Improving Self-Supervised Learning by Characterizing Idealized Representations

Anonymous Author(s)
Affiliation
Address
email

Abstract

Despite the empirical successes of self-supervised learning (SSL) methods, it is unclear what characteristics of their representations lead to high downstream accuracies. In this work, we characterize properties that SSL representations should *ideally* satisfy. Specifically, we prove necessary and sufficient conditions such that for any task invariant to given data augmentations, probes (e.g., linear or MLP) trained on that representation attain perfect accuracy. These requirements lead to a unifying conceptual framework for improving existing SSL methods and deriving new ones. For contrastive learning, our framework prescribes simple but significant improvements to previous methods such as using asymmetric projection heads. For non-contrastive learning, we use our framework to derive a simple and novel objective. Our resulting SSL algorithms outperform baselines on standard benchmarks, including SwAV+multicrops on linear probing of ImageNet.

1 Introduction

We study self-supervised learning (SSL), where the goal is to learn representations from minimal supervision, such that simple probes trained on these representations achieve high downstream accuracy. Recently, there has been many different SSL methods achieving impressive empirical results (e.g. SimCLR [1], SwAV [2]) using label-preserving augmentations (e.g. cropping or color jittering) as supervision. We dub this setting *invariant SSL* (ISSL). Despite these empirical successes, it remains unclear how these various SSL methods relate to one another, how to improve them, and how to derive new ones. Our goal is to provide a simple conceptual framework to think about those questions.

To derive such a framework, we ask ourselves: *what are the ideal requirements that ISSL representations should aim to satisfy?* We prove necessary and sufficient requirements to ensure that probes from a specified family, e.g. linear or multi-layer perceptron (MLP), perfectly classify any task that is invariant to desired data augmentations. This complements theoretical work in ISSL [3–8], which analyze specific ISSL algorithms. Our work instead focuses on properties of representations that should serve as a goal for any ISSL algorithm. These ideal properties are: (i) desired predictors should be able to distinguish positive and negative examples from the representation; (ii) the dimensionality of the representation should be sufficiently large; (iii) augmented inputs should map to the same representations.

Previous ISSL methods can be seen as approximations of our ideal requirements. Using our requirements, we derive simple improvements to those approximations as well as new ISSL objectives. Our theory thus results in a unifying conceptual ISSL framework, with practical prescriptions including:

- improvements to existing methods, such as increasing the dimensionality of representations and using asymmetric projections heads, which lead to 5% point gains on TinyImageNet;
- a novel non-contrastive ISSL objective that outperforms all baselines, including SwAV+multicrops, by at least 1% point on linear classification of ImageNet.
- extensions of ISSL algorithms to learn representations that are better suited for non-linear probes.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

2 Problem statement

2.1 Invariant Self-Supervised Learning

SSL learns an encoder ϕ that maps an input x from a finite space $\mathcal{X}$ (e.g., images of a given size) into a representation $\phi(x) \in \mathbb{R}^d$. Given the encoder and a dataset $\mathcal{D}_t$ drawn from some task of interest $p_t(X, Y) \in \mathcal{T}$, we fit a classifier f from a desired family of probes $\mathcal{F}$. Families of probes are sets of k-ary classifiers for any $k \geq 2$ such as linear or MLP probes. For clarity, we consider linear probes $\mathcal{F}$ until Sec. 5.

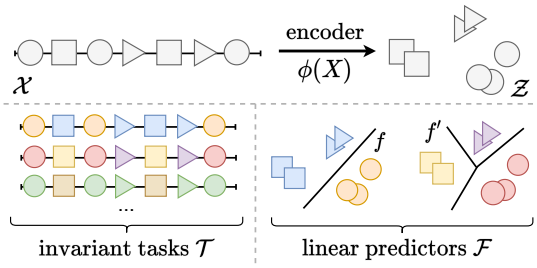


Figure 1: ISSL setting. (Top) 1D inputs, partitioned into 3 equivalence classes (shapes), are encoded by ϕ into a 2D representation. (Bot. left) 3 $\sim$-invariant tasks, where labels are the colors. (Bot. right) examples of probes for 2 of the invariant tasks.

Supervision for ISSL comes from unlabeled data p_X and label-preserving augmentations. Augmentations are ways of sampling "positive" x, x^+ examples that are equivalent for downstream classification tasks. We formalize this using an equivalence relation $x \sim x^+$ that partitions the inputs $\mathcal{X}$ into equivalent classes $[x] \in \mathcal{X}/\!\sim$, and we consider the following downstream tasks $\mathcal{T}$ whose labelings are deterministic and constant within these classes (we allow stochastic labeling in appendices).

Definition 1. The $\sim$-*invariant tasks* $\mathcal{T}$, is the set of all input-label distributions $p_t(X, Y)$ such that the labeling $p_t(Y|X)$ is deterministic and invariant to $\sim$, i.e.,

$$\text{for all } \; p_t \in \mathcal{T}, \; x, x^+ \in \mathcal{X}: \qquad x \sim x^+ \implies \arg\max_{y \in \mathcal{Y}} p_t(y|x) = \arg\max_{y \in \mathcal{Y}} p_t(y|x^+). \tag{1}$$

As an illustration, consider the 3 classes (triangle, square, and circle) shown in Fig. 1. Then $\mathcal{T}$ consists of all tasks that are predictable from those shapes, e.g., recognizing shapes with vertices (blue/orange in Fig. 1) or recognizing the shape (yellow/red/purple in Fig. 1). Note that equivalence relations defined here can model arbitrary transformations including cropping and adding Gaussian noise, which contrasts with typical restrictions to augmentations defined by group actions[9–11].

2.2 Idealized representations for ISSL

In this section, we define the optimal encoders, those that induce idealized representations that ISSL should be striving for. Although such encoders exist, they will likely not be learned in practice. Those idealized representations will nevertheless allow us to derive practical algorithms in Sec. 4.

The goal of ISSL is to learn representations from which (typically simple) probes $f \in \mathcal{F}$ classify well downstream tasks $p_t \in \mathcal{T}$, i.e., they achieve low 0-1 risk $\mathrm{R}_t(\phi, f) := \mathbb{E}_{p_t(X,Y)}[\mathbb{1}[Y \neq f(\phi(X))]]$. Of course, ideally, for any task of interest $p_t \in \mathcal{T}$ there will be a probe that can classify it perfectly.

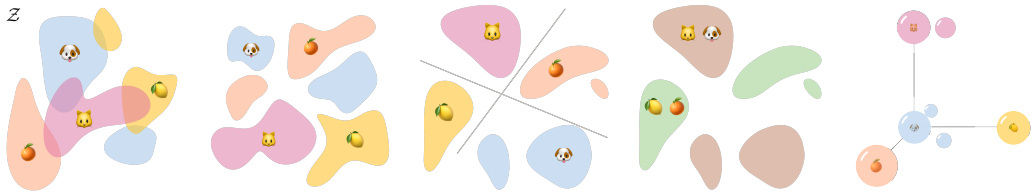
Definition 2. An encoder ϕ is *population optimal* for $\mathcal{T}, \mathcal{F}$, denoted as $\phi \in \Phi_{\text{pop}}$, iff predictors realize the Bayes error on any task of interest, i.e., for all $p_t \in \mathcal{T}$ we have $\inf_{f \in \mathcal{F}} \mathrm{R}_t(\phi, f) = 0$.

Population optimal encoders ensure the existence of perfect downstream probes. Such predictors would essentially be learned with infinite downstream data. In practice, however, predictors will be trained from finite datasets $\mathcal{D}_t$ of possibly small size n with empirical risk minimization. When n is small a fitted probe (ERM) $\hat{f} \in \widehat{\mathcal{F}}(\mathcal{D}_t, \phi) := \arg\min_{f \in \mathcal{F}} |\mathcal{D}_t|^{-1} \sum_{x,y \in \mathcal{D}_t} \mathbb{1}[y \neq f(\phi(x))]$ could be a terrible population predictor even when the underlying encoder is population optimal. Ideally, representations would thus also guarantee that *any* ERM performs as well as possible for *any* desired task and dataset size n. This suggests minimizing the following worst expected risk over tasks and ERMs.

$$\mathrm{W}_n(\phi, \mathcal{F}, \mathcal{T}) := \sup_{t \in \mathcal{T}} \mathbb{E}_{\mathcal{D}_t \overset{\text{iid}}{\sim} p_t^n(X,Y)} \left[\sup_{\hat{f} \in \widehat{\mathcal{F}}(\mathcal{D}_t, \phi)} \mathrm{R}_t(\phi, \hat{f}) \right]. \tag{2}$$

Definition 3. An encoder ϕ^* is *sample optimal* for $\mathcal{T}, \mathcal{F}$ iff if it is population optimal and minimizes the worst expected risk of ERMs for arbitrary sample sizes, i.e.,

$$\text{for all } \; n \geq 1: \qquad \phi^* \in \arg\min_{\phi \in \Phi_{\text{pop}}} \mathrm{W}_n(\phi, \mathcal{F}, \mathcal{T}). \tag{3}$$



(a) M not predictable (b) M non-lin. predictable (c) M linearly predictable (d) Tasks are not lin. pred. (e) Population optimal

Figure 2: Representations are *population optimal* for linear $\mathcal{F}$ iff their dimensionality is sufficiently large and a maximal invariant $M(X)$ is linearly predictable. The first 2 figures show representations from which $M(X)$ is (a) never or (b) only non-linearly predictable. Although (c) ensures linear predictability of $M(X)$, there exist invariant tasks, e.g. (d), that are non-linearly predictable. (e) Linear predictability of $M(X)$ ensures population optimality iff the dimensionality is at least $|\mathcal{X}/{\sim}| - 1$.

3 Theoretical framework for linear $\mathcal{F}$

3.1 Characterizing optimal representations for linear ISSL

In this section, we characterize all sample optimal encoders (Def. 3), with simple properties which give insights into the form of the idealized representation. The key to our theory is that there exists a *maximal invariant* for $\sim$, such that any $\sim$-invariant function h can be written as a composition between some function c_g and a *maximal invariant* $M : \mathcal{X} \to \mathcal{M}$, i.e., $g = c_g \circ M$ [12]. For simplicity, we consider maximal invariants that index the equivalence class, i.e., such that $\mathcal{M} := \{1, \dots, |\mathcal{X}/{\sim}|\}$ and,

$$\text{for any } x, x^+ \in \mathcal{X} : \quad x \sim x^+ \iff M(x) = M(x^+). \tag{4}$$

To build intuition let us first discuss population optimal encoders (Def. 2). By Def. 1, labels of downstream tasks $p_t \in \mathcal{T}$ are $\sim$-invariant. Labelings can thus be written as some function c_t of a $M(X)$, i.e., $\arg\max_y p_t(y|X) = c_t(M(X))$. This shows that $M(X)$ contains all and only information about downstream tasks. If probes were unconstrained, population optimal encoders would thus be those from which $M(X)$ is predictable, i.e., there exist an h_M such that $M(X) = h_M(\phi(X))$. This information about the downstream task will generally not be usable by linear probes, i.e., $c_t \circ h_M \notin \mathcal{F}$. As an illustration, consider 4 equivalence classes: cats, dogs, oranges, and lemons. Fig. 2b shows a representation from which $M(X)$ is predictable but invariant tasks are not linearly classifiable. In fact, even when $M(X)$ is linearly predictable $h_M \in \mathcal{F}$, as in Fig. 2c, downstream labels might not be, i.e., $c_t \circ h_M \notin \mathcal{F}$ as shown for the binary task from Fig. 2d. By standard VC dimension arguments [13, 14], such binary task is linearly predictable if the representation's dimensionality d is one less number of equivalence classes $|\mathcal{X}/{\sim}| - 1$, as in Fig. 2e. Building on this intuition we prove that population optimal encoders are essentially those that induce $d \geq |\mathcal{X}/{\sim}| - 1$ dimensional representations from which $M(X)$ is linearly predictable.

Population optimality does not ensure that ERMs perform well from finite samples. Even if good linear predictors exist, one might be unlucky and learn a bad ERM as shown on the left of Fig. 3. Intuitively, one can remove bad ERMs by mapping equivalent examples to the same representation, i.e., by using invariant encoders. Indeed, this ensures that ERMs that correctly predict one example in an equivalence class, also correctly predict all the others, as seen on the right of Fig. 3. We prove that such invariance of population optimal encoders is necessary and sufficient for sample optimality.

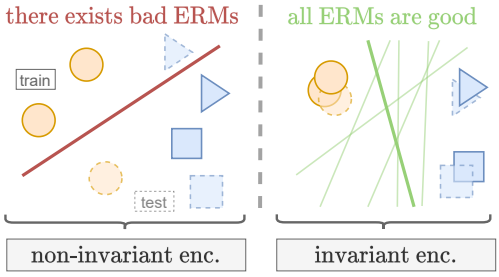


Figure 3: Invariant encoders ensure that there exists no bad ERM.

Theorem 1. An encoder ϕ^* is sample optimal for $\sim$-invariant tasks $\mathcal{T}$ and linear $\mathcal{F}$ if and only if

- **$\mathcal{F}$-predictability of** M: there exists a max. invariant M and an $f \in \mathcal{F}$ s.t. $M(X) = f(\phi^*(X))$;
- **Dimensionality**: the effective dimensionality of the representations is at least one less than the number of equivalence classes, i.e., $\dim(\text{span}(\{\phi^*(x) | x \in \mathcal{X}\})) \geq |\mathcal{X}/{\sim}| - 1$;
- **Invariance**: ϕ^* is $\sim$-invariant, i.e., for any $x, x^+ \in \mathcal{X}$ we have $x \sim x^+ \implies \phi^*(x) = \phi^*(x^+)$.

3.2 The impact of augmentations on downstream performance

Let us compute the worst-case excess risk $\mathrm{W}_n(\phi, \mathcal{F}, \mathcal{T})$ of sample optimal encoders and show its dependence on the invariance structure. The key is that since sample optimal encoders are invariant (Theorem 1), ERMs only need a single example per equivalence class. The risk then depends on the proportion of equivalence classes seen when training the probe. This proportion depends on the number of equivalence classes $|\mathcal{X}/{\sim}|$ and can be computed using classical occupancy results (see e.g., [15]).

Proposition 2. The worst expected risk of a sample optimal ϕ^* for $\sim$-invariant tasks $\mathcal{T}$ and any $\mathcal{F}$ is

$$\mathrm{W}_n(\phi^*, \mathcal{F}, \mathcal{T}) = \left(1 - \frac{1}{|\mathcal{X}/{\sim}|}\right)^n \tag{5}$$

Prop. 2 shows that fewer equivalence classes, i.e., coarser $\sim$, will lead to better downstream sample efficiency as ERMs only need to see a single example per class. In practice, this suggests using stronger data augmentations (those that induce coarser $\sim$) which are still label-preserving. Of course, the convergence rate will be slower for practical encoders than for the sample optimal ones.

4 Practical ISSL objectives

In this section, we derive simple objectives that learn optimal encoders in *ideal settings* (infinite data, perfect optimizers, universal function approximators). In the process, we shed light on why previously proposed ISSL methods work, and how to improve them in practice.

All our derivations use the fact that a sample optimal encoder can be achieved by training it to predict $M(X)$ using log loss in high dimensions. Intuitively, this is because logistic regression not only ensures linear predictability of $M(X)$ but also minimizes the variance within a class (i.e. invariance) and maximizes the angle between different classes (i.e. maximal effective dimensionality). Indeed, from the neural collapse literature [16–23], representations minizing the log loss will form a simplex equiangular tight frame (sETF) as in Fig. 4. We call the following loss of multinomial logistic regressors predicting $M(X)$ the *$\sim$-ISSL log loss*

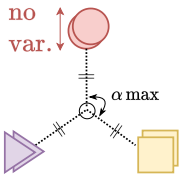


Figure 4: sETF.

$$\mathcal{L}_I(\phi; p_X) := \inf_{w \in W_1} \mathbb{E}_{p_X}\left[-\log \frac{\exp\big(w(M(X))^\top \phi(X)\big)}{\sum_{m'=1}^{|\mathcal{X}/{\sim}|} \exp(w(m')^\top \phi(X))}\right], \tag{6}$$

where w maps classes to weights, e.g., by indexing a weight matrix. Let $\mathcal{S}$ denote the $(d{-}1)$-sphere. Following previous work on neural collapse, we assume for this section that classes are equiprobable $p_X([x]) = 1/|\mathcal{X}/{\sim}|$, and that classifier's weights are unit-normalized $w \in W_1 := \mathcal{S}^{\mathcal{M}}$ as well as representations $\phi \in \Phi_1 := \mathcal{S}^{\mathcal{X}}$. (These assumptions are not necessary but simplify exposition.) We then have that any high dimension encoder trained to minimize the ISSL log loss will be sample optimal.

Proposition 3. Let p_X be a distribution whose support is $\mathcal{X}$ and such that equivalence classes are equiprobable $p_X([x]) = 1/|\mathcal{X}/{\sim}|$, $\forall x \in \mathcal{X}$. Any unit-normalized encoder that minimizes the $\sim$-ISSL log loss $\phi^* \in \arg\min_{\phi \in \Phi_1} \mathcal{L}_I(\phi; p_X)$ is optimal for $\sim$-invariant tasks and linear probes $\mathcal{F}$.

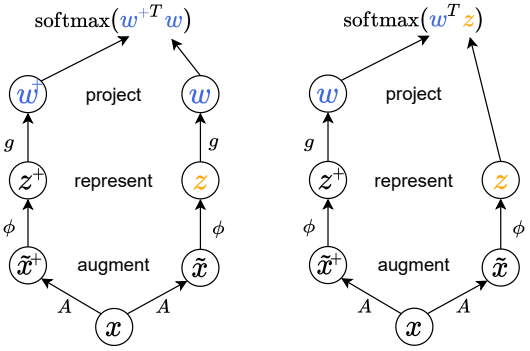
Prop. 3 shows that the ISSL log loss is a perfect pretext task in ideal settings. This suggests optimizing the ISSL log loss in practice. Unfortunately, we typically do not have access to the maximal invariant $M(X)$ nor the number of equivalence classes $|\mathcal{X}/{\sim}|$ required to compute the denominator of Eq. (6). Instead, knowledge about the desired equivalence structure comes from data augmentations $A(\tilde{X}|X)$ from which we can sample examples that are equivalent to the input $\tilde{X} \sim X$.

Having established our framework, we can re-interpret previous ISSL methods as practical objectives that approximate the ISSL log loss using data augmentations (SimCLR, SwAV, DINO [24], [2], SimSiam [25], MOCO [26]). These approximations are nevertheless suboptimal, in that none of them learn sample (nor population) optimal encoders in idealized settings. By directly deriving the ISSL methods from Eq. (6), we will prescribe improvements to previous ISSL objectives that ensure that sample idealized encoders are learned in idealized settings. We broadly categorize previous methods into two families depending on whether they select the desired number of equivalence classes $|\mathcal{X}/{\sim}|$ or whether it is implicitly inferred from the augmentations. We call these approaches distillation and contrastive ISSL respectively. For all derivations and theoretical claims see Appx. C.

4.1 Contrastive ISSL (CISSL)

Inspired by previous contrastive objectives [1, 27–29], we show how to optimize the ISSL log loss using data augmentations to implicitly replace the maximal invariant $M(X)$ and the number of classes $|\mathcal{X}/\!\sim\!|$. The resulting objective, dubbed CISSL, corresponds to SimCLR using only a projection head g on one branch. See Fig. 5. This asymmetry is necessary to learn population optimal encoders for linear $\mathcal{F}$.

CISSL bypasses the need for the maximal invariant by noticing it only appears in the ISSL log loss in the mapping to class weights $w(M(X))$. CISSL then aims to learn a function g that maps equivalent (augmented) examples $\tilde{X}$ to $w(M(\tilde{X}))$. Such mapping exists as M is invariant, e.g., $g := w \circ M$.



(a) SimCLR (b) CISSL

Figure 5: CISSL corresponds to SimCLR with a single (asymmetric) projection head to ensure linear predictability from $\phi(x)$ in downstream tasks.

When augmentations satisfy the Markov Chain $\tilde{X} - M(X) - X$, we show that we can replace the class weights $w(M(X))$ in the ISSL loss by $g(\tilde{X})$. Intuitively, this is because predicting $\tilde{X}$ contains only information about $M(X)$ due to the data processing inequality.

The only remaining challenge is computing the denominator of Eq. (6) without summing over the unknown classes $\mathcal{X}/\!\sim$. To do so we use ranking conditional noise contrastive estimation (NCE; [30–32]). NCE replaces classification of equivalence classes by classifications of positives in a batch $\tilde{\mathbf{X}} := \{\tilde{X}^+, \tilde{X}_1^-, \ldots, \tilde{X}_k^-\}$ where the positive $\tilde{X}^+$ is sampled from the conditional $A(\tilde{X} \mid X)$, while the k negatives $\tilde{X}_i^-$ come from the marginal $A(\tilde{X}) = \mathbb{E}_{p_X}[A(\tilde{X}|X)]$. Using Monte Carlo (MC) estimates with an unlabeled dataset $\mathcal{D} \overset{\text{i.i.d.}}{\sim} p_X$ we the get the empirical CISSL objective

$$\hat{\mathcal{L}}_C(\phi; \mathcal{D}) := \inf_{g \in \mathcal{S}^{\mathcal{X}}} \sum_{x \in \mathcal{D}} \mathbb{E}_{p(\tilde{\mathbf{X}}|x,\mathcal{D})} \left[-\log \frac{\exp g(\tilde{X}^+)^\top \phi(x)}{\sum_{\tilde{X}' \in \tilde{\mathbf{X}}} \exp g(\tilde{X}')^\top \phi(x)} \right]. \tag{7}$$

Using Prop. 3 and the consistency of ranking NCE for any $k \geq 1$ [32] we have that in ideal settings ($|\mathcal{D}| \to \infty$ and unconstrained g) we show that encoders trained with CISSL $\phi^* \in \arg\min_{\phi \in \Phi_1} \hat{\mathcal{L}}_C(\phi; \mathcal{D})$ are sample optimal for linear $\mathcal{F}$ and the $\sim$-invariant tasks when $\tilde{X} - M(X) - X$. While this holds for any $k \geq 1$, more negatives k will improve statistical efficiency [32].

Typically $\tilde{X}, X$ take value in the same space (e.g. images). If so we can tie parameters by encoding $\tilde{X}$ by ϕ before applying g, i.e., we can replace g by $g' \circ \phi$ where $g' : \mathbb{R}^d \to \mathcal{S}$ is called a projection head. The logits inside of Eq. (7) are then $g'(\phi(\tilde{X}))^\top \phi(x)$ as shown in Fig. 5b. This is strikingly similar to SimCLR's objective shown in Fig. 5a, which uses $g'(\phi(\tilde{X})))^\top g'(\phi(x))$. By projecting the current representation we show that *SimCLR generally learns encoders that are not even population optimality for linear $\mathcal{F}$*. In contrast, CISSL yields sample optimal encoders in idealized settings. Intuitively, this is because CISSL trains the representations in the same way as they will be used in downstream tasks. In downstream task p_t, the representations $\phi(X)$ will be dotted with the tasks' weight W_t to compute logits ${W_t}^\top \phi(x)$. In ISSL, representations $\phi(X)$, rather than their projections, should thus be dotted with the ISSL weights $g'(\phi(\tilde{X})) = w(M(X))$ to compute the logits $g(\phi(\tilde{X}))^\top \phi(x)$. CISSL thus derives from first principles the asymmetric use of projections.

4.2 Distillation ISSL (DISSL)

In practice, CISSL requires contrasting a large number of negatives k. To avoid NCE we can directly approximate $M(X)$, instead of the weight $w(M(X))$, so that the denominator of Eq. (6) can be computed exactly. The resulting method, dubbed DISSL, is a simplified and theoretically-grounded version of non-contrastive losses (e.g. SwAV, DINO, SimSiam). The top branch in Fig. 6 (the teacher) approximates $M(X)$ with $\hat{M}$, while the bottom branch (the student) tries to predict $\hat{M}$.

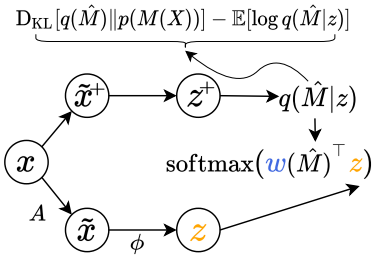


Figure 6: DISSL objective.

DISSL approximates the output of the maximal invariant $M(x)$ using a categorical distribution $q(\hat{M} \mid x)$ in C-dimensions. By replacing and rearranging terms in the ISSL log loss, we get

$$\inf_{w \in W_1} \mathbb{E}_{p_X q(\hat{M} \mid X)} \Big[-\log s_{\phi,w}(\hat{M} \mid X) \Big], \quad s_{\phi,w}(m \mid x) = \frac{\exp\big(w(m)^\top \phi(x)\big)}{\sum_{m=1}^{C} \exp(w(m)^\top \phi(x))}, \tag{8}$$

where differences with Eq. (6) are in red. To highlight similarities with previous methods [2, 24, 33] we refer to Eq. (8) as *distilling* the *teacher* $q(\hat{M} \mid X)$ into the student $s_{\phi,w}(\hat{M} \mid X)$. Note that if $C = |\mathcal{X}/{\sim}|$ and the teacher maps to the output of some maximal invariant M, i.e., $q(\hat{M} = M(X)|X) = 1$, then Eq. (8) is exactly the ISSL log loss. By Prop. 3 we thus have distillation ISSL gives sample optimal encoders when $C = |\mathcal{X}/{\sim}|$ and the teacher satisfies the following requirements from Eq. (4).

Deterministic the teacher is a deterministic distribution $\max_{m \in \{1,\dots,C\}} q(m \mid X) = 1$;
Invariant the teacher maps positives together $x \sim x^+ \implies q(\hat{M} \mid x) = q(\hat{M} \mid x)$;
Maximal the teacher maps negatives separately $x \not\sim x^- \implies q(\hat{M} \mid x) \neq q(\hat{M} \mid x^-)$.

There are many ways of enforcing these desired requirements for the teacher. In the following we use information-theoretic quantities (entropies and KL divergences) as those are interpretable can be computed in closed-form for categorical distributions such as our teacher and student.

Determinism and Invariance. Both properties hold if and only if for any equivalent example $x \sim x^+$ the cross-entropy of the theacher's outputs is minimized $\mathbb{E}_{q(\hat{M} \mid x)}[-\log q(\hat{M} \mid x^+)] = 0$. Indeed, zero cross-entropy simultaneously minimizes the conditional entropy $\mathrm{H}[\hat{M} \mid x] = 0$ (equivalent to determinism) and the KL divergence $\mathrm{D_{KL}}[q(\hat{M} \mid x) \| q(\hat{M} \mid x^+)] = 0$ (equivalent to invariance).

Maximality. A natural way of enforcing maximality is to train the teacher to predict differently any negative $x \not\sim x^-$. This requires access to batches of negatives, which we wanted to avoid with DISSL. Instead, assume that we know (or have a prior on) the true distribution of equivalence classes $p(M(X))$. Then a deterministic and invariant teacher is maximal if and only if the KL divergence between its marginal $q(\hat{M}) = \mathbb{E}_{p_X}[q(\hat{M}|X)]$ and the true one is minimized $\mathrm{D_{KL}}[q(\hat{M}) \| p(M(X))] = 0$. We can thus avoid contrasting negatives using (a prior on) the distribution of equivalence classes.

Putting all in Eq. (8), using Lagrangian multipliers λ, β, MC estimates, we get DISSL $\hat{\mathcal{L}}_D(\phi; \mathcal{D}) :=$

$$\inf_{q,w} \lambda \underbrace{\mathrm{D_{KL}}\Big[q(\hat{M}) \Big\| p(M(X))\Big]}_{\text{Maximality}} - \sum_{x \in \mathcal{D}} \mathbb{E}_{A(\tilde{X}, \tilde{X}^+ | x) q(\hat{M} \mid \tilde{X}^+)} \Big[\underbrace{\beta \log q(\hat{M} \mid \tilde{X})}_{\text{Det. and Inv.}} + \underbrace{\log s_{\phi,w}(\hat{M} \mid x)}_{\text{Distillation}} \Big]. \tag{9}$$

In ideal settings ($|\mathcal{D}| \to \infty$, unconstrained q, known $p(M(X))$) and for $\lambda, \beta \to \infty$ we show that encoders trained with DISSL are sample optimal for linear $\mathcal{F}$ and tasks invariant to the equivalences defined by the connected components of the augmentation graph [8]. If the marginal is unknown, we follow the max entropy principle [34] and use a uniform prior $p(M(X)) = \mathrm{Unif}(1, C)$.

As with CISSL, when $\tilde{X}, X$ are in the same space, we tie parameters by encoding $\tilde{X}$ before the teacher, i.e., $q(\hat{M} \mid X) = \mathrm{softmax}(g'(\phi(X)))$ for a head $g' : \mathbb{R}^d \to \mathbb{R}^C$. Algorithm 1 shows the result when the prior is uniform, and the $q(\hat{M})$ is computed from batches.

Algorithm 1 Batched DISSL

Require: $g, W, \phi, X, \beta, \lambda, A$
$\tilde{X}, \tilde{X}^+ \leftarrow \mathrm{sample}(A(\tilde{X} \mid X), 2)$
$q(\hat{M}|\tilde{X}) = \mathrm{softmax}(g(\phi(\tilde{X}))$
$q(\hat{M}|\tilde{X}^+) = \mathrm{softmax}(g(\phi(\tilde{X}^+))$
$s(\hat{M}|\tilde{X}^+) = \mathrm{softmax}(W^T \phi(\tilde{X}^+))$
$q(M) = \mathrm{batch_avg}(q(M|\tilde{X}))$
$\max = \sum_m q(m) \log q(m)$
$\mathrm{ce} = -\sum_m q(m|\tilde{X}) \log q(m|\tilde{X}^+)$
$\mathrm{dstl} = -\sum_m q(m|\tilde{X}) \log s(m|\tilde{X}^+)$
return $\lambda * \max + \mathrm{ce} + \beta * \mathrm{dstl}$

DISSL is one of the many distillation method that can be derived from our teacher's requirements. These requirements generally lead to a framework for deriving, comparing, and analysing distillation objectives. In Appx. E we a provide taxonomy of more than 10 previous SSL methods from this perspective — none of which recover sample optimal encoders. Typically, previous methods favor (i) determinism by "sharpening" the teacher $q(\hat{M} \mid X)$ with a temperature parameter; (ii) invariance by making the teacher's and student's similar for equivalent inputs $q(\hat{M} \mid x^+) \approx s_{\phi,w}(\hat{M} \mid x)$; (iii) maximality through optimization tricks (e.g. stop gradients or momentum teacher) to avoid a constant teacher, i.e., referred to as "collapsing" [25, 26, 35, 36]. Note that avoiding constant collapse $q(\hat{M} \mid x') = q(\hat{M} \mid x)$

for any $x, x' \in \mathcal{X}$ is insufficient, e.g., mapping half the inputs to a constant is also undesirable. Maximality formalizes the desired requirement: non-equivalent examples should not be mapped together.

5 ISSL for non-linear predictors $\mathcal{F}^+$

Linear probes $\mathcal{F}$ are standard for evaluating representations [28, 37, 38]. However, if the goal is to maximize performance it is natural to consider non-linear predictors $\mathcal{F}^+$. The question then becomes how should we learn optimal representations for any $\mathcal{F}^+$? In Appx. B.5 we extend our framework to certain non-linear probes, such as neural networks, that separate into a hidden component h and linear layer, i.e., $f(\cdot) = W^T h(\cdot)$. We informally summarize the theory and implications below.

Theory. There are two main differences between our characterization of optimal encoders for linear and non-linear probes. First, encoders should ensure predictability of $M(X)$ for the desired $\mathcal{F}^+$. Second, the dimensionality requirement decreases with the complexity of $\mathcal{F}^+$, e.g., for universal $\mathcal{F}^+$ it is $d \geq 1$. More generally, the necessary dimension and the sufficient dimension for optimality do not coincide and respectively depend on the predictors' VC dimension [13] and memory capacity [39].

Implications. Both theoretical differences lead to direct implications for non-linear ISSL. First, we can decrease the dimensionality when performing ISSL for more complex non-linear probes. Second, we should use a projection head that ensures the predictability of the maximal invariant with using the desired probes $\mathcal{F}^+$. Similarly to the linear case, we should match how the representation is used in the ISSL log loss and downstream tasks. We should thus apply the non-linear predictor h in a similar asymmetric way. For CISSL we should thus compute the right hand side of Fig. 5 as $g(\phi(\tilde{X}))h(\phi(x))$. For DISSL, we should change the line 4 of Algorithm 1 to $s(\hat{M}|\tilde{X}^+) = \text{softmax}(W^T h(\phi(\tilde{X}^+)))$.

6 Experiments

In our experiment, we focus on evaluating our frameworks' prescriptive implications, which can be summarized as: (i) use our objectives to ensure sample optimal encoders in ideal settings (Sec. 4); (ii) increase dimensionality d to ensure predictability of all invariant tasks (Theorem 1); (iii) use coarser label-preserving augmentations to improve sample efficiency log loss to ensure optimality (Prop. 2); and (iv) train to the smallest ISSL log loss to ensure optimality (Prop. 3). For experimental details see Appx. G. For additional experimental results see Appx. H.

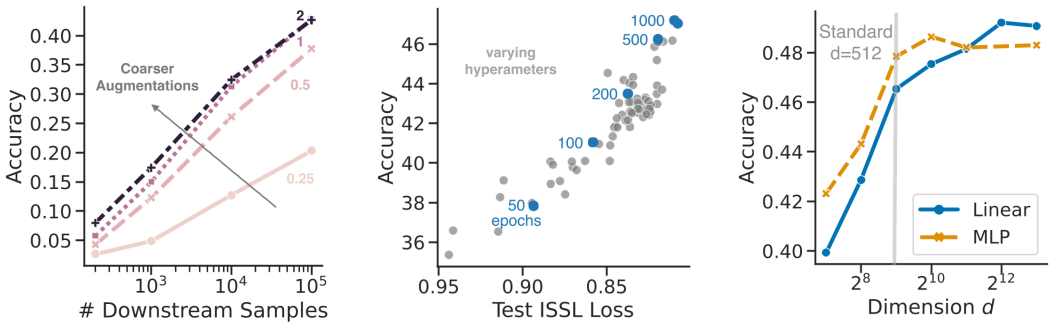
For the first experiments, we use TinyImageNet [40], 300 pretraining epochs, and ResNet18s. Note that for ResNet18s the dimensionality of the representation (`res5avg`) is $d = 512$. For contrastive baselines we use SimCLR, and for distilling baselines, we use the best over DINO, SwAV, SimSiam.

Can we improve ISSL for linear predictors? Table 1 shows that each of our prescriptions significantly improves distillation and contrastive ISSL. "Ours" vs "Baselines" shows that our simple asymmetric heads give a 1% gain in contrastive learning, while our new distillation objective achieves 3% gains compared to the best distillation baseline. "dim. ↑" shows that increasing the dimensionality of the representation from 512 to 2048 further improves linear probing by $1-2\%$. Finally, "coarse aug." shows that coarser augmentations (we double the strength of standard TinyImageNet augmentations [35] and added Gaussian blurring), we further improve performance by $3-4\%$. All together our framework's insights improves linear probing accuracy by an impressive 5% for contrastive learning and 8% for distillation.

Table 1: Our prescriptive implications improves ISSL for linear $\mathcal{F}$ on TinyImageNet. Models are ResNet18s trained for 300 epochs. 3 seeds.

	CONTR.	DISTIL.
BASE. (SIMCLR | DINO)	44.6 $\pm$0.1	41.7 $\pm$0.0
OURS (CISSL | DISSL)	45.8 $\pm$0.3	44.9 $\pm$0.0
+ DIM. ↑	47.5 $\pm$0.1	45.7 $\pm$0.1
+ COARSE AUG.	50.1 $\pm$0.1	49.7 $\pm$0.2

Effect of coarser augmentations? Prop. 2 suggests that coarser label-preserving augmentations improve downstream sample efficiency. We test that by introducing a hyperparameter α that controls the strength of standard augmentations. Specifically, α increases the probability of grayscaling (`p`=0.1α), each color jittering parameter (`brightness`=0.4α, `contrast`=0.1α, `saturation`=0.2α, `hue`=0.1α), and the cropping interval (`scale`=$[0.1/\alpha, 1]$). Fig. 7a shows four different DISSL encoders $\alpha \in \{0.25, 0.5, 1, 2\}$ evaluated at varying downstream samples sizes. As suggested by the theory, stronger label-preserving augmentations improve sample efficiency.



(a) Augmentations (b) ISSL log loss (c) Dimensionality

Figure 7: As suggested by our theory: (a) coarsening augmentations improves sample efficiency; (b) decreasing ISSL log loss improves representations, which can be achieved by longer training; and (c) increasing the representation's dimensionality improves performance for linear $\mathcal{F}$ but less so for MLPs $\mathcal{F}^+$. Y-axis is downstream TinyImageNet performance. Each point is obtained by sweeping: (a) augmentation strengths for DISSL and downstream sample sizes; (b) CISSL's optimization hyperparameters including epochs; and (c) the dimensionality of CISSL's representations.

Does ISSL log loss correlate with performance? Previous work [41] suggested that contrastive losses may not predict downstream performance. On the contrary, Prop. 3 shows that minimal ISSL log loss implies optimal encoders if d is large. This suggests that, for fixed augmentations, ISSL log loss is highly related to performance. To test this relation, we trained 80 CISSL models with various hyperparameters (e.g. optimizer, projection head, learning rate) while fixing augmentations and negatives k. Fig. 7b shows that the test ISSL loss and downstream accuracy are highly correlated (gray points). In Appx. H.3 we show similar correlation with the training ISSL loss.

Effect of longer training? We just showed that ISSL log loss correlates with downstream performance. A particularly efficient way of decreasing ISSL log loss is by longer training (blue points in Fig. 7b only vary in the number ISSL epochs). Prop. 3 thus provides an explanation of the well known performance gains of longer ISSL training [1, 42]. Namely, longer training decreases ISSL log loss, which results in representations that are closer to ETFs and thus optimality.

Effect of dimensionality for linear $\mathcal{F}$? Theorem 1 shows that linear ISSL requires a dimensionality of $d = |\mathcal{X}/\!\sim| - 1$ to ensure predictability of all invariant tasks. In practice, the number of equivalence classes is most likely very large. Although such dimensionality is impractical, it does suggest increasing the dimensionality beyond the $d = 512$ of ResNet18s. To test this, we train CISSL with a larger dimensionality by increasing the output channels before the average pooling layer. To keep the number of parameters comparable, we use bottlenecks or low-rank linear layers before and after the representations layer as detailed in Appx. G. Fig. 7c shows that increasing the dimensionality has a significant effect on downstream performance for linear $\mathcal{F}$ (blue). In Appx. H.3 we show that, contrary to popular beliefs, the gains when going from ResNet18s to ResNet50s are due to an increase in dimensionality (512 to 2048) rather than the number of parameters (11M to 23M).

Previous results concern linear $\mathcal{F}$ which are standard for ISSL evaluation. In practice, one would likely use more complex probes $\mathcal{F}^+$ to improve results. In Sec. 5 we discussed how to perform ISSL for $\mathcal{F}^+$. In the following, we evaluate ISSL for MLPs $\mathcal{F}^+$ with two hidden layers of 2048 units.

Can we perform ISSL for MLPs $\mathcal{F}^+$? Table 2 shows that MLP probes ("$\mathcal{F}_{\text{EVAL}}$: MLP") outperforms linear ones ("$\mathcal{F}_{\text{EVAL}}$: lin.") even in few-shot regimes (10 shot). Furthermore, when predicting with MLP probes it is desirable to train the ISSL encoders for MLPs ("$\mathcal{F}_{\text{ISSL}}$: MLP") instead of linear probes (row 4 vs 3). As discussed in Sec. 5, the only difference between DISSL for MLPs and for linear probes is that we project the representation using an MLP of the same architecture than downstream probes.

Table 2: MLP probes outperform linear ones on TinyImageNet when using MLP ISSL.

	$\mathcal{F}_{\text{ISSL}}$	$\mathcal{F}_{\text{EVAL}}$	ALL	10 SHOT
DINO	-	MLP	43.0 ±0.3	19.6 ±0.2
DISSL	LIN.	LIN.	44.9 ±0.0	21.7 ±0.0
	LIN.	MLP	45.6 ±0.1	21.9 ±0.0
	MLP	MLP	46.5 ±0.2	22.3 ±0.1

Effect dimensionality for MLPs $\mathcal{F}^+$? In Sec. 5 we saw that the optimal dimensionality d decreases for more complex predictors $\mathcal{F}^+$. Fig. 7c indeed shows that larger dimensions result in fewer gains when the training and evaluation probe are MLPs $\mathcal{F}^+$ (orange) compared to linear ones $\mathcal{F}$ (blue).

Given encouraging TinyImageNet results, we investigated whether our objectives can be used as replacement of standard objectives on ImageNet [43]. We specifically replaced SimCLR by our CISSL and SwAV our DISSL in the `VISSL`'s codebase [44] without any tuning. Due to computational constrains we mostly consider 100 epochs of pretraining and do not use augmentations to train the probes.

Table 3: Our models outperform baselines on ImageNet. ResNet50 encoders are trained using 100 epochs, 3072 batch size, standard augmentations. Our models increase dimensionality to 8192.

SIMCLR	SWAV	BARLOW T.	**CISSL**	**DISSL**
65.3	64.3	66.1	65.9	**67.3**

How do our objective perform on ImageNet? Table 3 shows that CISSL's small modifications of SimCLR lead to slight gains (0.6%). More importantly, our novel DISSL objective significantly outperforms all the considered baselines including SwAV by a 3% when using standard augmentations.

Given encouraging results on standard augmentations, we investigated the performance of our model compared to near a SOTA model: SwAV trained with their special multi-crop augmentations.

How does DISSL compare to SOTA models at scale? Table 4 shows that DISSL significantly outperforms SwAV on 100 epochs and performs at least as well for 400 epochs. Combined with Table 3 this suggests that DISSL is a robust method that works well in different settings. This is especially encouraging given the lack of hyperparameter tuning and the simplicity of DISSL.For example, SwAV runs a Sinkhorn-Knopp algorithm at each step, stores a queue of previous examples, uses stop-gradients, and freezes certain parameters during initial training steps. In contrast, DISSL is a simple objective that works out-of-the-box.

Table 4: Our DISSL outperforms SwAV using 2×160 + 4×96 multi-crops on ImageNet. 3072 batch size, ResNet50.

	100 EPOCHS	400 EPOCHS
SWAV	68.5	72.9
DISSL	**69.5**	**73.3**

7 Discussion and Outlook

We have presented a simple conceptual framework to understand, compare and improve ISSL methods. Standard theoretical advances in SSL typically analyse specific SSL algorithms [3, 5, 6, 8, 45–47] and / or focus on upper-bounding downstream performance [4]. These theoretical works have the advantage of explaining why specific ISSL alogithms work and providing performance guarantees. Those frameworks are nevertheless less well suited for improving and comparing different ISSL objectives as they do not have a general notion of better ISSL algorithms. Inspired by recent work on optimal representations for supervised learning [48] and domain adaptation [49], we provide a complementary view by characterizing ideal properties of ISSL representations. This characterization results in a simple unifying framework for improving and deriving ISSL algorithms. The downside is that by distancing itself from specific algorithms, our framework does not directly provide any concrete guarantees. For a detailed discussion of related work see Appx. E.

There are many limitations that should be addressed for a more realistic prescriptive framework for ISSL. Details in Appx. F. First, our theory is binary, e.g., augmentations are either label preserving or not, and encoders are optimal or not. Although our framework likely captures the right intuition, nuance would be more realistic .Second, we consider unconstrained encoders ϕ and infinite unlabeled data. Such assumptions follow recent empirical trends in increasing data and models [50, 51], but it means that our framework currently cannot improve the computational or data efficiency of the ISSL phase. Finally, we consider all invariant tasks $\mathcal{T}$ even though only a subset of those are meaningful. Optimal representations for this subset will be different, and defining this subset likely requires considering inductive biases of encoders [41]. Despite those limitations, we showed that our insights from our framework can lead to substantial improvements in ISSL performance.

References

[1] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, ser. Proceedings of Machine Learning Research, vol. 119. PMLR, 2020, pp. 1597–1607. [Online]. Available: http://proceedings.mlr.press/v119/chen20j.html

[2] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised Learning of Visual Features by Contrasting Cluster Assignments," *arXiv:2006.09882 [cs]*, Oct. 2020, arXiv: 2006.09882. [Online]. Available: http://arxiv.org/abs/2006.09882

[3] N. Saunshi, O. Plevrakis, S. Arora, M. Khodak, and H. Khandeparkar, "A Theoretical Analysis of Contrastive Unsupervised Representation Learning," in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 2019, pp. 5628–5637. [Online]. Available: http://proceedings.mlr.press/v97/saunshi19a.html

[4] Y. Bansal, G. Kaplun, and B. Barak, "For self-supervised learning, Rationality implies generalization, provably," *arXiv:2010.08508 [cs, stat]*, Oct. 2020, arXiv: 2010.08508. [Online]. Available: http://arxiv.org/abs/2010.08508

[5] J. D. Lee, Q. Lei, N. Saunshi, and J. Zhuo, "Predicting What You Already Know Helps: Provable Self-Supervised Learning," *arXiv:2008.01064 [cs, stat]*, Aug. 2020, arXiv: 2008.01064. [Online]. Available: http://arxiv.org/abs/2008.01064

[6] C. Tosh, A. Krishnamurthy, and D. Hsu, "Contrastive learning, multi-view redundancy, and linear models," in *Algorithmic Learning Theory, 16-19 March 2021, Virtual Conference, Worldwide*, ser. Proceedings of Machine Learning Research, V. Feldman, K. Ligett, and S. Sabato, Eds., vol. 132. PMLR, 2021, pp. 1179–1206. [Online]. Available: http://proceedings.mlr.press/v132/tosh21a.html

[7] Z. Wen and Y. Li, "Toward Understanding the Feature Learning Process of Self-supervised Contrastive Learning," *arXiv:2105.15134 [cs, stat]*, May 2021, arXiv: 2105.15134. [Online]. Available: http://arxiv.org/abs/2105.15134

[8] J. Z. HaoChen, C. Wei, A. Gaidon, and T. Ma, "Provable Guarantees for Self-Supervised Deep Learning with Spectral Contrastive Loss," *arXiv:2106.04156 [cs, stat]*, Jun. 2021, arXiv: 2106.04156. [Online]. Available: http://arxiv.org/abs/2106.04156

[9] T. Leen, "From Data Distributions to Regularization in Invariant Learning," in *Advances in Neural Information Processing Systems*, vol. 7. MIT Press, 1994. [Online]. Available: https://proceedings.neurips.cc/paper/1994/hash/7fa732b517cbed14a48843d74526c11a-Abstract.html

[10] S. Chen, E. Dobriban, and J. H. Lee, "A Group-Theoretic Framework for Data Augmentation," *arXiv:1907.10905 [cs, math, stat]*, Feb. 2020, arXiv: 1907.10905. [Online]. Available: http://arxiv.org/abs/1907.10905

[11] C. Lyle, M. v. d. Wilk, M. Kwiatkowska, Y. Gal, and B. Bloem-Reddy, "On the Benefits of Invariance in Neural Networks," *CoRR*, vol. abs/2005.00178, 2020, _eprint: 2005.00178. [Online]. Available: https://arxiv.org/abs/2005.00178

[12] Y. Dubois, B. Bloem-Reddy, K. Ullrich, and C. J. Maddison, "Lossy Compression for Lossless Prediction," *arXiv:2106.10800 [cs, math, stat]*, Jun. 2021, arXiv: 2106.10800. [Online]. Available: http://arxiv.org/abs/2106.10800

[13] V. N. Vapnik and A. Y. Chervonenkis, "On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities," *Theory of Probability & Its Applications*, vol. 16, no. 2, pp. 264–280, 1971, _eprint: https://doi.org/10.1137/1116025. [Online]. Available: https://doi.org/10.1137/1116025

[14] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 121–167, 1998, publisher: Springer.

[15] B. O'Neill, "The Classical Occupancy Distribution: Computation and Approximation," *The American Statistician*, vol. 75, no. 4, pp. 364–375, Oct. 2021, publisher: Taylor & Francis. [Online]. Available: https://amstat.tandfonline.com/doi/full/10.1080/00031305.2019.1699445

[16] V. Papyan, X. Y. Han, and D. L. Donoho, "Prevalence of Neural Collapse during the terminal phase of deep learning training," *Proceedings of the National Academy of Sciences*, vol. 117, no. 40, pp. 24 652–24 663, Oct. 2020, arXiv: 2008.08186. [Online]. Available: http://arxiv.org/abs/2008.08186

[17] Z. Zhu, T. Ding, J. Zhou, X. Li, C. You, J. Sulam, and Q. Qu, "A Geometric Analysis of Neural Collapse with Unconstrained Features," *arXiv:2105.02375 [cs, math, stat]*, May 2021, arXiv: 2105.02375. [Online]. Available: http://arxiv.org/abs/2105.02375

[18] T. Ergen and M. Pilanci, "Revealing the Structure of Deep Neural Networks via Convex Duality," in *Proceedings of the 38th International Conference on Machine Learning*. PMLR, Jul. 2021, pp. 3004–3014, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v139/ergen21b.html

[19] J. Lu and S. Steinerberger, "Neural Collapse with Cross-Entropy Loss," *arXiv:2012.08465 [cs, math]*, Jan. 2021, arXiv: 2012.08465. [Online]. Available: http://arxiv.org/abs/2012.08465

[20] F. Graf, C. Hofer, M. Niethammer, and R. Kwitt, "Dissecting Supervised Contrastive Learning," in *Proceedings of the 38th International Conference on Machine Learning*. PMLR, Jul. 2021, pp. 3821–3830, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v139/graf21a.html

[21] J. Zarka, F. Guth, and S. Mallat, "Separation and Concentration in Deep Networks," arXiv, Tech. Rep. arXiv:2012.10424, Mar. 2021, arXiv:2012.10424 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2012.10424

[22] W. Ji, Y. Lu, Y. Zhang, Z. Deng, and W. J. Su, "An Unconstrained Layer-Peeled Perspective on Neural Collapse," *arXiv:2110.02796 [cs, stat]*, Oct. 2021, arXiv: 2110.02796. [Online]. Available: http://arxiv.org/abs/2110.02796

[23] E. Weinan and S. Wojtowytsch, "On the emergence of simplex symmetry in the final and penultimate layers of neural network classifiers," *arXiv:2012.05420 [cs, stat]*, Jun. 2021, arXiv: 2012.05420. [Online]. Available: http://arxiv.org/abs/2012.05420

[24] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging Properties in Self-Supervised Vision Transformers," *arXiv:2104.14294 [cs]*, May 2021, arXiv: 2104.14294. [Online]. Available: http://arxiv.org/abs/2104.14294

[25] X. Chen and K. He, "Exploring Simple Siamese Representation Learning," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Nashville, TN, USA: IEEE, Jun. 2021, pp. 15 745–15 753. [Online]. Available: https://ieeexplore.ieee.org/document/9578004/

[26] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum Contrast for Unsupervised Visual Representation Learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA, USA: IEEE, Jun. 2020, pp. 9726–9735. [Online]. Available: https://ieeexplore.ieee.org/document/9157636/

[27] A. Mnih and K. Kavukcuoglu, "Learning word embeddings efficiently with noise-contrastive estimation," in *Advances in Neural Information Processing Systems*, vol. 26. Curran Associates, Inc., 2013. [Online]. Available: https://proceedings.neurips.cc/paper/2013/hash/db2b4182156b2f1f817860ac9f409ad7-Abstract.html

[28] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation Learning with Contrastive Predictive Coding," *arXiv:1807.03748 [cs, stat]*, Jan. 2019, arXiv: 1807.03748. [Online]. Available: http://arxiv.org/abs/1807.03748

[29] I. Misra and L. v. d. Maaten, "Self-Supervised Learning of Pretext-Invariant Representations," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 6706–6716. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/html/Misra_Self-Supervised_Learning_of_Pretext-Invariant_Representations_CVPR_2020_paper.html

[30] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, Mar. 2010, pp. 297–304, iSSN: 1938-7228. [Online]. Available: http://proceedings.mlr.press/v9/gutmann10a.html

[31] R. Jozefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu, "Exploring the Limits of Language Modeling," *arXiv:1602.02410 [cs]*, Feb. 2016, arXiv: 1602.02410. [Online]. Available: http://arxiv.org/abs/1602.02410

[32] Z. Ma and M. Collins, "Noise Contrastive Estimation and Negative Sampling for Conditional Models: Consistency and Statistical Efficiency," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 3698–3707. [Online]. Available: https://www.aclweb.org/anthology/D18-1405

[33] Z. Fang, J. Wang, L. Wang, L. Zhang, Y. Yang, and Z. Liu, "SEED: Self-supervised Distillation For Visual Representation," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=AHm3dbp7D1D

[34] E. T. Jaynes, "Information Theory and Statistical Mechanics," *Physical Review*, vol. 106, no. 4, pp. 620–630, May 1957, publisher: American Physical Society. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.106.620

[35] A. Ermolov, A. Siarohin, E. Sangineto, and N. Sebe, "Whitening for Self-Supervised Representation Learning," *arXiv:2007.06346 [cs, stat]*, May 2021, arXiv: 2007.06346. [Online]. Available: http://arxiv.org/abs/2007.06346

[36] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko, "Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M.-F. Balcan, and H.-T. Lin, Eds., 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/f3ada80d5c4ee70142b17b8192b2958e-Abstract.html

[37] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *arXiv:1206.5538 [cs]*, Apr. 2014, arXiv: 1206.5538. [Online]. Available: http://arxiv.org/abs/1206.5538

[38] R. Zhang, P. Isola, and A. A. Efros, "Colorful Image Colorization," in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*, ser. Lecture Notes in Computer Science, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., vol. 9907. Springer, 2016, pp. 649–666. [Online]. Available: https://doi.org/10.1007/978-3-319-46487-9_40

[39] T. M. Cover, "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition," *IEEE Transactions on Electronic Computers*, vol. EC-14, no. 3, pp. 326–334, Jun. 1965. [Online]. Available: http://ieeexplore.ieee.org/document/4038449/

[40] Y. Le and X. Yang, "Tiny ImageNet Visual Recognition Challenge," p. 6, 2015.

[41] N. Saunshi, J. Ash, S. Goel, D. Misra, C. Zhang, S. Arora, S. Kakade, and A. Krishnamurthy, "Understanding Contrastive Learning Requires Incorporating Inductive Biases," *arXiv:2202.14037 [cs]*, Feb. 2022, arXiv: 2202.14037. [Online]. Available: http://arxiv.org/abs/2202.14037

[42] Z. Wu, Y. Xiong, S. Yu, and D. Lin, "Unsupervised Feature Learning via Non-Parametric Instance-level Discrimination," *arXiv:1805.01978 [cs]*, May 2018, arXiv: 1805.01978. [Online]. Available: http://arxiv.org/abs/1805.01978

[43] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2009, pp. 248–255, iSSN: 1063-6919.

[44] P. Goyal, Q. Duval, J. Reizenstein, M. Leavitt, M. Xu, B. Lefaudeux, M. Singh, V. Reis, M. Caron, P. Bojanowski, A. Joulin, and I. Misra, "VISSL," 2021. [Online]. Available: https://github.com/facebookresearch/vissl

[45] C. Tosh, A. Krishnamurthy, and D. Hsu, "Contrastive estimation reveals topic posterior information to linear models," *arXiv:2003.02234 [cs, stat]*, Mar. 2020, arXiv: 2003.02234. [Online]. Available: http://arxiv.org/abs/2003.02234

[46] K. Nozawa and I. Sato, "Understanding Negative Samples in Instance Discriminative Self-supervised Representation Learning," *arXiv:2102.06866 [cs, stat]*, Oct. 2021, arXiv: 2102.06866. [Online]. Available: http://arxiv.org/abs/2102.06866

[47] Y. Tian, X. Chen, and S. Ganguli, "Understanding self-supervised Learning Dynamics without Contrastive Pairs," *arXiv:2102.06810 [cs]*, Oct. 2021, arXiv: 2102.06810. [Online]. Available: http://arxiv.org/abs/2102.06810

[48] Y. Dubois, D. Kiela, D. J. Schwab, and R. Vedantam, "Learning Optimal Representations with the Decodable Information Bottleneck," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M.-F. Balcan, and H.-T. Lin, Eds., 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/d8ea5f53c1b1eb087ac2e356253395d8-Abstract.html

[49] Y. Ruan, Y. Dubois, and C. J. Maddison, "Optimal Representations for Covariate Shift," Sep. 2021. [Online]. Available: https://openreview.net/forum?id=Rf58LPCwJj0

[50] P. Goyal, M. Caron, B. Lefaudeux, M. Xu, P. Wang, V. Pai, M. Singh, V. Liptchinsky, I. Misra, A. Joulin, and P. Bojanowski, "Self-supervised Pretraining of Visual Features in the Wild," *arXiv:2103.01988 [cs]*, Mar. 2021, arXiv: 2103.01988. [Online]. Available: http://arxiv.org/abs/2103.01988

[51] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, E. Brynjolfsson, S. Buch, D. Card, R. Castellon, N. Chatterji, A. Chen, K. Creel, J. Q. Davis, D. Demszky, C. Donahue, M. Doumbouya, E. Durmus, S. Ermon, J. Etchemendy, K. Ethayarajh, L. Fei-Fei, C. Finn, T. Gale, L. Gillespie, K. Goel, N. Goodman, S. Grossman, N. Guha, T. Hashimoto, P. Henderson, J. Hewitt, D. E. Ho, J. Hong, K. Hsu, J. Huang, T. Icard, S. Jain, D. Jurafsky, P. Kalluri, S. Karamcheti, G. Keeling, F. Khani, O. Khattab, P. W. Koh, M. Krass, R. Krishna, R. Kuditipudi, A. Kumar, F. Ladhak, M. Lee, T. Lee, J. Leskovec, I. Levent, X. L. Li, X. Li, T. Ma, A. Malik, C. D. Manning, S. Mirchandani, E. Mitchell, Z. Munyikwa, S. Nair, A. Narayan, D. Narayanan, B. Newman, A. Nie, J. C. Niebles, H. Nilforoshan, J. Nyarko, G. Ogut, L. Orr, I. Papadimitriou, J. S. Park, C. Piech, E. Portelance, C. Potts, A. Raghunathan, R. Reich, H. Ren, F. Rong, Y. Roohani, C. Ruiz, J. Ryan, C. Ré, D. Sadigh, S. Sagawa, K. Santhanam, A. Shih, K. Srinivasan, A. Tamkin, R. Taori, A. W. Thomas, F. Tramèr, R. E. Wang, W. Wang, B. Wu, J. Wu, Y. Wu, S. M. Xie, M. Yasunaga, J. You, M. Zaharia, M. Zhang, T. Zhang, X. Zhang, Y. Zhang, L. Zheng, K. Zhou, and P. Liang, "On the Opportunities and Risks of Foundation Models," *arXiv:2108.07258 [cs]*, Aug. 2021, arXiv: 2108.07258. [Online]. Available: http://arxiv.org/abs/2108.07258

Checklist

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] in Sec. 7.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] in Appx. F.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] in the paper the main paper we mention the important ones succintly, in Appx. A.2 we discuss each in detail.
   (b) Did you include complete proofs of all theoretical results? [Yes] in Appx. B.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] but we will open source our code.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] in Appx. G.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] for the TinyImageNet table. We did not for ImageNet due to computational constraints.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] at a high level in Appx. G.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] .
   (b) Did you mention the license of the assets? [No] .
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] .
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] .
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] .
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A] .
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] .
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A] .