\boldsymbol{g}^{-1}$ is automatically continuous as well. Therefore, any continuous function $\boldsymbol{h} : \mathcal{Z} \to \mathcal{Z}$ can take the role of $\boldsymbol{f} \circ \boldsymbol{g}$, by substituting $\boldsymbol{f} = \boldsymbol{h} \circ \boldsymbol{g}^{-1}$ continuous. Hence, minimizing $\mathcal{L}_{\mathrm{AINCE}}(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}})$ for $\boldsymbol{f}$ continuous is equivalent to minimizing the new, latent space loss $\tilde{\mathcal{L}}(\boldsymbol{h}, \hat{\boldsymbol{\Lambda}})$ for $\boldsymbol{h}$ continuous:

$$\min_{\substack{\hat{\boldsymbol{\Lambda}} \text{ PD diag} \\ \boldsymbol{f} \text{ cont}}} \mathcal{L}_{\mathrm{AINCE}}(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}}) = \min_{\substack{\hat{\boldsymbol{\Lambda}} \text{ PD diag} \\ \boldsymbol{h} \text{ cont}}} \tilde{\mathcal{L}}(\boldsymbol{h}, \hat{\boldsymbol{\Lambda}}). \qquad (29)$$

The new hypothesis class is $\tilde{\mathcal{U}} = \{u \mid u(\boldsymbol{z}, \boldsymbol{z}^+) = -\|\boldsymbol{h}(\boldsymbol{z}^+) - \boldsymbol{h}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2 \text{ for any pair } (\boldsymbol{h}, \hat{\boldsymbol{\Lambda}})\}$.

We claim that $\tilde{\mathcal{L}}$ can attain its Bayes- or unconditional optimum (Thm. B.1), i.e., as if there were no constraints on the similarity function u. According to Thm. B.1, an arbitrary similarity function u minimizes $\tilde{\mathcal{L}}$ amongst all possible measurable u's (attaining the Bayes-optimum) if and only if for a suitable function c:

$$u(\boldsymbol{z}, \tilde{\boldsymbol{z}}) = c(\boldsymbol{z}) + \ln \frac{p_{\boldsymbol{z}^+|\boldsymbol{z}}(\tilde{\boldsymbol{z}}|\boldsymbol{z})}{p_{\boldsymbol{z}^-|\boldsymbol{z}}(\tilde{\boldsymbol{z}}|\boldsymbol{z})} \quad \text{holds almost everywhere on } \mathcal{Z}. \qquad (30)$$

In our special case, u is restricted to take the form $u(\boldsymbol{z}, \tilde{\boldsymbol{z}}) = -\|\boldsymbol{h}(\tilde{\boldsymbol{z}}) - \boldsymbol{h}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2$. Besides that, $\boldsymbol{z}_j^- \sim Uni(\mathcal{Z})$ with a constant density function and $p_{\boldsymbol{z}^-|\boldsymbol{z}} = p_{\boldsymbol{z}^-} \equiv \text{const}$. We also plug in the anisotropic positive conditional distribution $p_{\boldsymbol{z}^+|\boldsymbol{z}}(\tilde{\boldsymbol{z}}|\boldsymbol{z}) \propto e^{-\|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}}}$. After merging every constant additive term (including normalization constants) into function c, we get the functional equation with unknowns $\boldsymbol{h}, \hat{\boldsymbol{\Lambda}}, c$:

$$\|\boldsymbol{h}(\tilde{\boldsymbol{z}}) - \boldsymbol{h}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2 = c(\boldsymbol{z}) + \|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}}^2 \quad \text{holds almost everywhere on } \mathcal{Z}. \qquad (31)$$

Observe that the equation is trivially satisfied for $\boldsymbol{h} = id_{\mathcal{Z}}, \hat{\boldsymbol{\Lambda}} = \boldsymbol{\Lambda}, c \equiv 0$. This special case corresponds to having $\boldsymbol{f} = \boldsymbol{g}^{-1}$, i.e., we inverted the exact generator and also reconstructed $\boldsymbol{\Lambda}$. Consequently, our loss function attains its Bayes-minimum even in our restricted hypothesis class. Now leveraging again Thm. B.1 in the other direction, the above shows us that all minimizers are, in fact, solutions of Eq. (31).

Let's take an arbitrary solution $\boldsymbol{h}, \hat{\boldsymbol{\Lambda}}, c$. Noticeably, the single-variable, measurable function c is equal (almost everywhere) to a two-variable continuous function. This is only possible if the latter function, depending on $(\boldsymbol{z}, \tilde{\boldsymbol{z}})$ is a single-variable, continuous function of $\boldsymbol{z}$. In this case, swapping c to this exact function will also reveal a good solution and, therefore, it is only required to solve Eq. (31) for c continuous and on the full space $\mathcal{Z}$. By taking the special case of $\tilde{\boldsymbol{z}} \leftarrow \boldsymbol{z}$, we see that $c(\boldsymbol{z}) = 0$ follows.

Step 2: Solving the functional equation. What is left to solve is the following functional equation in terms of the variables $\boldsymbol{h}, \hat{\boldsymbol{\Lambda}}$:

$$\|\boldsymbol{h}(\tilde{\boldsymbol{z}}) - \boldsymbol{h}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2 = \|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}}^2 \quad \text{holds for any } \tilde{\boldsymbol{z}}, \boldsymbol{z} \text{ on } \mathcal{Z}. \qquad (32)$$

For this, let $\boldsymbol{z}_0 \in \mathcal{Z}$ be fixed and let us define the following functions:

$$\boldsymbol{a}(\boldsymbol{z}) = \hat{\boldsymbol{\Lambda}}^{1/2}\big(\boldsymbol{h}(\boldsymbol{z}) - \boldsymbol{h}(\boldsymbol{z}_0)\big) \quad \text{and} \quad \boldsymbol{b}(\boldsymbol{z}) = \boldsymbol{\Lambda}^{1/2}(\boldsymbol{z} - \boldsymbol{z}_0). \tag{33}$$

In this case, the following properties are easily verified:

$$\|\boldsymbol{a}(\tilde{\boldsymbol{z}}) - \boldsymbol{a}(\boldsymbol{z})\|^2 = \|\boldsymbol{b}(\tilde{\boldsymbol{z}}) - \boldsymbol{b}(\boldsymbol{z})\|^2, \tag{34}$$

$$\boldsymbol{a}(\boldsymbol{z}_0) = \boldsymbol{b}(\boldsymbol{z}_0) = 0. \tag{35}$$

By substituting $\tilde{\boldsymbol{z}} \leftarrow \boldsymbol{z}_0$ into Eq. (34) and using Eq. (35), we get that

$$\|\boldsymbol{a}(\boldsymbol{z})\|^2 = \|\boldsymbol{b}(\boldsymbol{z})\|^2 \text{ holds for any } \boldsymbol{z} \in \mathcal{Z}. \tag{36}$$

After expanding Eq. (34) and using Eq. (36) twice, we get that, similarly to the distance and norm, the scalar product is equivalently transformed by $\boldsymbol{a}$ and $\boldsymbol{b}$:

$$\langle \boldsymbol{a}(\tilde{\boldsymbol{z}}), \boldsymbol{a}(\boldsymbol{z}) \rangle = \langle \boldsymbol{b}(\tilde{\boldsymbol{z}}), \boldsymbol{b}(\boldsymbol{z}) \rangle \text{ for any } \tilde{\boldsymbol{z}}, \boldsymbol{z} \in \mathcal{Z}. \tag{37}$$

From this, it can be proven that there exists an orthogonal linear transformation $\boldsymbol{O}$ of $\mathbb{R}^d$ such that:

$$\boldsymbol{a}(\boldsymbol{z}) = \boldsymbol{O}\boldsymbol{b}(\boldsymbol{z}) \text{ holds for any } \boldsymbol{z} \in \mathcal{Z}. \tag{38}$$

To see this, we first prove that the following mapping is well-defined on the linear span of $\boldsymbol{b}(\mathcal{Z})$:

$$\boldsymbol{O} : \sum_{i=1}^{n} \alpha_i \boldsymbol{b}(\boldsymbol{z}_i) \mapsto \sum_{i=1}^{n} \alpha_i \boldsymbol{a}(\boldsymbol{z}_i) \tag{39}$$

Assume there exist two equal expansions (with potentially zero coefficients):

$$\sum_{i=1}^{n} \alpha_i \boldsymbol{b}(\boldsymbol{z}_i) = \sum_{i=1}^{n} \beta_i \boldsymbol{b}(\boldsymbol{z}_i). \tag{40}$$

In this case the difference and the distance square vanish:

$$\left\| \sum_{i=1}^{n} (\alpha_i - \beta_i)\boldsymbol{b}(\boldsymbol{z}_i) \right\|^2 = 0. \tag{41}$$

However, due to the equivalent transformation of the scalar product (Eq. (37)), after the expansion of Eq. (41) we can essentially change every occurrence of $\boldsymbol{b}$ to $\boldsymbol{a}$ and receive that:

$$\left\| \sum_{i=1}^{n} (\alpha_i - \beta_i)\boldsymbol{a}(\boldsymbol{z}_i) \right\|^2 = 0, \tag{42}$$

which may hold if and only if

$$\sum_{i=1}^{n} \alpha_i \boldsymbol{a}(\boldsymbol{z}_i) = \sum_{i=1}^{n} \beta_i \boldsymbol{a}(\boldsymbol{z}_i), \tag{43}$$

proving that $\boldsymbol{O}$ in Eq. (39) is well-defined on the linear span of $\boldsymbol{b}(\mathcal{Z})$.

Secondly, the facts that $\boldsymbol{O}$ is linear and that $\boldsymbol{a}(\boldsymbol{z}) = \boldsymbol{O}\boldsymbol{b}(\boldsymbol{z})$ hold trivially. We also see that the image of $\boldsymbol{b}$, i.e. $\boldsymbol{b}(\mathcal{Z}) = \boldsymbol{b}(\mathbb{S}^{d-1})$ is full dimensional and hence $\boldsymbol{O}$ is defined on the entire $\mathbb{R}^d$. The orthogonality is a direct consequence of the scalar product preservation property (Eq. (37)) and the definition of $\boldsymbol{O}$ (Eq. (39)).

Consequently, we have proven that there exists an orthogonal linear transformation $\boldsymbol{O}$ of $\mathbb{R}^d$ such that:

$$\hat{\boldsymbol{\Lambda}}^{1/2}(\boldsymbol{h}(\boldsymbol{z}) - \boldsymbol{h}(\boldsymbol{z}_0)) = \boldsymbol{O}\boldsymbol{\Lambda}^{1/2}(\boldsymbol{z} - \boldsymbol{z}_0) \text{ holds for any } \boldsymbol{z} \in \mathcal{Z}. \tag{44}$$

As $\hat{\boldsymbol{\Lambda}}$ is positive definite and invertible, it follows (after merging constant terms containing $\boldsymbol{z}_0$), that:

$$\boldsymbol{h}(\boldsymbol{z}) = \hat{\boldsymbol{\Lambda}}^{-1/2}\boldsymbol{O}\boldsymbol{\Lambda}^{1/2}\boldsymbol{z} + \boldsymbol{c} \text{ holds for any } \boldsymbol{z} \in \mathcal{Z}, \text{ for some constant } c. \tag{45}$$

Let us denote $\boldsymbol{H} = \hat{\boldsymbol{\Lambda}}^{-1/2} \boldsymbol{O} \boldsymbol{\Lambda}^{1/2}$. To complete the proof, we have to show that $\boldsymbol{c} = 0$ and $\boldsymbol{H}$ is orthogonal. For this, we use the fact that $\boldsymbol{h}$ is a transformation of the unit hypersphere $\mathcal{Z} = \mathbb{S}^{d-1}$:

$$\langle \boldsymbol{H}\boldsymbol{z}, \boldsymbol{c} \rangle = \frac{1}{4} \Big(\|\boldsymbol{H}\boldsymbol{z} + \boldsymbol{c}\|^2 - \| -\boldsymbol{H}\boldsymbol{z} + \boldsymbol{c}\|^2 \Big) \tag{46}$$

$$= \frac{1}{4} \Big(\|\boldsymbol{h}(\boldsymbol{z})\|^2 - \|\boldsymbol{h}(-\boldsymbol{z})\|^2 \Big) = \frac{1}{4}(1-1) = 0, \text{ for any } \boldsymbol{z} \in \mathcal{Z}. \tag{47}$$

As $\boldsymbol{H}$ is invertible and, thus, with a full range of $\mathbb{R}^d$, $\boldsymbol{c} = 0$ is concluded. Therefore, $\boldsymbol{h}$ is linear. As unit vectors are mapped to unit vectors, $\boldsymbol{h}$ is norm-preserving and, thus, $\boldsymbol{h} = \boldsymbol{f} \circ \boldsymbol{g}$ is orthogonal.

Now, rewriting $\boldsymbol{H} = \hat{\boldsymbol{\Lambda}}^{-1/2} \boldsymbol{O} \boldsymbol{\Lambda}^{1/2}$ gives us:

$$\boldsymbol{O}\boldsymbol{\Lambda}^{1/2} = \hat{\boldsymbol{\Lambda}}^{1/2} \boldsymbol{H} = \boldsymbol{H} \big(\boldsymbol{H}^\top \hat{\boldsymbol{\Lambda}}^{1/2} \boldsymbol{H} \big), \tag{48}$$

where the first and last formulas provide polar decompositions of the same operator. As the polar decomposition of invertible operators is unique, we conclude that $\boldsymbol{O} = \boldsymbol{H}$, and

$$\boldsymbol{\Lambda}^{1/2} = \boldsymbol{H}^\top \hat{\boldsymbol{\Lambda}}^{1/2} \boldsymbol{H}. \tag{49}$$

After squaring both sides, we get:

$$\boldsymbol{\Lambda} = \boldsymbol{H}^\top \hat{\boldsymbol{\Lambda}} \boldsymbol{H}. \tag{50}$$

Then the right-hand side of Eq. (50) is the eigendecomposition of $\boldsymbol{\Lambda}$. Due to the uniqueness of eigenvalues, there must exist a permutation $\mathbf{P}_{\boldsymbol{h}}$ such that $\hat{\boldsymbol{\Lambda}} = \mathbf{P}_{\boldsymbol{h}} \boldsymbol{\Lambda} \mathbf{P}_{\boldsymbol{h}}^\top$. Then

$$\boldsymbol{\Lambda} = \boldsymbol{H}^\top \mathbf{P}_{\boldsymbol{h}} \boldsymbol{\Lambda} \mathbf{P}_{\boldsymbol{h}}^\top \boldsymbol{H} \tag{51}$$

$$\Leftrightarrow \mathbf{P}_{\boldsymbol{h}}^\top \boldsymbol{H} \boldsymbol{\Lambda} = \boldsymbol{\Lambda} \mathbf{P}_{\boldsymbol{h}}^\top \boldsymbol{H}. \tag{52}$$

In other words, $\mathbf{P}_{\boldsymbol{h}}^\top \boldsymbol{H}$ and $\boldsymbol{\Lambda}$ commute. Then $\mathbf{P}_{\boldsymbol{h}}^\top \boldsymbol{H}$ must be a diagonal matrix if all values on the diagonal of $\boldsymbol{\Lambda}$ are distinct. Should some values be the same, then the corresponding submatrix of $\mathbf{P}_{\boldsymbol{h}}^\top \boldsymbol{H}$ is orthogonal.

To see this more formally, we define $\mathbf{O} := \mathbf{P}_{\boldsymbol{h}}^\top \boldsymbol{H}$. Then, for all i, j,

$$\sum_k \mathrm{O}_{ik} \lambda_k \delta_{kj} = \sum_k \lambda_i \delta_{ik} \mathrm{O}_{kj}, \tag{53}$$

$$\Leftrightarrow \mathrm{O}_{ij} \lambda_j = \lambda_i \mathrm{O}_{ij}, \tag{54}$$

$$\Leftrightarrow \mathrm{O}_{ij}(\lambda_j - \lambda_i) = 0. \tag{55}$$

Hence, for distinct eigenvalues, all off-diagonal entries of $\mathbf{O}$ are zero. Within the space of equivalent eigenvalues, $\mathbf{O}$ is only constrained by the requirement to be orthogonal.

From $\mathbf{O} = \mathbf{P}_{\boldsymbol{h}}^\top \boldsymbol{H}$ we then know that $\boldsymbol{H}$ is — up to permutation — an orthogonal block-diagonal matrix where the size of each block corresponds to the multiplicity of the values on the diagonal of $\boldsymbol{\Lambda}$.

$\square$

C.2 Including hard negative sampling

Definition C.3 (Anisotropic DGP with hard negatives (HN)). Let $\boldsymbol{g} : \mathcal{Z} := \mathbb{S}^{d-1} \to \mathcal{X} \subseteq \mathbb{R}^D$ be an invertible and continuous function. Let $\boldsymbol{\Lambda}^+, \boldsymbol{\Lambda}^- \in \mathbb{R}^{d \times d}$ be positive definite diagonal matrices such that $\boldsymbol{\Lambda}_{ii}^+ > \boldsymbol{\Lambda}_{ii}^-$ (or equivalently, $\boldsymbol{\Lambda}^+ - \boldsymbol{\Lambda}^-$ is still PD).

The following process is called an *anisotropic DGP with hard negatives (HN)*, characterized by the triple $(\boldsymbol{g}, \boldsymbol{\Lambda}^+, \boldsymbol{\Lambda}^-)$:

Step 1) $\boldsymbol{z} \sim Uni(\mathcal{Z})$

Step 2) $\boldsymbol{z}^+ \sim p_{\boldsymbol{z}^+|\boldsymbol{z}}(\cdot|\boldsymbol{z})$, where $p_{\boldsymbol{z}^+|\boldsymbol{z}}(\boldsymbol{z}^+|\boldsymbol{z}) \propto e^{-\|\boldsymbol{z}^+ - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^+}^2}$

Step 3) for all $j \in \{1, \dots, M\}, \boldsymbol{z}_j^- \sim p_{\boldsymbol{z}^-|\boldsymbol{z}}(\cdot|\boldsymbol{z})$, where $p_{\boldsymbol{z}^-|\boldsymbol{z}}(\boldsymbol{z}^-|\boldsymbol{z}) \propto e^{-\|\boldsymbol{z}^- - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^-}^2}$

Step 4) $\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_j^-\} \xleftarrow{\boldsymbol{g}} \boldsymbol{z}, \boldsymbol{z}^+, \{\boldsymbol{z}_j^-\}$.

Corollary 3.1b (Identifiability of anisotropic DGPs with HN). *Let an ADGP with HN (Defn. C.3) be characterized by $(\boldsymbol{g}, \boldsymbol{\Lambda}^+, \boldsymbol{\Lambda}^-)$ with latent and observed data being $(\boldsymbol{z}, \boldsymbol{z}^+, \{\boldsymbol{z}_j^-\})$ and $(\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_j^-\})$. Let $\mathcal{L}_{\mathrm{AINCE}}$ denote the AnInfoNCE loss (Defn. C.2), evaluated w.r.t. $(\boldsymbol{x}, \boldsymbol{x}^+, \{\boldsymbol{x}_j^-\})$.*

If a pair $(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}})$ (globally) minimizes the $\mathcal{L}_{\mathrm{AINCE}}$ loss, then:

  i) $\hat{\boldsymbol{\Lambda}}$ is equal to $\boldsymbol{\Lambda}^+ - \boldsymbol{\Lambda}^-$ up to a permutation of elements and

  ii) $\boldsymbol{f} \circ \boldsymbol{g}$ is an orthogonal transformation, or latent $\boldsymbol{z}$ is identified up to an orthogonal transformation.

Proof. We are following the steps of the Proof of Thm. 3.1b.

First, we again rewrite the $\mathcal{L}_{\mathrm{AINCE}}$ by plugging in $\boldsymbol{x} = \boldsymbol{g}(\boldsymbol{z})$ and receiving $\mathcal{L}_{\mathrm{AINCE}}(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}}) = \tilde{\mathcal{L}}(\boldsymbol{f} \circ \boldsymbol{g}, \hat{\boldsymbol{\Lambda}})$, with optimization over $\boldsymbol{f}$ (and $\hat{\boldsymbol{\Lambda}}$). As $\boldsymbol{h} = \boldsymbol{f} \circ \boldsymbol{g}$ attains all possible continuous functions, we may optimize $\tilde{\mathcal{L}}(\boldsymbol{h}, \hat{\boldsymbol{\Lambda}})$. The hypothesis class becomes, again, $\tilde{\mathcal{U}} = \{u \mid u(\boldsymbol{z}, \boldsymbol{z}^+) = -\|\boldsymbol{h}(\boldsymbol{z}^+) - \boldsymbol{h}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2 \text{ for any pair } (\boldsymbol{h}, \hat{\boldsymbol{\Lambda}})\}$.

Secondly, according to Thm. B.1, an arbitrary similarity function u minimizes $\tilde{\mathcal{L}}$ amongst all possible measurable u's (attaining the Bayes-optimum) if and only if for a suitable function c:

$$u(\boldsymbol{z}, \tilde{\boldsymbol{z}}) = c(\boldsymbol{z}) + \ln \frac{p_{\boldsymbol{z}^+|\boldsymbol{z}}(\tilde{\boldsymbol{z}}|\boldsymbol{z})}{p_{\boldsymbol{z}^-|\boldsymbol{z}}(\tilde{\boldsymbol{z}}|\boldsymbol{z})} \quad \text{holds almost everywhere on } \mathcal{Z}. \tag{56}$$

We are about to plug in all our constraints. The major difference compared to the Proof of Thm. 3.1b is that besides the positive conditional distribution being $p_{\boldsymbol{z}^+|\boldsymbol{z}}(\tilde{\boldsymbol{z}}|\boldsymbol{z}) \propto e^{-\|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^+}^2}$, the negative conditional also changes to $p_{\boldsymbol{z}^-|\boldsymbol{z}}(\tilde{\boldsymbol{z}}|\boldsymbol{z}) \propto e^{-\|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^-}^2}$. After plugging in the restricted form of u and merging every constant normalizing term into function c, Eq. (56) becomes:

$$-\|\boldsymbol{h}(\tilde{\boldsymbol{z}}) - \boldsymbol{h}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2 = c(\boldsymbol{z}) + \ln \frac{e^{-\|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^+}^2}}{e^{-\|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^-}^2}} \tag{57}$$

$$= c(\boldsymbol{z}) - \|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^+}^2 + \|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^-}^2 \tag{58}$$

$$= c(\boldsymbol{z}) - \langle \tilde{\boldsymbol{z}} - \boldsymbol{z}, \boldsymbol{\Lambda}^+(\tilde{\boldsymbol{z}} - \boldsymbol{z}) \rangle + \langle \tilde{\boldsymbol{z}} - \boldsymbol{z}, \boldsymbol{\Lambda}^-(\tilde{\boldsymbol{z}} - \boldsymbol{z}) \rangle \tag{59}$$

$$= c(\boldsymbol{z}) - \langle \tilde{\boldsymbol{z}} - \boldsymbol{z}, (\boldsymbol{\Lambda}^+ - \boldsymbol{\Lambda}^-)(\tilde{\boldsymbol{z}} - \boldsymbol{z}) \rangle = c(\boldsymbol{z}) - \|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^+ - \boldsymbol{\Lambda}^-}^2. \tag{60}$$

Hence, we get the functional equation with unknowns $\boldsymbol{h}, \hat{\boldsymbol{\Lambda}}, c$:

$$\|\boldsymbol{h}(\tilde{\boldsymbol{z}}) - \boldsymbol{h}(\boldsymbol{z})\|_{\hat{\boldsymbol{\Lambda}}}^2 = c(\boldsymbol{z}) + \|\tilde{\boldsymbol{z}} - \boldsymbol{z}\|_{\boldsymbol{\Lambda}^+ - \boldsymbol{\Lambda}^-}^2 \quad \text{holds almost everywhere on } \mathcal{Z}. \tag{61}$$

We conclude our proof by continuing the Proof of Thm. 3.1b by substituting $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^+ - \boldsymbol{\Lambda}^-$ PD. $\qquad \square$

Remark C.3 (Subsuming the uniform marginal in the negative conditional with $\boldsymbol{\Lambda}^- = 0$). When using a negative conditional in the form of Eq. (2), $\boldsymbol{\Lambda}^- = 0$ accounts for a uniform marginal.

C.3 Identifiability of a Loss Ensemble

In this setting the observations $\boldsymbol{x}$ come from k different DGPs, with a *shared* generator $\boldsymbol{g}$ and separate concentration parameters $\{\boldsymbol{\Lambda}^i\}_{i=1}^k$. That is, this setting is akin to works investigating identifiability in multiple environments (Hyvärinen and Morioka, 2016; Gresele et al., 2019; Rajendran et al., 2023) and reflects the practice in CL of optimizing multiple loss components with different data augmentations (Eastwood et al., 2023; Xiao et al., 2021; Zhang and Ma, 2022).

Corollary 3.2b (Identifiability of ensemble AnInfoNCE). *Assume k DGPs, each satisfying Assum. 1. Assume that they share $\boldsymbol{g}$, but have different concentration parameters $\{\boldsymbol{\Lambda}^i\}$, for $1 \leq i \leq k$. Assume a shared encoder $\boldsymbol{f}$ and k learnable $\hat{\boldsymbol{\Lambda}}^i$ parameters. Then, the tuple $(\boldsymbol{f}, \{\hat{\boldsymbol{\Lambda}}^i\})$ minimizing the ensemble loss $\sum_i^k \mathcal{L}_{\mathrm{AINCE}}(\boldsymbol{f}, \hat{\boldsymbol{\Lambda}}^i)$ identifies the latents up to a block-orthogonal transformation.*

Proof. $\boldsymbol{f} = \boldsymbol{g}^{-1}$ together with $\hat{\boldsymbol{\Lambda}}^i = \boldsymbol{\Lambda}^i$ minimizes the ith loss term. As $\boldsymbol{g}$ is shared, the individual losses can attain their minima simultaneously. Hence, this also holds for any minimum of the ensemble loss. Applying Thm. 3.1 to either of them concludes the proof. $\qquad \square$
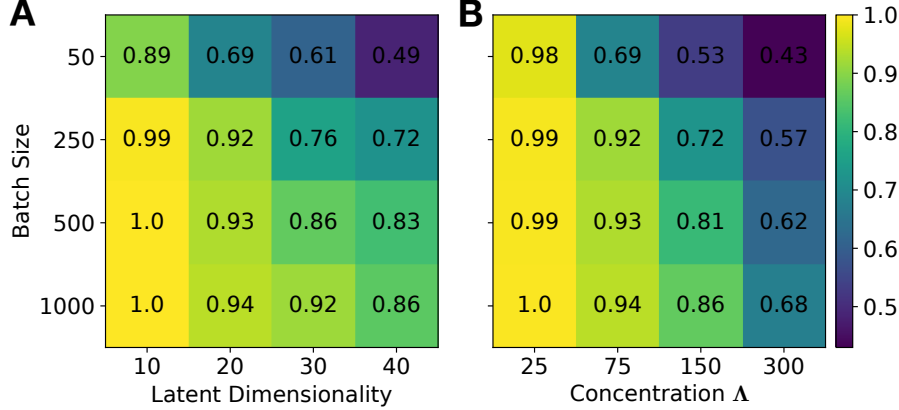
A

Figure 4: **Latent Dimensionality, Concentration, and Batch Size Influence Identifiability.** Linear identifiability, quantified by the R^2 score between reconstructed and ground-truth latents, degrades with **A:** higher latent dimensionality d; and **B:** larger concentration parameter $\mathbf{\Lambda}$ in the ground-truth positive conditional. Both detrimental effects can be countered by increasing the batch size.

Remark C.4 (Sufficiency condition for identifiability up to permutations for the ensemble AnInfoNCE loss). Ensemble AnInfoNCE is identifiable up to permutations if, under the conditions of Cor. 3.2, there are sufficiently many DGPs such that for all distinct latent factors z_k, z_l, there exists a DGP with concentration parameter $\mathbf{\Lambda}^i$ such that $\mathbf{\Lambda}_{kk}^i \neq \mathbf{\Lambda}_{ll}^i$.

D   Additional Analysis: Ablations and Parameter Sensitivity

In Section 5 of the main manuscript, we identified the use of augmentations as the primary issue in observing a trade-off between accuracy and identifiability on real-world data. In this Section, we analyze other reasons why scaling AnInfoNCE to real-world data is challenging. In particular, we observe that the interplay between the batch size, the latent dimensionality and the value of the concentration parameter plays a crucial role. To demonstrate this, we run controlled experiments on synthetic data to investigate the assumptions and properties of our theory. Since our analysis equally applies to the isotropic and anisotropic cases, we use the isotropic one for our experiments for simplicity but consider the general anisotropic case in our additional theoretical results.

D.1   Issues when Scaling up AnInfoNCE

Batch Size Needs to Scale with the Dimensionality.   One potential difference between the fully-controlled and the real-world experiments is the latent dimensionality, which will (most likely) be substantially higher for more complex image data. Therefore, we investigate the influence of the latent dimension and batch size on the linear identifiability score (Fig. 4A). We observe a significant degradation of performance with a smaller batch size and higher latent dimensionality, corroborating the theoretical predictions of Wang et al. (2022) and in line with the empirical observation that CL performs better with a higher batch size (Chen et al., 2020a). Even for a modest dimensionality, a rather high batch size is required. Extrapolating our results, feature encoders on the order of what is typically used on CIFAR10 and ImageNet (512 to 2048 dimensions) would require extremely large batch sizes to reach high identifiability scores even if all assumptions match our theory.

Batch Size Needs to Scale with the Concentration $\mathbf{\Lambda}$.   Another important difference between our controlled and real-world experiments concerns the choice of concentration parameters and its influence on the batch size for identifiability. More concretely, the more concentrated the conditionals are, and the higher the dimensionality, the less training signal CL has to structure the representation (Wang et al., 2022). This was denoted by Wang et al. (2022) as the role of *augmentation overlap*. Our anisotropic positive conditional distribution could help explain the role of augmentation overlap from an identifiability perspective via the concentration parameter $\mathbf{\Lambda}$, whose effect is illustrated in Fig. 5. Larger $\mathbf{\Lambda}$ means a more concentrated conditional and requires larger batch sizes for sufficient augmentation overlap; otherwise, the R^2 score will degrade (see Fig. 4B for $d = 20$). In real-world experiments, the positive pairs distribution is determined by augmentations. Likely, the shape or the class of an
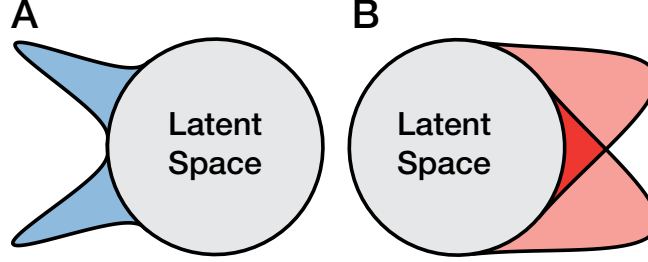
Rusak*, Reizinger*, Juhos*, Bringmann, Zimmermann[†], Brendel[†]

Figure 5: **Concentration of Positive Pairs Influences Augmentation Overlap**. We figuratively visualize conditional distributions with large (**A**) and small (**B**) concentration parameters $\mathbf{\Lambda}$. For large concentration values, samples from the conditional distribution of two anchor points do not overlap, signifying missing augmentation overlap. This is not the case for small $\mathbf{\Lambda}$ values.
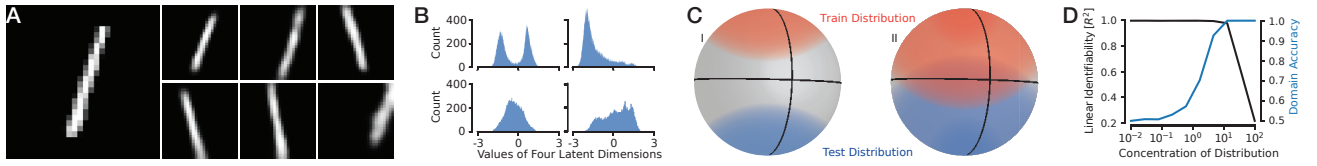

Figure 6: **Data Augmentation Can Violate Our Data Distribution Assumptions.** We train a VAE on augmented MNIST images. **A.** We show an MNIST sample and six augmented versions generated by the VAE. **B.** We project augmented versions of an MNIST sample into the VAE's latent space and visualize the distribution of four dimensions. This demonstrates a violated conditional assumption of a normalized Gaussian. **C.** We visualize violations of the uniformity assumption for the marginal distribution during training and testing. **D.** We observe degrading identifiability scores when the train and test domains can be distinguished. Since an almost perfect domain classifier can be trained on real-world datasets, we conclude that the uniformity assumption is likely violated in practice.

object are minimally affected by the augmentations, which corresponds to those latent dimensions having a large concentration parameter, thereby again increasing the necessary batch size to reach high identifiability scores.

Augmentations can Violate our Conditional Assumption. Next, we consider the role of different positive pair distributions. In § 3, we assume a normalized Gaussian distribution over a hypersphere. However, it is unclear whether practical augmentation strategies follow such a conditional. We test our assumption's validity by training a Variational Autoencoder (VAE) on augmented images from MNIST with only crops and horizontal flips as augmentations since MNIST images are grayscale. Training an encoder on MNIST using these augmentations leads to a KNN accuracy of 96.7%, validating our augmentation choice. We visualize an unaugmented image and VAE reconstructions of its augmented versions in Fig. 6A. To show the conditional distribution, we project augmented images onto the learned VAE latent space and demonstrate in Fig. 6B that the conditional distribution does not follow a normalized Gaussian distribution for most dimensions and can even be bimodal; we show more examples in Fig. 7. This holds even though the latent space of the VAE across all augmented data samples is Gaussian by construction. Our theory does not cover this case and so this mismatch presents a fruitful direction for future research on closing the gap between theory and practice.

Augmentations can Violate our Uniformity Assumption. Our theory assumes the same uniform marginal $p(\boldsymbol{z})$ for both training and testing. However, this assumption's validity is unclear in practice: During training, even anchor points are produced by data augmentation, creating a potential mismatch to the test domain which lacks these augmentations. We test the influence of violations of this assumption by interpolating the marginal between a uniform and a vMF distribution with varying concentration. We center the vMF at the north (south) pole during training (testing) (Fig. 6C). We train a binary classifier to distinguish samples from the training and test distributions. For different concentrations, we compute the domain classification accuracy and the identifiability score (Fig. 6D), showing a stark anticorrelation. Our domain classification results with regular SimCLR augmentations (MNIST: 99.1%, CIFAR10: 99.2%, and ImageNet 98.7%) demonstrate that binary

classifiers can almost perfectly distinguish augmented and non-augmented samples on real-world data. Connecting this to observed anticorrelation with identifiability, we conclude that the mismatch between distributions hurts the quality of representations.

The Conditional's Effect on Learning Dynamics. Finally, the generative process of real-world image data is most likely substantially richer compared to the previously studied synthetic DGPs, resulting in a generally harder learning problem requiring longer training time. In particular, we hypothesize that the conditional distribution's concentration partly influences the learning speed; this aligns with previous observations of flat loss landscapes for Noise Contrastive Estimation (NCE) (Liu et al., 2022). Our anisotropic conditional of positive pairs provides a possible explanation: We hypothesize that similar to Saxe et al. (2014), the factors controlling the variances (in our case, the concentration Λ) affect the loss landscape, making some latents *slow* to learn. Thus, it is conceivable that more style-like (ground-truth) latents are learned slower (thus, harder) compared to more content-like ones (see, e.g., in Fig. 2B).

We demonstrate the influence of the concentration parameter on the learning dynamics as follows. We define two DGPs using the synthetic data from Section 4.1. We assume a uniform marginal and an isotropic normalized Gaussian conditional with two different Λ parameters: $\Lambda = 25$ and $\Lambda = 75$. A higher concentration parameter corresponds to a more narrow conditional distribution and thus, lower variation between the anchor and its conditional sample in the latent space. We analyze how Λ affects learning dynamics (Fig. 8) and observe that learning the latents follows a step-like behavior, corroborating Simon et al. (2023), where latents with larger Λ are learned slower (Fig. 8 A & D). Alternatively, reaching a target R^2 score requires more gradient steps unless the batch size is increased (Fig. 8 B & E). The cosine similarities' distribution for negative and positive pairs clearly shows that large Λ leads to insufficient augmentation overlap: Since the histograms are disjoint for large Λ, the anchor's nearest neighbor is always a positive sample (Fig. 8 C & F). This makes distinguishing positive from negative samples trivial and allows for a short-cut solution. However, augmentation overlap cannot fully explain CL in practice: Saunshi et al. (2022) show that practical augmentations often provide no significant overlap; nonetheless, CL still succeeds, potentially due to (architectural) inductive biases.

D.2   Dependence of Identifiability on Batch Size, Training Duration, and Dimensionality

We now further analyze the dependence of the batch size, training duration, and dimensionality of the latent space on the identifiability scores. To this end, we train models with different batch sizes on data with various latent dimensionality and investigate their identifiability scores. We use 3 random seeds per configuration and train all models for 500k iterations using the same hyperparameters as described above. We consider the usual conditional distribution of positive pairs with a concentration value $\Lambda = 75$.

Fig. 9 clearly shows that for higher dimensional latent spaces, one needs to train (exponentially) longer and use an increased batch size. Note that even for the fairly low dimensional settings investigated here, large batch sizes and long training runs were necessary to achieve high identifiability scores. Comparing this to the dimensionalities usually assumed/used in practice, e.g., 512 dimensions when training on images, we argue that the usual batch sizes and training durations are too small to achieve high identifiability scores. This observation would explain the mixed performance observed earlier when testing our loss on real-world image data despite its identifiability guarantees.

D.3   Hard Negative Mining and Ensemble AnInfoNCE

Hard Negative Mining. In the previous section, we observed that high concentration parameters require higher batch sizes for successful learning. HN mining can help counter highly concentrated conditionals by increasing the sample density close to the anchors. First, we demonstrate in our synthetic setting for $d = 20$ that with HN mining, the optimum for the loss in terms of $\hat{\Lambda}$ is indeed at the predicted value of $\Lambda = \Lambda^+ - \Lambda^-$ (Fig. 10A). We also finetune an encoder using AnInfoNCE by adding three hard negatives for each sample. We observe better linear identifiability scores compared to continued regular training, supporting our Cor. 3.1, (Fig. 10B).

Training With Ensemble AnInfoNCE. To test the benefit of ensembling AnInfoNCE for training, we define two DGPs with $d = 20$. The DGPs share a uniform marginal distribution and differ in their respective conditional distributions. To model latents that cannot be changed by either DGP, such as object class or shape, we set a high concentration parameter of 400 for three latents shared between both DGPs. For the remaining seventeen latents, we choose two Λ values of 15 and 250, modeling strongly and weakly varying latents and alternate them

between the DGPs:

$$\boldsymbol{\Lambda}_1 = \mathrm{diag}\,(400, 400, 400, 15, \ldots, 15, 250, \ldots, 250)$$
$$\boldsymbol{\Lambda}_2 = \mathrm{diag}\,(400, 400, 400, 250, \ldots, 250, 15, \ldots, 15)\,.$$

We sample data with both DGPs and use two losses with two learnable concentration parameters $\hat{\boldsymbol{\Lambda}}_1, \hat{\boldsymbol{\Lambda}}_2$ to model both DGPs. We train the encoder on the sum of both losses. We see improved linear identifiability scores when training the encoder on the loss ensemble compared to InfoNCE or AnInfoNCE (see also Fig. 11A):

Loss	InfoNCE	AnInfoNCE	Ens. AnInfoNCE
Lin. ident. $[R^2]$	0.48	0.93	0.98

Further, we observe a specialization in the learned $\hat{\Lambda}$ values (Fig. 11B+C), where same colors indicate the same $\hat{\Lambda}$ dimension): We observe an anti-correlation between $\hat{\Lambda}_1$ and $\hat{\Lambda}_2$. Optimizing an ensemble loss makes it more likely that for each latent dimension, there exists an augmentation making the dimensions more content-like with a higher $\boldsymbol{\Lambda}$.

D.4 Comparison to TriCL (Zhang et al., 2023)

We first discuss the conceptual differences, theoretical results and assumptions betwen TriCL and AnInfoNCE. TriCL extends and theoretically analyzes the spectral contrastive loss formulation (their extension to InfoNCE does not have a proof). Our paper extends InfoNCE and provides an identifiability result in that case. Note that the extensions are also different: We weigh the differences of latent coordinates instead of the latents themselves. The TriCL paper provides a result on encoder output equivalence (similar to Roeder et al. (2021), meaning that all encoders optimizing the TriCL loss are linearly related, or, more precisely $f_1(x) = A f_2(x)$. However, the encoder's output is not related to ground-truth latents in any way. Our paper provides a result on the equivalence of the output with the ground-truth latents, meaning that $f \circ g(z) = Oz$. On one hand, our type of result subsumes the TriCL type of result. Moreover, our result is well-suited for the qualitative assessment of inferred latent features, which is not possible with the latter. The theoretical result from TriCL requires that an additional decorrelation regularizer completely vanishes, whereas our method does not have any such term.

We also show an experimental comparison between TriCL and AnInfoNCE below. First, we noticed a small discrepancy between the paper's definition of TriCL and its experimental implementation in how the loss is computed. Based on the used implementation, the scale parameter is not applied to all terms in the sum but only to half of them. We are not sure how how this discrepancy affects the experiments.

We compare the evolution of linear identifiability scores when training with TriCL or AnInfoNCE in our synthetic experiment in Fig. 12. We set the latent dimensionality to 20 and test two cases for the positive conditional distribution. (1) Uniform Λ set to 100 (top row); (2) Dual partitioned Λ, such that half of the dimensions have a low Λ of 15 ("style"-like dimensions) and the other half high Λ of 250 ("content"-like dimensions) (bottom row). In both cases, we observe that AnInfoNCE behaves much more stably compared to TriCL and reaches higher linear identifiability scores, especially when there is an anisotropic conditional.

D.5 Training Details for VAE Training in MNIST

The VAE consists of three fully connected layers with Leaky ReLU activations and is trained on MNIST for 50 epochs. The latent dimensionality d of the VAE is set to 10. We use the trained VAE to generate training images in the following way: We first sample a latent vector z from the marginal distribution and then sample a vector conditioned on z. Since both vectors are normalized to the sphere surface, we multiply them with $\sqrt{d}$ to project them to a Gaussian vector. Finally, we use the VAE decoder to generate images which we train the encoder with the AnInfoNCE loss on.

To provide an intuition for the quality of samples generated by the VAE, we show images generated from random samples from the latent space in Fig. 13 (left). For the highlighted sample in the top-right corner, we also show samples from the conditional distribution for two different Λ values: $\Lambda = 5$ (middle) and $\Lambda = 50$ (right).

D.6 Training details and additional experiments for C-3DIdent

The Causal 3D-Ident dataset (von Kügelgen et al., 2021) is an extension of the 3D-Ident dataset (Zimmermann et al., 2021) to six more classes. The training split has 252000 images and the test split has 25200 images. We use

the original code base from (von Kügelgen et al., 2021) with the original training hyperparameters. The latent dimensionality is ten and the latent space of C-3DIdent is a hypercube. Following von Kügelgen et al. (2021), we use cosine similarity as our similarity measure which restricts the inferred latents to a hypersphere. To enable our encoder learning all hypercube latents, we set the latent dimensionality to eleven. We train all models for 100K gradient steps.

E   Compute requirements

Synthetic experiments.   For our synthetic experiments, as described in Sections 4.1, 4.2 as well as in Appendix D, we used a single NVIDIA GeForce RTX 2080 Ti with 11GB of RAM for each experiment. We used 8 CPUs and 8 workers per experiment. The convergence time for each experiment varied, e.g. higher batch size or a lower concentration parameter $\hat{\mathbf{\Lambda}}$ generally imply faster convergence: We show the relationship between the batch size and the concentration parameter in Fig. 8. We set the run time for synthetic experiments to ten hours to make sure that all runs converge; the main experiments shown in Figs. 2 and 3 train within one hour to convergence.

Real-world experiments.   On CIFAR10, we trained the ResNet18 model on a single NVIDIA A100 GPU with 40GB RAM and 8 CPUs (with 8 workers). The training took about seven hours per run. The linear readout evaluation as well as the binary augmentations readout took about 30 minutes on the same compute setup.

On ImageNet, we trained a ResNet50 on 8 NVIDIA A100 GPUs with 40GB RAM and 64 CPUs. We set the number of workers to 16. The training took about 15 hours on this setup. The linear readout evaluation as well as the binary augmentations readout took about 3 hours using one GPU.

On C-3DIdent, we trained a ResNet18 model on a single NVIDIA A100 GPU with 40GB RAM and 32 CPUs (with 32 workers). The training took about fourteen hours per run.

Compute costs per experiment, main paper.   While we used different GPUs between our synthetic and real-world experiments, we will use a metric of generic "GPU-hours" for both cases. As described below, both GPU types require about the same amount of power.

  i) Synthetic data, Fig. 2: Experiments showing the influence of linear identifiability scores when varying $\mathbf{\Lambda}$. We show results for 6 values of $\mathbf{\Lambda}$ for InfoNCE and AnInfoNCE. The GPU run time for this experiment is $6 \cdot 2 \cdot 10h = 120h$.
  ii) Synthetic data, Fig. 3: Experiments showing how AnInfoNCE behaves in a slightly more difficult experimental setting on MNIST. We show results for 3 values of $\mathbf{\Lambda}$ for InfoNCE and AnInfoNCE in Fig 3 A+B, and for 2 values of $\mathbf{\Lambda}$ when training with AnInfoNCE in Fig 3 C+D. The GPU run time for this experiment is $3 \cdot 2 \cdot 10h + 2 \cdot 10h = 80h$.
  iii) Real-world data, Table 1: We show a trade-off between augmentations readout and downstream accuracy on CIFAR10 and ImageNet. We show results for training with InfoNCE and AnInfoNCE both on CIFAR10 and ImageNet. On CIFAR10, we additionally show results for training with AnInfoNCE without ℓ_2 normalization of the output before calculating the loss. We also conduct the linear readout evaluation on both the backbone and the post-projection features. The compute break-down is as follows:
     i) CIFAR10: Training: $3 \times 7h = 21h$, Linear readout: $3 \cdot 6 \cdot 0.5h \cdot 2 = 18h$. Total: $39h$
     ii) ImageNet: Training: $2 \times 15h \cdot 8$ GPUs$= 240h$, Linear readout: $2 \cdot 3h \cdot 6 \cdot 2 = 72h$. Total: $312h$
    The total compute for the real-world experiments is then 351 GPU hours.
  iv) MNIST, inset Table in Sec.5; showing the accuracy-identifiability trade-off. The run time for this experiment has been: $2 \cdot 10h=20h$.
  v) C-3DIdent, Tab. 2; we compare InfoNCE with AnInfoNCE on C-3DIdent and investigate the accuracy and identifiability trade-off. We tested 2 view generation strategies (via data augmentation and via ground-truth sampling) and ran 3 random seeds for two different loss functions. The total run time for this experiment has been: $2 \cdot 3 \cdot 2 \cdot 14h = 168h$.

Compute costs per experiment, Appendix.
  i) Synthetic data, Fig. 4. We show ablation experiments for varying the batch size, the latent dimensionality and the concentration. We show results for 32 different parameter combinations. The GPU run time for this experiment is $32 \cdot 10h = 320h$.
  ii) Synthetic data, Fig. 8. We analyze the interplay between higher dimensional latent spaces and batch size in

    more detail. We ran 3 random seeds in this experiment, analyzed 9 different batch sizes across 5 different latent dimensionalities. The total run time for this experiment is $3 \cdot 5 \cdot 9 \cdot 10h = 1350h$.

iii) Synthetic data, Fig. 9. We utilize the previous runs from computing results for Fig. 4; this Figure does not incur additional compute costs.

iv) Hard-negative mining, Fig. 10. We show that hard-negative mining can improve identifiability. We finetune one model trained with regular InfoNCE on AnInfoNCE with hard negative examples. The run time for this experiment is $3h$.

v) Synthetic data, Fig. 11. We show that ensembling can improve upon InfoNCE and AnInfoNCE. The run time for this experiment is $3 \cdot 10h = 30h$.

vi) Synthetic data, Fig. 12. We show that AnInfoNCE is more stable and reaches higher linear identifiability scores compared to the TriCL loss. The run times for this experiment is $2 \cdot 10h = 20h$

Total compute estimation. Given the previous paragraphs, the training time estimation is 2774 GPU hours. The maximum power consumption for the 2080Ti GPUs is 250W (Techpowerup, 2024) and 300W for the A100 GPUs (FOLDING.LAR.SYSTEMS, 2024). We assume an upper bound on power consumption of 300W per GPU. Then, the energy spent by the GPUs for the experiments shown in this paper is 832.2kWh. We estimate an additional factor of 20 in terms of energy spent during the development stage of this project. Therefore, the total estimated project cost in terms of energy is about 16.6MWh. This corresponds to 3.4t CO_2 emissions (Rensmart, 2024) which is roughly equivalent to six single-person flights from London to New York (Travel Navigator, 2024).

F   Software stack

We use different open source software packages for our experiments, most notably Docker (Merkel, 2014), Singularity (Sylabs Inc, 2018), scipy and numpy (Virtanen et al., 2020), PyTorch (Paszke et al., 2017), and torchvision (Marcel and Rodriguez, 2010).

G   Additional related works

G.1   An LVM view of SSL

Saunshi et al. (2019) developed generalization bounds for downstream classification in contrastive learning, with one or more negative samples, assuming the presence of latent class variables and (for some results considering intra-class concentration) sub-Gaussian conditionals. Results also show that more negative samples can hurt performance when the negative samples have the same class (c.f. their Sec. 4.2). Zimmermann et al. (2021) connected SimCLR to nonlinear ICA and proves that the SimCLR loss is equivalent to the cross entropy between ground-truth and approximate conditionals. Based on this result, identifiability results are provided for latent spaces structured as hyperspheres or convex bodies - our theory extends the results of Zimmermann et al. (2021). von Kügelgen et al. (2021) introduced the content-style terminology and proved that content variables can be recovered with non-contrastive SSL under certain assumptions. Daunhawer et al. (2023) extended these results to the multi-modal case where some latent variables are modality-specific. Lyu et al. (2021) extended von Kügelgen et al. (2021) to get guarantees on the identifiability of style variables as well, though they require invertible encoders, whereas von Kügelgen et al. (2021) have results for the non-invertible case. However, Lyu et al. (2021) admit different encoders across views. Lachapelle et al. (2022) investigated the role of sparsity for identifiability in the multi-task scenario, i.e., when the representation is used to solve multiple downstream tasks. This setting is similar to that in Dubois et al. (2022), since the authors used downstream tasks that rely on all latent factors, i.e., for good performance all of them need to be identified. Inducing sparsity in the linear classification head yields an optimal and disentangled representation with identifiability guarantees up to permutation and scaling. Cui et al. (2022), building on Zimmermann et al. (2021); von Kügelgen et al. (2021), introduced the AggNCE loss function and proves that with multiple views, the InfoNCE loss is identifiable up to an invertible matrix. Lee et al. (2021b) quantified how specific pretext tasks can guarantee learning a good representation, i.e., one which can solve the downstream task by only using a linear layer on top. The authors relied on approximate conditional independence between the input and the pretext target (conditioned on the downstream task label and the latents). Eastwood et al. (2023) proposed a new SSL objective inspired from invariance-based methods that allow for the disentanglement of style variables alongside content. This paper can also be seen as a more rigorous treatment of Xiao et al. (2021). However, their identifiability results differ from ours as they mostly rely on multiple sets of augmentations, and a different model and loss pair. **?** unified and extended the results from von Kügelgen et al. (2021); Daunhawer et al. (2023) to the multi-view case with partial observability, showing

that any shared information across arbitrary subsets of views and modalities can be learned up to a smooth bijection using contrastive learning.

G.2   The role of data augmentations

The (empirical) study of different augmentation strategies relates to our work as the underlying data generating process (DGP) is affected by them and our work studies SSL from the DGP's perspective. Within this category, we distinguish two subcategories:

G.2.1   Augmentation strategies with a single latent space

Wu et al. (2020) empirically evaluated collision-free augmentations (i.e., when different data points cannot have the same view—e.g., changing brightness is collision-free, but cropping can introduce collisions). They conclude that collision-free augmentations are not useful for the downstream task of classification on ImageNet (c.f. their Figs. 1 and 5) and CIFAR10 (their Fig. 5)—whereas the ones with collision can improve performance. **?** proposed a more flexible class of data augmentations based on Gaussian random fields and showed that it improves performance on ImageNet and iNaturalist, but their method is sensitive to hyperparameters (strong transformations can degrade the image). Bendidi et al. (2023) extensively studied the effect of data augmentations in natural (biological) images w.r.t. downstream (classification) performance and clustering structure. They emphasize that the choice of data augmentation strategies is a form of weak supervision, and its design requires domain expertise to achieve better performance on a given task, since different augmentations can affect the classes differently. Wagner et al. (2022) introduced the GroupAugment augmentation strategy that searches over a very general space of both augmentations and sampling strategies. The authors also empirically study the role of data augmentation hyperparameters, highlighting that it dramatically impacts performance in SSL. Ericsson et al. (2021) studied learning invariances in contrastive learning and their transfer from synthetic to real-world scenarios. The authors consider two augmentation categories: spatial augmentations change the objects' position, whereas appearance mostly affects color and texture. The authors empirically demonstrate that using augmentations from one of the above groups leads to increased invariance w.r.t. the same augmentation types—these only transfer to some extent from synthetic settings to the real world. However, there is a trade-off in all studied models, i.e., there is no reliable means to learn invariances to both augmentation types. Since different downstream tasks are shown to rely on different invariances, the authors advocate for a fusion of multiple representations, i.e., leveraging multiple representations that were trained by using different types of augmentations.

G.2.2   Augmentation strategies with combined latent spaces

Xiao et al. (2021) introduced LooC (Leave-one-out Contrastive Learning) using multiple embeddings, where each embedding is trained with a different (sub)set of augmentations. The authors argued and empirically demonstrated that by inducing different invariances in the different embeddings, the combined latent space outperforms all considered baselines on multiple downstream tasks. This setting corresponds to our ensemble approach, with the only difference that we share the encoder, thus, we only have a single latent space. Zhang and Ma (2022) proposed a similar strategy to LooC with an add-one augmentation strategy (i.e., their augmentation subsets can be ordered, where the first includes two augmentations, the second the first two plus another one, etc.). Furthermore, they calculate the contrastive loss at different hierarchical levels in the siamese encoder—the authors motivate this step by conjecturing that this way invariances are imposed at different levels. Another key difference is that the embeddings of augmentation parameters are also included in the representation before the projection head (i.e., the loss has information about the data augmentation).

G.3   The effect of normalization

Tian (2023, Lem. 2) shed light into why normalized representations are favorable: the author shows that the Frobenius norm of all weight matrices below an ℓ_2- or layer normalization Ba et al. (2016) terms remains constant. The theory holds for nonlinear MLP with reversible activations, including ReLU; and, interestingly, provides a connection of why we could feasibly replace ℓ_2-normalization with layer normalization. Wen and Li (2022) explained dimensionality collapse for non-contrastive SimSiam (Chen and He, 2021) with a two-layer nonlinear model and output normalization, and analyze the role of the projector. Empirically, when the projector is initialized to the identitiy matrix and only the off-diagonal entries are trainable, the authors obtain competitive representations. They also point out that batch normalization (by enforcing a nonzero variance) is less prone to collapse and conjecture that a complete collapse is more probable with ℓ_2-normalization but do not prove the

conjecture. Their results are in unison with ours since both works state that the content variables are learned first.

Ermolov et al. (2021) proposed a non-contrastive SSL loss where an MSE objective is applied to whitened features and show empirical improvements regarding dimensionality collapse; furthermore, their empirical results suggest that using the InfoNCE loss with ℓ_2-normalization and whitened features leads to divergence. Intuitively, there are some similarities between their normalization strategy and AnInfoNCE , though they are distinct as we normalize the latent factors before calculating the weighted MSE, whereas the authors whiten before normalization. Partially, this can explain why we do not have divergence in the case of ℓ_2-normalization and weighting via the diagonal scaling matrix.

Halvagal et al. (2022) proposed an isotropic loss to "flatten out" the eigenvalue spectrum of the inferred latent covariance, which is remarkably similar to our strategy of using different temperature values.

G.4   The role of negatives and hard negative sampling

Saunshi et al. (2019) showed that more negative samples can hurt performance when the negative samples have the same class. Chuang et al. (2020) made the same observation, i.e., that sampling negative examples truly different (downstream) labels improves performance in a synthetic setting where labels are available. They uncover a NPC in InfoNCE and propose to resolve it with debiasing, resulting in DCL that corrects for the sampling of same-label datapoints, even without knowledge of the true labels. They also bound the difference between the SimCLR and the debiased contrastive losses. Robinson et al. (2021b) extended Chuang et al. (2020) and propose a strategy to draw hard negative samples in an unsupervised way—using the data density and positive conditional distribution—for contrastive learning and analyze the resulting loss theoretically. The authors connect hard negative samples to adversarial samples and analyze the loss in the infinite sample limit in (c.f. their Thm. 4). They also bound the misclassification risk in their Thm. 5. Liu and Wang (2023) extended (Chuang et al., 2020) into the Bayesian framework; the authors propose a hard negative sampling strategy with importance sampling.

Several works tackle the problem of designing (hard) negative sampling strategies. Wu et al. (2020) generalized the negative sampling in InfoNCE, their method include more "exotic" negative sampling schemes, e.g., from a "ring" on a sphere. Robinson et al. (2021a) proposed IFM as a solution that disturbs both positive and negative pairs, though in an implicit manner (it can be thought of as implicit hard negative sampling): it makes positive samples less, negative samples more similar (measured by cosine similarity) to the anchor point. This perturbation is done in the latent space (c.f. Fig. 4 of their paper) and improves linear readout accuracy on multiple image data sets (Fig. 6; without the trade-offs induced by changing τ) and also improves performance on medical data (Tab. 2). The authors also demonstrate that SimCLR combined with IFM improves robustness in the adversarial setting (Fig. 8).

Chuang et al. (2022) proposed RINCE, a symmetric drop-in replacement for InfoNCE, with a theoretical connection to robust symmetric losses in noisy binary classification. The loss (approximately) takes the qth exponent of the positive and negative loss terms, implementing a trade-off between exploration and exploitation. In the limit of $q = 0$, it is the same as InfoNCE. Thus, the authors conclude, InfoNCE puts more weight on hard positives, whereas RINCE on easy positives. RINCE can be thought as an implicit hard/easy positive mining scheme with an exponential weighting factor on uniformity.
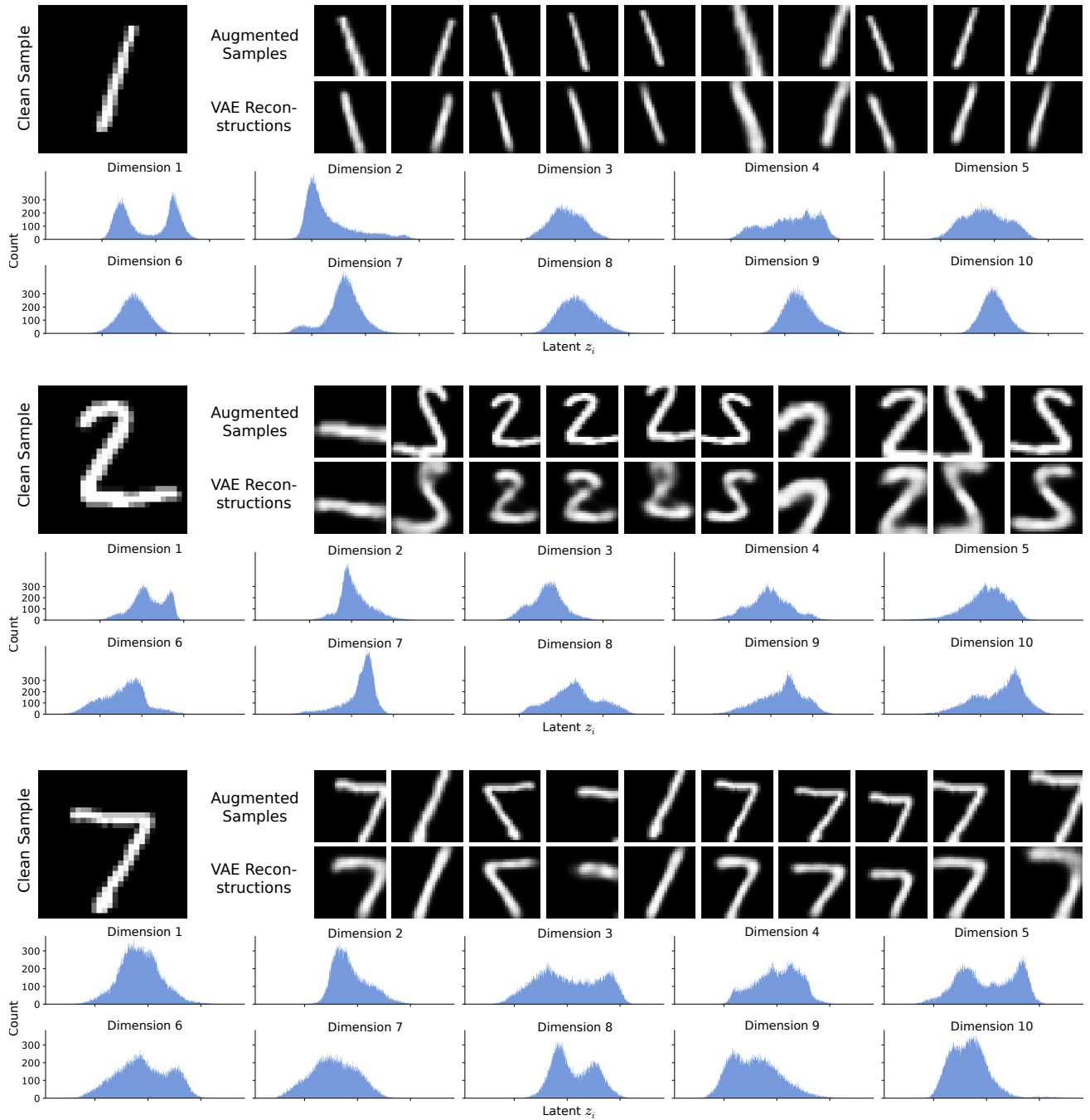
Figure 7: **When Using Augmentations to Generate Views for CL, the Conditional Generally no Longer Follows a Gaussian Distribution.** We show augmented images as well as their reconstructions using a VAE trained on augmented images. We project the augmented images into the latent space of the trained VAE and observe that their distribution does not resemble a Gaussian distribution and can even be bimodal.
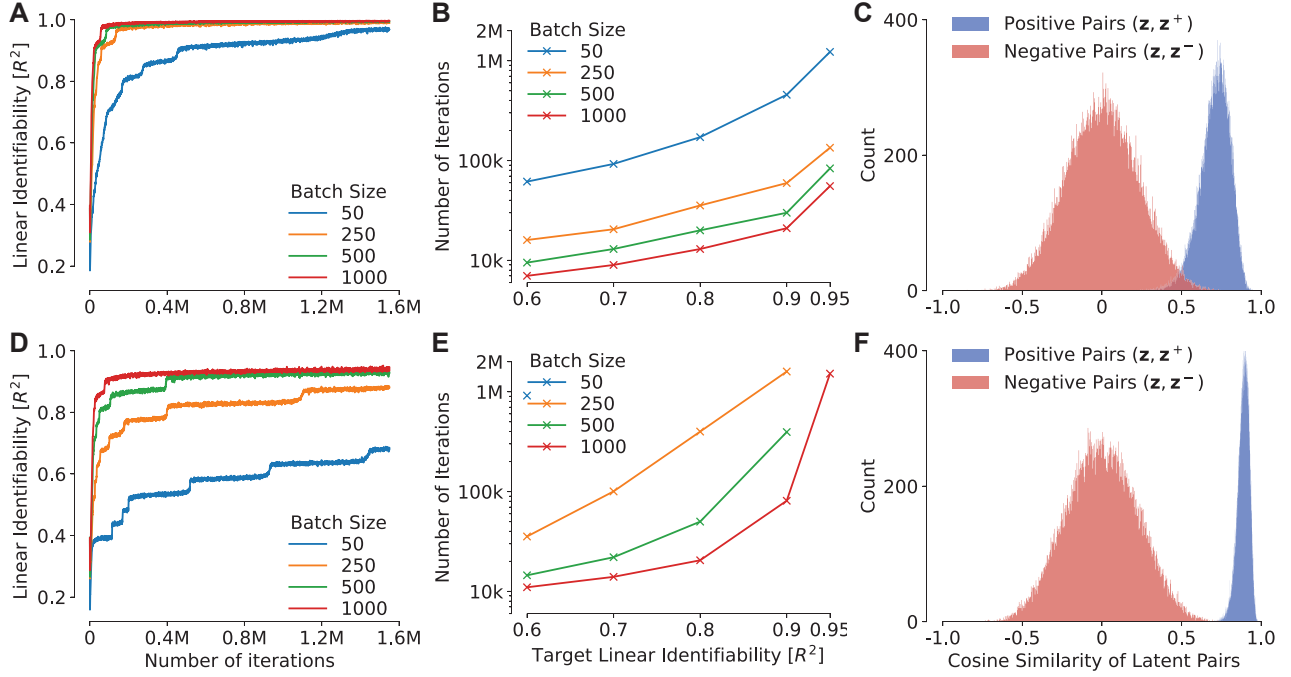
Figure 8: **Analysis of Learning Dynamics for a Small ($\Lambda = 25$, A-C) and a Large ($\Lambda = 75$, D-F) Concentration Parameter. A & D**: Linear identifiability measured by the R^2 scores show step-like behavior and generally need more training iterations to reach a certain R^2 score for higher Λ. **B & E**: The number of training iterations needed for a certain target R^2 score grows exponentially. **C & F**: We explain degrading R^2 scores for higher concentration parameters Λ with missing overlap in the latent positive and negative distributions.
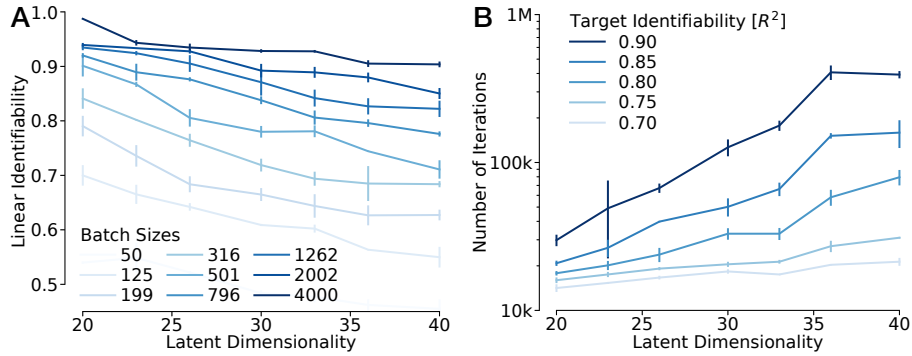
Figure 9: **Higher Dimensional Latent Spaces Require Longer Training with Larger Batch Sizes. A**: The linear identifiability achieved after training for 500k steps decreases with an increasing latent dimensionality but increases with larger batch sizes. **B**: We display the required number of iterations to achieve various target linear identifiability scores (0.70 to 0.90) for a fixed batch size of 4000. Note that the number of iterations required grows exponentially with the latent dimensionality.
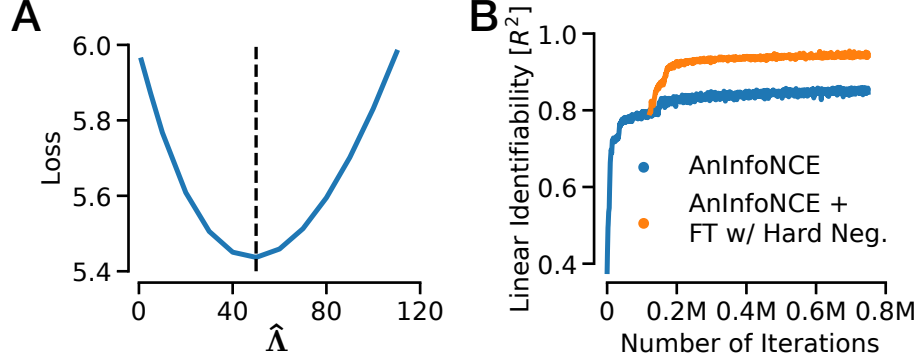
Figure 10: **Hard Negative Mining Improves Identifiability. A:** For $\mathbf{\Lambda}^+ = 150$ and $\mathbf{\Lambda}^- = 100$, we find a loss optimum at $\mathbf{\Lambda} = \mathbf{\Lambda}^+ - \mathbf{\Lambda}^- = 50$ as predicted by Cor. 3.1; **B:** Finetuning (FT) an encoder on a concatenation of hard negative samples and regular negative samples improves the identifiability score.
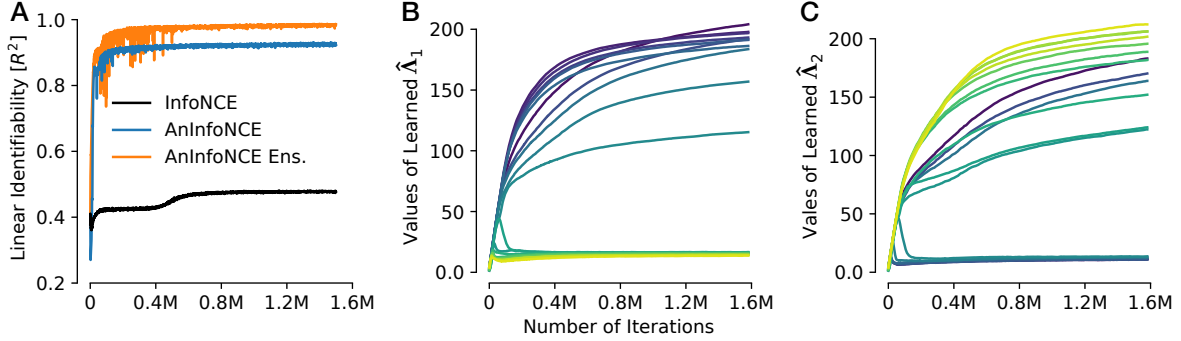

Figure 11: **Ensemble AnInfoNCE Improves Upon InfoNCE and AnInfoNCE:** We observe higher linear identifiability scores when training with Ensemble AnInfoNCE. When examining the learned $\hat{\Lambda}$ values, we see a specialization in the learned $\hat{\Lambda}$ values: The color scheme is chosen such that the same color indicates the same dimension, and we find that the learned $\hat{\Lambda}$-values for the two DGPs are anti-correlated with respect to each other.
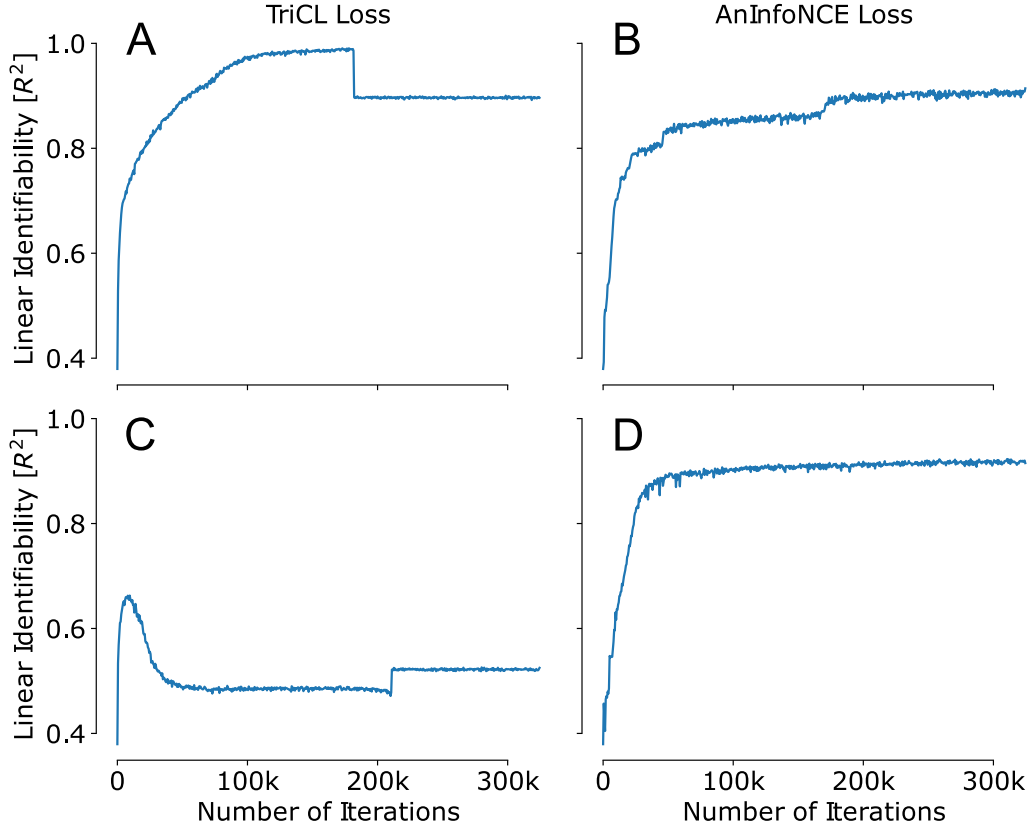
Figure 12: **Synthetic Experiments for a Latent Dimensionality of d=20 When Training With TriCL or AnInfoNCE.** Top row: Uniform Λ of 100. Bottom row: Dual partitioned Λ, such that half of the dimensions are varied according to a low Λ of 15 and the other half is varied with a high Λ of 250. In both cases, we observe that AnInfoNCE behaves much more stably compared to TriCL and reaches higher linear identifiability scores, especially when there is an anisotropic positive conditional.
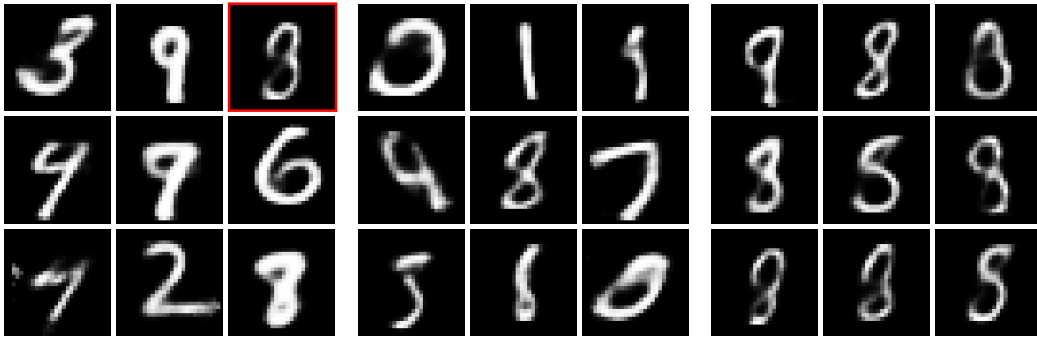

Figure 13: **Samples Generated by the MNIST-VAE.** **Left**: Samples from the marginal distribution, used as anchor samples for the contrastive objective. **Middle & Right**: For the highlighted sample, we show potential positive samples from the conditional distribution for two different values of Λ. A value of 5 leads to too much variation (middle) and leads to obvious class changes in the conditional samples, whereas a value of 50 leads to reasonable variation. We note that training an encoder on images which were sampled according to a DGP with either ground-truth Λ-value leads to perfect linear disentanglement scores. In other words, a low concentration parameter of $\Lambda = 5$ which changes the class of the conditional view does not pose an issue.