
Perfectly Balanced: Improving Transfer and Robustness of Supervised Contrastive Learning

Mayee F. Chen^{*1} Daniel Y. Fu^{*1} Avanika Narayan¹ Michael Zhang¹
Zhao Song² Kayvon Fatahalian¹ Christopher Ré¹

Abstract

An ideal learned representation should display transferability and robustness. Supervised contrastive learning (SupCon) is a promising method for training accurate models, but produces representations that do not capture these properties due to class collapse—when all points in a class map to the same representation. Recent work suggests that “spreading out” these representations improves them, but the precise mechanism is poorly understood. We argue that creating spread alone is insufficient for better representations, since spread is invariant to permutations within classes. Instead, both the correct degree of spread and a mechanism for breaking this invariance are necessary. We first prove that adding a weighted class-conditional InfoNCE loss to SupCon controls the degree of spread. Next, we study three mechanisms to break permutation invariance: using a constrained encoder, adding a class-conditional autoencoder, and using data augmentation. We show that the latter two encourage clustering of latent subclasses under more realistic conditions than the former. Using these insights, we show that adding a properly-weighted class-conditional InfoNCE loss and a class-conditional autoencoder to SupCon achieves 11.1 points of lift on coarse-to-fine transfer across 5 standard datasets and 4.7 points on worst-group robustness on 3 datasets, setting state-of-the-art on CelebA by 11.5 points.

1. Introduction

Learning a representation with a favorable geometry is a critical challenge for modern machine learning. Good

^{*}Equal contribution ¹Department of Computer Science, Stanford University ²Adobe Research. Correspondence to: Mayee F. Chen, Daniel Y. Fu <{mfchen, danfu}@cs.stanford.edu>.

geometries can engender strong downstream transfer performance and robustness to subgroup imbalances, whereas poor geometries may have low transferability and be brittle (Islam et al., 2021; Sohoni et al., 2020). However, producing—or even characterizing—a good geometry can be difficult.

We focus on the challenges in doing so with supervised contrastive learning (SupCon). SupCon is a promising method for training accurate machine learning models (Khosla et al., 2020), but suffers from class collapse—wherein each point in the same class has the same representation, as in Figure 1 far left (Graf et al., 2021). Collapsed representations cannot distinguish fine-grained details within classes—in particular *latent subclasses*—resulting in poor transferability and robustness. Modifications to SupCon that heuristically “spread out” its representations have shown empirical promise (Islam et al., 2021), but a precise understanding of spread—how separated individual points are in representation space—and how to control it is lacking.

Furthermore, spread alone is not sufficient to explain improved representations. We observe that modifications to SupCon that increase spread are invariant to *class-fixing permutations*. That is, the loss value does not change when points of the same class are arbitrarily permuted in representation space. For example, Figure 1 right visualizes two geometries that both have spread but differ in representation quality, as suggested by the significant gap in transfer learning performance (35.4 points). Thus, while spread may be important, another mechanism is needed to break class-fixing permutation invariance for good performance.

We argue that these are the two key challenges to improving SupCon’s representations: creating the correct degree of spread, and breaking class-fixing permutation invariance. This paper makes progress on these challenges.

Challenge 1: Balancing Spread. We first prove a simple result that a class-collapsed representation cannot have good transfer performance, which motivates spread. We then analyze whether L_{spread} , a loss function that combines SupCon with a class-conditional InfoNCE loss, can induce spread.

We find that previous approaches for analyzing contrastive losses encounter a technical challenge because SupCon

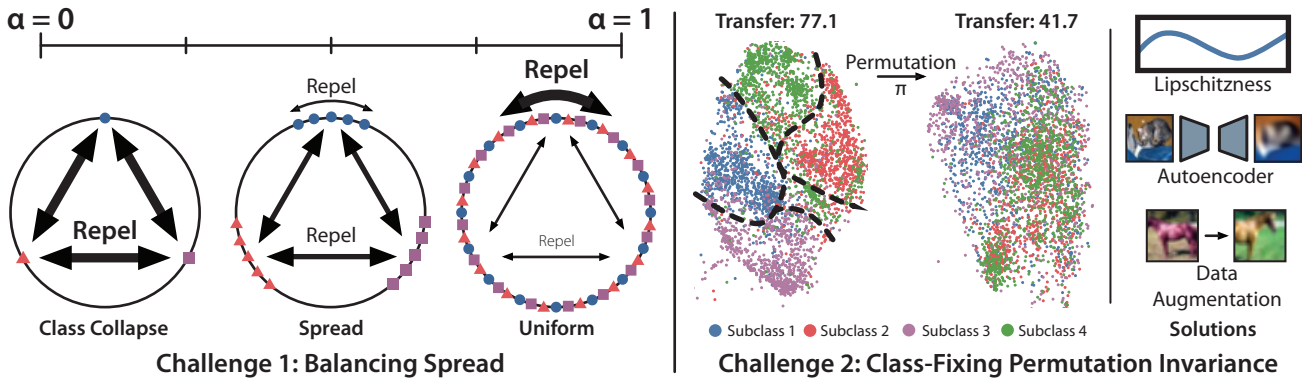


Figure 1. There are two key challenges to achieving better representations with SupCon. Left: The first challenge is balancing multiple contrastive loss terms with competing geometries. We show that adding a weighted class-conditional InfoNCE term can balance the geometries and induce spread in the representation geometry. Right: The second challenge is that representation geometries may be invariant to class-fixing permutations. The two example geometries shown both have spread, but vastly different coarse-to-fine transfer performance. We analyze three mechanisms for addressing this challenge: constraining the encoder, adding a class-conditional autoencoder, and using data augmentation. Best viewed in color.

and InfoNCE have incompatible optimal geometries (class collapse and uniformity on the hypersphere, respectively). For example, Wang & Isola (2020) analyze individual loss components in isolation, but doing so risks drawing misleading conclusions when the loss components are incompatible. Further, finding exact solutions to optimization problems on the hypersphere is fundamentally difficult; a classic example is the Thomson problem (Thomson, 1897), which has evaded an exact solution after a century of study.

We bypass these problems by constructing a distribution that is neither collapsed nor uniform and analyzing its loss. We introduce $s_f(y)$, a notion of class variance, to measure spread on this distribution. We show that this distribution has an intermediate degree of spread by deriving bounds for the weight α on the class-conditional InfoNCE loss within which this distribution attains lower loss than either extreme. While this result does not fully characterize the geometry, it suggests that setting α properly can induce an optimal distribution with appropriate spread—which we validate with measurements on CIFAR10.

Challenge 2: Breaking Permutation Invariance. Our first result demonstrates that L_{spread} can induce spread but does not give insight on class-fixing permutation invariance. We formally define class-fixing permutation invariance and prove that L_{spread} is subject to it absent other interventions.

This motivates the question: how should we break class-fixing permutation invariance? We show that inducing an inductive bias towards clustering of latent subclasses can break permutation invariance—and more importantly, can result in good coarse-to-fine transfer performance. We introduce $\sigma_f(z)$, a measure of subclass clustering in representation space, and show that coarse-to-fine

generalization error scales with $\sigma_f(z)/s_f(y)$.

A standard approach to controlling $\sigma_f(z)$ is assuming Lipschitzness of the model. However, Lipschitzness is a strong assumption for modern deep networks, which are powerful enough to memorize random noise (Zhang et al., 2016). In empirical measurements, we find that modern deep networks display poor Lipschitzness, and thus the Lipschitzness assumption is insufficient for inducing clustered subclass representations.

We thus propose two alternatives that can bound $\sigma_f(z)$ under more realistic assumptions: directly encoding fine-grained details by concatenating the representations from a class-conditional autoencoder, and using data augmentation in the class-conditional InfoNCE loss. The former only requires a “reverse Lipschitz” decoder to upper bound $\sigma_f(z)$, and can do so by a constant factor tighter than a general (non-conditional) autoencoder. The latter only requires the encoder to be Lipschitz over data augmentations to induce subclass clustering—and can also explain observations from prior work (Islam et al., 2021). We validate these findings by measuring Lipschitzness constants and $\sigma_f(z)/s_f(y)$ on real data; we find that these alternate assumptions are more realistic than overall Lipschitzness, and that data augmentation and autoencoders help induce subclass clustering.

Empirical Validation Using our theoretical insights, we propose THANOS: adding a class-conditional InfoNCE loss and a class-conditional autoencoder to SupCon. We evaluate THANOS on two tasks designed to evaluate how well it preserves subclasses:

- **Coarse-to-fine transfer learning** trains a model to classify superclasses but use the representations to distinguish subclasses. THANOS outperforms SupCon by

11.1 points on average across 5 standard datasets.

- **Worst-group robustness** evaluates how well a model can identify underperforming sub-groups and maintain high performance on them. THANOS identifies underperforming sub-groups 7.7 points better than previous work (Sohoni et al., 2020) and achieves 4.7 points of lift on worst-group robustness across 3 datasets, setting state-of-the-art on CelebA by 11.5 points. THANOS can even outperform GroupDRO (Sagawa et al., 2019), a state-of-the-art robustness algorithm that uses *ground-truth* sub-group labels.

2. Background

Section 2.1 presents our data model and the coarse-to-fine transfer task. Section 2.2 presents L_{spread} , a simple variant of SupCon that adds a weighted class-conditional InfoNCE loss. Section 2.3 discusses geometry of contrastive losses.

2.1. Data Setup

Input data $x \in \mathcal{X}$ are drawn from a distribution \mathcal{P} with deterministic class $y = h(x)$, where $y \in \mathcal{Y} = \{0, \dots, K-1\}$. We assume that the data is class-balanced such that $\Pr(y=i) = \frac{1}{K}$ for all $i \in \mathcal{Y}$.

Data points also belong to *latent subclasses*. Following Sohoni et al. (2020), we denote a subclass as a latent discrete variable $z \in \mathcal{Z}$. \mathcal{Z} can be partitioned into disjoint subsets S_0, \dots, S_{K-1} such that if $z \in S_k$, then its corresponding y label is equal to k . For simplicity, we assume that there are two subclasses for each label k , e.g. $|S_k| = 2$. The data generating process proceeds as follows: first, the latent subclass z is sampled with proportion $p(z)$. Then, x is sampled from the distribution $\mathcal{P}_z = p(\cdot|z)$, and its corresponding deterministic label is denoted $y = S(z)$. Let $h_s(x) : \mathcal{X} \rightarrow \mathcal{Z}$ denote x 's subclass.

We have a class-balanced labeled training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ where points (x_i, z_i, y_i) are drawn i.i.d, and the value of each z_i is unknown during training time. Denote $\mathcal{D}_y = \{x \in \mathcal{D} : h(x) = y\}$ and $\mathcal{D}_z = \{x \in \mathcal{D} : h_s(x) = z\}$, and denote their sizes by $n_y = |\mathcal{D}_y|$ and $n_z = |\mathcal{D}_z|$.

Contrastive learning trains an encoder $f : \mathcal{X} \rightarrow \mathbb{R}^d$ on \mathcal{D} that maps inputs to representations in an embedding space \mathbb{R}^d .

Coarse-to-Fine Transfer Coarse-to-fine transfer evaluates how well an embedding trained on coarse classes \mathcal{Y} distinguishes fine classes (subclasses) \mathcal{Z} . Fix a y and smake-uppose that $S_y = \{z, z'\}$. The task is to classify z versus z' using the encoder f learned on \mathcal{D} . We are given a dataset of subclass labels, $\mathcal{D}_s = \{(x_i, z_i)\}_{i=1}^m$. Denote $\mathcal{D}_{s,z} = \{x \in \mathcal{D}_s : h_s(x) = z\}$ and $m_z = |\mathcal{D}_{s,z}|$. We learn linear weights $W_z, W_{z'} \in \mathbb{R}^d$ and construct an estimate $\hat{p}(z|f(x))$ by using softmax scores $\hat{p}(z|f(x)) = \frac{\exp(f(x)^\top W_{h_s(x)})}{\exp(f(x)^\top W_z) + \exp(f(x)^\top W_{z'})}$, where f is fixed. We use the mean classifier to construct

W , following prior work (Arora et al., 2019). That is, $W_z = \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f(x)$, and $W_{z'}$ is similarly defined.

We evaluate the performance of coarse-to-fine transfer with a γ -margin loss, defined on a point (x, z) as

$$\ell_{\gamma, f}(x, z) = 1 - \mathbb{1}\{\hat{p}(z|f(x)) \geq \gamma \hat{p}(z'|f(x))\} \quad (1)$$

for $\gamma > 1$. That is, we want the model to output the correct subclass label at least γ times more likely than the incorrect one. Define the γ -margin generalization error on z as $L_{\gamma, f}(z) = \mathbb{E}_{x \sim \mathcal{P}_z} [\ell_{\gamma, f}(x, z)]$.

2.2. A Modified Supervised Contrastive Loss

Contrastive learning trains an encoder to produce representations of the data by pushing together similar points (positive pairs) and pulling apart different points (negative pairs). We consider L_{spread} , a weighted sum of a supervised contrastive loss L_{sup} (Khosla et al., 2020) and a class-conditional InfoNCE loss L_{cNCE} .

Let B be a batch of data from \mathcal{D} . Define $P(i, B) = \{x^+ \in B \setminus i : h(x^+) = h(x_i)\}$ as the points with the same label as x_i and $N(i, B) = \{x^- \in B \setminus i : h(x^-) \neq h(x_i)\}$ as points with a different label. Let $a(x_i)$ be an augmentation of x_i , and assume that augmentations of each sample are disjoint. Denote $\sigma_f(x, x') = \exp(f(x)^\top f(x') / \tau)$ with temperature hyperparameter τ . For $\alpha \in [0, 1]$, $\hat{L}_{\text{spread}}(f, x, B)$ on x belonging to B is:

$$\hat{L}_{\text{spread}}(f, x, B) = (1 - \alpha) \hat{L}_{\text{sup}}(f, x, B) + \alpha \hat{L}_{\text{cNCE}}(f, x, B),$$

where

$$\hat{L}_{\text{sup}}(f, x_i, B) = -1/|P(i, B)| \times \quad (2)$$

$$\sum_{x^+ \in P(i, B)} \log \frac{\sigma_f(x_i, x^+)}{\sigma_f(x_i, x^+) + \sum_{x^- \in N(i, B)} \sigma_f(x_i, x^-)},$$

$$\hat{L}_{\text{cNCE}}(f, x_i, B) = -\log \frac{\sigma_f(x_i, a(x_i))}{\sum_{x^+ \in P(i, B)} \sigma_f(x_i, x^+)}. \quad (3)$$

The overall loss $\hat{L}_{\text{spread}}(f, B)$ is averaged over all points in B . L_{sup} is a variant of the SupCon loss (Khosla et al., 2020). L_{cNCE} is a class-conditional version of the InfoNCE loss, where the positive distribution consists of augmentations and the negative distribution consists points from the same class, intuitively encouraging them to be spread apart.

2.3. Geometries of Contrastive Losses

We present a series of standard theoretical assumptions for analyzing contrastive geometry, and define two important distributions—class collapse and class uniformity.

Assumptions We make several standard theoretical assumptions (Graf et al., 2021; Wang & Isola, 2020; Robinson et al., 2020): 1) restrict the encoder f 's output space to be \mathcal{S}^{d-1} , the unit hypersphere (i.e. normalized outputs); 2) assume that $K \leq d+1$, such that a K -regular simplex inscribed in \mathcal{S}^{d-1} exists; 3) assume that the encoder is *infinitely powerful*,

meaning that any distribution on \mathcal{S}^{d-1} is realizable by $f(x)$. We define the pushforward measure of the class-conditional distribution of $p(\cdot|h(x) = y)$ via f as μ_y for $y \in \mathcal{Y}$, where $\mu_y \in \mathcal{M}(\mathcal{S}^{d-1})$ is over all Borel probability measures on the hypersphere. Define $\boldsymbol{\mu} = \{\mu_y\}_{y \in \mathcal{Y}}$ as the overall pushforward measure corresponding to $\mathcal{P} \circ f^{-1} \in \mathcal{M}(\mathcal{S}^{d-1})$.

Class Collapse Distribution Define $\mathbf{v} = \{v_y\}_{y \in \mathcal{Y}} \in \mathcal{S}^{d-1}$ as the set of vectors forming the regular simplex inscribed in the hypersphere, satisfying: a) $\sum_y v_y = \vec{0}$; b) $\|v_y\|_2 = 1 \forall y$; and c) $\exists c_K \in \mathbb{R}$ s.t. $v_y^\top v_{y'} = c_K$ for $y \neq y'$. Let δ_{v_y} be the probability measure on \mathcal{S}^{d-1} with all mass on v_y , and let $\boldsymbol{\delta}_v = \{\delta_{v_y}\}_{y \in \mathcal{Y}}$ be the *class-collapsed measure* such that $\mu_y = \delta_{v_y}$ and $f(x) = v_y$ almost surely whenever $h(x) = y$. Graf et al. (2021) show that $\boldsymbol{\mu} = \boldsymbol{\delta}_v$ minimizes the SupCon loss.

Class Uniform Distribution Denote σ_{d-1} as the normalized surface area measure on \mathcal{S}^{d-1} . $\boldsymbol{\mu} = \boldsymbol{\sigma}_{d-1}$ is the *class-uniform measure* when $\mu_y = \sigma_{d-1}$ for all $y \in \mathcal{Y}$. Wang & Isola (2020) show that σ_{d-1} minimizes the InfoNCE loss.

3. Controlling Spread

In Section 3.1, we demonstrate the importance of spread—having distinguishable representations of points in a class—by showing that SupCon results in poor coarse-to-fine transfer. In Section 3.2, we begin to explore whether L_{spread} can result in more spread out geometries. We define the asymptotic form of L_{spread} and apply the approach from Wang & Isola (2020) to analyze individual loss terms. We find that the optimal geometries of the individual terms are incompatible. In Section 3.3, we analyze the asymptotic L_{spread} as a whole using a nuanced approach that compares the loss over different geometries. We conclude that the optimal geometry is neither class-collapsed nor class-uniform for a range of α . This result suggests that spread can be carefully controlled, and we capture this property by introducing a notion of intra-class variance, $s_f(y)$. All proofs for the paper are in Appendix C.

3.1. The Importance of Spread

SupCon exhibits class collapse on the training data and does not spread out representations in a class. We use standard generalization bounds and show that this geometry results in poor coarse-to-fine generalization error: asymptotically, the error obtains its maximum possible value.

Define $f_{SC} \in \mathcal{F}$ to be the encoder trained with SupCon satisfying class collapse, $f_{SC}(x) = v_y$ for all $x \in \mathcal{D}$ where $h(x) = y$. Let $f(x)[j]$ be the j th entry of $f(x)$. For function class \mathcal{F} , let $\mathcal{F}_j = \{f(\cdot)[j] : f \in \mathcal{F}\}$ be the elementwise class. Let $\mathfrak{R}_n(\mathcal{F}_j)$ denote \mathcal{F}_j 's Rademacher complexity on n samples, and define $\mathfrak{R}_n(\mathcal{F}) = \sum_{j=1}^d \mathfrak{R}_n(\mathcal{F}_j)$.

Theorem 1. For γ where $\log \gamma \geq 8 \max\{\mathfrak{R}_{n_z}(\mathcal{F}), \mathfrak{R}_{n_{z'}}(\mathcal{F})\}$,

SupCon's coarse-to-fine error is at least

$$L_{\gamma, f_{SC}}(z) \geq 1 - \delta(n_z, \mathcal{F}, \gamma) - \delta(n_{z'}, \mathcal{F}, \gamma) - \xi(m_z \wedge m_{z'}, \gamma),$$

where $\delta(n_z, \mathcal{F}, \gamma) = d \exp\left(-\frac{n_z}{32d^2}(\log \gamma - 8\mathfrak{R}_{n_z}(\mathcal{F}))^2\right)$ bounds generalization error of f_{SC} and $\xi(m_z \wedge m_{z'}, \gamma) = 4d \exp\left(-\frac{(m_z \wedge m_{z'}) \log^2 \gamma}{32d}\right)$ bounds the noise from \mathcal{D}_s .

As $n \wedge m$ increases, error approaches 1—its maximum value—and the model will almost surely predict the correct subclass γ times less often than the incorrect one. This result motivates studying whether L_{spread} can encourage spread.

3.2. Asymptotic L_{spread}

We present the asymptotic version of L_{spread} . For a given anchor $x \sim \mathcal{P}$, define a positive pair $x^+ \sim p(\cdot|h(x^+) = h(x))$ from the same class and a negative pair using $x^- \sim p(\cdot|h(x^-) \neq h(x))$ from a different class. Let $a(x)$ be an augmentation of x drawn from a distribution $p_a(\cdot|x)$, where each $p_a(\cdot|x)$ has disjoint support.

Definition 1. Define $L_{\text{spread}}(f, \alpha)$ as

$$L_{\text{spread}}(f, \alpha) = (1 - \alpha)L_{\text{align}}(f) + \alpha L_{\text{aug}}(f) + (1 - \alpha)L_{\text{diff}}(f) + \alpha L_{\text{same}}(f), \quad (4)$$

where

$$\begin{aligned} L_{\text{align}}(f) &= \mathbb{E}_{x, x^+} [\|f(x) - f(x^+)\|^2 / 2\tau] \\ L_{\text{aug}}(f) &= \mathbb{E}_{x, a(x)} [\|f(x) - f(a(x))\|^2 / 2\tau] \\ L_{\text{diff}}(f) &= \mathbb{E}_x [\log \mathbb{E}_{x^-} [\exp(-\|f(x) - f(x^-)\|^2 / 2\tau)]] \\ L_{\text{same}}(f) &= \mathbb{E}_x [\log \mathbb{E}_{x^+} [\exp(-\|f(x) - f(x^+)\|^2 / 2\tau)]] \end{aligned}$$

The derivation of $L_{\text{spread}}(f, \alpha)$ is in Appendix C.1. Next, we analyze individual terms, similar to Wang & Isola (2020)'s approach. For simplicity, we present the binary setting $K = 2$. We abuse notation and use f and $\boldsymbol{\mu}$, the pushforward measure of x on the hypersphere, interchangeably in $L_{\text{spread}}(f, \alpha)$ as well as in the loss components in Definition 1.

Proposition 1 (Individual losses). $L_{\text{align}}(f)$ and $L_{\text{aug}}(f)$ are minimized when $f(x) = f(x^+)$ and $f(x) = f(a(x))$ almost surely, respectively. $L_{\text{diff}}(\boldsymbol{\mu})$ is minimized when $\boldsymbol{\mu} = \boldsymbol{\delta}_v$. $L_{\text{same}}(\boldsymbol{\mu})$ is minimized when $\boldsymbol{\mu} = \boldsymbol{\sigma}_{d-1}$.

When $\alpha = 0$, the ‘‘active’’ loss terms are L_{align} and L_{diff} , whose optima are jointly realizable and yield $\boldsymbol{\mu} = \boldsymbol{\delta}_v$ overall. When $\alpha = 1$, the terms L_{aug} and L_{same} are also compatible, yielding $\boldsymbol{\mu} = \boldsymbol{\sigma}_{d-1}$ and augmentations with the same embedding as their original point.

Neither of these distributions has good coarse-to-fine transfer performance on its own: $\boldsymbol{\delta}_v$ loses information within classes, and $\boldsymbol{\sigma}_{d-1}$ allows points of different classes to be close together (Figure 1 left). To avoid both $\boldsymbol{\delta}_v$ and $\boldsymbol{\sigma}_{d-1}$, $\alpha \in (0, 1)$ must achieve a balance between the two loss terms. But the behavior of the weighted loss overall is unclear from the result in Proposition 1. It is also unclear whether there even exists an intermediate distribution that minimizes $L_{\text{spread}}(\boldsymbol{\mu}, \alpha)$.

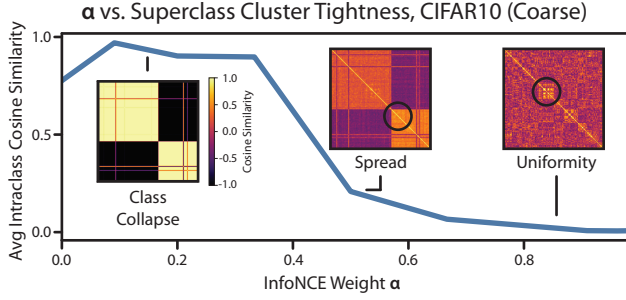


Figure 2. Measure of cluster tightness vs. α . Clusters are collapsed for low values of α , display spread for a small region, and then dissolve into uniformity for high values of α . Inserts: heatmaps of cosine similarity between points, sorted by class and subclass. Circles: apparent subclass clusters within spread.

3.3. Our Spread Result

We seek to analyze the geometry of the overall loss. Explicitly characterizing the optimal geometry is challenging, so we design a family of measures on the hypersphere and examine when such measures obtain lower loss than collapsed or uniform measures. We perform analysis for $K = 2$ and consider $K = 3$ in Appendix D. Synthetic experiments are in Appendix H.

The measure we study, μ_θ , assigns mass evenly on two points that are close to v_y , a vertex of the regular simplex, but separated by some angle θ for each μ_y (see Figure 4 in Appendix H). Formally, define a block-diagonal rotation matrix $R_\theta \in \mathbb{R}^{d \times d}$ consisting of submatrices $\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$ and I_{d-2} on the diagonal. For $\theta \in (0, \pi/2]$, define the measure $\mu_\theta = \{\mu_{0,\theta}, \mu_{1,\theta}\}$, where $\mu_{0,\theta} = \frac{1}{2}\delta_{R_\theta v_0} + \frac{1}{2}\delta_{R_{-\theta} v_0}$, and similarly $\mu_{1,\theta} = \frac{1}{2}\delta_{R_\theta v_1} + \frac{1}{2}\delta_{R_{-\theta} v_1}$. We present a technical result on the range of α for which μ_θ attains lower loss than class-collapsed or class-uniform measures.

Theorem 2. Let $c_{\tau,d} = \frac{2 + \frac{1}{\tau} - \sqrt{\frac{1}{\tau}(-2 + \frac{1}{\tau}) - 2\log W_{1/2\tau}(S^{d-1})}}{3}$, where $W_{1/2\tau}(S^{d-1})$ is a constant depending on τ and d (see Appendix C.1 for exact value). Then, when $\alpha \in (2/3, c_{\tau,d})$, $\theta^* = \arcsin \sqrt{\frac{\tau}{2} \log \frac{3\alpha-1}{3-3\alpha}}$ minimizes $L_{\text{spread}}(\mu_\theta, \alpha)$ and satisfies $L_{\text{spread}}(\mu_{\theta^*}, \alpha) \leq \min_{\mu \in \{\delta_v, \sigma_{d-1}\}} L_{\text{spread}}(\mu, \alpha)$.

Our result does not define the exact optimal geometry since it constrains the measures we optimize to be over μ_θ . For $\alpha \notin (2/3, c_{\tau,d})$, it also does not specify the optimal geometry—we only know that the optimal geometry is not of form μ_θ .

However, our result yields a high-level insight: there exists a range of α for which the optimal geometry that minimizes $L_{\text{spread}}(\mu, \alpha)$ spreads out points on the hypersphere. Concretely, define the spread of class y under f as $s_f(y) = \mathbb{E}_{h(x)=y} [\|f(x) - \mathbb{E}_{h(x)=y}[f(x)]\|]$.

Corollary 1. If $\alpha \in (2/3, c_{\tau,d})$ and $f(x)$ has measure μ_{θ^*} ,

the spread for y under f is $s_f(y) = \sqrt{\frac{\tau}{2} \log \frac{3\alpha-1}{3-3\alpha}} \sim \omega(1)$.

In other words, L_{spread} can yield an extent of spread $s_f(y)$ that is controlled by α . Experiments on CIFAR10 support our result (Figure 2); the geometry is collapsed for low values of α , followed by a region of spread, followed by uniformity.

Finally, we remark on two deliberate aspects of our analysis. First, to avoid issues of non-convexity, we directly compare the overall loss of our measures with those of the two extrema δ_v and σ_{d-1} . Second, general distributions beyond the regular simplex and normalized surface measure are hard to compute contrastive losses over, and such computations are not largely studied to the extent of our knowledge. This inherently restricts analysis to simple distributions like μ_θ .

4. Breaking Permutation Invariance

Our analysis in the previous section shows that L_{spread} can obtain an optimal geometry that is neither collapsed nor uniform. However, this result does not completely explain improved transfer performance because L_{spread} under the previous setup is *class-fixing permutation invariant*, a property we define in Section 4.1. Inducing an inductive bias can break such an invariance. We argue that an inductive bias that encourages clustering of latent subclasses can be particularly useful. In Section 4.2, we show that generalization error on coarse-to-fine transfer learning depends on both $s_f(y)$ and a notion of subclass clustering $\sigma_f(z)$. We thus discuss three approaches for controlling $\sigma_f(z)$: one standard, and two alternatives with more realistic assumptions (Section 4.3).

4.1. Class-Fixing Permutation Invariance

First, we define class-fixing permutation invariance.

Definition 2 (Class-Fixing Permutation Invariance). Let \mathcal{F} be a class of encoders. Let $L(f, B)$ be a loss function over an encoder $f \in \mathcal{F}$ and a set of n points $B = \{x_1, \dots, x_n\}$. Define $S_{h,B}$ as the set of class-fixing permutations such that $\pi \in S_{h,B} : [n] \rightarrow [n]$ satisfies $h(x_{\pi(i)}) = h(x_i)$ for all $i \in [n]$. Then, L is invariant on class-fixing permutations under \mathcal{F} if, for any batch B , permutation $\pi \in S_{h,B}$, and encoder $f \in \mathcal{F}$, there exists another encoder $f^\pi \in \mathcal{F}$ such that $f^\pi(x_i) = f(x_{\pi(i)})$ for all $i \in [n]$ and $L(f, B) = L(f^\pi, B)$.

We find that L_{spread} is invariant on class-fixing permutations under the infinite encoder assumption from Section 2.3.

Proposition 2. Let \mathcal{F} be the set of infinite encoders. Then L_{spread} is invariant on class-fixing permutations under \mathcal{F} .

Under class-fixing permutation invariance, data points can be arbitrarily mapped to representations within their classes, suggesting that the mapping that minimizes L_{spread} is not unique. However, not all these mappings achieve the same performance on downstream tasks. Therefore, while our

result from Section 3 provides insight into L_{spread} 's geometry under an infinitely powerful encoder, it cannot completely explain representation quality.

4.2. Inductive Bias for Improved Coarse-to-fine Transfer

Inducing an inductive bias can break permutation invariance (see Lemma 4 in Appendix D for a simple proof of how smoothness of f is a sufficient condition for breaking invariance). We argue that inducing subclass clustering can be particularly helpful for transfer performance. We measure subclass clustering in embedding space via the expected distance to the center of the subclass, $\sigma_f(z) = \mathbb{E}_{x \sim \mathcal{P}_z} [\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z} [f(x)]\|]$. We show that this quantity $\sigma_f(z)$, along with degree of spread $s_f(y)$, is critical for the generalization error of coarse-to-fine transfer.

To present our result on coarse-to-fine generalization error, we define some additional terms. Let y denote the class label corresponding to z, z' . Define the quantity $\delta_f(z, z') = \frac{1}{p(z|y)p(z'|y)} (s_f(y) - p(z|y)^2 \sigma_f(z) - p(z'|y)^2 \sigma_f(z'))$ as a notion of separation between z and z' . $\delta_f(z, z')$ is large when there is spread (large $s_f(y)$) and sufficient subclass clustering (low $\sigma_f(z), \sigma_f(z')$). Define the variance of a subclass as $\text{Var}_f[z] = \mathbb{E}_{x \sim \mathcal{P}_z} [\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z} [f(x)]\|^2]$. We assume that for all $x \sim \mathcal{P}_z$, there exists a $c > 0$ such that $\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z} [f(x)]\| \geq c \cdot \mathbb{E}_{x \sim \mathcal{P}_z, x' \sim \mathcal{P}_{z'}} [\|f(x) - f(x')\|]$ (i.e., no point from z is equal to the center of z').

Theorem 3. Denote $r_f(z, z') = c^2 \delta_f(z, z')^2 - |\text{Var}_f[z] - \text{Var}_f[z']|$. With probability $1 - \delta$, the coarse-to-fine error is at most

$$L_{\gamma, f}(z) \leq \frac{\sigma_f(z)}{\sqrt{r_f(z, z') - 2 \log \gamma}} + \mathcal{O}\left(\left(\frac{d \log(d/\delta)}{m_z \wedge m_{z'}}\right)^{1/4}\right).$$

under the boundary condition that $r_f(z, z') - 2 \log \gamma \geq 16 \sqrt{\frac{2d \log(8d/\delta)}{m_z \wedge m_{z'}} + \frac{2d \log(8d/\delta)}{m_z}}$.

The generalization error depends on the sampling error, γ , and three quantities intrinsic to the distribution of $f(x)$:

- $s_f(y)$: the bound scales inversely in $s_f(y)$; points in a class must be spread out in order for subclasses to be distinguishable. Corollary 1 and empirical measurements (Figure 2) suggest that spread is non-zero when using L_{spread} with α set properly. Note that under SupCon, $s_f(y)$ is asymptotically equal to 0 and this bound is vacuous (refer to Theorem 1 for SupCon's generalization error).
- $\sigma_f(z)$: the bound scales in $\sigma_f(z)$, confirming that spread alone is insufficient. Subclasses also need to be clustered tightly to achieve good transfer performance.
- $|\text{Var}_f[z] - \text{Var}_f[z']|$: distinguishing z versus z' may be difficult when only one subclass is clustered. When both $\sigma_f(z)$ and $\sigma_f(z')$ are small, this quantity is negligible.

Altogether, the generalization error scales in $\frac{\sigma_f(z)}{s_f(y)}$. There-

fore, in addition to having sufficient spread $s_f(y)$, it is critical that $\sigma_f(z)$ is bounded. We thus explore techniques for inducing an inductive bias that can control this quantity.

4.3. Techniques for Inducing Subclass Clustering

We analyze three mechanisms on f for inducing an inductive bias that can cluster subclasses: a constrained encoder, a class-conditional autoencoder, and data augmentations. These three mechanisms use Lipschitzness assumptions of varying strength to bound $\sigma_f(z)$. We assume that subclasses are ‘‘clustered’’ in input space; i.e. there exists some σ_z such that $\mathbb{E}_{x, x' \sim \mathcal{P}_z} [\|x - x'\|] \leq \sigma_z$, and we study how these mechanisms on f allow us to control $\sigma_f(z)$ in terms of σ_z . For each mechanism, we show that $\sigma_f(z) \sim K \sigma_z$ for some particular Lipschitzness constant K . The lower the Lipschitzness constant, the better each mechanism can induce subclass clustering.

We summarize the assumptions of each mechanism in Section 4.3.4 and report empirical estimates of Lipschitz constants in Table 1. Section 4.3.4 also reports estimates of $\frac{\sigma_f(z)}{s_f(y)}$, the ratio that governs the generalization error in Theorem 3, showing how these mechanisms impact this quantity.

4.3.1. LIPSCHITZ ENCODER

One common method for incorporating inductive bias is to suppose that the class of encoders \mathcal{F} is Lipschitz smooth. We show that assuming \mathcal{F} to be the class of K_L -Lipschitz encoders can explain subclass clustering of representations.

Lemma 1. Let \mathcal{F}_{K_L} be the class of K_L -Lipschitz encoders. Then for any $f_{K_L} \in \mathcal{F}_{K_L}$, $\sigma_{f_{K_L}}(z) \leq K_L \sigma_z$.

Lipschitzness with a sufficiently low constant K_L is realistic for simple function classes, such as MLPs with bounded norms. However, modern deep networks are not Lipschitz, as they are powerful enough to memorize random noise (Zhang et al., 2016). In Table 1 we confirm that the Lipschitz constant estimated empirically from our model's encoder on real data is relatively high. Therefore, since encoders with deep architectures are not Lipschitz, we consider other more realistic setups that can encourage subclass clustering.

4.3.2. CLASS-CONDITIONAL AUTOENCODER

To encourage embeddings to preserve properties of the input space without assuming Lipschitzness over the encoder, we propose concatenating embeddings from separate ‘‘class-conditional’’ autoencoders, each consisting of an encoder $f_{AE} \in \mathcal{F}_{AE}$ and a decoder $g \in \mathcal{G}$, to the embeddings learned from L_{spread} . An autoencoder for class y aims to minimize the class reconstruction loss $\hat{L}_{AE}(\mathcal{D}_y) = \frac{1}{n_y} \sum_{x \in \mathcal{D}_y} \|g(f_{AE}(x)) - x\|^2$. These K per-class autoencoders thus intuitively learn distinctions within classes.

Table 1. Three mechanisms for inductive bias and empirical measurements of their associated Lipschitzness constants. Higher K is a worse Lipschitzness constant, which suggests the assumptions are less realistic

Mechanism	Assumptions	Lipschitzness Constant
Encoder (Constrained)	Lipschitz f	$K_L = 0.058$
Autoencoder	Decoder g reverse Lipschitz	$K_g = 0.041$
Augmentations	Lipschitz f on augmentations	$K_{aug} = 0.040$

Define a notion of Rademacher complexity $\mathfrak{R}_n^p(\mathcal{F}_1, \mathcal{F}_2) = \mathbb{E}_\sigma \left[\sup_{f_1, f_2 \in \mathcal{F}_1, \mathcal{F}_2} \frac{1}{n} \sum_{i=1}^n \sigma_i \|f_1(x_i) - f_2(x_i)\|^p \right]$ for Rademacher random variables $\sigma = \{\sigma_1, \dots, \sigma_n\}$.

Lemma 2. For any $g \in \mathcal{G}$, suppose there exists a $K_g > 0$ such that g is “reverse Lipschitz”, satisfying $\|f_{AE}(x) - f_{AE}(x')\| \leq K_g \|g(f_{AE}(x)) - g(f_{AE}(x'))\|$, and there exists finite b such that the reconstruction loss satisfies $\max_x \|g(f_{AE}(x)) - x\|^2 \leq b$.

Then with probability at least $1 - \delta$,

$$\sigma_{f_{AE}}(z) \leq \frac{2K_g}{p(z|y)} \left(\hat{L}_{AE}(\mathcal{D}_y) + 2\mathfrak{R}_{n_y}^2(\mathcal{G} \circ \mathcal{F}_{AE}, id_{\mathcal{X}}) + b \sqrt{\log(1/\delta)/2n_y} \right)^{1/2} + K_g \sigma_z, \quad (5)$$

where $id_{\mathcal{X}}$ is the identity function on \mathcal{X} , and $p(z|y) = \frac{p(z)}{p(y)}$ is the probability that x drawn from $p(\cdot|y)$ has label z .

There are no explicit assumptions on f_{AE} ; instead, a condition on the decoder is used for clustering subclasses. In Appendix C.2, we show that for an autoencoder trained on \mathcal{D} instead of \mathcal{D}_y , $p(z)$ replaces $p(z|y)$, and n replaces n_y . That is, while a general autoencoder is learned on more data, individual subclasses comprise a smaller proportion of the data and thus could be harder to learn meaningful representations of. This suggests that $\sigma_{f_{AE}}(z)$ is roughly a constant factor larger with a general autoencoder when n_y and n are both large, and thus a class-conditional autoencoder can better cluster subclasses.

4.3.3. DATA AUGMENTATION

Another way of inducing inductive bias for subclass clustering is data augmentations, which we use in L_{spread} and which play a prominent role in contrastive learning overall. Define $\mathcal{A} : \mathcal{X} \rightarrow \mathcal{X}$ as the function class of augmentations and \mathcal{F}_{aug} as the class of encoders.

Lemma 3. For $a \in \mathcal{A}$ and any $x, x' \in \mathcal{X}$, suppose that $f_{aug} \in \mathcal{F}_{aug}$ satisfies $\|f_{aug}(a(x)) - f_{aug}(a(x'))\| \leq K_{aug} \|a(x) - a(x')\|$ for some K_{aug} and that $f(a(x)) = f(x)$ for $x \in \mathcal{D}$. Denote $\sigma_z^{aug} = \mathbb{E}_{x, x' \sim \mathcal{P}_z} [\|a(x) - a(x')\|]$. Then with probability at least $1 - \delta$,

$$\sigma_{f_{aug}}(z) \leq \frac{2}{p(z)} \left(2\mathfrak{R}_n^1(\mathcal{F}_{aug}, \mathcal{F}_{aug} \circ \mathcal{A}) + \sqrt{2 \log(1/\delta)/n} \right) + K_{aug} \sigma_z^{aug}.$$

Our result assumes Lipschitzness *only on the augmentations*, which is consistent with literature such as Dao et al. (2019), and that the model can align augmented and original training data pairs. $\sigma_f(z)$ scales with how close augmentations of points within a subclass are, σ_z^{aug} . This quantity can actually be less than σ_z under assumptions in prior work on characterizing augmentations (Huang et al., 2021), which results in tighter embedding clusters. Our result can also explain why prior works (Islam et al., 2021) observe that modified losses (that include augmentations) result in better transfer. The empirical findings in Figure 1 (right), where the subclass embedding visualization are L_{spread} with and without augmentations, support this result.

4.3.4. OVERALL TAKEAWAYS

Our results from Lemmas 1, 2, and 3 show that $\sigma_f(z)$, which is critical for transfer performance as demonstrated in Theorem 3, can be controlled. Table 1 summarizes our results on how a standard encoder, an autoencoder, and data augmentations can encourage subclass clustering under various assumptions. We report empirical measures of K_L , K_g and K_{aug} on real datasets, and find that the autoencoder and data augmentation assumptions are more realistic (lower values of K).

Figure 2 demonstrates these effects on real data; apparent clusters begin forming under L_{spread} (which is trained with data augmentation). We also measure the ratio $\frac{\sigma_f(z)}{\sigma_f(y)}$ and find that it can range as high as 1.94 for SupCon. For L_{spread} with augmentations and the autoencoder, the maximum values are 1.05 and 1.03, respectively—suggesting that these modifications control subclass clustering better, and should result in better coarse-to-fine transfer.

5. Experiments

In this section, we evaluate how well adding a class-conditional InfoNCE loss and a class-conditional autoencoder improves the representations produced by supervised contrastive learning. We call our overall method THANOS. This section is primarily designed to evaluate two claims:

- We use coarse-to-fine transfer learning to evaluate how well the representations maintain subclass information. THANOS achieves 11.1 lift on average across five datasets.

Table 2. Summary of the datasets we use for evaluation.

Dataset	K_{coarse}	K_{fine}	Notes
CIFAR10	2	10	Coarse labels are animal vs. vehicle
CIFAR100	20	100	Standard coarse labels
CIFAR100-U	20	100	CIFAR100, imbalanced fine classes
MNIST	2	10	Coarse labels are < 5 and ≥ 5
TinyImageNet	67	200	Coarse labels from ImageNet hierarchy
Waterbirds	2	3	Bird images (Sagawa et al., 2019)
ISIC	2	3	Skin lesions (Codella et al., 2019)
CelebA	2	3	Celebrity faces (Liu et al., 2015)

- We evaluate how well THANOS can improve worst-group robustness in the unlabeled setting. THANOS detects low-performing sub-groups 6.2 points better than SupCon across three datasets. THANOS sets state-of-the-art worst-group robustness without sub-group labels by 11.5 points on CelebA—and even outperforms an algorithm that has access to ground-truth sub-group labels in some cases.

We also present ablations. Additional experiments on overall model quality, additional baselines, and additional datasets are in Appendix G. Although we focus on coarse-to-fine transfer and robustness here, we note that our method also produces lift on overall model quality.

THANOS Method We summarize the THANOS method.¹ THANOS consists of adding a class-conditional InfoNCE loss and a class-conditional autoencoder to the supervised contrastive loss with standard data augmentations used in Chen et al. (2020a). We implement the former by training an encoder with L_{spread} . To implement the latter, we train a single autoencoder with a joint MSE reconstruction loss and a cross entropy loss. We then concatenate the autoencoder representation to the representation of the encoder trained with L_{spread} . Details on architectures and hyperparameters in Appendix F.

Datasets Table 2 lists the datasets we use in our evaluation. We use coarse versions of CIFAR10, CIFAR100, MNIST, and TinyImageNet to study coarse-to-fine transfer. We use Waterbirds, ISIC and CelebA for robustness (Sagawa et al., 2019; Codella et al., 2019; Liu et al., 2015; Sohoni et al., 2020).

Coarse-to-Fine Transfer We use coarse-to-fine transfer learning to isolate how well representations separate subclasses in an ideal setting. In coarse-to-fine transfer, we train models on coarse labels, freeze the weights, and then train a linear probe over the final layer on fine labels. Note that this setting is more challenging than the self-supervised setting, since it requires maintaining high performance on the coarse classes while also being transferrable to the fine classes. We focus on transfer numbers in this section, but Table 6 in the Appendix presents results on coarse accuracy.

¹Our code is available at <https://github.com/HazyResearch/thanos-code/>.

For the autoencoder experiments, we train an autoencoder separately and concatenate its embedding layer with the contrastive embedding for the linear probe. We jointly optimize all contrastive losses and the class-conditional autoencoder with a cross-entropy loss head. We train all models with dropout as well as label smoothing on the cross-entropy loss heads.

We report four variants of THANOS: the class-conditional autoencoder on its own, SupCon modified with a class-conditional InfoNCE loss, SupCon modified with a class-conditional autoencoder, and SupCon with both modifications. We report 3 baselines from previous work on the transferability of SupCon (Islam et al., 2021): SupCon, SupCon plus an InfoNCE loss, and the InfoNCE loss on its own.

THANOS significantly outperforms SupCon on coarse-to-fine transfer learning—by an average of 11.1 points across all tasks. 7.3 points can be attributed to the class-conditional InfoNCE loss on average, but mileage varies between tasks (25.3 points of lift for CIFAR10, vs. 2.6 for CIFAR100). The difference is the number of coarse classes: CIFAR10 only has two coarse classes, whereas CIFAR100 has 20. Fewer coarse classes makes it easier to achieve class collapse, so spread is more necessary. Finally, we also note that combining the autoencoder with the other components outperforms using the autoencoder on its own, by 2.7 points on average. This suggests that each component is helpful.

Worst-Group Robustness We use robustness to measure how well THANOS can recover hidden subgroups in an unsupervised setting. For these models, we train contrastive losses on their own. We follow the methodology from Sohoni et al. (2020). We first train a model with class labels. We then cluster the embeddings to produce pseudolabels for subclasses, which we use as input to the GroupDRO algorithm to optimize worst-group robustness (Sagawa et al., 2019).

Our primary evaluation metric is robustness, but we also evaluate a subgroup recovery metric since prior work has suggested that it is important for robustness. Subgroup recovery also acts as a proxy for *unsupervised* group recovery. We compare subgroup recovery against SupCon and Sohoni et al. (2020). We compare worst-group robustness against Sohoni et al. (2020) and JTT (Liu et al., 2021), as well as using subgroup labels from SupCon. We also report the performance of GroupDRO with ground-truth subclass labels.

Table 4 shows the results. THANOS outperforms both SupCon and Sohoni et al. (2020) on subgroup recovery. THANOS further achieves state-of-the-art worst-group robustness, outperforming JTT by 4.7 points and Sohoni et al. (2020) by 11.7 points on average—and setting state-of-the-art on CelebA by 11.5 points. Surprisingly, THANOS can even outperform GroupDRO with ground-truth subgroup labels in two cases.

Table 3. Coarse-to-fine transfer learning performance. Best in bold.

Method		CIFAR10	CIFAR100	CIFAR100-U	MNIST	TinyImageNet
Baselines	InfoNCE (Chen et al., 2020a)	77.6 ± 0.1	60.5 ± 0.1	56.4 ± 0.3	98.4 ± 0.1	44.9 ± 0.1
	SupCon (Khosla et al., 2020)	51.8 ± 1.2	56.1 ± 0.1	49.8 ± 0.3	95.4 ± 0.1	43.9 ± 0.1
	SupCon + InfoNCE (Islam et al., 2021)	77.6 ± 0.1	55.7 ± 0.1	48.0 ± 0.2	98.6 ± 0.1	46.1 ± 0.1
Ours	cAuto	71.4 ± 0.1	62.9 ± 0.1	58.7 ± 0.5	98.7 ± 0.1	47.1 ± 0.1
	SupCon + cNCE (L_{spread})	77.1 ± 0.1	58.7 ± 0.2	53.5 ± 0.4	98.5 ± 0.1	45.8 ± 0.1
	SupCon + cAuto	71.7 ± 0.1	63.8 ± 0.6	59.8 ± 0.3	98.7 ± 0.1	49.3 ± 0.1
	SupCon + cNCE + cAuto (THANOS)	79.1 ± 0.2	65.0 ± 0.2	59.7 ± 0.3	99.0 ± 0.1	49.6 ± 0.1

Table 4. Unsupervised subclass recovery (top, F1), and worst-group performance (AUROC for ISIC, Acc for others). Best in bold.

Method	Group Labels	Waterbirds	ISIC	CelebA
Sub-Group Recovery				
Sohoni et al. (2020)	✗	56.3	74.0	24.2
SupCon	✗	47.1	92.5	19.4
THANOS	✗	59.0	93.8	24.8
Worst-Group Robustness				
Sohoni et al. (2020)	✗	88.4	92.0	55.0
JTT (Liu et al., 2021)	✗	83.8	91.8	77.9
SupCon	✗	86.8	93.3	66.1
THANOS	✗	88.6	92.6	89.4
GroupDRO	✓	90.7	92.3	88.9

Subgroup recovery and worst-group robustness are correlated but not causal: (Sohoni et al., 2020) observed inconsistencies between them, and so do we (i.e., our method outperforms GroupDRO, an approach with “perfect” subgroup labels). This phenomenon deserves further exploration.

Ablations We summarize two ablations (Appendix G.5). First, we validate Lemma 2 and find that using a generic autoencoder underperforms a class-conditional autoencoder by 30.0 points on average—and furthermore does not improve the performance of SupCon as well (2.0 points of lift compared to 11.0 points). Second, we validate Lemma 3 and confirm that data augmentation is crucial; removing data augmentation degrades performance by 35.4 points.

6. Related Work and Discussion

We present an abbreviated related work. A full treatment can be found in Appendix A. Our theoretical work relates to theory on the geometry of contrastive learning (Wang & Isola, 2020; Graf et al., 2021; Robinson et al., 2020; Zimmermann et al., 2021), collapsed representations (Galanti et al., 2021; Jing et al., 2021), autoencoders (Epstein & Meir, 2019; Le et al., 2018), data augmentation (HaoChen et al., 2021; Guo et al., 2018; Abavisani et al., 2020), and robustness (Sohoni et al., 2020). Our use of L_{spread} and an autoencoder draws from a wave of empirical work on contrastive learning (Chen et al., 2020a; Khosla et al., 2020), and its properties (Islam

et al., 2021; Bukchin et al., 2021).

In aggregate, we study how to improve the quality of representations trained with supervised contrastive learning. We identify controlling spread and inducing subclass clustering as two key challenges and show how two modifications to supervised contrastive learning improve transfer and robustness.

Authors’ Note The first two authors contributed equally. Co-first authors can prioritize their names when adding this paper’s reference to their resumes.

Acknowledgments

We thank Beidi Chen, Tri Dao, Karan Goel, and Albert Gu for their helpful comments on early drafts of this paper. We gratefully acknowledge the support of NIH under No. U54EB020405 (Mobilize), NSF under Nos. CCF1763315 (Beyond Sparsity), CCF1563078 (Volume to Velocity), and 1937301 (RTML); ONR under No. N000141712266 (Unifying Weak Supervision); ONR N00014-20-1-2480: Understanding and Applying Non-Euclidean Geometry in Machine Learning; N000142012275 (NEPTUNE); the Moore Foundation, NXP, Xilinx, LETI-CEA, Intel, IBM, Microsoft, NEC, Toshiba, TSMC, ARM, Hitachi, BASF, Accenture, Ericsson, Qualcomm, Analog Devices, the Okawa Foundation, American Family Insurance, Google Cloud, Salesforce, Total, the HAI-GCP Cloud Credits for Research program, the Stanford Data Science Initiative (SDSI), Department of Defense (DoD) through the National Defense Science and Engineering Graduate Fellowship (NDSEG) Program, and members of the Stanford DAWN project: Facebook, Google, and VMware. The Mobilize Center is a Biomedical Technology Resource Center, funded by the NIH National Institute of Biomedical Imaging and Bioengineering through Grant P41EB027060. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views, policies, or endorsements, either expressed or implied, of NIH, ONR, or the U.S. Government.

References

- Abavisani, M., Naghizadeh, A., Metaxas, D. N., and Patel, V. M. Deep subspace clustering with data augmentation. In *Thirty-fourth Conference on Neural Information Processing Systems*, 2020.
- Arora, S., Khandeparkar, H., Khodak, M., Plevrakis, O., and Saunshi, N. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019.
- Borodachov, S. V., Hardin, D. P., and Saff, E. B. *Discrete energy on rectifiable sets*. Springer, 2019.
- Bostock, M. Imagenet hierarchy, 2018. URL <https://observablehq.com/@mbostock/imagenet-hierarchy>.
- Bukchin, G., Schwartz, E., Saenko, K., Shahar, O., Feris, R., Giryas, R., and Karlinsky, L. Fine-grained angular contrastive learning with coarse labels. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Jun 2021.
- Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems*, 2020.
- Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 2020a.
- Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020b.
- Codella, N., Rotemberg, V., Tschandl, P., Celebi, M. E., Dusza, S., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M., et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.
- Dao, T., Gu, A., Ratner, A., Smith, V., De Sa, C., and Ré, C. A kernel theory of modern data augmentation. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1528–1537. PMLR, 09–15 Jun 2019.
- d’Eon, G., d’Eon, J., Wright, J. R., and Leyton-Brown, K. The spotlight: A general method for discovering systematic errors in deep learning models. *arXiv preprint arXiv:2107.00758*, 2021.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.
- Duchi, J., Hashimoto, T., and Namkoong, H. Distributionally robust losses for latent covariate mixtures. *arXiv preprint arXiv:2007.13982*, 2020.
- Epstein, B. and Meir, R. Generalization bounds for unsupervised and semi-supervised learning with autoencoders. *arXiv preprint arXiv:1902.01449*, 2019.
- Falcon, W. and Cho, K. A framework for contrastive self-supervised learning and designing a new approach. *arXiv preprint arXiv:2009.00104*, 2020.
- Galanti, T., György, A., and Hutter, M. On the role of neural collapse in transfer learning. *arXiv preprint arXiv:2112.15121*, 2021.
- Goel, K., Gu, A., Li, Y., and Re, C. Model patching: Closing the subgroup performance gap with data augmentation. In *International Conference on Learning Representations*, 2020.
- Goyal, P., Caron, M., Lefauveux, B., Xu, M., Wang, P., Pai, V., Singh, M., Liptchinsky, V., Misra, I., Joulin, A., et al. Self-supervised pretraining of visual features in the wild. *arXiv preprint arXiv:2103.01988*, 2021.
- Graf, F., Hofer, C., Niethammer, M., and Kwitt, R. Dissecting supervised contrastive learning. In *International Conference on Machine Learning*, pp. 3821–3830. PMLR, 2021.
- Guo, X., Zhu, E., Liu, X., and Yin, J. Deep embedded clustering with data augmentation. In *Asian conference on machine learning*, pp. 550–565. PMLR, 2018.
- Han, X. Y., Pappas, V., and Donoho, D. L. Neural collapse under mse loss: Proximity to and dynamics on the central path, 2021.
- HaoChen, J. Z., Wei, C., Gaidon, A., and Ma, T. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *arXiv preprint arXiv:2106.04156*, 2021.
- He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.
- Hoffmann, A., Kwok, R., and Compton, P. Using subclasses to improve classification learning. In *European Conference on Machine Learning*, pp. 203–213. Springer, 2001.
- Huang, W., Yi, M., and Zhao, X. Towards the generalization of contrastive self-supervised learning, 2021.

- Hui, L., Belkin, M., and Nakkiran, P. Limitations of neural collapse for understanding generalization in deep learning, 2022.
- Islam, A., Chen, C.-F., Panda, R., Karlinsky, L., Radke, R., and Feris, R. A broad study on the transferability of visual representations with contrastive learning. *arXiv preprint arXiv:2103.13517*, 2021.
- Jing, L., Vincent, P., LeCun, Y., and Tian, Y. Understanding dimensional collapse in contrastive self-supervised learning. *arXiv preprint arXiv:2110.09348*, 2021.
- Khosla, A., Jayadevaprakash, N., Yao, B., and Li, F.-F. Novel dataset for fine-grained image categorization: Stanford dogs. In *Proc. CVPR workshop on fine-grained visual categorization (FGVC)*, volume 2. Citeseer, 2011.
- Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Mschinot, A., Liu, C., and Krishnan, D. Supervised contrastive learning. In *Thirty-Fourth Conference on Neural Information Processing Systems*, 2020.
- Kothapalli, V., Rasromani, E., and Awatramani, V. Neural collapse: A review on modelling principles and generalization, 2022.
- Le, L., Patterson, A., and White, M. Supervised autoencoders: Improving generalization performance with unsupervised regularizers. In *Thirty-second Conference on Neural Information Processing Systems*, 2018.
- Le, Y. and Yang, X. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.
- Liu, E. Z., Haghighi, B., Chen, A. S., Raghunathan, A., Koh, P. W., Sagawa, S., Liang, P., and Finn, C. Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning*, pp. 6781–6792. PMLR, 2021.
- Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.
- Lu, J. and Steinerberger, S. Neural collapse with cross-entropy loss, 2020.
- Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2018.
- Oakden-Rayner, L., Dunnmon, J., Carneiro, G., and Ré, C. Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In *Proceedings of the ACM conference on health, inference, and learning*, pp. 151–159, 2020.
- Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- Papayan, V., Han, X., and Donoho, D. L. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020.
- Robinson, J., Chuang, C.-Y., Sra, S., and Jegelka, S. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592*, 2020.
- Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2019.
- Sohoni, N., Dunnmon, J., Angus, G., Gu, A., and Ré, C. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *Thirty-fourth Conference on Neural Information Processing Systems*, 2020.
- Thomson, J. J. XI. cathode rays. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 44(269):293–316, 1897.
- Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. What makes for good views for contrastive learning? *arXiv preprint arXiv:2005.10243*, 2020.
- Tsai, Y.-H. H., Wu, Y., Salakhutdinov, R., and Morency, L.-P. Self-supervised learning from a multi-view perspective. In *International Conference on Learning Representations*, 2020.
- Tschannen, M., Djolonga, J., Rubenstein, P. K., Gelly, S., and Lucic, M. On mutual information maximization for representation learning. In *International Conference on Learning Representations*, 2020.
- Wang, T. and Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020.
- Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., and Perona, P. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010a.
- Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., and Perona, P. Caltech-ucsd birds 200. 2010b.
- Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2016.

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. Learning deep features for scene recognition using places database. *Twenty-eighth Conference on Neural Information Processing Systems*, 2014.

Zimmermann, R. S., Sharma, Y., Schneider, S., Bethge, M., and Brendel, W. Contrastive learning inverts the data generating process. *arXiv preprint arXiv:2012.08850*, 2021.

We present a full treatment of related work in Appendix A. We present a glossary in Appendix B. We present proofs in Appendix C, additional theoretical results in Appendix D, and auxiliary lemmas in Appendix E. We present additional experimental details in Appendix F, additional results in Appendix G, and synthetics in Appendix H.

A. Related Work

We presented an extended treatment of related work.

From work in contrastive learning, we take inspiration from Arora et al. (2019), who use a latent classes view to study self-supervised contrastive learning. Similarly, Zimmermann et al. (2021) considers how minimizing the InfoNCE loss recovers a latent data generating model. Recent work has also analyzed contrastive learning from the information-theoretic perspective (Oord et al., 2018; Tian et al., 2020; Tsai et al., 2020), but does not fully explain practical behavior (Tschannen et al., 2020). On the geometric side, we are inspired by the theoretical tools from Wang & Isola (2020) and Graf et al. (2021), who study representations on the hypersphere along with Robinson et al. (2020). There has been work studying other notions of collapsed representations. Jing et al. (2021) examines dimension collapse in contrastive learning, which occurs when the learned representations span a subspace of the representation space. Our definition of class collapse can also be viewed as Neural Collapse (Papayan et al., 2020), which started as an empirical observation about when models are trained beyond 0 training error using cross-entropy or MSE loss (Lu & Steinerberger, 2020; Han et al., 2021). Recent works on neural collapse have studied the transferrability of collapsed representations (Galanti et al., 2021), and Hui et al. (2022); Kothapalli et al. (2022) have identified its limitations in this setting. We offer another perspective on the relationship between collapse and embedding quality, and offer techniques to mitigate the effects of collapse in transfer learning.

Our work builds on the recent wave of empirical interest in contrastive learning (Chen et al., 2020a; He et al., 2019; Chen et al., 2020b; Goyal et al., 2021; Caron et al., 2020) and supervised contrastive learning (Khosla et al., 2020). There has also been empirical work analyzing the transfer performance of contrastive representations and the role of intra-class variability in transfer learning. Islam et al. (2021) find that combining supervised and self-supervised contrastive loss improves transfer learning performance, and they hypothesize that this is due to both inter-class separation and intra-class variability. Bukchin et al. (2021) find that combining cross entropy and a class-conditional self-supervised contrastive loss improves coarse-to-fine transfer, also motivated by preserving intra-class variability.

Our use of L_{spread} and a class-conditional autoencoder arises from similar motivations to losses proposed in these works, and we further theoretically study their implications for spread. Our theoretical analysis of autoencoders draws from previous work (Epstein & Meir, 2019; Le et al., 2018). Our study of data augmentation similarly builds on recent theoretical analysis of the role of data augmentation in contrastive learning (HaoChen et al., 2021; Huang et al., 2021) and clustering (Guo et al., 2018; Abavisani et al., 2020).

Our treatment of subclasses is strongly inspired by Sohoni et al. (2020) and Oakden-Rayner et al. (2020), who document empirical consequences of hidden strata. We are inspired by empirical work that has demonstrated that detecting subclasses can be important for performance (Hoffmann et al., 2001; d’Eon et al., 2021) and robustness (Duchi et al., 2020; Sagawa et al., 2019; Goel et al., 2020; Liu et al., 2021).

B. Glossary

The glossary is given in Table 5 below.

Symbol	Used for
x	Input data $x \in \mathcal{X}$ with distribution \mathcal{P} .
y	Class label $y \in \mathcal{Y} = \{0, \dots, K-1\}$, where $h(x)$ is x 's class label.
z	Latent subclass $z \in \mathcal{Z}$.
S_y	The set of all subclasses corresponding to class label y .
$p(z)$	The proportion of subclass z over \mathcal{Z} .
\mathcal{P}_z	The distribution of input data belonging to subclass z , i.e. $\mathcal{P}_z = p(\cdot z)$.
$S(z)$	The label corresponding to subclass z .
$h_s(x)$	The subclass that x belongs to.
\mathcal{D}	Training dataset of n points $\{(x_i, y_i)\}_{i=1}^n$.
\mathcal{D}_y	Training data with label y , $\mathcal{D}_y = \{x \in \mathcal{D} : h(x) = y\}$ of size n_y .
\mathcal{D}_z	Training data with latent subclass z , $\mathcal{D}_z = \{x \in \mathcal{D} : h_s(x) = z\}$ of size n_z .
f	The encoder $f : \mathcal{X} \rightarrow \mathbb{R}^d$ that maps input data to an embedding space with dimension d .
\mathcal{D}_s	A dataset of m points with subclass labels $\mathcal{D}_s = \{(x_i, z_i)\}_{i=1}^m$ used for coarse-to-fine transfer.
$\mathcal{D}_{s,z}$	The subset of \mathcal{D}_s with subclass z , $\mathcal{D}_{s,z} = \{x \in \mathcal{D}_s : h_s(x) = z\}$ of size m_z .
W_z	Linear weight W_z for model used in coarse-to-fine transfer.
$\hat{p}(z f(x))$	Softmax score output by linear model for coarse-to-fine transfer.
$L_{\gamma,f}(z)$	The γ -margin generalization error on subclass z in coarse-to-fine transfer.
B	Batch of input data.
$P(i, B)$	Points in B with the same label as x_i , $\{x^+ \in B \setminus i : h(x^+) = h(x_i)\}$.
$N(i, B)$	Points in B with a label different from that of x_i , $\{x^- \in B \setminus i : h(x^-) \neq h(x_i)\}$.
$a(x_i)$	An augmentation of x_i , where $a : \mathcal{X} \rightarrow \mathcal{X}$.
$\sigma_f(x, x')$	Notation for $\exp\left(\frac{f(x)^\top f(x')}{\tau}\right)$.
τ	Temperature hyperparameter in contrastive loss.
$\hat{L}_{\text{spread}}(f, B)$	The contrastive loss we study (on batch B with encoder f), a weighted sum of a SupCon and class-conditional InfoNCE loss.
α	Weight parameter for L_{spread} .
L_{sup}	SupCon loss that is used in L_{spread} that pushes points of a class together (see (2)).
L_{cNCE}	Class-conditional InfoNCE loss that is used in L_{spread} to pull apart points within a class (see (3)).
\mathcal{S}^{d-1}	The unit hypersphere in \mathbb{R}^d .
μ_y	The pushforward measure of the class-conditional distribution $p(\cdot h(x)=y)$ via f , where $\mu_y \in \mathcal{M}(\mathcal{S}^{d-1})$, the set of all Borel probability measures on the hypersphere.
$\boldsymbol{\mu}$	$\boldsymbol{\mu} = \{\mu_y\}_{y \in \mathcal{Y}}$ is the overall pushforward measure $\mathcal{P} \circ f^{-1} \in \mathcal{M}(\mathcal{S}^{d-1})$.
\boldsymbol{v}	$\boldsymbol{v} = \{v_y\}_{y \in \mathcal{Y}} \in \mathcal{S}^{d-1}$ is the regular simplex inscribed in the hypersphere.
δ_{v_y}	The probability measure on \mathcal{S}^{d-1} with all mass on v_y .
$\boldsymbol{\delta}_v$	The class-collapsed measure $\boldsymbol{\delta}_v = \{\delta_{v_y}\}_{y \in \mathcal{Y}}$ where $f(x) = v_y$ almost surely whenever $h(x) = y$.
σ_{d-1}	The normalized surface area measure on \mathcal{S}^{d-1} .
$\boldsymbol{\sigma}_{d-1}$	The class-uniform measure where $\mu_y = \sigma_{d-1}$ for all $y \in \mathcal{Y}$.
f_{SC}	The encoder trained with SupCon, satisfying $f_{SC}(x) = v_y$ for all $x \in \mathcal{D}$ where $h(x) = y$.
x^+	Point for x 's positive pair, drawn from distribution $p(\cdot h(x^+) = h(x))$.
x^-	Point for x 's negative pair, drawn from distribution $p(\cdot h(x^-) \neq h(x))$.
$L_{\text{spread}}(f, \alpha)$	Asymptotic version of L_{spread} that we analyze (see Definition 1, also referred to as $L_{\text{spread}}(\boldsymbol{\mu}, \alpha)$).
R_θ	Rotation matrix $R_\theta \in \mathbb{R}^{d \times d}$ that rotates by angle θ in two dimensions.
$\boldsymbol{\mu}_\theta$	A measure $\boldsymbol{\mu}_\theta = \{\mu_{0,\theta}, \mu_{1,\theta}\}$ on the hypersphere that we compare against $\boldsymbol{\delta}_v$ and $\boldsymbol{\sigma}_{d-1}$. In particular, $\mu_{0,\theta} = \frac{1}{2} \delta_{R_\theta^\top v_0} + \frac{1}{2} \delta_{R_\theta^\top v_1}$ and similarly for $\mu_{1,\theta}$.
$c_{\tau,d}$	Constant that upper bounds the range of α for which some $\boldsymbol{\mu}_\theta$ attains lower $L_{\text{spread}}(\boldsymbol{\mu}, \alpha)$ than $\boldsymbol{\delta}_v$ or $\boldsymbol{\sigma}_{d-1}$.
$s_f(y)$	Notion of spread in embedding space, defined as $s_f(y) = \mathbb{E}_{h(x)=y} [\ f(x) - \mathbb{E}_{h(x)=y}[f(x)]\]$.
$\sigma_f(z)$	Notion of subclass clustering in embedding space, defined as $\sigma_f(z) = \mathbb{E}_{x \sim \mathcal{P}_z} [\ f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\]$.
$\text{Var}_f[z]$	Notion of subclass variance, defined as $\text{Var}_f[z] = \mathbb{E}_{x \sim \mathcal{P}_z} [\ f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\ ^2]$.
σ_z	How clustered a subclass is in input space, defined as $\sigma_z = \mathbb{E}_{x, x' \sim \mathcal{P}_z} [\ x - x'\]$.
K_L	The Lipschitzness constant of a Lipschitz encoder from function class \mathcal{F}_{K_L} .
f_{AE}, g	Autoencoder with encoder $f_{AE} \in \mathcal{F}_{AE}$ and decoder $g \in \mathcal{G}$.
\hat{L}_{AE}	The autoencoder's reconstruction loss (mean squared error).
$\mathfrak{R}_n^p(\mathcal{F}_1, \mathcal{F}_2)$	Notion of Rademacher complexity defined as $\mathfrak{R}_n^p(\mathcal{F}_1, \mathcal{F}_2) = \mathbb{E}_\sigma [\sup_{f_1, f_2 \in \mathcal{F}_1, \mathcal{F}_2} \frac{1}{n} \sum_{i=1}^n \sigma_i \ f_1(x_i) - f_2(x_i)\ ^p]$.
K_g	"Reverse Lipschitzness" constant of the decoder, e.g. $\ f_{AE}(x) - f_{AE}(x')\ \leq K_g \ g(f_{AE}(x)) - g(f_{AE}(x'))\ $.
\mathcal{A}	Function class of augmentations $\mathcal{A} : \mathcal{X} \rightarrow \mathcal{X}$.
\mathcal{F}_{aug}	Function class of encoders that are trained on augmentations.
K_a	The Lipschitzness constant on augmentations for $f_{aug} \in \mathcal{F}_{aug}$, e.g. $\ f_{aug}(a(x)) - f_{aug}(a(x'))\ \leq K_a \ a(x) - a(x')\ $ for any $x', x \in \mathcal{X}$ and $a \in \mathcal{A}$.
σ_z^{aug}	How clustered augmentations of a subclass are in input space.

Table 5. Glossary of variables and symbols used in this paper.

C. Proofs

C.1. Proofs for Section 3

Theorem 1. For γ where $\log \gamma \geq 8\max\{\mathfrak{R}_{n_z}(\mathcal{F}), \mathfrak{R}_{n_{z'}}(\mathcal{F})\}$, SupCon's coarse-to-fine error is at least

$$L_{\gamma, f_{SC}}(z) \geq 1 - \delta(n_z, \mathcal{F}, \gamma) - \delta(n_{z'}, \mathcal{F}, \gamma) - \xi(m_z \wedge m_{z'}, \gamma),$$

where $\delta(n_z, \mathcal{F}, \gamma) = d \exp\left(-\frac{n_z}{32d^2}(\log \gamma - 8\mathfrak{R}_{n_z}(\mathcal{F}))^2\right)$ bounds generalization error of f_{SC} and $\xi(m_z \wedge m_{z'}, \gamma) = 4d \exp\left(-\frac{(m_z \wedge m_{z'}) \log^2 \gamma}{32d}\right)$ bounds the noise from \mathcal{D}_s .

Proof. $L_{\gamma, f}(z) = \Pr_{x \sim \mathcal{P}_z}(\hat{p}(z|f(x)) \leq \gamma \hat{p}(z'|f(x)))$, and by definition of the linear softmax classifier, we have that

$$\begin{aligned} L_{\gamma, f}(z) &= \Pr_{x \sim \mathcal{P}_z} \left(\frac{\exp(f(x)^\top W_z)}{\exp(f(x)^\top W_z) + \exp(f(x)^\top W_{z'})} \leq \gamma \frac{\exp(f(x)^\top W_{z'})}{\exp(f(x)^\top W_z) + \exp(f(x)^\top W_{z'})} \right) \\ &= \Pr_{x \sim \mathcal{P}_z} (f(x)^\top (W_z - W_{z'}) \leq \log \gamma). \end{aligned} \quad (6)$$

To lower bound this quantity, we focus on upper bounding $f(x)^\top (W_z - W_{z'})$. We can bound $f_{SC}(x)^\top (W_z - W_{z'}) \leq \|W_z - W_{z'}\| \leq \|\mathbb{E}_{x \sim \mathcal{P}_z}[f_{SC}(x)] - \mathbb{E}_{x' \sim \mathcal{P}_{z'}}[f_{SC}(x')]\| + \xi_z + \xi_{z'}$, where $\xi_z = \left\| \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f_{SC}(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f_{SC}(x)] \right\|$ can be bounded via standard concentration inequalities and $\xi_{z'}$ is similarly constructed.

Because SupCon yields collapsed training embeddings within any class, we know that $f_{SC}(x) = f_{SC}(x')$ for $x, x' \in \mathcal{D}$ where $h_s(x) = z$ and $h_s(x') = z'$. Therefore, it holds that

$$\begin{aligned} &\|\mathbb{E}_{x \sim \mathcal{P}_z}[f_{SC}(x)] - \mathbb{E}_{x' \sim \mathcal{P}_{z'}}[f_{SC}(x')]\| \\ &= \left\| \mathbb{E}_{x \sim \mathcal{P}_z}[f_{SC}(x)] - \frac{1}{n_z} \sum_{x \in \mathcal{D}_z} f_{SC}(x) + \frac{1}{n_{z'}} \sum_{x' \in \mathcal{D}_{z'}} f_{SC}(x') - \mathbb{E}_{x' \sim \mathcal{P}_{z'}}[f_{SC}(x')] \right\| \\ &\leq \left\| \mathbb{E}_{x \sim \mathcal{P}_z}[f_{SC}(x)] - \frac{1}{n_z} \sum_{x \in \mathcal{D}_z} f_{SC}(x) \right\| + \left\| \frac{1}{n_{z'}} \sum_{x' \in \mathcal{D}_{z'}} f_{SC}(x') - \mathbb{E}_{x' \sim \mathcal{P}_{z'}}[f_{SC}(x')] \right\| \\ &\leq \sup_{f \in \mathcal{F}} \left\| \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)] - \frac{1}{n_z} \sum_{x \in \mathcal{D}_z} f(x) \right\| + \sup_{f \in \mathcal{F}} \left\| \frac{1}{n_{z'}} \sum_{x' \in \mathcal{D}_{z'}} f(x') - \mathbb{E}_{x' \sim \mathcal{P}_{z'}}[f(x')] \right\|. \end{aligned}$$

Define $\epsilon(z, \mathcal{D}, \mathcal{F}) = \sup_{f \in \mathcal{F}} \|\mathbb{E}_{x \sim \mathcal{P}_z}[f(x)] - \frac{1}{n_z} \sum_{x \in \mathcal{D}_z} f(x)\|$ and similarly $\epsilon(z', \mathcal{D}, \mathcal{F})$. Therefore, our loss in (6) satisfies

$$\begin{aligned} L_{\gamma, f_{SC}}(z) &\geq \Pr(\epsilon(z, \mathcal{D}, \mathcal{F}) + \epsilon(z', \mathcal{D}, \mathcal{F}) + \xi_z + \xi_{z'} \leq \log \gamma) \\ &\geq \Pr\left(\epsilon(z, \mathcal{D}, \mathcal{F}) \leq \frac{\log \gamma}{4}\right) \Pr\left(\epsilon(z', \mathcal{D}, \mathcal{F}) \leq \frac{\log \gamma}{4}\right) \Pr\left(\xi_z \leq \frac{\log \gamma}{4}\right) \Pr\left(\xi_{z'} \leq \frac{\log \gamma}{4}\right), \end{aligned} \quad (7)$$

where independence comes from the fact that data is i.i.d. sampled for each subclass and each \mathcal{D} and \mathcal{D}_s , and that we are taking the supremum over \mathcal{F} . Next, we bound $\epsilon(z, \mathcal{D}, \mathcal{F})$. Since $\|f(x)\| \leq 1$, we have that by Lemma 6 that with probability $1 - \delta$,

$$\epsilon(z, \mathcal{D}, \mathcal{F}) \leq 2\mathfrak{R}_{n_z}(\mathcal{F}) + d \sqrt{\frac{2 \log(d/\delta)}{n_z}}.$$

Setting $\epsilon := 2\mathfrak{R}_{n_z}(\mathcal{F}) + d \sqrt{\frac{2 \log(d/\delta)}{n_z}}$, we can write $\delta = d \exp\left(-\frac{n_z}{2d^2}(\epsilon - 2\mathfrak{R}_{n_z}(\mathcal{F}))^2\right)$. Therefore, for $\log \gamma \geq 8\mathfrak{R}_{n_z}(\mathcal{F})$, we have that

$$\Pr\left(\epsilon(z, \mathcal{D}, \mathcal{F}) \leq \frac{\log \gamma}{4}\right) \geq 1 - d \exp\left(-\frac{n_z}{32d^2}(\log \gamma - 8\mathfrak{R}_{n_z}(\mathcal{F}))^2\right). \quad (8)$$

Next, we bound ξ_z . We can write

$$\left\| \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f_{SC}(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f_{SC}(x)] \right\| = \left(\sum_{j=1}^d \left(\frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f_{SC}(x)[j] - \mathbb{E}_{x \sim \mathcal{P}_z}[f_{SC}(x)[j]] \right)^2 \right)^{1/2},$$

where $f_{SC}(x)[j]$ is the j th element of $f_{SC}(x)$. Using Hoeffding's inequality, we have that $\Pr\left(\frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f_{SC}(x)[j] - \mathbb{E}_{x \sim \mathcal{P}_z}[f_{SC}(x)[j]] \geq \frac{\log \gamma}{4m_z}\right) \leq 2 \exp\left(-\frac{m_z (\log \gamma)^2}{32d}\right)$.

$\mathbb{E}_{x \sim \mathcal{P}_z} [f_{SC}(x)[j]]^2 \geq \epsilon \leq 2\exp(-\frac{m_z \epsilon}{2})$, and therefore

$$\begin{aligned} \Pr\left(\sum_{j=1}^d \left(\frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f_{SC}(x)[j] - \mathbb{E}_{x \sim \mathcal{P}_z} [f_{SC}(x)[j]]\right)^2 \leq d\epsilon\right) &\geq \Pr\left(\bigcap_{j=1}^d \left(\frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f_{SC}(x)[j] - \mathbb{E}_{x \sim \mathcal{P}_z} [f_{SC}(x)[j]]\right)^2 \leq \epsilon\right) \\ &\geq \left(1 - 2\exp\left(-\frac{m_z \epsilon}{2}\right)\right)^d \geq 1 - 2d\exp\left(-\frac{m_z \epsilon}{2}\right). \end{aligned}$$

That is, $\Pr(\xi_z \leq \sqrt{d\epsilon}) \geq 1 - 2d\exp\left(-\frac{m_z \epsilon}{2}\right)$.

Setting $\frac{\log \gamma}{4} = \sqrt{d\epsilon}$ gives us $\Pr(\xi_z \leq \frac{\log \gamma}{4}) \geq 1 - 2d\exp\left(-\frac{m_z \cdot \log^2 \gamma / 16d}{2}\right) = 1 - 2d\exp\left(-\frac{m_z \cdot \log^2 \gamma}{32d}\right)$.

We put this expression and (8) back into (7) and use the fact that $(1 - \delta_1)(1 - \delta_2) \geq 1 - \delta_1 - \delta_2$ for any $\delta_1, \delta_2 > 0$. Therefore, we have for SupCon,

$$\begin{aligned} L_{\gamma, f_{SC}}(z) &\geq 1 - d\exp\left(-\frac{n_z}{32d^2}(\log \gamma - 8\mathfrak{R}_{n_z}(\mathcal{F}))^2\right) - d\exp\left(-\frac{n_{z'}}{32d^2}(\log \gamma - 8\mathfrak{R}_{n_{z'}}(\mathcal{F}))^2\right) \\ &\quad - 2d\exp\left(-\frac{m_z \log^2 \gamma}{32d}\right) - 2d\exp\left(-\frac{m_{z'} \log^2 \gamma}{32d}\right) \\ &\geq 1 - d\exp\left(-\frac{n_z}{32d^2}(\log \gamma - 8\mathfrak{R}_{n_z}(\mathcal{F}))^2\right) - d\exp\left(-\frac{n_{z'}}{32d^2}(\log \gamma - 8\mathfrak{R}_{n_{z'}}(\mathcal{F}))^2\right) \\ &\quad - 4d\exp\left(-\frac{(m_z \wedge m_{z'}) \log^2 \gamma}{32d}\right) \\ &= 1 - \delta(n_z, \mathcal{F}, \gamma) - \delta(n_{z'}, \mathcal{F}, \gamma) - \xi(m_z \wedge m_{z'}, \gamma). \end{aligned}$$

□

Derivation of $L_{\text{spread}}(f, \alpha)$ We explain how we arrive at the asymptotic form of L_{spread} . Let $L_{\text{spread}}(f, n^+, n^-)$ be the population-level version of $\hat{L}_{\text{spread}}(f, B)$, where n^+, n^- are the number of negatives in the denominators of L_{CNCE} and L_{sup} , respectively.

$$L_{\text{spread}}(f, n^+, n^-) = (1 - \alpha)L_{\text{sup}}(f, n^-) + \alpha L_{\text{CNCE}}(f, n^+) \quad (9)$$

$$L_{\text{sup}}(f, n^-) = -\mathbb{E} \left[\log \frac{\sigma_f(x, x^+)}{\sigma_f(x, x^+) + \sum_{i=1}^{n^-} \sigma_f(x, x_i^-)} \right] \quad (10)$$

$$L_{\text{CNCE}}(f, n^+) = -\mathbb{E} \left[\log \frac{\sigma_f(x, a(x))}{\sigma_f(x, a(x)) + \sum_{i=1}^{n^+} \sigma_f(x, x_i^+)} \right] \quad (11)$$

We now demonstrate how minimizing $\lim_{n^+, n^- \rightarrow \infty} L_{\text{spread}}(f, n^+, n^-)$ is equivalently to minimizing $L_{\text{spread}}(f, \alpha)$. In L_{sup} , we divide the numerator and denominator by n^- , and in L_{CNCE} we divide the numerator and denominator by n^+ :

$$\begin{aligned} L_{\text{sup}}(f, n^-) &= \mathbb{E} \left[-\log \frac{\exp(\sigma_f(x, x^+))}{\frac{1}{n^-} \exp(\sigma_f(x, x^+)) + \frac{1}{n^-} \sum_{i=1}^{n^-} \exp(\sigma_f(x, x_i^-))} \right] + \log n^- \\ L_{\text{CNCE}}(f, n^+) &= \mathbb{E} \left[-\log \frac{\exp(\sigma_f(x, a(x)))}{\frac{1}{n^+} \exp(\sigma_f(x, a(x))) + \frac{1}{n^+} \sum_{i=1}^{n^+} \exp(\sigma_f(x, x_i^+))} \right] + \log n^+ \end{aligned}$$

We can thus write $L_{\text{spread}}(f, n^+, n^-)$ as

$$\begin{aligned} L_{\text{spread}}(f, n^+, n^-) - \alpha \log n^- - (1 - \alpha) \log n^+ &= -\alpha \mathbb{E} [\sigma_f(x, x^+)] - (1 - \alpha) \mathbb{E} [\sigma_f(x, a(x))] \\ &\quad + \alpha \mathbb{E} \left[\log \left(\frac{1}{n^-} \exp(\sigma_f(x, x^+)) + \frac{1}{n^-} \sum_{i=1}^{n^-} \exp(\sigma_f(x, x_i^-)) \right) \right] \\ &\quad + (1 - \alpha) \mathbb{E} \left[\log \left(\frac{1}{n^+} \exp(\sigma_f(x, a(x))) + \frac{1}{n^+} \sum_{i=1}^{n^+} \exp(\sigma_f(x, x_i^+)) \right) \right]. \end{aligned}$$

Taking the limit $n^+, n^- \rightarrow \infty$ yields

$$\begin{aligned} \lim_{n^+, n^- \rightarrow \infty} L_{\text{spread}}(f, n^+, n^-) &= \alpha \log n^- - (1-\alpha) \log n^+ \\ &= -\alpha \mathbb{E}[\sigma_f(x, x^+)] - (1-\alpha) \mathbb{E}[\sigma_f(x, a(x))] \\ &\quad + \alpha \mathbb{E}_x[\log \mathbb{E}_{x^-}[\exp(\sigma_f(x, x^-))]] + (1-\alpha) \mathbb{E}_x[\log \mathbb{E}_{x^+}[\exp(\sigma_f(x, x^+))]]. \end{aligned}$$

Lastly, we use the fact that $\sigma_f(x, x') = \exp(f(x)^\top f(x')/\tau) = \exp(-\|f(x) - f(x')\|^2/2\tau)$, since $f(x), f(x') \in \mathcal{S}^{d-1}$ to get that

$$\begin{aligned} \lim_{n^+, n^- \rightarrow \infty} L_{\text{spread}}(f, n^+, n^-) &= \alpha \log n^- - (1-\alpha) \log n^+ \\ &= (1-\alpha)L_{\text{align}}(f) + \alpha L_{\text{aug}}(f) + (1-\alpha)L_{\text{diff}}(f) + \alpha L_{\text{same}}(f) \\ &= L_{\text{spread}}(f, \alpha). \end{aligned}$$

Proposition 1 (Individual losses). $L_{\text{align}}(f)$ and $L_{\text{aug}}(f)$ are minimized when $f(x) = f(x^+)$ and $f(x) = f(a(x))$ almost surely, respectively. $L_{\text{diff}}(\mu)$ is minimized when $\mu = \delta_v$. $L_{\text{same}}(\mu)$ is minimized when $\mu = \sigma_{d-1}$.

Proof. We analyze each term's optimal measure on the hypersphere.

$L_{\text{align}}(f), L_{\text{aug}}(f)$ For both $L_{\text{align}}(f)$ and $L_{\text{aug}}(f)$, the minimum value of the expression is 0, which is obtained when $f(x) = f(x^+)$ and $f(x) = f(a(x))$ almost surely, respectively.

$L_{\text{diff}}(f)$ We show in Lemma 5 that the measure that minimizes $L_{\text{diff}}(f)$ also minimizes $\log \mathbb{E}_{x, x^-}[-\exp(-\|f(x) - f(x^-)\|^2/2\tau)]$ (note this is not identical to the approach in Wang & Isola (2020)). We can thus equivalently consider minimizing $\mathbb{E}_{x, x^-}[\exp(-\|f(x) - f(x^-)\|^2/2\tau)]$. Note that $\max_{f(x), f(x') \in \mathcal{S}^{d-1}} \|f(x) - f(x')\| = 2$, and so $\inf_{\mathbb{E}_{x, x^-}}[\exp(-\|f(x) - f(x^-)\|^2/2\tau)] = \exp(-2/\tau)$. For $\mu = \delta_v$,

$$\begin{aligned} \mathbb{E}_{x, x^-}[\exp(-\|f(x) - f(x^-)\|^2/2\tau)] &= \frac{1}{2} \mathbb{E}_{h(x)=0, h(x')=1}[\exp(-\|v_0 - v_1\|^2/2\tau)] \\ &\quad + \frac{1}{2} \mathbb{E}_{h(x)=1, h(x')=0}[\exp(-\|v_1 - v_0\|^2/2\tau)] \\ &= \exp(-\|v_0 - v_1\|^2/2\tau) = \exp(-2/\tau). \end{aligned} \quad (12)$$

The first equality follows from class balance, and the third equality follows from the definition of the regular simplex. Therefore, $\mu = \delta_v$ minimizes $L_{\text{diff}}(f)$.

$L_{\text{same}}(f)$ For $L_{\text{same}}(f)$, we can directly use the proof of Theorem 1 in Wang & Isola (2020) to show that the optimal measure that minimizes L_{same} also minimizes $\log \mathbb{E}_{x, x^+}[\exp(-\|f(x) - f(x^+)\|^2/2\tau)]$. We equivalently consider minimizing $\mathbb{E}_{x, x^+}[\exp(-\|f(x) - f(x^+)\|^2/2\tau)]$. We can write this as

$$\begin{aligned} \mathbb{E}_{x, x^+}[\exp(-\|f(x) - f(x^+)\|^2/2\tau)] &= \frac{1}{2} \mathbb{E}_{h(x)=h(x')=0}[\exp(-\|f(x) - f(x^+)\|^2/2\tau)] \\ &\quad + \frac{1}{2} \mathbb{E}_{h(x)=h(x')=1}[\exp(-\|f(x) - f(x^+)\|^2/2\tau)]. \end{aligned} \quad (13)$$

Using the infinite encoder assumption, we can equivalently consider the following minimization problem over the hypersphere, where $u, u' \in \mathcal{S}^{d-1}$:

$$\text{minimize}_{\mu_0, \mu_1} \frac{1}{2} \int \int \exp(-\|u - u'\|^2/2\tau) d\mu_0(u) d\mu_0(u') + \frac{1}{2} \int \int \exp(-\|u - u'\|^2/2\tau) d\mu_1(u) d\mu_1(u') \quad (14)$$

Each of these integrals can be minimized individually, and the problem becomes equivalent to having both μ_0 and μ_1 minimize the Gaussian $\frac{1}{2\tau}$ -energy. Using Proposition 4.4.1 and Theorem 6.2.1 of (Borodachov et al., 2019), the optimal solution is $\mu_0 = \mu_1 = \sigma_{d-1}$, the normalized surface area measure. Therefore, $\mu = \sigma_{d-1}$. □

Theorem 2. Let $c_{\tau, d} = \frac{2 + \frac{1}{\tau} - \sqrt{\frac{1}{\tau}(-2 + \frac{1}{\tau}) - 2 \log W_{1/2\tau}(\mathcal{S}^{d-1})}}{3}$, where $W_{1/2\tau}(\mathcal{S}^{d-1})$ is the Wiener constant of the Gaussian $\frac{1}{2\tau}$ -energy on \mathcal{S}^{d-1} , which is defined as

$$W_{1/2\tau}(\mathcal{S}^{d-1}) = \frac{2^{d-2} \Gamma(d/2)}{\sqrt{\pi} \Gamma((d-1)/2)} \int_0^1 \exp\left(-\frac{2u}{\tau}\right) (u(1-u))^{(d-3)/2} du,$$

where the Gamma function is $\Gamma(z) = \int_0^\infty x^{z-1} e^{-x} dx$ for $z > 0$. Then, when $\alpha \in (2/3, c_{\tau,d})$, $\theta^* = \arcsin \sqrt{\frac{\tau}{2} \log \frac{3\alpha-1}{3-3\alpha}}$ minimizes $L_{\text{spread}}(\boldsymbol{\mu}_\theta, \alpha)$ and satisfies $L_{\text{spread}}(\boldsymbol{\mu}_{\theta^*}, \alpha) \leq \min_{\boldsymbol{\mu} \in \{\boldsymbol{\delta}_v, \boldsymbol{\sigma}_{d-1}\}} L_{\text{spread}}(\boldsymbol{\mu}, \alpha)$.

Proof. Because the augmentations only play a role in $L_{\text{aug}}(f)$ and are disjoint, the condition that $f(x) = f(a(x))$ a.s. is compatible with any of the other three losses in Proposition 1. Therefore, we focus on analyzing the combined weighted loss $(1-\alpha)L_{\text{align}}(f) + (1-\alpha)L_{\text{diff}}(f) + \alpha L_{\text{same}}(f)$. We restate the loss:

$$L(\boldsymbol{\mu}, \alpha) = (1-\alpha) \mathbb{E}_x \left[\log \mathbb{E}_{x^-} \left[\exp \left(-\frac{1}{2\tau} \|f(x) - f(x^-)\|^2 \right) \right] \right] \\ + \alpha \mathbb{E}_x \left[\log \mathbb{E}_{x^+} \left[\exp \left(-\frac{1}{2\tau} \|f(x) - f(x^+)\|^2 \right) \right] \right] + (1-\alpha) \mathbb{E}_{x,x^+} \left[\frac{1}{2\tau} \|f(x) - f(x^+)\|^2 \right].$$

We restate the definition of the measure $\boldsymbol{\mu}_\theta = \{\mu_{0,\theta}, \mu_{1,\theta}\}$ for $\theta \in (0, \pi/2]$ that involves ‘‘splitting’’ $\boldsymbol{\delta}_v$ by angle θ . Without loss of generality, suppose that $v_0 = e_1$ and $v_1 = -e_1$, where $e_1 \in \mathbb{R}^d$ is a standard basis vector $[1, 0, \dots, 0]$ where all but the first two elements are always 0. For v_0 , we consider the vectors $v_{0,\theta} = [\cos\theta, \sin\theta, \dots, 0]$ and $v_{0,-\theta} = [\cos\theta, -\sin\theta, \dots, 0]$. For v_1 , we consider the vectors $v_{1,\theta} = [-\cos\theta, -\sin\theta, \dots, 0]$ and $v_{1,-\theta} = [-\cos\theta, \sin\theta, \dots, 0]$. Let $\mu_{0,\theta} = \frac{1}{2}\boldsymbol{\delta}_{v_{0,\theta}} + \frac{1}{2}\boldsymbol{\delta}_{v_{0,-\theta}}$ and let $\mu_{1,\theta} = \frac{1}{2}\boldsymbol{\delta}_{v_{1,\theta}} + \frac{1}{2}\boldsymbol{\delta}_{v_{1,-\theta}}$. That is, each class-conditional measure is a mixture on two points separated by θ .

The first step is to show that for some range of α , $\min_\theta L(\boldsymbol{\mu}_\theta, \alpha) < L(\boldsymbol{\delta}_v, \alpha)$. We have that

$$L(\boldsymbol{\delta}_v, \alpha) = (1-\alpha) \left(\frac{1}{2} \log \exp \left(-\frac{4}{2\tau} \right) + \frac{1}{2} \log \exp \left(-\frac{4}{2\tau} \right) \right) + \alpha \log \exp(0) + (1-\alpha) \cdot 0 = -\frac{2(1-\alpha)}{\tau}. \quad (15)$$

and

$$L(\boldsymbol{\mu}_\theta, \alpha) = (1-\alpha) \log \left(\frac{1}{2} \exp \left(-\frac{4}{2\tau} \right) + \frac{1}{2} \exp \left(-\frac{4\cos^2\theta}{2\tau} \right) \right) + \alpha \log \left(\frac{1}{2} + \frac{1}{2} \exp \left(-\frac{4\sin^2\theta}{2\tau} \right) \right) \\ + \frac{1-\alpha}{2\tau} \cdot \frac{1}{2} (4\sin^2\theta) \\ = -\log 2 - \frac{2(1-\alpha)}{\tau} + (1-\alpha) \log \left(1 + \exp \left(\frac{2\sin^2\theta}{\tau} \right) \right) + \alpha \log \left(1 + \exp \left(-\frac{2\sin^2\theta}{\tau} \right) \right) \\ + \frac{(1-\alpha)\sin^2\theta}{\tau}. \quad (16)$$

We now compute the derivative $\frac{\partial L(\boldsymbol{\mu}_\theta, \alpha)}{\partial \theta}$ to find local minima:

$$\frac{\partial L(\boldsymbol{\mu}_\theta, \alpha)}{\partial \theta} = (1-\alpha) \frac{\exp(2\sin^2\theta/\tau) \cdot 4\sin\theta\cos\theta/\tau}{1 + \exp(2\sin^2\theta/\tau)} + \alpha \frac{\exp(-2\sin^2\theta/\tau) \cdot -4\sin\theta\cos\theta/\tau}{1 + \exp(-2\sin^2\theta/\tau)} + \frac{(1-\alpha)4\sin\theta\cos\theta}{\tau} \\ = \frac{4\sin\theta\cos\theta}{\tau} \left((1-\alpha) \frac{\exp(2\sin^2\theta/\tau)}{1 + \exp(2\sin^2\theta/\tau)} - \alpha \frac{\exp(-2\sin^2\theta/\tau)}{1 + \exp(-2\sin^2\theta/\tau)} + \frac{1-\alpha}{2} \right).$$

Note that $\sin\theta$ and $\cos\theta$ are positive for $\theta \in (0, \pi/2]$. Next, for notational simplicity let $x = \frac{2\sin^2\theta}{\tau}$. Then, we can equivalently evaluate

$$(1-\alpha) \frac{e^x}{1+e^x} - \alpha \frac{e^{-x}}{1+e^{-x}} + \frac{1-\alpha}{2} = (1-\alpha) \frac{e^x}{1+e^x} - \alpha \frac{1}{1+e^x} + \frac{1-\alpha}{2} \\ = \frac{e^x - \alpha(1+e^x)}{1+e^x} + \frac{1-\alpha}{2} \\ = \frac{e^x}{1+e^x} + \frac{1}{2} - \frac{3\alpha}{2}.$$

Setting this equal to 0, we get that $\alpha = \frac{3e^x+1}{3e^x+3}$ and $x = \log \frac{3\alpha-1}{3-3\alpha}$. Since $x \in (0, 2/\tau]$, we have that if $\alpha \in \left(\frac{2}{3}, \frac{3\exp(2/\tau)+1}{3 \cdot 3\exp(2/\tau)+3} \right)$, there exists a local optima over $\theta \in (0, \pi/2]$.

Moreover, we observe that when $\alpha \leq 2/3$, we have that $\frac{e^x}{1+e^x} + \frac{1}{2} - \frac{3\alpha}{2} \geq \frac{e^x}{1+e^x} - \frac{1}{2} \geq 0$, which means that $L(\boldsymbol{\mu}_\theta, \alpha)$ increases in θ for $\alpha \leq 2/3$. Therefore, when $\alpha \leq 2/3$, class collapse is *always better* no matter the angle, and $L(\boldsymbol{\delta}_v, \alpha) \leq \min_\theta L(\boldsymbol{\mu}_\theta, \alpha)$.

Next, we consider when $\alpha \geq \frac{3\exp(2/\tau)+1}{3\exp(2/\tau)+3}$. In this case, $\frac{e^x}{1+e^x} + \frac{1}{2} - \frac{3\alpha}{2} < 0$, so $L(\boldsymbol{\mu}_\theta, \alpha)$ is decreasing in θ . This means that any nonzero θ in this setting is going to result in a smaller loss than the class-collapsed loss.

Lastly, we consider the intermediate case of $\alpha \in \left(\frac{2}{3}, \frac{3\exp(2/\tau)+1}{3}\right)$. Plugging back in $x = \log \frac{3\alpha-1}{3-3\alpha}$ back into $L(\boldsymbol{\mu}_\theta, \alpha)$ in (16), we have

$$\begin{aligned} L(\boldsymbol{\mu}_\theta, \alpha) &= -\log 2 - \frac{2(1-\alpha)}{\tau} + (1-\alpha)\log\left(1 + \frac{3\alpha-1}{3-3\alpha}\right) + \alpha\log\left(1 + \frac{3-3\alpha}{3\alpha-1}\right) + \frac{1-\alpha}{2} \cdot \log \frac{3\alpha-1}{3-3\alpha} \\ &= -\log 2 - \frac{2(1-\alpha)}{\tau} + (1-\alpha)\log \frac{2}{3-3\alpha} + \alpha\log \frac{2}{3\alpha-1} + \frac{1-\alpha}{2}\log(3\alpha-1) - \frac{1-\alpha}{2}\log(3-3\alpha) \\ &= -\frac{2(1-\alpha)}{\tau} - \frac{3-3\alpha}{2}\log(3-3\alpha) - \frac{3\alpha-1}{2}\log(3\alpha-1). \end{aligned} \quad (17)$$

Note that $(3-3\alpha)\log(3-3\alpha) + (3\alpha-1)\log(3\alpha-1)$ equals 0 at $\alpha = 2/3$ and is increasing in α . Therefore, we have that $-\frac{2(1-\alpha)}{\tau} - \frac{3-3\alpha}{2}\log(3-3\alpha) - \frac{3\alpha-1}{2}\log(3\alpha-1) \leq -\frac{2(1-\alpha)}{\tau} = L(\boldsymbol{\delta}_v, \alpha)$. Therefore, for $\alpha \in \left(\frac{2}{3}, \frac{3\exp(2/\tau)+1}{3}\right)$, the optimal θ^* satisfies $L(\boldsymbol{\mu}_\theta, \alpha) \leq L(\boldsymbol{\delta}_v, \alpha)$. In particular, solving $\frac{2\sin^2\theta^*}{\tau} = \log \frac{3\alpha-1}{3-3\alpha}$ gives us $\theta^* = \arcsin \sqrt{\frac{\tau}{2} \log \frac{3\alpha-1}{3-3\alpha}}$.

Therefore, our analysis in these three ranges of α suggest that the optimal embedding geometry is *not collapsed* when $\alpha \geq \frac{2}{3}$.

Next, we want to understand when the optimal embedding geometry is not $\boldsymbol{\sigma}_{d-1}$. A sufficient condition for this is to show that there exists an $\alpha \geq \frac{2}{3}$ where $\min_\theta L(\boldsymbol{\mu}_\theta, \alpha) \leq L(\boldsymbol{\sigma}_{d-1}, \alpha)$. We first compute an upper bound on $\min_\theta L(\boldsymbol{\mu}_\theta, \alpha)$ for $\alpha > \frac{2}{3}$. Recall that our loss from (17) can be written as

$$L(\boldsymbol{\mu}_\theta, \alpha) = -\frac{2(1-\alpha)}{\tau} - \log 2 - \frac{3-3\alpha}{2}\log\left(\frac{3-3\alpha}{2}\right) - \frac{3\alpha-1}{2}\log\left(\frac{3\alpha-1}{2}\right). \quad (18)$$

For ease of notation, let $x = \frac{3-3\alpha}{2} \in (0, 1)$. We show that $f(x) = x\log x + (1-x)\log(1-x)$ can be lower bounded quadratically. Performing a Taylor expansion at $x = 0.5$, we have that $x\log x + (1-x)\log(1-x) \approx -\log 2 + 2(x-1/2)^2$. We claim that $x\log x + (1-x)\log(1-x) \geq -\log 2 + 2(x-1/2)^2$. Note that the two sides are equal when $x = 1/2$, so proving this inequality is equivalent to showing that $f'(x) \geq 4(x-1/2)$ for $x \geq 1/2$ and $f'(x) < 4(x-1/2)$ for $x < 1/2$. $f'(x) = \log \frac{x}{1-x}$ is equal to $4(x-1/2)$ at $x = 1/2$, so we want to show that $f''(x) \geq 4$ for all x . $f''(x) = \frac{1}{x(1-x)}$ satisfies this inequality. Therefore, (18) becomes

$$L(\boldsymbol{\mu}_\theta, \alpha) \leq -\frac{2(1-\alpha)}{\tau} - 2\left(1 - \frac{3\alpha}{2}\right)^2.$$

Next, we compute $L(\boldsymbol{\sigma}_{d-1}, \alpha)$. With $\boldsymbol{\sigma}_{d-1}$, $f(x)$, $f(x^+)$, and $f(x^-)$ are all uniformly distributed on the hypersphere. Therefore,

$$L(\boldsymbol{\sigma}_{d-1}, \alpha) = \mathbb{E}_x \left[\log \mathbb{E}_{x^+} \left[\exp(-\|f(x) - f(x^+)\|^2/2\tau) \right] \right] - \frac{1-\alpha}{2\tau} \iint -\|u - u'\|^2 d\sigma_{d-1}(u) d\sigma_{d-1}(u'). \quad (19)$$

From Wang & Isola (2020), we know that when $f(x)$ and $f(x^+)$ are drawn from a distribution with measure $\boldsymbol{\sigma}_{d-1}$, it holds that $\mathbb{E}_x \left[\log \mathbb{E}_{x^+} \left[\exp(-\|f(x) - f(x^+)\|^2/2\tau) \right] \right] = \log \mathbb{E}_{x, x^+} \left[\exp(-\|f(x) - f(x^+)\|^2/2\tau) \right]$.

Define the Gaussian $\frac{1}{2\tau}$ -energy of σ_{d-1} on \mathcal{S}^{d-1} as $I_{1/2\tau}[\sigma_{d-1}] = \int_{\mathcal{S}^{d-1}} \int_{\mathcal{S}^{d-1}} \exp\left(-\frac{1}{2\tau}\|u - u'\|^2\right) d\sigma_{d-1}(u) d\sigma_{d-1}(u')$. Then, (19) becomes

$$L(\boldsymbol{\sigma}_{d-1}, \alpha) = (1-\alpha)\log I_{1/2\tau}[\sigma_{d-1}] + \alpha\log I_{1/2\tau}[\sigma_{d-1}] - \frac{1-\alpha}{2\tau} \iint -\|u - u'\|^2 d\sigma_{d-1}(u) d\sigma_{d-1}(u'). \quad (20)$$

From Theorem 4.6.5 of Borodachov et al. (2019), any measure μ that has mass centered at the origin, i.e. $\int u d\mu(u) = 0$, minimizes the energy $I_{-2}[\mu] = \int \int -\|u - u'\|^2 d\mu(u) d\mu(u')$. Therefore, $\int \int -\|u - u'\|^2 d\sigma_{d-1}(u) d\sigma_{d-1}(u')$ is equivalent to the energy $I_{-2}[\frac{1}{2}\delta_{v_0} + \frac{1}{2}\delta_{v_1}] = -\frac{1}{2}(2^2) = -2$, since a measure on two points with probability 1/2 each has mass centered at the origin. Plugging this back into $L(\boldsymbol{\sigma}_{d-1}, \alpha)$ in (20), we have

$$L(\boldsymbol{\sigma}_{d-1}, \alpha) = \log I_{1/2\tau}[\sigma_{d-1}] + \frac{1-\alpha}{\tau}. \quad (21)$$

From Proposition 4.4.1 and Theorem 6.2.1 of Borodachov et al. (2019), σ_{d-1} is the unique equilibrium measure for the Gaussian $\frac{1}{2\tau}$ kernel on \mathcal{S}^{d-1} , and as a result $I_{1/2\tau}[\sigma_{d-1}]$ is equal to the Wiener constant $W_{1/2\tau}(\mathcal{S}^{d-1})$. By Proposition A.11.2 of (Borodachov et al., 2019), this has the value

$$W_{1/2\tau}(\mathcal{S}^{d-1}) = \frac{2^{d-2}\Gamma(d/2)}{\sqrt{\pi}\Gamma((d-1)/2)} \int_0^1 \exp\left(-\frac{2u}{\tau}\right) (u(1-u))^{(d-3)/2} du, \quad (22)$$

where the Gamma function is $\Gamma(z) = \int_0^\infty x^{z-1} e^{-x} dx$ for $z > 0$.

Therefore, to prove that there exists a μ_θ that has lower loss than σ_{d-1} , we must find $\alpha > 2/3$ that satisfies

$$2\left(1 - \frac{3\alpha}{2}\right)^2 + \frac{3(1-\alpha)}{\tau} + \log W_{1/2\tau}(\mathcal{S}^{d-1}) \geq 0.$$

This expression is quadratic in α , and we solve it to get that $c(\tau, d) \leq \frac{2 + \frac{1}{\tau} - \sqrt{\frac{1}{\tau}(-2 + \frac{1}{\tau}) - 2\log W_{1/2\tau}(\mathcal{S}^{d-1})}}{3}$.

□

C.2. Proofs for Section 4

Proposition 2. *Let \mathcal{F} be the set of infinite encoders. Then L_{spread} is invariant on class-fixing permutations under \mathcal{F} .*

Proof. We know that any $f^\pi \in \mathcal{F}$ can satisfy $f^\pi(x_i) = f(x_{\pi(i)})$, since the infinite encoder assumption means that f^π can be arbitrarily fit to any data. Therefore, we only need to show that L_{spread} does not change when f^π , which permutes within classes, is used instead of f .

For a given batch B and f , L_{spread} is constructed as defined in Section 2.2. The numerator of L_{sup} can be written as $\frac{1}{|B|} \frac{1}{|P(i, B)|} \sum_{i=1}^{|B|} \sum_{x^+ \in P(i, B)} \log \sigma_f(x_i, x^+)$. This is a summation over the representations of all positive pairs in the batch. Therefore, a permutation π within each class that changes the assignments to the representations will not change the value of this quantity, and $\frac{1}{|B|} \frac{1}{|P(i, B)|} \sum_{i=1}^{|B|} \sum_{x^+ \in P(i, B)} \log \sigma_{f^\pi}(x_i, x^+) = \frac{1}{|B|} \frac{1}{|P(i, B)|} \sum_{i=1}^{|B|} \sum_{x^+ \in P(i, B)} \log \sigma_f(x_i, x^+)$.

Next, the denominator of L_{sup} can be written as $\frac{1}{|P(i, B)|} \sum_{i=1}^{|B|} \sum_{x^+ \in P(i, B)} \log \left(\sigma_f(x_i, x^+) + \sum_{x^- \in N(i, B)} \sigma_f(x_i, x^-) \right)$. Every single positive pair and negative pair is included in this expression, so this quantity is class-fixing permutation invariant.

The numerator of L_{CNCE} is $\frac{1}{|B|} \sum_{i=1}^{|B|} \log \sigma_f(x_i, a(x_i))$. Since an augmentation of x_i is a function of x_i and is disjoint from augmentations of other points, this quantity is class-fixing permutation invariant.

Lastly, the denominator of L_{CNCE} is $\frac{1}{|B|} \sum_{i=1}^{|B|} \log \left(\sum_{x^+ \in P(i, B)} \sigma_f(x_i, x^+) \right)$. From the same logic as the numerator of L_{sup} , any permutation within the class will still allow each x_i to be compared with all other points in x_i 's class, hence being class-fixing permutation invariant.

Therefore, under the infinite encoder assumption where all f^π are valid, L_{spread} is permutation invariant. □

Theorem 3. *Denote $r_f(z, z') = c^2 \delta_f(z, z')^2 - |\text{Var}_f[z] - \text{Var}_f[z']|$. With probability $1 - \delta$, the coarse-to-fine error is at most*

$$L_{\gamma, f}(z) \leq \frac{\sigma_f(z)}{\sqrt{r_f(z, z') - 2\log\gamma}} + \mathcal{O}\left(\left(\frac{d\log(d/\delta)}{m_z \wedge m_{z'}}\right)^{1/4}\right).$$

under the boundary condition that $r_f(z, z') - 2\log\gamma \geq 16\sqrt{\frac{2d\log(8d/\delta)}{m_z \wedge m_{z'}}} + \frac{2d\log(8d/\delta)}{m_z}$.

Proof. From (6), our loss function can be written as $L_{\gamma, f}(z) = \Pr_{x \sim \mathcal{P}_z}(f(x)^\top (W_z - W_{z'}) \leq \log\gamma)$. Note that $\|f(x) - W_z\|^2 = \|f(x)\|^2 + \|W_z\|^2 - 2f(x)^\top W_z = 1 + \|W_z\|^2 - 2f(x)^\top W_z$, which means that $f(x)^\top W_z = \frac{1}{2}(1 + \|W_z\|^2 - \|f(x) - W_z\|^2)$. We can thus write our loss as

$$\begin{aligned} L_{\gamma, f}(z) &= \Pr_{x \sim \mathcal{P}_z}(f(x)^\top (W_z - W_{z'}) \leq \log\gamma) \\ &= \Pr_{x \sim \mathcal{P}_z}(\|f(x) - W_z\| \geq (\|f(x) - W_{z'}\|^2 - 2\log\gamma + \|W_z\|^2 - \|W_{z'}\|^2)^{1/2}). \end{aligned} \quad (23)$$

We bound terms in this probability individually. First, we can write

$$\|f(x) - W_z\| \leq \|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\| + \xi_z, \quad (24)$$

where ξ_z again is constructed as $\xi_z = \|W_z - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\| = \left\| \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s, z}} f_{SC}(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f_{SC}(x)] \right\|$.

Next, by the reverse triangle inequality we can write

$$\begin{aligned}
 \|f(x) - W_{z'}\|^2 &\geq \left| \|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\| - \|\mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)] - W_{z'}\| \right|^2 \\
 &\geq \|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\|^2 - 2\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\| \xi_{z'} \\
 &\geq \|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\|^2 - 4\xi_{z'} \\
 &\geq c^2 \cdot \mathbb{E}_{x \sim \mathcal{P}_z, x' \sim \mathcal{P}_{z'}}[\|f(x) - f(x')\|]^2 - 4\xi_{z'}.
 \end{aligned} \tag{25}$$

Note the following decomposition by Jensen's inequality:

$$\begin{aligned}
 s_f(y) &= \mathbb{E}_{h(x)=y}[\|f(x) - \mathbb{E}_{h(x)=y}[f(x)]\|] \leq p(z|y)^2 \mathbb{E}_{x \sim \mathcal{P}_z}[\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\|] \\
 &\quad + p(z'|y)^2 \mathbb{E}_{x \sim \mathcal{P}_{z'}}[\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\|] + p(z|y)p(z'|y) \mathbb{E}_{x \sim \mathcal{P}_z}[\|f(x) - \mathbb{E}_{x' \sim \mathcal{P}_{z'}}[f(x')]\|] \\
 &\quad + p(z|y)p(z'|y) \mathbb{E}_{x \sim \mathcal{P}_{z'}}[\|f(x) - \mathbb{E}_{x' \sim \mathcal{P}_z}[f(x')]\|] \\
 &\leq p(z|y)^2 \sigma_f(z) + p(z'|y)^2 \sigma_f(z') + 2p(z|y)p(z'|y) \mathbb{E}_{x \sim \mathcal{P}_z, x' \sim \mathcal{P}_{z'}}[\|f(x) - f(x')\|],
 \end{aligned}$$

and thus, $\mathbb{E}_{x \sim \mathcal{P}_z, x' \sim \mathcal{P}_{z'}}[\|f(x) - f(x')\|] \geq \delta_f(z, z') = \frac{1}{p(z|y)p(z'|y)} \cdot (s_f(y) - p(z|y)^2 \sigma_f(z) - p(z'|y)^2 \sigma_f(z'))$. Then, (25) becomes

$$\|f(x) - W_{z'}\|^2 \geq c^2 \cdot \delta_f(z, z')^2 - 4\xi_{z'}. \tag{26}$$

Finally, we bound $\|W_z\|^2 - \|W_{z'}\|^2$. Recall that $\|W_z\|^2 = \left\| \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f(x) \right\|^2$. We can write the empirical variance $\frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} \|f(x) - \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f(x)\|^2$ as $1 + \|W_z\|^2 - \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f(x)^\top \frac{2}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f(x) = 1 - \|W_z\|^2$, and therefore,

$$\begin{aligned}
 \left| \|W_z\|^2 - \|W_{z'}\|^2 \right| &\leq \left| \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} \|f(x) - \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f(x)\|^2 - \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} \|f(x) - \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} f(x)\|^2 \right| \\
 &\leq \left| \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} \left\| f(x) - \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} f(x) \right\|^2 - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\|^2] \right| \\
 &\quad + \left| \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} \left\| f(x) - \frac{1}{m_z} \sum_{x \in \mathcal{D}_{s,z}} f(x) \right\|^2 - \mathbb{E}_{x \sim \mathcal{P}_z}[\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\|^2] \right| \\
 &\quad + |\text{Var}_f[z] - \text{Var}_f[z']|.
 \end{aligned} \tag{27}$$

where $\text{Var}_f[z] = \mathbb{E}_{x \sim \mathcal{P}_z}[\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\|^2]$, and $\text{Var}_f[z']$ is similarly defined. We decompose the first term in (27) and bound it by

$$\begin{aligned}
 &\left| \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} \left\| f(x) - \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} f(x) \right\|^2 - \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} \|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\|^2 \right| \\
 &+ \left| \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} \|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\|^2 - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\|^2] \right| \\
 &= \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} \left| \left\| f(x) - \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} f(x) \right\|^2 - \|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\|^2 \right| + \zeta_{z'} \\
 &\leq \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} 4 \left| \left\| f(x) - \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} f(x) \right\| - \|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\| \right| + \zeta_{z'} \\
 &\leq \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} 4 \left| \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)] \right| + \zeta_{z'} \\
 &\leq 4\xi_{z'} + \zeta_{z'}.
 \end{aligned}$$

where $\zeta_{z'} = \left| \frac{1}{m_{z'}} \sum_{x \in \mathcal{D}_{s,z'}} \left\| f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)] \right\|^2 - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_{z'}}[f(x)]\|^2] \right|$ can be bounded by standard concentration inequalities. Therefore, $\left| \|W_z\|^2 - \|W_{z'}\|^2 \right|$ is bounded by

$$\left| \|W_z\|^2 - \|W_{z'}\|^2 \right| \leq |\text{Var}_f[z] - \text{Var}_f[z']| + 4\xi_z + 4\xi_{z'} + \zeta_z + \zeta_{z'}. \tag{28}$$

Then, combining (24), (26), and (28), the loss in (23) becomes

$$L_{\gamma, f}(z) \leq \Pr_{x \sim \mathcal{P}_z} (\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\| \geq (r_f(z, z') - 8\xi_{z'} - 4\xi_z - \zeta_z - \zeta_{z'} - 2\log\gamma)^{1/2} - \xi_z). \quad (29)$$

Next, we note that $\xi_z \leq \sqrt{\frac{2d\log(2d/\delta)}{m_z}}$ with probability at least $1 - \delta$. Applying Hoeffding's inequality on ζ_z gives us $\zeta_z \leq \sqrt{\frac{8\log(2/\delta)}{m_z}}$ with probability at least $1 - \delta$. Applying a union bound, we have that with probability $1 - \delta$, (29) satisfies

$$L_{\gamma, f}(z) \leq \Pr_{x \sim \mathcal{P}_z} \left(\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\| \geq \left(r_f(z, z') - 10\sqrt{\frac{2d\log(8d/\delta)}{m_{z'}}} - 6\sqrt{\frac{2d\log(8d/\delta)}{m_z}} - 2\log\gamma \right)^{1/2} - \sqrt{\frac{2d\log(8d/\delta)}{m_z}} \right).$$

Finally, we can apply Markov's inequality under the condition that $r_f(z, z') - 2\log\gamma \geq \frac{2d\log(8d/\delta)}{m_z} + 6\sqrt{\frac{2d\log(8d/\delta)}{m_z}} + 10\sqrt{\frac{2d\log(8d/\delta)}{m_{z'}}$, or equivalently $r_f(z, z') - 2\log\gamma \geq \frac{2d\log(8d/\delta)}{m_z} + 16\sqrt{\frac{2d\log(8d/\delta)}{m_z \wedge m_{z'}}$. With probability at least $1 - \delta$, our loss is bounded by

$$\begin{aligned} L_{\gamma, f}(z) &\leq \frac{\mathbb{E}_{x \sim \mathcal{P}_z}[f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]]}{\sqrt{r_f(z, z') - 2\log\gamma - 16\sqrt{\frac{2d\log(8d/\delta)}{m_z \wedge m_{z'}}} - \sqrt{\frac{2d\log(8d/\delta)}{m_z}}} \\ &\leq \frac{\sigma_f(z)}{\sqrt{r_f(z, z') - 2\log\gamma}} + \mathcal{O}\left(\left(\frac{d\log(d/\delta)}{m_z \wedge m_{z'}}\right)^{1/4}\right). \end{aligned}$$

□

Lemma 1. Let \mathcal{F}_{K_L} be the class of K_L -Lipschitz encoders. Then for any $f_{K_L} \in \mathcal{F}_{K_L}$, $\sigma_{f_{K_L}}(z) \leq K_L \sigma_z$.

Proof. Using Jensen's inequality and then Lipschitzness of f ,

$$\begin{aligned} \sigma_f(z) &= \mathbb{E}_{x \sim \mathcal{P}_z} [\|f(x) - \mathbb{E}_{x \sim \mathcal{P}_z}[f(x)]\|] = \mathbb{E}_{x \sim \mathcal{P}_z} \left[\left\| \int (f(x) - f(x'))p(x'|z)dx' \right\| \right] \\ &\leq \mathbb{E}_{x \sim \mathcal{P}_z} \left[\int \|f(x) - f(x')\|p(x'|z)dx' \right] \leq K_L \cdot \mathbb{E}_{x \sim \mathcal{P}_z} \left[\int \|x - x'\|p(x'|z)dx' \right] \\ &= K_L \cdot \mathbb{E}_{x, x' \sim \mathcal{P}_z} [\|x - x'\|] = K_L \delta_z. \end{aligned}$$

□

Lemma 2. For any $g \in \mathcal{G}$, suppose there exists a $K_g > 0$ such that g is "reverse Lipschitz", satisfying $\|f_{AE}(x) - f_{AE}(x')\| \leq K_g \|g(f_{AE}(x)) - g(f_{AE}(x'))\|$, and there exists finite b such that the reconstruction loss satisfies $\max_x \|g(f_{AE}(x)) - x\|^2 \leq b$.

Then with probability at least $1 - \delta$,

$$\sigma_{f_{AE}}(z) \leq \frac{2K_g}{p(z|y)} \left(\hat{L}_{AE}(\mathcal{D}_y) + 2\mathfrak{R}_{n_y}^2(\mathcal{G} \circ \mathcal{F}_{AE}, id_{\mathcal{X}}) + b \sqrt{\frac{\log(1/\delta)}{2n_y}} \right)^{1/2} + K_g \sigma_z,$$

where $id_{\mathcal{X}}$ is the identity function on \mathcal{X} , and $p(z|y) = \frac{p(z)}{p(y)}$ is the probability that x drawn from $p(\cdot|y)$ has label z .

Proof. We can decompose $\sigma_{f_{AE}}(z)$ into the following using the assumption on the decoder g :

$$\begin{aligned} \sigma_{f_{AE}}(z) &= \mathbb{E}_{x \sim \mathcal{P}_z} [\|f_{AE}(x) - \mathbb{E}_{x' \sim \mathcal{P}_z}[f_{AE}(x')]\|] \leq \mathbb{E}_{x \sim \mathcal{P}_z} \left[\int \|f_{AE}(x) - f_{AE}(x')\|p(x'|z)dx' \right] \\ &\leq \mathbb{E}_{x \sim \mathcal{P}_z} \left[\int K_g \cdot \|g(f_{AE}(x)) - g(f_{AE}(x'))\|p(x'|z)dx' \right] \\ &\leq \mathbb{E}_{x \sim \mathcal{P}_z} \left[\int K_g (\|g(f_{AE}(x)) - x\| + \|x - x'\| + \|x' - g(f_{AE}(x'))\|)p(x'|z)dx' \right] \\ &= 2K_g \mathbb{E}_{x \sim \mathcal{P}_z} [\|g(f_{AE}(x)) - x\|] + K_g \mathbb{E}_{x, x' \sim \mathcal{P}_z} [\|x - x'\|]. \end{aligned} \quad (30)$$

Note that $\mathbb{E}_{x \sim \mathcal{P}_y} [\|g(f_{AE}(x)) - x\|] = \sum_{k \in S_y} \mathbb{E}_{x \sim \mathcal{P}_k} [\|g(f_{AE}(x)) - x\|] p(k|y) \geq \mathbb{E}_{x \sim \mathcal{P}_z} [\|g(f_{AE}(x)) - x\|] p(z|y)$, so (30) becomes

$$\begin{aligned} \sigma_{f_{AE}}(z) &\leq \frac{2K_g}{p(z|y)} \mathbb{E}_{x \sim \mathcal{P}_y} [\|g(f_{AE}(x)) - x\|] + K_g \mathbb{E}_{x, x' \sim \mathcal{P}_z} [\|x - x'\|] \\ &\leq \frac{2K_g}{p(z|y)} \sqrt{\mathbb{E}_{x \sim \mathcal{P}_y} [\|g(f_{AE}(x)) - x\|^2]} + K_g \mathbb{E}_{x, x' \sim \mathcal{P}_z} [\|x - x'\|] \\ &= \frac{2K_g}{p(z|y)} \sqrt{\hat{L}_{AE}(n_y) + \mathbb{E}_{x \sim \mathcal{P}_y} [\|g(f_{AE}(x)) - x\|^2] - \hat{L}_{AE}(n_y)} + K_g \sigma_z. \end{aligned} \quad (31)$$

where $\hat{L}_{AE}(\mathcal{D}_y) = \frac{1}{n_y} \sum_{x \in \mathcal{D}_y} \|g(f_{AE}(x)) - x\|^2$ is the reconstruction error on the training data \mathcal{D}_y . We bound the generalization error $\mathbb{E}_{x \sim \mathcal{P}_y} [\|g(f_{AE}(x)) - x\|^2] - \hat{L}_{AE}(\mathcal{D}_y)$ using Theorem 3.3 of Mohri et al. (2018) to get that with probability at least $1 - \delta$,

$$\sigma_{f_{AE}}(z) \leq \frac{2K_g}{p(z|y)} \left(\hat{L}_{AE}(\mathcal{D}_y) + 2\mathfrak{R}_{n_y}^2(\mathcal{G} \circ \mathcal{F}_{AE}, \text{id}_{\mathcal{X}}) + b \sqrt{\frac{\log(1/\delta)}{2n_y}} \right)^{1/2} + K_g \sigma_z.$$

Finally, we compare against using a general autoencoder trained on the entire dataset of n points. This yields a bound

$$\sigma_{f_{AE}}(z) \leq \frac{2K_g}{p(z)} \left(\hat{L}_{AE}(\mathcal{D}) + 2\mathfrak{R}_n^2(\mathcal{G} \circ \mathcal{F}_{AE}, \text{id}_{\mathcal{X}}) + b \sqrt{\frac{\log(1/\delta)}{2n}} \right)^{1/2} + K_g \sigma_z,$$

where the only change in the result is that $p(z|y)$ is replaced with $p(z)$, the overall proportion of the subclass, and n_y is replaced with n . This highlights a tradeoff: $p(z|y) > p(z)$, but $n_y < n$. A class-conditional autoencoder may suffer from poorer generalization due to lower sample size, but its relative worst case performance on z in expectation is better. On the other hand, a general autoencoder is learned on more data, but its relative worst case performance on z in expectation is worse since the subclass is more rare w.r.t. the training data. \square

Lemma 3. For $a \in \mathcal{A}$ and any $x, x' \in \mathcal{X}$, suppose that $f_{aug} \in \mathcal{F}_{aug}$ satisfies $\|f_{aug}(a(x)) - f_{aug}(a(x'))\| \leq K_{aug} \|a(x) - a(x')\|$ for some K_{aug} and that $f(a(x)) = f(x)$ for $x \in \mathcal{D}$. Denote $\sigma_z^{aug} = \mathbb{E}_{x, x' \sim \mathcal{P}_z} [\|a(x) - a(x')\|]$. Then with probability at least $1 - \delta$,

$$\sigma_{f_{aug}}(z) \leq \frac{2}{p(z)} \left(2\mathfrak{R}_n^1(\mathcal{F}_{aug}, \mathcal{F}_{aug} \circ \mathcal{A}) + \sqrt{\frac{2\log(1/\delta)}{n}} \right) + K_{aug} \sigma_z^{aug}.$$

Proof. We can decompose $\sigma_{f_{aug}}(z)$ into

$$\begin{aligned} \mathbb{E}_{x \sim \mathcal{P}_z} [\|f_{aug}(x) - \mathbb{E}_{x' \sim \mathcal{P}_z} [f_{aug}(x')]\|] &\leq \mathbb{E}_{x \sim \mathcal{P}_z} \left[\int \|f_{aug}(x) - f_{aug}(x')\| p(x'|z) dx' \right] \\ &\leq \mathbb{E}_{x \sim \mathcal{P}_z} \left[\int \left(\|f_{aug}(x) - f_{aug}(a(x))\| + \|f_{aug}(a(x)) - f_{aug}(a(x'))\| + \|f_{aug}(a(x')) - f_{aug}(x')\| \right) p(x'|z) dx' \right] \\ &\leq 2\mathbb{E}_{x \sim \mathcal{P}_z} [\|f_{aug}(x) - f_{aug}(a(x))\|] + K_{aug} \sigma_z^{aug}. \end{aligned} \quad (32)$$

We can bound $\mathbb{E}_{x \sim \mathcal{P}_z} [\|f_{aug}(x) - f_{aug}(a(x))\|] \leq \frac{1}{p(z)} \mathbb{E} [\|f_{aug}(x) - f_{aug}(a(x))\|]$. We assume that the encoder is able to satisfy $f_{aug}(x) = f_{aug}(a(x))$ for all training data $x \in \mathcal{D}$, so (32) becomes

$$\sigma_{f_{aug}}(z) \leq \frac{2}{p(z)} \left(\mathbb{E}_x [\|f_{aug}(x) - f_{aug}(a(x))\|] - \frac{1}{n} \sum_{i=1}^n \|f(x_i) - f(a(x_i))\| \right) + K_{aug} \sigma_z^{aug}. \quad (33)$$

Then, using Theorem 3.3 from Mohri et al. (2018), with probability at least $1 - \delta$

$$\mathbb{E} [\|f_{aug}(x) - f_{aug}(a(x))\|] - \frac{1}{n} \sum_{i=1}^n \|f_{aug}(x_i) - f_{aug}(a(x_i))\| \leq 2\mathfrak{R}_n^1(\mathcal{F}_{aug}, \mathcal{F}_{aug} \circ \mathcal{A}) + \sqrt{\frac{2\log(1/\delta)}{n}}.$$

Therefore, (33) becomes

$$\sigma_f(z) \leq \frac{2}{p(z)} \left(2\mathfrak{R}_n^1(\mathcal{F}_{aug}, \mathcal{F}_{aug} \circ \mathcal{A}) + \sqrt{\frac{2\log(1/\delta)}{n}} \right) + K_{aug} \sigma_z^{aug}.$$

\square

D. Additional Theoretical Results

D.1. Optimal L_{spread} geometry for $K=3$

We provide a proof sketch that when there are $K=3$ classes and $d \geq 3$, there exists a distribution μ_θ that obtains lower loss than the uniform or collapsed distributions. Synthetic experiments for this setting are in Appendix H (see Figure 7).

For simplicity, let's consider when $d=3$. Without loss of generality, denote the 3-simplex as $v_0 = [1, 0, 0]$, $v_1 = [-1/2, 0, \sqrt{3}/2]$, $v_2 = [-1/2, 0, -\sqrt{3}/2]$. We will perform a rotation in the “free” dimension (2), in particular rotating v_0 by θ in the direction orthogonal to the subspace that the simplex is in.

In particular, we construct the rotation matrix $R_\theta = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$. Then,

$$R_\theta v_0 = \begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix} \quad R_\theta v_1 = \begin{bmatrix} -\cos\theta/2 \\ -\sin\theta/2 \\ \sqrt{3}/2 \end{bmatrix} \quad R_\theta v_2 = \begin{bmatrix} -\cos\theta/2 \\ -\sin\theta/2 \\ -\sqrt{3}/2 \end{bmatrix},$$

and we make a distribution where $\mu_{0,\theta} = \frac{1}{2}\delta_{v_0} + \frac{1}{2}\delta_{R_\theta v_0}$, $\mu_{1,\theta} = \frac{1}{2}\delta_{v_1} + \frac{1}{2}\delta_{R_\theta v_1}$, $\mu_{2,\theta} = \frac{1}{2}\delta_{v_2} + \frac{1}{2}\delta_{R_\theta v_2}$. That is, similar to the binary setting, we take a mixture of a simplex and that simplex rotated by θ in a dimension orthogonal to its subspace.

Now, we want to compute what the loss is. Recall that our asymptotic loss function is

$$\begin{aligned} L_{\text{spread}}(\mu, \alpha) &= (1-\alpha)\mathbb{E}_x \left[\log \mathbb{E}_{x^-} \left[\exp\left(-\frac{1}{2\tau} \|f(x) - f(x^-)\|^2\right) \right] \right] \\ &\quad + \alpha \mathbb{E}_x \left[\log \mathbb{E}_{x^+} \left[\exp\left(-\frac{1}{2\tau} \|f(x) - f(x^+)\|^2\right) \right] \right] + (1-\alpha)\mathbb{E}_{x,x^+} \left[\frac{1}{2\tau} \|f(x) - f(x^+)\|^2 \right]. \end{aligned} \quad (34)$$

For the class collapsed embeddings, note that for $K=3$ the simplex side length is $\sqrt{3}$. Therefore,

$$L_{\text{spread}}(\delta_v, \alpha) = (1-\alpha) \log \exp\left(-\frac{1}{2\tau} \cdot 3\right) = \frac{-3 \cdot (1-\alpha)}{2\tau}.$$

Next, we compute the loss for our intermediate distribution. We note the following:

$$\begin{aligned} \|v_0 - R_\theta v_1\|^2 &= \|R_\theta v_0 - v_1\|^2 = \|v_0 - R_\theta v_2\|^2 = \|R_\theta v_0 - v_2\|^2 = \left(1 + \frac{\cos\theta}{2}\right)^2 + \frac{\sin^2\theta}{4} + \frac{3}{4} \\ &= 2 + \cos\theta \\ \|v_1 - R_\theta v_2\| &= \|R_\theta v_1 - v_2\| = \left(-\frac{1}{2} + \frac{\cos\theta}{2}\right)^2 + \frac{\sin^2\theta}{4} + 3 = \frac{7 - \cos\theta}{2} \\ \|v_0 - R_\theta v_0\| &= (1 - \cos\theta)^2 + \sin^2\theta = 2 - 2\cos\theta \\ \|v_1 - R_\theta v_1\| &= \|v_2 - R_\theta v_2\| = \left(-\frac{1}{2} + \frac{\cos\theta}{2}\right)^2 + \frac{\sin^2\theta}{4} = \frac{1 - \cos\theta}{2} \end{aligned}$$

and recall that $\|v_i - v_j\|^2 = \|R_\theta v_i - R_\theta v_j\|^2 = 3$. Plugging these back in, we have

$$\begin{aligned} \mathbb{E}_x \left[\log \mathbb{E}_{x^-} \left[\exp(-\|f(x) - f(x^-)\|^2/2\tau) \right] \right] &= \frac{1}{3} \log \left(\frac{1}{2} \exp\left(-\frac{3}{2\tau}\right) + \frac{1}{2} \exp\left(-\frac{2 + \cos\theta}{2\tau}\right) \right) \\ &\quad + \frac{2}{3} \log \left(\frac{1}{2} \exp\left(-\frac{3}{2\tau}\right) + \frac{1}{4} \exp\left(-\frac{7 - \cos\theta}{4\tau}\right) + \frac{1}{4} \exp\left(-\frac{2 + \cos\theta}{2\tau}\right) \right) \\ \mathbb{E}_x \left[\log \mathbb{E}_{x^+} \left[\exp(-\|f(x) - f(x^+)\|^2/2\tau) \right] \right] &= \frac{1}{3} \log \left(\frac{1}{2} + \frac{1}{2} \exp\left(-\frac{1 - \cos\theta}{\tau}\right) \right) + \frac{2}{3} \log \left(\frac{1}{2} + \frac{1}{2} \exp\left(-\frac{1 - \cos\theta}{4\tau}\right) \right) \\ \mathbb{E}_{x,x^+} \left[\frac{1}{2\tau} \|f(x) - f(x^+)\|^2 \right] &= \frac{1}{3} \cdot \frac{1 - \cos\theta}{4\tau} + \frac{1}{6} \cdot \frac{1 - \cos\theta}{\tau} = \frac{1 - \cos\theta}{4\tau} \end{aligned}$$

We use the above expressions to simplify (34) and numerically check that there exists θ for $\alpha \gtrsim 0.6$ such that $L(\mu_\theta, \alpha) \leq L(\delta_v, \alpha)$. We then numerically check there exists (θ, α) that also satisfies $L(\mu_\theta, \alpha) \leq L(\sigma_{d-1}, \alpha)$, where $L(\sigma_{d-1}, \alpha)$ is defined in (21).

D.2. Permutation Invariance

This result is a simple example of a sufficient condition under which L_{spread} does not exhibit class-fixing permutation invariance.

Lemma 4. *Let $\phi : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ be a monotonically increasing function. Suppose that for $x, x' \in \mathcal{X}$, all $f \in \mathcal{F}$ satisfy $\|f(x) - f(x')\| \leq \phi(\|x - x'\|)$. Then, L_{spread} is not invariant on class-fixing permutations under \mathcal{F} .*

Proof. First, note that L_{spread} in (2) has terms in the numerator of L_{sup} of the form $\|f(x) - f(x^+)\|$, where $h(x) = h(x^+)$. We show how to break permutation invariance using this quantity.

Fix two vectors in the embedding space \mathcal{S}^{d-1} , u_a and u_b . For x_1, x_2 in a given class, suppose that $f(x_1) = u_a$ and $f(x_2) = u_b$. We select a third point x_3 that is very close to x_1 , satisfying $\phi(\|x_3 - x_1\|) < \|u_a - u_b\|$ (this property must hold for some x_3 since ϕ is monotonic).

We construct a permutation π where $\pi(1) = 3, \pi(2) = 1, \pi(3) = 1$. We know that $\|u_a - u_b\| \leq \phi(\|x_1 - x_2\|)$. Suppose that the mapping f^π satisfies $\|f^\pi(x_3) - f^\pi(x_1)\| = \|u_a - u_b\|$. However, this implies that $\|u_a - u_b\| \leq \phi(\|x_3 - x_1\|)$, which is a contradiction. Therefore, no $f^\pi \in \mathcal{F}$ exists that is able to map the permutation to the same value as f does. As this holds for a single term in L_{spread} , it applies to L_{spread} overall, demonstrating that such an assumption on \mathcal{F} (which we find is true for a Lipschitz encoder, the autoencoder, and data augmentations) is able to break permutation invariance. \square

E. Auxiliary Lemmas

Lemma 5. *Under the infinite encoder assumption, the following statement holds for $L_{\text{diff}}(f)$:*

$$\min_{\mu \in \{\mathcal{S}^{d-1}\}^K} \mathbb{E}_x [\log \mathbb{E}_{x^-} [\exp(-\|f(x) - f(x^-)\|^2/2\tau)]] \equiv \min_{\mu \in \{\mathcal{S}^{d-1}\}^K} \log \mathbb{E}_{x, x^-} [\exp(-\|f(x) - f(x^-)\|^2/2\tau)].$$

Proof. Conditioning on the label of x and using the definition of x^- for $K=2$, we can write $L_{\text{diff}}(f)$ as

$$\begin{aligned} L_{\text{diff}}(f) &= \mathbb{E}_y [\mathbb{E}_{x|h(x)=y} [\log \mathbb{E}_{x^-|h(x^-) \neq y} [\exp(\sigma_f(x, x^-))]]]] \\ &= \frac{1}{2} \int p(x|y=0) \left(\log \int p(x^-|y=1) \exp(\sigma_f(x, x^-)) dx^- \right) dx \\ &\quad + \frac{1}{2} \int p(x|y=1) \left(\log \int p(x^-|y=0) \exp(\sigma_f(x, x^-)) dx^- \right) dx. \end{aligned}$$

Since the encoder is assumed to be infinitely powerful, we optimize over the class-conditional measures μ_0 and μ_1 in $\mathcal{M}(\mathbb{S}^{d-1})$, the set of Borel probability measures on \mathbb{S}^{d-1} . The optimization problem is now

$$\text{minimize}_{\mu_0, \mu_1} \int \left(\log \int \exp(\sigma(x, x^-)) d\mu_1(x) \right) d\mu_0(x) + \int \left(\log \int \exp(\sigma(x, x^-)) d\mu_0(x) \right) d\mu_1(x).$$

Next, define

$$U_\mu(u) = \int \exp(u^\top v / \tau) d\mu(v).$$

The expression we want to minimize is thus

$$\text{minimize}_{\mu_i, \mu_{-i}} \int \log U_{\mu_1}(u) d\mu_0(u) + \int \log U_{\mu_0}(u) d\mu_1(u). \quad (35)$$

Following the approach of Wang & Isola (2020), we analyze the measures μ_0^*, μ_1^* that minimize this expression in two steps. First, we show that the minimum of (35) exists, i.e. the infimum is attained for some two measures. Second, we show that $U_{\mu_0^*}$ is constant μ_1^* -almost surely, and vice versa. This will allow us to interchange the outer expectation over x and the log in $L_{\text{diff}}(f)$.

1. Minimizers of (35) exist.

Let m be a sequence such that

$$\begin{aligned} &\lim_{m \rightarrow \infty} \int \log U_{\mu_1^m}(u) d\mu_0^m(u) + \int \log U_{\mu_0^m}(u) d\mu_1^m(u) \\ &= \inf_{\mu_0, \mu_1} \int \log U_{\mu_1}(u) d\mu_0(u) + \int \log U_{\mu_0}(u) d\mu_1(u). \end{aligned}$$

Using Helly's Selection Theorem twice, there exists a subsequence n such that $\{(\mu_0^n, \mu_1^n)\}_n$ converges to a weak cluster point (μ_0^*, μ_1^*) . Because $\{\log U_{\mu_0^n}\}_n$ is uniformly bounded and continuously convergent to $\log U_{\mu_0^*}$ and same for μ_1^n and μ_1^* , it holds that

$$\begin{aligned} & \int \log U_{\mu_1^*}(u) d\mu_0^*(u) + \int \log U_{\mu_0^*}(u) d\mu_1^*(u) \\ &= \lim_{n \rightarrow \infty} \int \log U_{\mu_1^n}(u) d\mu_0^n(u) + \int \log U_{\mu_0^n}(u) d\mu_1^n(u). \end{aligned}$$

and therefore μ_0^*, μ_1^* achieve the infimum of (35).

2. $U_{\mu_1^*}$ is constant μ_0^* -almost surely and $U_{\mu_0^*}$ is constant μ_1^* -almost surely, for any minimizer (μ_0^*, μ_1^*) of (35).

Formally, define (μ_0^*, μ_1^*) to be a solution of (35), i.e.

$$\mu_0^*, \mu_1^* \in \operatorname{argmin}_{\mu_0, \mu_1} \int \log U_{\mu_1}(u) d\mu_0(u) + \int \log U_{\mu_0}(u) d\mu_1(u).$$

Define the Borel sets where μ_i^* has positive measure to be $\mathcal{T}_i = \{T \in \mathcal{M}(\mathbb{S}^{d-1}) : \mu_i^*(T) > 0\}$. Define the conditional distribution of μ_i^* on T for some $T \in \mathcal{T}_i$ as $\mu_{i,T}^*$, where $\mu_{i,T}^*(A) = \frac{\mu_i^*(A \cap T)}{\mu_i^*(T)}$.

Now we consider a mixture $(1-\alpha)\mu_0^* + \alpha\mu_{0,T}^*$. The first variation of μ_0^* states that

$$\begin{aligned} 0 &= \frac{\partial}{\partial \alpha} \left[\int \log U_{\mu_1^*}(u) d((1-\alpha)\mu_0^* + \alpha\mu_{0,T}^*)(u) + \int \log U_{(1-\alpha)\mu_0^* + \alpha\mu_{0,T}^*} d\mu_1^*(u) \Big|_{\alpha=0} \right] \\ &= \int \log U_{\mu_1^*}(u) d(\mu_{0,T}^* - \mu_0^*)(u) + \int \frac{U_{\mu_{0,T}^*}(u) - U_{\mu_0^*}(u)}{U_{\mu_0^*}(u)} d\mu_1^*(u), \end{aligned}$$

Where we've used the fact that $\frac{\partial}{\partial \alpha} U_{(1-\alpha)\mu_0^* + \alpha\mu_{0,T}^*}(u) \Big|_{\alpha=0} = \frac{\partial}{\partial \alpha} \int \exp(u^\top v / \tau) d((1-\alpha)\mu_0^* + \alpha\mu_{0,T}^*)(v) \Big|_{\alpha=0} = U_{\mu_{0,T}^*}(u) - U_{\mu_0^*}(u)$. Therefore, due to symmetry the optimality conditions using the first variation are

$$\int \log U_{\mu_1^*}(u) d(\mu_{0,T}^* - \mu_0^*)(u) + \int \frac{U_{\mu_{0,T}^*}(u)}{U_{\mu_0^*}(u)} d\mu_1^*(u) = 1 \quad (36)$$

$$\int \log U_{\mu_1^*}(u) d(\mu_{1,T}^* - \mu_1^*)(u) + \int \frac{U_{\mu_{1,T}^*}(u)}{U_{\mu_1^*}(u)} d\mu_0^*(u) = 1 \quad (37)$$

Now, let $\{T_0^n\}_{n=1}^\infty$ be a sequence of sets in \mathcal{T}_0 such that

$$\lim_{n \rightarrow \infty} \int U_{\mu_1^*}(u) d\mu_{0,T_0^n}^*(u) = \sup_{T_0 \in \mathcal{T}_0} \int U_{\mu_1^*}(u) d\mu_{0,T_0}^*(u) = U_{1,0}^*$$

and similarly let $\{T_1^n\}_{n=1}^\infty$ be a sequence of sets in \mathcal{T}_1 such that

$$\lim_{n \rightarrow \infty} \int U_{\mu_0^*}(u) d\mu_{1,T_1^n}^*(u) = \sup_{T_1 \in \mathcal{T}_1} \int U_{\mu_0^*}(u) d\mu_{1,T_1}^*(u) = U_{0,1}^*$$

It holds that $\mu_0^*(\{u : U_{\mu_1^*}(u) \geq U_{1,0}^*\}) = 0$, $\mu_{0,T_0^n}^*(\{u : U_{\mu_1^*}(u) \geq U_{1,0}^*\}) = 0$ and similarly $\mu_1^*(\{u : U_{\mu_0^*}(u) \geq U_{0,1}^*\}) = 0$, $\mu_{1,T_1^n}^*(\{u : U_{\mu_0^*}(u) \geq U_{0,1}^*\}) = 0$.

This implies that asymptotically $U_{\mu_0^*}$ is constant $\mu_{1,T_1^n}^*$ -almost surely:

$$\begin{aligned} & \int \left| U_{\mu_0^*}(u) - \int U_{\mu_0^*}(u') d\mu_{1,T_1^n}^*(u') \right| d\mu_{1,T_1^n}^*(u) \\ &= 2 \int \max \left(0, U_{\mu_0^*}(u) - \int U_{\mu_0^*}(u') d\mu_{1,T_1^n}^*(u') \right) d\mu_{1,T_1^n}^*(u) \\ &\leq 2 \left(U_{0,1}^* - \int U_{\mu_0^*}(u) d\mu_{1,T_1^n}^*(u) \right) \rightarrow 0. \end{aligned}$$

And the same holds that $U_{\mu_1^*}$ is constant $\mu_{0,T_0^n}^*$ -almost surely. As a result, $\lim_{n \rightarrow \infty} \int \log U_{\mu_0^*}(u) d\mu_{1,T_1^n}^*(u) = \log U_{0,1}^*$ and $\lim_{n \rightarrow \infty} \int \log U_{\mu_1^*}(u) d\mu_{0,T_0^n}^*(u) = \log U_{1,0}^*$.

We now revisit (36) with a mixture over μ_0^* and μ_{0,T_0^n} :

$$\begin{aligned} 1 &= \int \log U_{\mu_1^*}(u) d(\mu_{0,T_0^n}^* - \mu_0^*)(u) + \int \frac{U_{\mu_{0,T_0^n}^*}(u)}{U_{\mu_0^*}(u)} d\mu_1^*(u) \\ &\geq \int \log U_{\mu_1^*}(u) d(\mu_{0,T_0^n}^* - \mu_0^*)(u) + \frac{1}{U_{0,1}^*} \int U_{\mu_1^*}(u) d\mu_{0,T_0^n}^*(u). \end{aligned}$$

Taking the limit of both sides as $n \rightarrow \infty$, we get

$$1 \geq \log U_{1,0}^* - \int \log U_{\mu_1^*}(u) d\mu_0^*(u) + \frac{1}{U_{0,1}^*} U_{1,0}^*.$$

and rearranging and doing the same to (37) yields

$$\left(1 - \frac{U_{1,0}^*}{U_{0,1}^*}\right) \geq \log U_{1,0}^* - \int \log U_{\mu_1^*}(u) d\mu_0^*(u)$$

$$\left(1 - \frac{U_{0,1}^*}{U_{1,0}^*}\right) \geq \log U_{0,1}^* - \int \log U_{\mu_0^*}(u) d\mu_1^*(u)$$

Note that Jensen's inequality and the definition of $U_{1,0}^*$ tell us that $\int \log U_{\mu_1^*}(u) d\mu_0^*(u) \leq \log \int U_{\mu_1^*}(u) d\mu_0^*(u) \leq \log U_{1,0}^*$, which means that $\left(1 - \frac{U_{1,0}^*}{U_{0,1}^*}\right) \geq 0$. However, applying the same logic also tells us that $\left(1 - \frac{U_{0,1}^*}{U_{1,0}^*}\right) \geq 0$. The only case in which this is possible is when $U_{1,0}^* = U_{0,1}^*$, in which case equality is obtained. Therefore, this means that for optimal μ_0^*, μ_1^* , it holds that

$$\int \log U_{\mu_1^*}(u) d\mu_0^*(u) = \log \int U_{\mu_1^*}(u) d\mu_0^*(u) \quad (38)$$

$$\int \log U_{\mu_0^*}(u) d\mu_1^*(u) = \log \int U_{\mu_0^*}(u) d\mu_1^*(u) \quad (39)$$

Using (38) and (39), minimizing (35) is equivalent to minimizing

$$\log \int U_{\mu_1}(u) d\mu_0(u),$$

where we use the fact that μ_0 and μ_1 are interchangeable in the above expression. This expression can be written as $\log \mathbb{E}_{x, x^-} [\exp(-\|f(x) - f(x^-)\|^2/2\tau)]$, which completes our proof. \square

Lemma 6. Suppose \mathcal{F} is a family of functions mapping from \mathcal{X} to \mathbb{R}^d . Define $f(x)[j]$ as the j th element of $f(x)$ and suppose that for all j , $|f(x)[j]| \leq b$. Define the element-wise class $\mathcal{F}_j = \{f(\cdot)[j] : f \in \mathcal{F}\}$. Then, with probability at least $1 - \delta$ over n i.i.d. samples $\{x_i\}_{i=1}^n$,

$$\left\| \mathbb{E}[f(x)] - \frac{1}{n} \sum_{i=1}^n f(x_i) \right\| \leq 2\mathfrak{R}_n(\mathcal{F}) + bd \sqrt{\frac{\log(d/\delta)}{2n}} \quad \forall f \in \mathcal{F},$$

where $\mathfrak{R}_n(\mathcal{F}) = \sum_{j=1}^d \mathfrak{R}_n(\mathcal{F}_j)$.

Proof. Using the triangle inequality,

$$\left\| \mathbb{E}[f(x)] - \frac{1}{n} \sum_{i=1}^n f(x_i) \right\| = \left(\sum_{j=1}^d \left(\mathbb{E}[f(x)[j]] - \frac{1}{n} \sum_{i=1}^n f(x_i)[j] \right)^2 \right)^{1/2} \leq \sum_{j=1}^d \left| \mathbb{E}[f(x)[j]] - \frac{1}{n} \sum_{i=1}^n f(x_i)[j] \right|.$$

Using Theorem 3.3 of Mohri et al. (2018), we know that with probability at least $1 - \delta$,

$$\mathbb{E}[f(x)[j]] \leq \frac{1}{n} \sum_{i=1}^n f(x_i)[j] + 2\mathfrak{R}_n(\mathcal{F}_j) + b \sqrt{\frac{2 \log(1/\delta)}{n}}.$$

Applying a union bound, we have that with probability at least $1 - \delta$,

$$\left\| \mathbb{E}[f(x)] - \frac{1}{n} \sum_{i=1}^n f(x_i) \right\| \leq \sum_{j=1}^d \left(2\mathfrak{R}_n(\mathcal{F}_j) + b \sqrt{\frac{\log(d/\delta)}{2n}} \right) = 2\mathfrak{R}_n(\mathcal{F}) + bd \sqrt{\frac{2 \log(d/\delta)}{n}}.$$

\square

F. Additional Experimental Details

We describe details about the datasets, model architectures, and hyperparameters.

F.1. Datasets

We first describe all the datasets in more detail:

- **CIFAR10**, **CIFAR100**, and **MNIST** are all the standard computer vision datasets.
- **CIFAR10-Coarse** consists of two superclasses: animals (dog, cat, deer, horse, frog, bird) and vehicles (car, truck, plane, boat).
- **CIFAR100-Coarse** consists of twenty superclasses. We artificially imbalance subclasses to create **CIFAR100-Coarse-U**. For each superclass, we select one subclass to keep all 500 points, select one subclass to subsample to 250 points, select one subclass to subsample to 100 points, and select the remaining two to subsample to 50 points. We use the original CIFAR100 class index to select which subclasses to subsample: the subclass with the lowest original class index keeps all 500 points, the next subclass keeps 250 points, etc.
- **TinyImageNet-Coarse** (Le & Yang, 2015) consists of 67 superclasses constructed from the ImageNet class hierarchy (Bostock, 2018). The 67 superclasses are as follows: arachnid, armadillo, bear, bird, bug, butterfly, cat, coral, crocodile, crustacean, dinosaur, dog, echinoderms, ferret, fish, flower, frog, fruit, fungus, hog, lizard, marine mammals, marsupial, mollusk, mongoose, monotreme, person, plant, primate, rabbit, rodent, salamander, shark, sloth, snake, trilobite, turtle, ungulate, vegetable, wild cat, wild dog, accessory, aircraft, ball, boat, building, clothing, container, cooking, decor, electronics, fence, food, furniture, hat, instrument, lab equipment, other, outdoor scene, paper, sports equipment, technology, tool, toy, train, vehicle and weapon.
- **MNIST-Coarse** consists of two superclasses: <5 and ≥ 5 .
- **Waterbirds** (Sagawa et al., 2019) is a robustness dataset designed to evaluate the effects of spurious correlations on model performance. The waterbirds dataset is constructed by cropping out birds from photos in the Caltech-UCSD Birds dataset (Welinder et al., 2010a), and pasting them on backgrounds from the Places dataset (Zhou et al., 2014). It consists of two categories: water birds and land birds. The water birds are heavily correlated with water backgrounds and the land birds with land backgrounds, but 5% of the water birds are on land backgrounds, and 5% of the land birds are on water backgrounds. These form the (imbalanced) hidden strata.
- **ISIC** is a public skin cancer dataset for classifying skin lesions (Codella et al., 2019) as malignant or benign. 48% of the benign images contain a colored patch, which form the hidden strata.
- **CelebA** is an image dataset commonly used as a robustness benchmark (Liu et al., 2015; Sagawa et al., 2019). The task is blonde/not blonde classification. Only 6% of blonde faces are male, which creates a rare stratum in the blonde class.

F.2. Model Architectures

We use a ViT model (Dosovitskiy et al., 2020) (4 x 4 patch size, 7 multi-head attention layers with 8 attention heads and hidden MLP size of 256, final embedding size of 128) as the encoder for the transfer learning experiments and a ResNet50 for the robustness experiments. For the ViT models, we jointly optimize the contrastive loss with a cross-entropy loss head. For the ResNets, we train the contrastive loss on its own and use linear probing on the final layer.

For the autoencoder, we use the same encoder backbone as the main model, and use a ResNet18 in reverse order for the decoder. The convolutions are replaced with resize convolutions. We use the implementation in PyTorch Lightning Bolts² (Falcon & Cho, 2020).

²https://github.com/PyTorchLightning/lightning-bolts/blob/master/pl_bolts/models/autoencoders/components.py

Dataset	End Model Perf.		
	InfoNCE	L_{sup}	L_{spread}
CIFAR10	89.7	90.9	91.5
CIFAR10-Coarse	97.7	96.5	98.1
CIFAR100	68.0	67.5	69.1
CIFAR100-Coarse	76.9	77.2	78.3
CIFAR100-Coarse-U	72.1	71.6	72.4
MNIST	99.1	99.3	99.2
MNIST-Coarse	99.1	99.4	99.4
Waterbirds	77.8	73.9	77.9
ISIC	87.8	88.7	90.0

Table 6. End model performance training with L_{spread} on various datasets compared against contrastive baselines. All metrics are accuracy except for ISIC (AUROC). L_{spread} produces the best performance in 7 out of 9 cases, and matches the best performance in 1 case.

Table 7. Coarse-to-fine transfer learning performance (expanded table). Best in bold.

	Method	CIFAR10	CIFAR100	CIFAR100-U	MNIST	TinyImageNet
Baselines	Cross Entropy	71.1 \pm 0.2	54.2 \pm 0.2	56.4 \pm 0.4	98.7 \pm 0.1	44.4 \pm 0.1
	InfoNCE (Chen et al., 2020a)	77.6 \pm 0.1	60.5 \pm 0.1	56.4 \pm 0.3	98.4 \pm 0.1	44.9 \pm 0.1
	SupCon (Khosla et al., 2020)	51.8 \pm 1.2	56.1 \pm 0.1	49.8 \pm 0.3	95.4 \pm 0.1	43.9 \pm 0.1
	SupCon + InfoNCE (Islam et al., 2021)	77.6 \pm 0.1	55.7 \pm 0.1	48.0 \pm 0.2	98.6 \pm 0.1	46.1 \pm 0.1
Ours	cAuto	71.4 \pm 0.1	62.9 \pm 0.1	58.7 \pm 0.5	98.7 \pm 0.1	47.1 \pm 0.1
	SupCon + cNCE (L_{spread})	77.1 \pm 0.1	58.7 \pm 0.2	53.5 \pm 0.4	98.5 \pm 0.1	45.8 \pm 0.1
	SupCon + cAuto	71.7 \pm 0.1	63.8 \pm 0.6	59.8 \pm 0.3	98.7 \pm 0.1	49.3 \pm 0.1
	SupCon + cNCE + cAuto (THANOS)	79.1 \pm 0.2	65.0 \pm 0.2	59.7 \pm 0.3	99.0 \pm 0.1	49.6 \pm 0.1

F.3. Hyperparameters

For the coarse dataset training, all models were trained for 600 epochs with an initial learning rate of 0.0003, a cosine annealing learning rate scheduler with T_{max} set to 100 and the AdamW optimizer. A dropout rate of 0.05 was used. We did not use weight decay. For each coarse dataset, we trained 5 separate models which jointly optimize a cross-entropy loss head with either a contrastive loss (InfoNCE, SupCon, SupCon + InfoNCE, SupCon + Class-conditional InfoNCE) or a reconstruction loss (mean squared error).

In the coarse-to-fine transfer experiments, we trained 5 separate models for each of the configurations reported in Table 3 using 5 random seeds (42, 32, 64, 128 and 72). All models were trained for 100 epochs with an initial learning rate of 0.001, a cosine annealing learning rate scheduler with T_{max} set to 100 and the AdamW optimizer. All transfer experiments were run using Tesla V100 machines.

All experiments were run using a batch size of 128 for both training and evaluation.

G. Additional Experimental Results

We present additional experimental results on end model accuracy, transfer with cross entropy, more datasets, more baselines, and full ablations.

G.1. End Model Accuracy

See Table 6 for raw accuracy. We confirm that using L_{spread} instead of L_{sup} does not degrade end model performance.

G.2. Additional Transfer Results

We reproduce Table 3 and additionally report the performance of training with cross entropy loss (Table 7).

Table 8. Coarse-to-fine transfer learning performance on two additional datasets. Best in bold.

Method	Caltech-UCSD Birds	Stanford Dogs
Cross Entropy	8.2	14.9
SupCon (Khosla et al., 2020)	7.8	15.0
cAuto	8.8	17.7
SupCon + cNCE (L_{spread})	7.5	16.5
SupCon + cAuto	9.1	19.8
SupCon + cNCE + cAuto (THANOS)	8.8	20.8

Table 9. Coarse-to-fine transfer learning performance on CIFAR10 with two additional baselines. Best in bold.

Method	CIFAR10
Clip Positives	35.3
Weighted Pos in Denom.	56.3
THANOS	79.1

Table 10. Ablations on the autoencoder and data augmentation on CIFAR10 coarse-to-fine transfer.

General vs. Class-Conditional Autoencoder	
gAuto	41.4 ± 0.2
cAuto	71.4 ± 0.1
SupCon	51.8 ± 1.2
SupCon + gAuto	55.4 ± 0.4
SupCon + cAuto	71.7 ± 0.1
SupCon + cNCE	77.1 ± 0.1
SupCon + cNCE + gAuto	77.4 ± 0.1
SupCon + cNCE + cAuto (THANOS)	79.1 ± 0.2
cNCE With and Without Augmentation	
SupCon + cNCE - augmentation	41.7 ± 0.2
SupCon + cNCE	77.1 ± 0.1

G.3. Additional Datasets

Table 8 report the performance of C2F transfer on two additional datasets, Caltech-UCSD Birds (Welinder et al., 2010b) and Stanford Dogs (Khosla et al., 2011).

G.4. Additional Baselines

Table 9 reports C2F transfer performance with two additional baselines—a) clipping the values of the positives in the numerator L_{sup} , and b) upweighting the negatives in the denominator of L_{sup} . Both these methods underperform THANOS.

G.5. Ablations and Sensitivity Studies

In this section, we validate our specific theoretical claims on the class-conditional autoencoder, data augmentation, and the Lipschitzness of the decoder.

We use two ablations to validate our claims that the class-conditional autoencoder outperforms a generic autoencoder, and that data augmentation in the class-conditional InfoNCE loss is critical for inducing subclass clustering. Table 10 reports the results:

- Lemma 2 claims that a class-conditional autoencoder should outperform a generic autoencoder in coarse-to-fine transfer. Indeed, we find that using a generic autoencoder underperforms a class-conditional autoencoder by 30.0 points on CIFAR10 coarse-to-fine transfer. Furthermore, the generic autoencoder does not improve performance of SupCon or its variants as well; we observe average lift of 2.0 points, compared to 11.0 points for the class-conditional autoencoder.

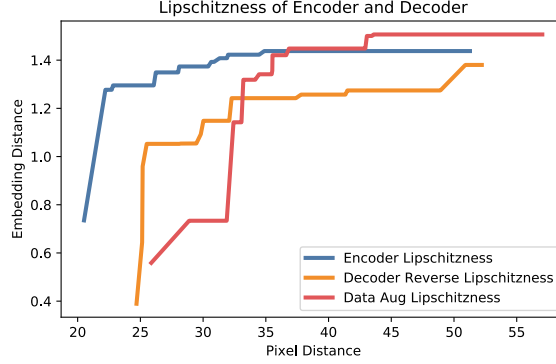


Figure 3. Measures of Lipschitzness for three ways to break permutation invariance. Encoder Lipschitzness reports pixel distance on the X axis and embedding distance on the Y axis. The Decoder reports pixel distance of the reconstruction. For augmentations, we run ten augmentations and pick the pair with the smallest ratio of pixel distance to embedding distance. The decoder is more Lipschitz than the encoder, and the encoder is more Lipschitz under augmentations than under traditional Lipschitzness.

- Lemma 3 claims that data augmentation in the class-conditional InfoNCE loss is key to break the permutation invariance. Removing data augmentation degrades performance by 35.4 points (and produces the permutation shown in Figure 1).

Finally, Figure 3 measures the Lipschitzness of an encoder trained with L_{spread} and the reverse Lipschitzness of the decoder from a class-conditional autoencoder. The encoder displays a high Lipschitzness constant (not very Lipschitz). However, it displays a low Lipschitzness constant over augmentations. The decoder displays a lower reverse Lipschitzness constant. This suggests that the assumptions in Lemmas 2 and 3 are reasonable.

To measure Lipschitzness of an encoder, we measure distance in embedding space of an encoder trained with L_{spread} vs. distance in pixel space of two images (blue line). To measure reverse Lipschitzness of a decoder, we make the same measurement, but over pixel distance of decoded images from an autoencoder (orange line). To measure Lipschitzness under data augmentations, we measure the minimum ratio between embedding distance and pixel distance for 10 randomly-generated augmentations of two images (red line). The Lipschitzness constants K_L , K_g , and K_{aug} in Table 1 are the slopes of the lines tangent to each of the curves in Figure 3 from the origin.

H. Synthetic Experiments

We conduct synthetic experiments to understand the optimal geometry that minimizes the asymptotic loss $L_{\text{spread}}(\mu, \alpha)$ as defined in Section 3.2.

Setup We minimize an empirical estimate of the asymptotic loss over a set of unit vectors $\{u_i\}_{i=1}^{Kn_y}$. Denote u as a unit vector, and denote $h(u)$ as its class label. The loss we minimize is

$$(1-\alpha) \frac{1}{Kn_y} \sum_{i=1}^{Kn_y} \log \left(\frac{1}{(K-1)n_y} \sum_{j: h(u_i) \neq h(u_j)} \exp(-\|u_i - u_j\|^2 / 2\tau) \right) \quad (40)$$

$$+\alpha \cdot \frac{1}{Kn_y} \sum_{i=1}^{Kn_y} \log \left(\frac{1}{n_y} \sum_{j: h(u_i) = h(u_j)} \exp(-\|u_i - u_j\|^2 / 2\tau) \right) + (1-\alpha) \frac{1}{Kn_y^2} \sum_{h(u) = h(u')} \|u - u'\|^2 / 2\tau. \quad (41)$$

We use `scipy.minimize` and the Sequential Least Squares Programming (SLSQP) option. We report the set of vectors that obtain the lowest loss over 5 runs with random initializations (seeds 0–4) as the optimal geometry.

We compute an empirical estimate of $s_f(y)$ as $\frac{1}{n_y} \sum_{u: h(u)=y} \|u - \frac{1}{n_y} \sum_{u': h(u')=y} u'\|$, and average over all classes.

Matching μ and the optimal geometry Figure 4 displays our constructed distribution μ_θ as well as simulations on S^1 for $K=2$. In particular, the right figure consists of the optimal geometry for $n_y=20$, $\tau=0.5$. We see that for $\alpha=0.6$ (which is below our threshold in Theorem 2), that the optimal geometry is collapsed. For $\alpha=0.7$, the optimal geometry appears

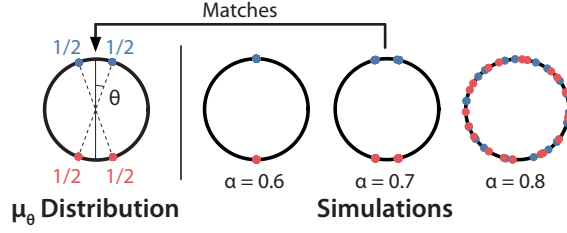


Figure 4. Left: Distribution family μ_θ . Right: Simulations of optimal geometry for binary setting on S^1 .

to closely match the parametrization of μ_θ . For $\alpha = 0.8$, which is above the theoretical threshold, the optimal geometry is uniform per class.

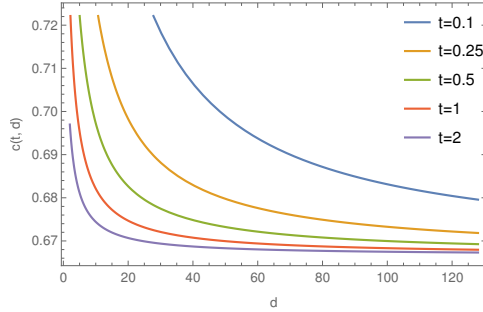


Figure 5. The value of $c(\tau, d)$, which determines the range of α for which μ_θ obtains lower loss than δ_v and σ_{d-1} .

Computing $c(\tau, d)$ Next, we compute the value of $c(\tau, d)$, the constant in Theorem 2, over values of τ and d to verify that $\alpha \in (2/3, c(\tau, d))$ is a valid range for which the optimal geometry is neither collapsed nor uniform. This quantity depends on the Wiener constant of the Gaussian $\frac{1}{2\tau}$ -energy on S^{d-1} , which does not have a closed form expression (see (22)). Figure 5 shows that $c(\tau, d) > 2/3$ for d up to 128 (which is the dimension of our embedding space) and for $\tau = 0.1, 0.25, 0.5, 1, 2$.

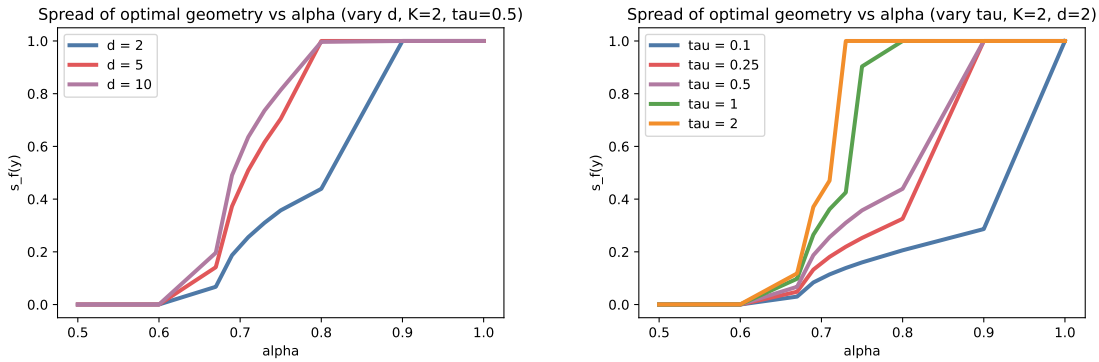


Figure 6. The spread $s_f(y)$ of the optimal geometry for a given α in the binary setting, $n_y = 8$. Left: how optimal spread changes based on dimension d of the embedding space. Right: how optimal spread changes based on the temperature hyperparameter τ .

Varying τ and d for $K = 2$ We plot α versus spread $s_f(y)$ for the optimal geometry in the binary setting, and in particular we vary τ and d in Figure 6. We compute the optimal geometries over $\alpha = 0.5, 0.6, 0.67, 0.69, 0.71, 0.73, 0.75, 0.8, 0.9$. Figure 6 left shows how the spread changes as α increases for dimensions $d = 2, 5, 10$ and $n_y = 8$ samples with $\tau = 0.25$, and the right shows how the spread changes as α increases for $\tau = 0.1, 0.25, 0.5, 1, 2$ with $d = 2$. Note that for all dimensions and all τ , the optimal geometry has nonzero spread starting at $\alpha = 0.67$, matching our theoretical findings. The point at which the uniform distribution

becomes optimal is less clear but follows the trends we note from Figure 5 for $c(\tau, d)$. This figure matches our findings that an α that induces appropriate spread exists over a certain range, outside of which behavior is strictly collapsed or uniform.

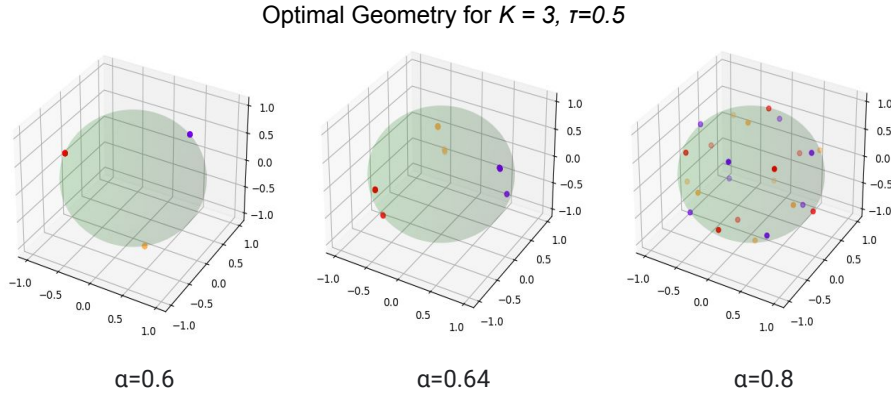


Figure 7. Visualizations of the optimal geometry for $n_y = 8, K = 3$ in \mathcal{S}^2 across various α .

Multiclass Analysis for $K = 3$ Similar to Figure 4, we show for $K = 3$ and \mathcal{S}^2 that the optimal geometry is collapsed for low α , sufficiently spread for a particular range, and uniform for high α . Figure 7 displays the optimal geometry for $n_y = 8, \tau = 0.5$ across $\alpha = 0.6, 0.64, 0.8$, suggesting that the multiclass case exhibits similar behavior as α varies.

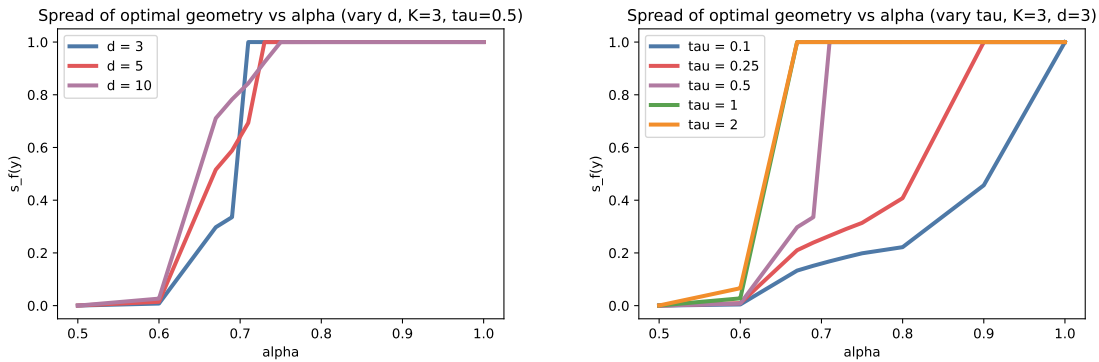


Figure 8. The spread $s_f(y)$ of the optimal geometry for a given α when $K = 3, n_y = 8$. Left: how optimal spread changes based on dimension d of the embedding space. Right: how optimal spread changes based on the temperature hyperparameter τ .

We plot α versus spread $s_f(y)$ for the optimal geometry in the multiclass setting, and again we vary τ and d in Figure 8. We see that the behavior of the optimal geometry's spread $s_f(y)$ across α is roughly similar to that of $K = 2$ in Figure 6.