A. Proofs of theoretical results

The aim of this section is to detail the proofs of the theoretical results presented in the main manuscript. The key theoretical tools driving our analysis are prepared separately in Section C.

Throughout our analysis, we assume that all spaces (e.g., $\mathcal{A}$ and $\mathcal{X}$) are subspaces of Euclidean space and therefore admit a Lebesgue measure. We also assume that all distributions (e.g., $a \sim \mathcal{A}$ and $x \sim \mathcal{X}$) admit a density with respect to the Lebesgue measure. With these conditions in mind, we recall the loss function that is the main object of study:

$$\mathcal{L}_{\text{equi}}(f) = \mathbb{E}_{a\sim\mathcal{A}}\mathbb{E}_{x,x'\sim\mathcal{X}}\left[f(a(x'))^\top f(a(x)) - f(x)^\top f(x')\right]^2 \tag{1}$$

Next, we re-state and prove Proposition 1, our first key result.

Proposition 1. *Suppose $\mathcal{L}_{equi}(f) = 0$. Then for almost every $a \in \mathcal{A}$, there is an orthogonal matrix $R_a \in O(d)$ such that $f(a(x)) = R_a f(x)$ for almost all $x \in \mathcal{X}$.*

Proof. Suppose that $\mathcal{L}_{\text{equi}}(f) = 0$. This means that $f(a(x'))^\top f(a(x)) = f(x)^\top f(x')$ for almost all $a \in G$, and $x, x' \in \mathcal{X}$. Setting $g_a(x) = f(a(x))$, we have that $g_a(x')^\top g_a(x) = f(x)^\top f(x')$. The continuous version of the First Fundamental Theorem of invariant theory for the orthogonal group (see Proposition 4) implies that there is an $R_a \in O(d)$ such that $f(a(x)) = g_a(x) = R_a f(x)$. □

As discussed in greater detail in the main manuscript, these results show that minimizing $\mathcal{L}_{\text{equi}}$ produces a model where an augmentation a corresponds to a single orthogonal transformation of embeddings R_a, independent of the input. This result is continuous in flavor as it studies the loss over the full data distribution $p(x)$. There exists a corresponding result for the finite sample loss

$$\mathcal{L}_{\text{equi},n}(f) = \mathbb{E}_{a\sim\mathcal{A}} \sum_{i,j=1}^{n} \left[f(a(x_j))^\top f(a(x_i)) - f(x_i)^\top f(x_j)\right]^2.$$

Proposition 2. *Suppose $\mathcal{L}_{equi,n}(f) = 0$. Then for almost every $a \in \mathcal{A}$, there is an orthogonal matrix $R_a \in O(d)$ such that $f(a(x_i)) = R_a f(x_i)$ for all $i = 1, \dots, n$.*

As for the population counterpart, the proof of this result directly follows from the application of the First Fundamental Theorem of invariant theory for the orthogonal group.

Proof of Proposition 2. Suppose that $\mathcal{L}_{\text{equi}}(f) = 0$. This means that for almost every $a \in G$, and every $i, j = 1, \dots, n$ we have $f(a(x_j))^\top f(a(x_i)) = f(x_i)^\top f(x_j)$. In other words $AA^T = BB^T$ where $A, B \in \mathbb{R}^{n\times d}$ are matrices whose ith rows are $A_i = f(a(x_i))^\top$ and $B_i = f(x_i)^\top$ respectively. This implies, by the First Fundamental Theorem of invariant theory for the orthogonal group (see Corollary 2), that there is an $R_a \in O(d)$ such that $A = BR_a$. Considering only the ith rows of A and B leads us to conclude that $f(a(x_i)) = R_a f(x_i)$. □

A corollary of Proposition 1 is that compositions of augmentations correspond to compositions of rotations.

Corollary 1. *If $\mathcal{L}_{equi}(f) = 0$, then $\rho : \mathcal{A} \to O(d)$ given by $\rho(a) = R_a$ satisfies $\rho(a' \circ a) = \rho(a')\rho(a)$ for almost all a, a'. That is, ρ defines a group action on $\mathbb{S}^{d-1}$ up to a set of measure zero.*

Proof. Applying Proposition 1 on $a' \circ a$ as the sampled augmentation, we have that $f(a' \circ a(x_i)) = R_{a'\circ a} f(x_i) = \rho(a' \circ a) f(x_i)$. However, taking $\bar{x} = a(x_i)$ and applying Proposition 1 twice we also know that $f(a' \circ a(x_i)) = f(a'(\bar{x})) = R_a f(\bar{x}) = R_{a'} f(a(x_i)) = R_{a'} R_a f(x) = \rho(a')\rho(a) f(x_i)$. That is, $\rho(a' \circ a) f(x_i) = f(a' \circ a(x_i)) = \rho(a')\rho(a) f(x_i)$. Since this holds for all i, we have that $\rho(a' \circ a) = \rho(a')\rho(a)$. □

This corollary requires us to assume that $\mathcal{A}$ is a semi-group. That is, $\mathcal{A}$ is closed under compositions, but group elements do not necessarily have inverses and it does not need to include an identity element.

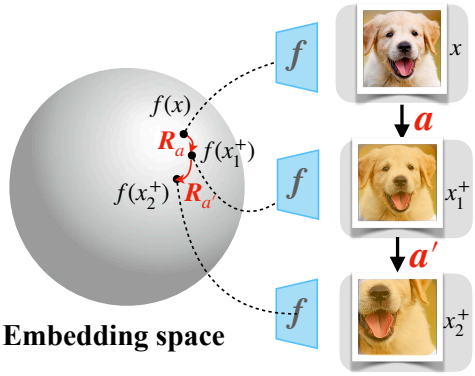

Figure 5: When $\mathcal{L}_{\text{equi}} = 0$, compositions of augmentations correspond to compositions of rotations.

B. Background on role of augmentations in self supervised learning

Given access only to samples from a marginal distribution $p(x)$ on some input space $\mathcal{X}$ such as images, the goal of representation learning is commonly to train a feature extracting model $f : \mathcal{X} \to \mathbb{S}^{d-1}$ mapping to the unit sphere $\mathbb{S}^{d-1} = \{z \in \mathbb{R}^d : \|z\|_2 = 1\}$. A common strategy to automatically generate supervision from the data is to additionally introduce a space of augmentations $\mathcal{A}$, containing maps $a : \mathcal{X} \to \mathcal{X}$ which slightly perturb inputs $\bar{x}$ (blurring, cropping, jittering, etc.). Siamese self-supervised methods learn representation spaces that reflect the relationship between the embeddings of $x = a(\bar{x})$ and $x^+ = a^+(\bar{x})$, commonly by training f to be invariant or equivariant to the augmentations in the input space (Chen & He, 2021).

Invariance to augmentation. One approach is to train f to embed x and x^+ nearby—i.e., so that $f(x) = f(x^+)$ is *invariant* to augmentations. The InfoNCE loss (van den Oord et al., 2018; Gutmann & Hyvärinen, 2010) used in contrastive learning achieves precisely this:

$$\mathcal{L}_{\text{InfoNCE}}(f) = \mathbb{E}_{x,x^+,\{x_i^-\}_{i=1}^N}\left[-\log \frac{e^{f(x)^\top f(x^+)/\tau}}{e^{f(x)^\top f(x^+)/\tau} + \sum_{i=1}^N e^{f(x)^\top f(x_i^-)/\tau}}\right], \tag{2}$$

where $\tau > 0$ is a temperature hyperparameter, and $x_i^- \sim p$ are negative samples from the marginal distribution on $\mathcal{X}$. As noted by (Wang & Isola, 2020), the contrastive training mechanism balances invariance to augmentations with a competing objective: uniformly distributing embeddings over the sphere, which rules out trivial solutions such as constant functions.

Whilst contrastive learning has produced considerable advances in large-scale learning (Radford et al., 2021), several lines of work have begun to probe the fundamental role of invariance in contrastive learning. Two key conclusions of recent investigations include: 1) invariance limits the expressive power of features learned by f, as it removes information about features or transformations that may be relevant in fine-grained tasks (Lee et al., 2021; Xie et al., 2022), and 2) contrastive learning actually benefits from not having exact invariance. For instance, a critical role of the projection head is to expand the feature space so that f is not fully invariant (Jing et al., 2022), suggesting that it is preferable for the embeddings of x and x^+ to be close, but not identical.

Equivariance to augmentation. To address the limitations of invariance, recent work has additionally proposed to control *equivariance* (i.e., sensitivity) of f to data transformations (Dangovski et al., 2022; Devillers & Lefort, 2023; Garrido et al., 2023). Prior works can broadly be viewed as training a set of features f (sometimes alongside the usual invariant features) so that $f(a(x)) \approx T_a f(x)$ for samples $x \sim p$ from the data distribution where T_a is some transformation of the embedding space. A common choice is to take $T_a f(x) = \text{MLP}(f(x), a)$, a learnable feed-forward network, and optimize a loss $\|\text{MLP}(f(x), a) - f(a(x))\|_2$. Whilst a learnable MLP ensures that information about a is encoded into the embedding of $a(x)$, it permits complex non-linear relations between embeddings and hence does not necessarily encode relations in a linearly separable way. Furthermore, it does not enjoy the beneficial properties of equivariance in the formal group-theoretic sense, such as consistency under compositions in general: $T_{a_2 \circ a_1} f(x) \neq T_{a_2} T_{a_1} f(x)$.

C. Background on invariance theory for the orthogonal group

This section recalls some classical theory on orthogonal groups and an extension that we use for proving results over continuous data distributions.

A function $f : (\mathbb{R}^d)^n \to \mathbb{R}$ is said to be $O(d)$-invariant if $f(Rv_1, \dots, Rv_n) = f(v_1, \dots, v_n)$ for all $R \in O(d)$. Throughout this section, we are especially interested in determining easily computed statistics that *characterize* an $O(d)$ invariant function f. In other words, we would like to write f as a function of these statistics. The following theorem was first proved by Hermann Weyl using Capelli's identity (Weyl, 1946) and shows that the inner products $v_i^\top v_j$ suffice.

Theorem 3 (First fundamental theorem of invariant theory for the orthogonal group)**.** *Suppose that $f : (\mathbb{R}^d)^n \to \mathbb{R}$ is $O(d)$-invariant. Then there exists a function $g : \mathbb{R}^{n \times n} \to \mathbb{R}$ for which*

$$f(v_1, \dots, v_n) = g\big([v_i^\top v_j]_{i,j=1}^n\big).$$

In other words, to compute f at a given input, it is not necessary to know all of $v_1, \dots, v_n$. Computing the value of f at a point can be done using only the inner products $v_i^\top v_j$, which are invariant to $O(d)$. Letting V be the $n \times d$ matrix whose ith row is $v_i^\top$, we may also write $f(v_1, \dots, n_n) = g(VV^\top)$. The map $V \mapsto VV^\top$ is known as the orthogonal projection of V.

A corollary of this result has recently been used to develop $O(d)$ equivariant architectures in machine learning (Villar et al., 2021).

Corollary 2. *Suppose that A, B are $n \times d$ matrices and $AA^\top = BB^\top$. Then $A = BR$ for some $R \in O(d)$.*

(Villar et al., 2021) use this characterization of orthogonally equivariant functions to *parameterize* function classes of neural networks that have the same equivariance. This result is also useful in our context; However, we put it to use for a very different purpose: studying $\mathcal{L}_{\text{equi}}$.

Intuitively this result says the following: given two point clouds A, B of unit length vectors with some fixed correspondence (bijection) between each point in A and a point in B, if the *angles* between the ith and jth points in cloud A always equal the angle between the ith and jth point in cloud B, then A and B are the same up to an orthogonal transformation.

This is the main tool we use to prove the finite sample version of the main result for our equivariant loss (Proposition 2). However, to analyze the population sample loss $\mathcal{L}_{\text{equi}}$ (Proposition 1), we require an extended version of this result to the continuous limit as $n \to \infty$. To this end, we develop a simple but novel extension to Theorem 3 to the case of continuous data distributions. This result may be useful in other contexts independent of our setting.

Proposition 4. *Let $\mathcal{X}$ be any set and $f, h : \mathcal{X} \to \mathbb{R}^d$ be functions on $\mathcal{X}$. If $f(x)^\top f(y) = h(x)^\top h(y)$ for all $x, y \in \mathcal{X}$, then there exists $R \in O(d)$ such that $Rf(x) = h(x)$ for all $x \in \mathcal{X}$.*

The proof of this result directly builds on the finite sample version. The key idea of the proof is that since the embedding space $\mathbb{R}^d$ is finite-dimensional we may select a set of points $\{f(x_i)\}_i$ whose span has maximal rank in the linear space spanned by the outputs of f. This means that any arbitrary point $f(x)$ can be written as a linear combination of the $f(x_i)$. This observation allows us to apply the finite sample result on each $f(x_i)$ term in the sum to conclude that $f(x)$ is also a rotation of a sum of $h(x_i)$ terms. Next, we give the formal proof.

Proof of Proposition 4. Choose $x_1, \dots, x_n \in \mathcal{X}$ such that $F = [f(x_1) \mid \dots \mid f(x_n)]^\top \in \mathbb{R}^{n \times d}$ and $h = [h(x_1) \mid \dots \mid h(x_n)]^\top \in \mathbb{R}^{n \times d}$ have maximal rank. Note we use "|" to denote the column-wise concatenation of vectors. Note that such x_i can always be chosen. Since we have $FF^\top = HH^\top$, we know by Corollary 2 that $F = HR$ for some $R \in O(d)$.

Now consider an arbitrary $x \in \mathcal{X}$ and define $\tilde{F} = [F \mid f(x)]^\top$ and $\tilde{H} = [H \mid h(x)]^\top$, both of which belong to $\mathbb{R}^{(n+1) \times d}$. Note that again we have $\tilde{F}\tilde{F}^\top = \tilde{H}\tilde{H}^\top$ so also know that $\tilde{F} = \tilde{H}\tilde{R}$ for some $\tilde{R} \in O(d)$. Since x_i were chosen so that F and H are of maximal rank, we know that $h(x) = \sum_{i=1}^n c_i h(x_i)$ for some coefficients $c_i \in \mathbb{R}$, since if this were not the case then we would have $\text{rank}(\tilde{H}) = \text{rank}(H) + 1$.

From this, we know that

$$
\begin{aligned}
R^\top h(x) &= \sum_{i=1}^{n} c_i R^\top h(x_i) \\
&= \sum_{i=1}^{n} c_i f(x_i) \\
&= \sum_{i=1}^{n} c_i \tilde{R}^\top h(x_i) \\
&= \tilde{R}^\top \sum_{i=1}^{n} c_i h(x_i) \\
&= \tilde{R}^\top h(x) \\
&= f(x).
\end{aligned}
$$

So we have that $Rf(x) = RR^\top h(x) = h(x)$ for all $x \in \mathcal{X}$. □

D. Extensions to other groups: further discussion

In Section 3.2, we explore the possibility of formulating an equivariant loss $\mathcal{L}_{\text{equi}}$ for pairs of points that fully captures equivariance by requiring the group to be the stabilizer of a bilinear form. In this context, the invariants are generated by polynomials of degree two in two variables, and the equivariant functions can be obtained by computing gradients of these invariants (Blum-Smith & Villar, 2022). Section 3.2 notes that this holds true not only for the orthogonal group, which is the primary focus of our research but also for the Lorentz group and the symplectic group, suggesting natural extensions of our approach.

It is worth noting that the group of rotations $SO(d)$ does not fall into this framework. It can be defined as the set of transformations that preserve both inner products (a 2-form) and determinants (a d-form). Consequently, some of its generators have degree 2 while others have degree d (see (Weyl, 1946), Section II.A.9).

Weyl's theorem states that if a group acts on n copies of a vector space (in our case, $(\mathbb{R}^d)^n$ for consistency with the rest of the paper), its action can be characterized by examining how it acts on k copies (i.e., $(\mathbb{R}^d)^k$) when the maximum degree of its irreducible components is k (refer to Section 6 of (Schmid, 2006) for a precise statement of the theorem). Since our interest lies in understanding equivariance in terms of pairs of objects, we desire invariants that act on pairs of points. One way to guarantee this is to restrict ourselves to groups that act through representations where the irreducible components have degrees of at most two (though this is not necessary in all cases, such as the orthogonal group $O(d)$ that we consider in the main paper). An example of such groups is the product of finite subgroups of the unitary group $U(2)$, which holds relevance in particle physics. According to Weyl's theorem, the corresponding invariants can be expressed as *polarizations* of degree-2 polynomials on two variables. Polarizations represent an algebraic construction that enables the expression of homogeneous polynomials in multiple variables by introducing additional variables to polynomials with fewer variables. In our case, the base polynomials consist of degree-2 polynomials in two variables, while the polarizations incorporate additional variables. Notably, an interesting open problem lies in leveraging this formulation for contrastive learning.

E. Related work

Geometry of representations. Equivariance is a key tool for encoding geometric structure—e.g., symmetries—into neural network representations (Cohen & Welling, 2016; Bronstein et al., 2021). Whilst hard-coding equivariance into model architectures is very successful, approximate learned equivariance (Kaba et al., 2022; Shakerinava et al., 2022), has certain advantages: 1) when the symmetry is provided only by data, with no closed-form expression, 2) can still be used when it is unclear how to hard code equivariance into the architecture, and 3) can exploit standard high capacity architectures (He et al., 2016; Dosovitskiy et al., 2021), benefiting from considerable engineering efforts to optimize their performance. (Shakerinava et al., 2022) also consider learning orthogonal equivariance, but consider problems where both input and embedding space are acted on by $O(d)$. Our setting differs from this in two key ways: 1) we consider a very different set of transforms of input space—jitter, crops, etc.—and 2) can be naturally integrated into contrastive learning, and 3) theoretically study the

minima of the angle-preserving loss. A related line of work, *mechanistic interpretability*, hypothesizes that algorithmic structure—possibly including group symmetries—emerge naturally within network connections during training (Chughtai et al., 2023). Our approach is very different from this as we directly *train* models to have the desired structure without relying on implicit processes. Finally, the geometry of representation space has been used in a very different sense in prior contrastive learning approaches, for instance bootstrapping useful negatives (Chuang et al., 2020; Robinson et al., 2021) based on their location in embedding space during training.

Self-superised learning. Prior equivariant contrastive learning approaches extend the usual setup of learning invariance by learning *sensitivity* to certain features known to be important for downstream tasks. A subset of these works performs additional tasks of learning sensitivity to augmentation while learning invariances. For instance, (Dangovski et al., 2022) learns to predict the augmentation applied but only considers a discrete group of 4-fold rotations. (Lee et al., 2021) learns the difference of augmentation parameters and (Xiao et al., 2021) constructs separate embedding sub-spaces that capture invariances to all but one augmentation. However, these approaches do not offer a meaningful structure to the embedding space. Others attempt to control how this sensitivity occurs. Specifically, (Devillers & Lefort, 2023; Garrido et al., 2023; Bhardwaj et al., 2023) learn a mapping from one latent representation to another, predicting how data augmentation affects the resulting embedding. However, none of these approaches constrain the group action in the embedding space, resulting in complex non-linear augmentation maps.

F. Implementation details

Algorithm 1 presents pytorch-based pseudocode for implementing CARE. This implementation introduces the idea of using a smaller batch size for the equivariance loss compared to the InfoNCE loss. Specifically, by definition, the equivariance loss is defined as a double expectation, one over data pairs and the other over augmentations. Empirical observations reveal that sampling one augmentation per batch leads to unstable yet superior performance when compared to standard invariant-based baselines such as SimCLR. Since these invariant-based contrastive benchmarks generally perform well with large batch sizes, we adopt the approach of splitting a batch into multiple chunks to efficiently sample multiple augmentations per batch for the equivariance loss. Each chunk of the batch is associated with a new pair of augmentations, ensuring a large batch size for the InfoNCE loss and a smaller batch size for the equivariance loss.

Algorithm 1 PyTorch based pseudocode for CARE

Notations: f represents the backbone encoder network, λ is the weight on CARE loss, `apply_same_aug` function applies the same augmentation to all samples in the input batch
for minibatch x in `dataloader` **do**
  draw *two batches* of augmentation functions $a_1, a_2 \in \mathcal{A}$
  /* Functions a_1, a_2 apply different augmentation to each sample in batch x */
  $z_1^{\text{inv}}, z_2^{\text{inv}} = f(a_1(x)), f(a_2(x))$
  divide x into `n_split` chunks to form x_{chunks}
  /* Module for calculating orthogonal equivariance loss */
  for c_i in x_{chunks} in parallel **do**
    draw *two* augmentation functions $\tilde{a}_1, \tilde{a}_2 \in \mathcal{A}$
    /* Functions $\tilde{a}_1, \tilde{a}_2$ apply same augmentation to each sample in batch c_i */
    $\tilde{z}_{i1}, \tilde{z}_{i2} = f(\texttt{apply\_same\_aug}(c_i, \tilde{a}_1)), f(\texttt{apply\_same\_aug}(c_i, \tilde{a}_2))$
  /* Concatenate embedding vectors corresponding to all chunks */
  merge $\tilde{z}_{i1}, \tilde{z}_{i2}$ into $z_1^{\text{equiv}}, z_2^{\text{equiv}}$ respectively
  /* Loss computation */
  $\mathcal{L}_{\text{InfoNCE}}(f) = \texttt{infonce\_loss}(z_1^{\text{inv}}, z_2^{\text{inv}})$
  $\mathcal{L}_{\text{equiv}}(f) = \texttt{orthogonal\_equivariance\_loss}(z_1^{\text{equiv}}, z_2^{\text{equiv}}, \texttt{n\_split})$
  $\mathcal{L}_{\text{CARE}}(f) = \mathcal{L}_{\text{InfoNCE}}(f) + \lambda \cdot \mathcal{L}_{\text{equiv}}(f)$
  /* Optimization step */
  $\mathcal{L}_{\text{CARE}}(f)$.backward()
  optimizer.step()
=0

G. Supplementary experimental details and assets disclosure

G.1. Assets

We do not introduce new data in the course of this work. Instead, we use publicly available widely used image datasets for the purposes of benchmarking and comparison.

G.2. Hardware and setup

All experiments were performed on an HPC computing cluster using 4 NVIDIA Tesla V100 GPUs with 32GB accelerator RAM for a single training run. The CPUs used were Intel Xeon Gold 6248 processors with 40 cores and 384GB RAM. All experiments use the PyTorch deep learning framework (Paszke et al., 2019).

G.3. Experimental protocols

We first outline the training protocol adopted for training our proposed approach on a variety of datasets, namely CIFAR10, CIFAR100, STL10, and ImageNet100.

CIFAR10, CIFAR100 and STL10 All encoders have ResNet-50 backbones and are trained for 400 epochs with temperature $\tau = 0.5$ for SimCLR and $\tau = 0.1$ for MoCo-v2 [1]. The encoded features have a dimension of 2048 and are further processed by a two-layer MLP projection head, producing an output dimension of 128. A batch size of 256 was used for all datasets. For CIFAR10 and CIFAR100, we employed the Adam optimizer with a learning rate of $1e^{-3}$ and weight decay of $1e^{-6}$. For STL10, we employed the SGD optimizer with a learning rate of 0.06, utilizing cosine annealing and a weight decay of $5e^{-4}$, with 10 warmup steps. We use the same set of augmentations as in SimCLR (Chen et al., 2020). To train the encoder using $\mathcal{L}_{\text{CARE-SimCLR}}$, we use the same hyper-parameters for InfoNCE loss. Additionally, we use 4, 8 and 16 batch splits for CIFAR100, STL10 and CIFAR10, respectively. This allows us to sample multiple augmentations per batch, effectively reducing the batch size of equivariance loss whilst retaining the same for InfoNCE loss. Furthermore, for the equivariant term, we find it optimal to use a weight of $\lambda = 0.01, 0.001$, and 0.01 for CIFAR10, CIFAR100, and STL10, respectively.

ImageNet100 We use ResNet-50 as the encoder architecture and pretrain the model for 200 epochs. A base learning rate of 0.8 is used in combination with cosine annealing scheduling and a batch size of 512. For MoCo-v2, we use 0.99 as the momentum and $\tau = 0.2$ as the temperature. All remaining hyperparameters were maintained at their respective official defaults as in the official MoCo-v2 code. While training with $\mathcal{L}_{\text{CARE-SimCLR}}$ and $\mathcal{L}_{\text{CARE-MoCo}}$, we find it optimal to use splits of 4 and 8 and weight of $\lambda = 0.005$ and 0.01 respectively on the equivariant term.

Linear evaluation We train a linear classifier on frozen features for 100 epochs with a batch size of 512 for CIFAR10, CIFAR100, and STL10 datasets. To optimize the classifier, we employ the Adam optimizer with a learning rate of $1e^{-3}$ and a weight decay of $1e^{-6}$. In the case of ImageNet100, we train the linear classifier for 60 epochs using a batch size of 128. We initialize the learning rate to 30.0 and apply a step scheduler with an annealing rate of 0.1 at epochs 30, 40, and 50. The remaining hyper-parameters are retained from the official code.

G.4. Reproducibility statement

Algorithm 1 provides the pseudocode for implementing our work using the PyTorch framework. It serves as the primary public code for our proposed method CARE. To ensure reproducibility, Section G.3 details all the experimental configurations employed in our work. Additionally, we have included our code as supplementary material and will make it publicly available.

H. Additional experiments

H.1. Qualitative assessment of equivariance

A key property promised by equivariant contrastive models is sensitivity to specific augmentations. To qualitatively evaluate the sensitivity, or equivariance, of our models, we consider an image retrieval task on the Flowers-102 dataset (Nilsback &

[1] https://github.com/facebookresearch/moco

Zisserman, 2008), as considered by (Bhardwaj et al., 2023). Specifically, when presented with an input image x, we extract the top 5 nearest neighbors based on the Euclidean distance of $f(x)$ and $f(a(x))$, where $a \in \mathcal{A}$. We report the results of using color jitter as a transformation of the input, comparing the invariant (SimCLR) and our equivariant (CARE) models in Figure 6. We see that retrieved results for the CARE model exhibit greater variability in response to a change in query color compared to the SimCLR model. Notably, the color of the retrieved results for all queries in the SimCLR model remains largely invariant, thereby confirming its robustness to color changes.

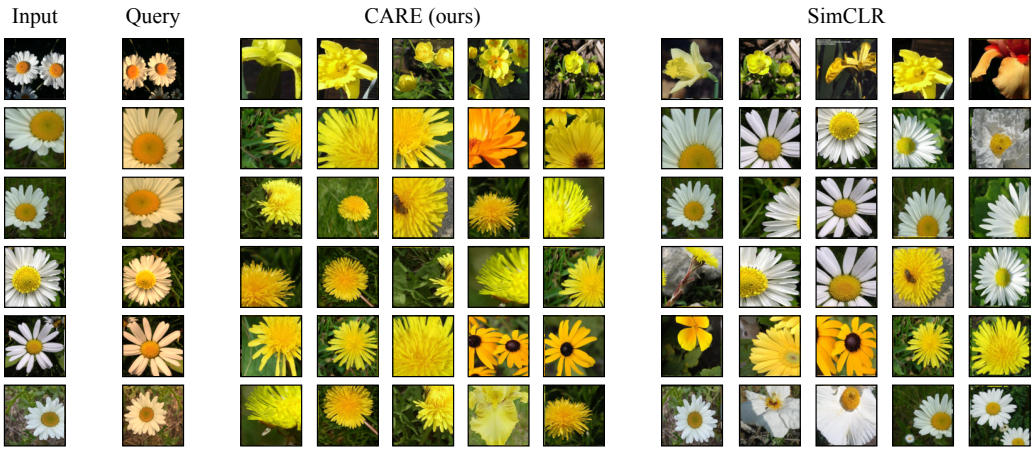

Figure 6: CARE exhibits sensitivity to features that invariance-based contrastive methods (e.g., SimCLR) do not. For each input we apply color jitter to produce the query image. We then retrieve the 5 nearest neighbors in the embedding space of CARE and SimCLR.

H.2. Quantitative assessment of equivariance

H.2.1. Relative rotational equivariance.

Optimizing for the CARE objective may potentially result in learning invariance rather than equivariance. Specifically, for input image x, $f(a(x)) = f(x)$ for $a \in \mathcal{A}$ is a trivial optimal solution of $\arg\min_f \mathcal{L}_{\text{equi}}(f)$. To check that our model is learning non-trivial equivariance, we consider a metric similar to one proposed by (Bhardwaj et al., 2023) for measuring the equivariance *relative* to the invariance of f:

$$\gamma_f = \mathbb{E}_{a\sim\mathcal{A}}\mathbb{E}_{x,x'\sim\mathcal{X}}\left\{\frac{(\|f(a(x')) - f(a(x))\|^2 - \|f(x') - f(x)\|^2)^2}{(\|f(a(x')) - f(x')\|^2 + \|f(a(x)) - f(x)\|^2)^2}\right\}. \tag{3}$$

Here, the denominator measures the invariance of the representation, with smaller values corresponding to greater invariance to the augmentations. The numerator, on the other hand, measures equivariance and can be simplified to $\left[f(a(x'))^\top f(a(x)) - f(x)^\top f(x')\right]^2$ (i.e., $\mathcal{L}_{\text{equi}}(f)$) up to a constant, because f maps to the unit sphere. The ratio γ_f of these two terms measures the non-trivial equivariance, with a lower value implying greater non-trivial orthogonal equivariance.

We measure the relative rotational equivariance for both CARE and SimCLR over the course of pretraining by following the approach outlined in Section 4. Specifically, we compare ResNet-18 models trained using CARE and SimCLR on CIFAR10. From Figure 7, we observe that both the models produce embeddings with comparable non-zero invariance loss $\mathcal{L}_{\text{inv}}$, indicating approximate invariance. However, they differ in their sensitivity to augmentations, with CARE attaining a much lower relative equivariance error. Importantly, this shows that CARE is *not* achieving lower equivariance error $\mathcal{L}_{\text{equi}}$ by collapsing to invariance, a trivial form of equivariance.

H.2.2. Analyzing structure on a 2D manifold.

To further study $\mathcal{L}_{\text{equi}}$, we train an encoder f that projects the input onto $\mathbb{S}^1$, the unit circle in the 2D plane. In this case, orthogonal transformations are characterized by *angles*. We sample an augmentation $a \sim \mathcal{A}$ and measure the cosine of the angle between pairs $f(x)$ and $f(a(x))$ for all x in the test set. This process is repeated for 20 distinct sampled augmentations, and the density of all recorded cosine angles is recorded in Figure 8. Both CARE and SimCLR exhibit high density close

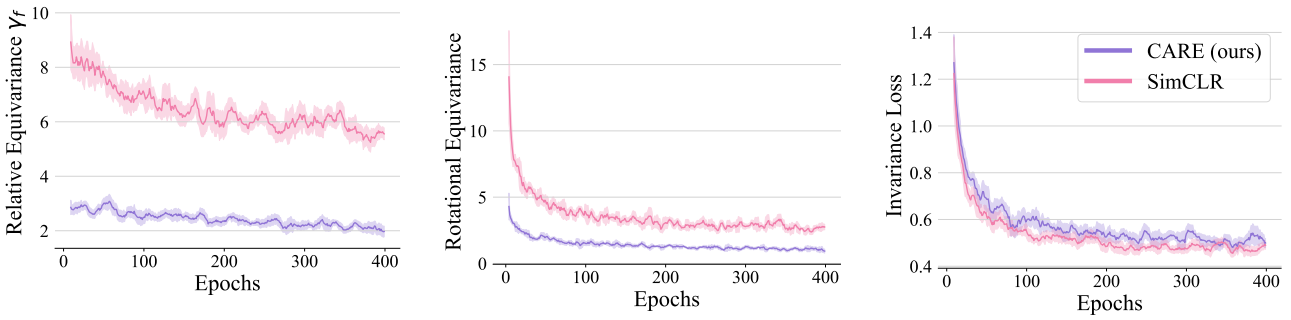

Figure 7: **Relative rotational equivariance** (lower is more equivariant). Both CARE and invariance-based contrastive methods (e.g., SimCLR) produce *approximately* invariant embeddings. However, they differ in their residual sensitivity to augmentations. CARE learns a considerably more rotationally structured embedding space. We note that this is in part because CARE is less invariant to augmentations (higher invariance loss).

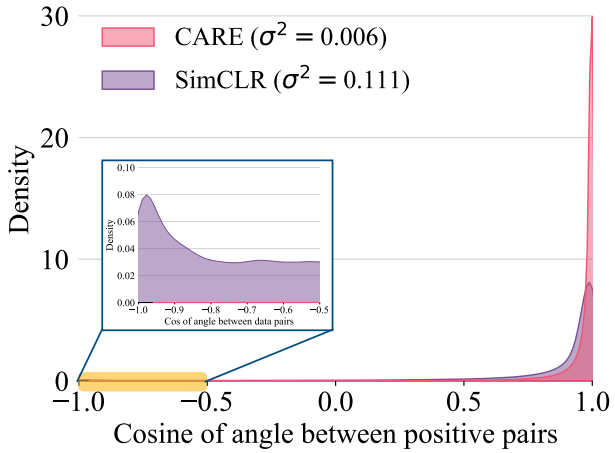

Figure 8: Histogram of the cosine of angles between data pairs for CARE and SimCLR. CARE exhibits a significantly lower variance of cosine similarity values compared to SimCLR.

to 1, demonstrating approximate invariance. However, unlike CARE, SimCLR exhibits non-zero density in the region -0.5 to -1.0, indicating that the application of augmentations significantly displaces the embeddings. Additionally, CARE consistently exhibits lower variance σ^2 of the cosine angles between $f(x)$ and $f(a(x))$ for a fixed augmentation, as expected given that it is supposed to transform all embeddings in the same way.

H.2.3. HISTOGRAM FOR LOSS ABLATION.

To accompany Figure 2, this section plots the cosine similarity between positive pairs. We provide two plots for each experiment: the first plots the *histogram* of similarities of positive pairs drawn from the test set; the second plots the *average* positive cosine similarity throughout training. The results are reported in Figures 9, 10, 11, 12, 13, 14.

I. Discussion

Converting transformations that are complex in input space into simple transformations in embedding space has many potential uses. For instance, modifying data (e.g., in order to reason about counterfactuals) can be viewed as transforming one embedding to another. If the sought after transformation was *simple* and *predictable*, it may be easier to find. Similarly, generalizing out-of-distribution is easier when extrapolating linearly (Xu et al., 2021), suggesting that linear transformations of embedding space may facilitate more reliable generalization. This work considers several design principles that may be broadly relevant: 1) *learned* equivariance preserves the expressivity of backbone architectures, and in some cases may be

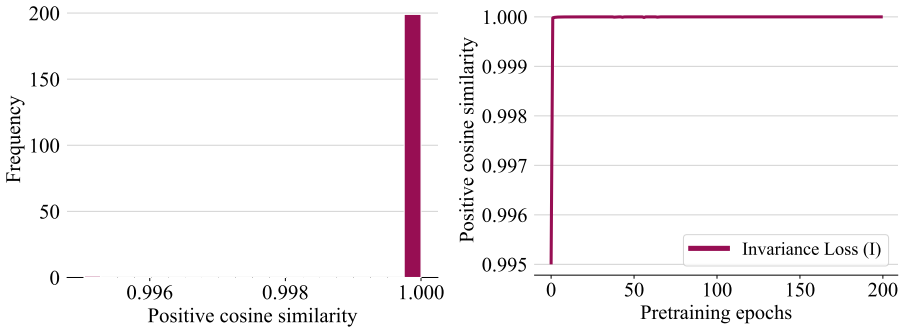

Figure 9: (left) Histogram of positive cosine similarity values at the end of pre-training using the invariance loss; (right) Evolution of positive cosine similarity values over pre-training epochs using the invariance loss

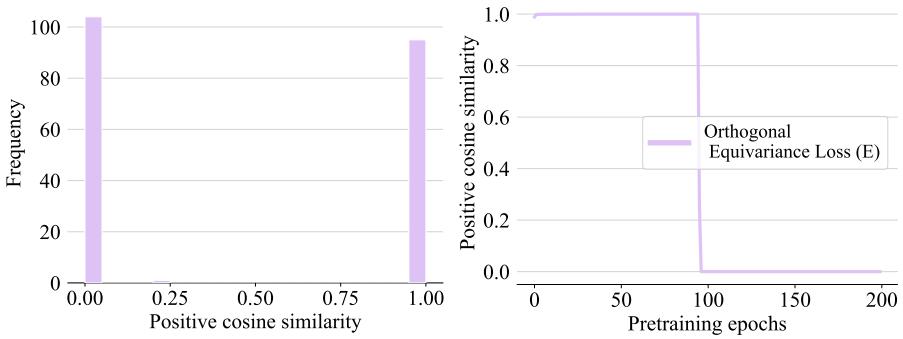

Figure 10: (left) Histogram of positive cosine similarity values at the end of pre-training using the orthogonal equivariance loss; (right) Evolution of positive cosine similarity values over pre-training epochs using the orthogonal equivariance loss

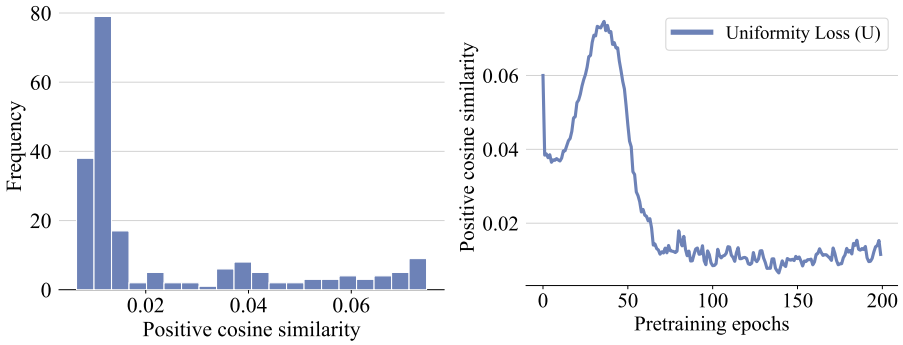

Figure 11: (left) Histogram of positive cosine similarity values at the end of pre-training using the uniformity loss; (right) Evolution of positive cosine similarity values over pre-training epochs using the uniformity loss

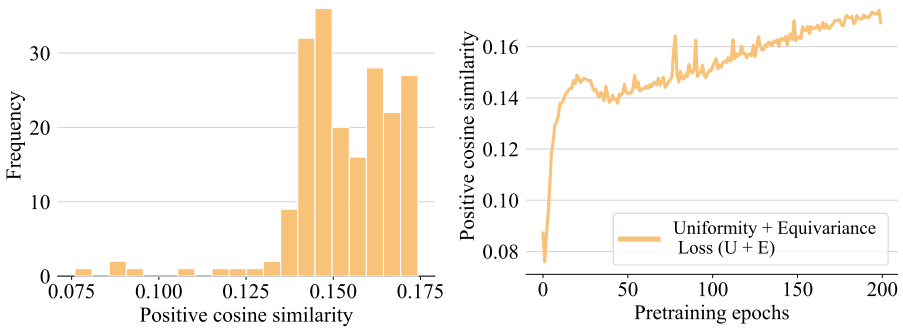

Figure 12: (left) Histogram of positive cosine similarity values at the end of pre-training using the Uniformity + Equivariance loss; (right) Evolution of positive cosine similarity values over pre-training epochs using the Uniformity + Equivariance loss

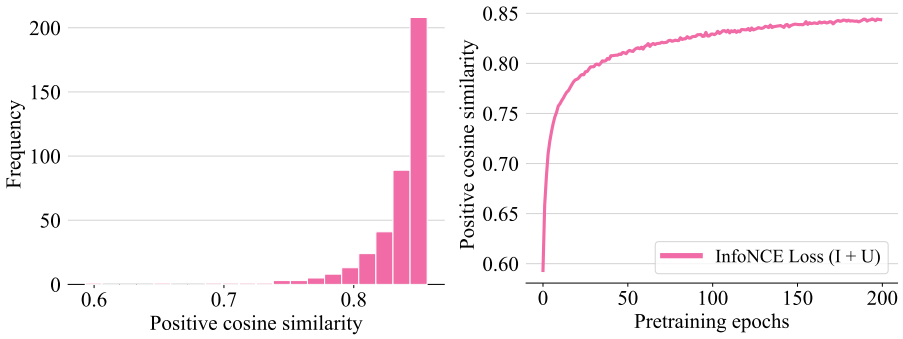

Figure 13: (left) Histogram of positive cosine similarity values at the end of pre-training using the InfoNCE (invariance + uniformity) loss; (right) Evolution of positive cosine similarity values over pre-training epochs using the InfoNCE loss

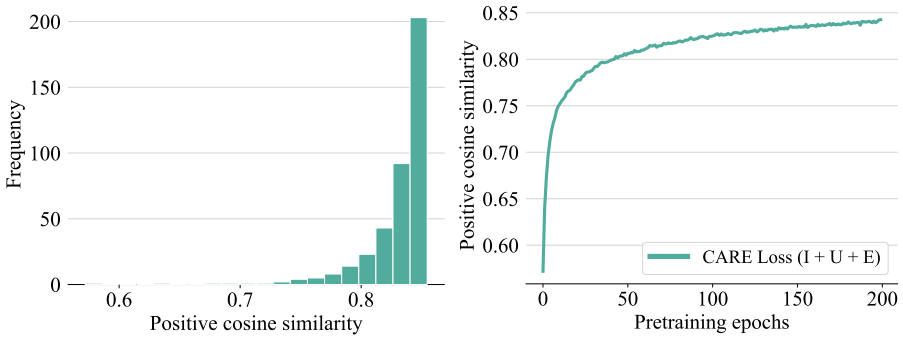

Figure 14: (left) Histogram of positive cosine similarity values at the end of pre-training using the CARE (InfoNCE + orthogonal equivariance) loss; (right) Evolution of positive cosine similarity values over pre-training epochs using the CARE loss

easier for model design than hard-coded equivariance, 2) linear group actions are desirable, but require carefully designed objectives (similar in spirit to the principle of *parsimony* (Ma et al., 2022), also advocated for by (Shakerinava et al., 2022)), and 3) orthogonal (and related) symmetries are a promising structure for Siamese network training as they can be efficiently learned using *pair-wise* data comparisons.

Limitations. While our method, CARE, learns embedding spaces with many advantages over prior contrastive learning embedding spaces, there are certain limitations that we acknowledge here. First, we do not provide a means to directly identify the rotation corresponding to a specific transformation. Instead, our approach allows the recovery of the rotation by solving Wahba's problem. However, this requires solving an instance of Wahba's for each augmentation of interest. Future improvements that develop techniques for quickly and easily (i.e., without needing to solve an optimization problem) identifying specific rotations would be a valuable improvement, enhancing the steerability of our models. Second, it is worth noting that equivariant contrastive methods, including CARE, only achieve approximate equivariance. This is a fundamental challenge shared by all such methods, as it is unclear how to precisely encode exact equivariance. The question remains open as to a) whether this approximate equivariance should be considered damaging in the first place, and if so, b) whether scaling techniques can sufficiently produce reliable approximate equivariance to enable the diverse applications that equivariance promises. Addressing this challenge is a crucial area for future research and exploration in the field. Each of these limitations points to valuable directions for future work.

Broader impact. Through our self-supervised learning method CARE we explore foundational questions regarding the structure and nature of neural network representation spaces. Currently, our approaches are exploratory and not ready for integration into deployed systems. However, this line of work studies self-supervised learning and therefore has the potential to scale and eventually contribute to systems that do interact with humans. In such cases, it is crucial to consider the usual safety and alignment considerations. However, beyond this, CARE, offers insights into algorithmic approaches for controlling and moderating model behavior. Specifically, CARE identifies a simple rotation of embedding space that corresponds to a change in the attribute of the data. In principle, this transformation could be used to "canonicalize" data, preventing the model from relying on certain attributes in decision-making. Additionally, controlled transformations of embeddings could be used to debias model responses and achieve desired variations in output. It is important to note that while our focus is on the core methodology, we do not explore these possibilities in this particular work.

References

[A1] Bhardwaj, S., McClinton, W., Wang, T., Lajoie, G., Sun, C., Isola, P., and Krishnan, D. Steerable equivariant representation learning. *preprint arXiv:2302.11349*, 2023.

[A2] Blum-Smith, B. and Villar, S. Equivariant maps from invariant functions. *preprint arXiv:2209.14991*, 2022.

[A3] Bronstein, M. M., Bruna, J., Cohen, T., and Veličković, P. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *preprint arXiv:2104.13478*, 2021.

[A3] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, pp. 1597–1607. PMLR, 2020.

[A5] Chen, X. and He, K. Exploring simple siamese representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15750–15758, 2021.

[A6] Chuang, C.-Y., Robinson, J., Lin, Y.-C., Torralba, A., and Jegelka, S. Debiased contrastive learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 8765–8775, 2020.

[A7] Chughtai, B., Chan, L., and Nanda, N. A toy model of universality: Reverse engineering how networks learn group operations. In *ICLR Workshop on Physics for Machine Learning*, 2023.

[A8] Cohen, T. and Welling, M. Group equivariant convolutional networks. In *International Conference on Machine Learning (ICML)*, pp. 2990–2999. PMLR, 2016.

[A5] Dangovski, R., Jing, L., Loh, C., Han, S., Srivastava, A., Cheung, B., Agrawal, P., and Soljačić, M. Equivariant contrastive learning. In *International Conference on Learning Representations (ICLR)*, 2022.

[A6] Devillers, A. and Lefort, M. Equimod: An equivariance module to improve self-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2023.

[A11] 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 (ICLR)*, 2021.

[A7] Garrido, Q., Najman, L., and Lecun, Y. Self-supervised learning of split invariant equivariant representations. *preprint arXiv:2302.10283*, 2023.

[A13] Gutmann, M. and Hyvärinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 297–304, 2010.

[A14] He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[A15] Jing, L., Vincent, P., LeCun, Y., and Tian, Y. Understanding dimensional collapse in contrastive self-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2022.

[A16] Kaba, S.-O., Mondal, A. K., Zhang, Y., Bengio, Y., and Ravanbakhsh, S. Equivariance with learned canonicalization functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[A17] Lee, H., Lee, K., Lee, K., Lee, H., and Shin, J. Improving transferability of representations via augmentation-aware self-supervision. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 17710–17722, 2021.

[A9] Ma, Y., Tsao, D., and Shum, H.-Y. On the principles of parsimony and self-consistency for the emergence of intelligence. *Frontiers of Information Technology & Electronic Engineering*, 23(9):1298–1323, 2022.

[A19] Nilsback, M.-E. and Zisserman, A. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pp. 722–729. IEEE, 2008.

[A20] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

[A10] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pp. 8748–8763. PMLR, 2021.

[A22] Robinson, J., Chuang, C.-Y., Sra, S., and Jegelka, S. Contrastive learning with hard negative samples. In *International Conference on Learning Representations (ICLR)*, 2021.

[A23] Schmid, B. J. Finite groups and invariant theory. In *Topics in Invariant Theory: Séminaire d'Algèbre P. Dubreil et M.-P. Malliavin 1989–1990 (40ème Année)*, pp. 35–66. Springer, 2006.

[A12] Shakerinava, M., Mondal, A. K., and Ravanbakhsh, S. Structuring representations using group invariants. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[A25] van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *preprint arXiv:1807.03748*, 2018.

[A26] Villar, S., Hogg, D. W., Storey-Fisher, K., Yao, W., and Blum-Smith, B. Scalars are universal: Equivariant machine learning, structured like classical physics. pp. 28848–28863, 2021.

[A13] Wang, T. and Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning (ICML)*, pp. 9929–9939. PMLR, 2020.

[A28] Weyl, H. *The classical groups: their invariants and representations*. Number 1. Princeton university press, 1946.

[A29] Xiao, T., Wang, X., Efros, A. A., and Darrell, T. What should not be contrastive in contrastive learning. In *International Conference on Learning Representations (ICLR)*, 2021.

[A30] Xie, Y., Wen, J., Lau, K. W., Rehman, Y. A. U., and Shen, J. What should be equivariant in self-supervised learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4111–4120, 2022.

[A31] Xu, K., Zhang, M., Li, J., Du, S. S., Kawarabayashi, K.-i., and Jegelka, S. How neural networks extrapolate: From feedforward to graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2021.