The Role of Entropy and Reconstruction in Multi-View Self-Supervised Learning

Borja Rodríguez-Gálvez [1 2] **Arno Blaas** [2] **Pau Rodríguez** [2] **Adam Goliński** [2]
Xavier Suau [2] **Jason Ramapuram** [2] **Dan Busbridge** [2] **Luca Zappella** [2]

Abstract

The mechanisms behind the success of multi-view self-supervised learning (MVSSL) are not yet fully understood. Contrastive MVSSL methods have been studied through the lens of InfoNCE, a lower bound of the Mutual Information (MI). However, the relation between other MVSSL methods and MI remains unclear. We consider a different lower bound on the MI consisting of an entropy and a reconstruction term (ER), and analyze the main MVSSL families through its lens. Through this ER bound, we show that clustering-based methods such as DeepCluster and SwAV maximize the MI. We also re-interpret the mechanisms of distillation-based approaches such as BYOL and DINO, showing that they explicitly maximize the reconstruction term and implicitly encourage a stable entropy, and we confirm this empirically. We show that replacing the objectives of common MVSSL methods with this ER bound achieves competitive performance, while making them stable when training with smaller batch sizes or smaller exponential moving average (EMA) coefficients.

Github repo: apple/ml-entropy-reconstruction.

1. Introduction

Representation learning tackles the problem of learning lower dimensional representations of data which capture the data's semantic information. To achieve this, many representation learning methods aim to maximize the *mutual information* (MI) between the input data and the learned representations (Linsker, 1988; Belghazi et al., 2018; Hjelm et al., 2019), while inducing biases in the model that steer the learned information to be semantically meaningful (Alemi et al., 2017; van den Oord et al., 2018; Velickovic et al., 2019). As such, MI has played a crucial role in understanding the performance of many representation learning methods (Tishby et al., 1999; Rodríguez Gálvez et al., 2020; Goldfeld & Polyanskiy, 2020).

Recently, multi-view self-supervised learning (MVSSL), where the loss enforces the model to produce similar representations for different views of the same data, has proven to be a successful approach for representation learning (Bachman et al., 2019; Tian et al., 2020a; He et al., 2020; Caron et al., 2021). The success of MVSSL has motivated the research of several families of MVSSL approaches, such as *contrastive* (Chen et al., 2020a), *clustering-* (Caron et al., 2018), and *distillation*-based methods (Grill et al., 2020). However, the effort to understand all of them under a common umbrella lags behind the development of new methods. In this work, we aim to further our understanding of MVSSL methods by identifying any mechanisms contributing to maximizing MI, and to what extent they do so.

The connection of the contrastive MVSSL methods to MI maximization is well established through the `InfoNCE` bound (van den Oord et al., 2018; Poole et al., 2019), which, in the MVSSL context, lower bounds the MI between the learned representations of different views. Tian et al. (2020b) and Tsai et al. (2020) argue that maximizing this MI is attractive as a representation learning target since, when the views are selected carefully, it extracts task-relevant and discards task-irrelevant information.

The interest in the MI perspective on representation learning, and MVSSL in particular, has been undermined following the work of Tschannen et al. (2020), whose key result is showing that maximizing MI alone is not sufficient for learning good representations. Yet, it is empirically evident that methods based on MI lower bound maximization are competitive with state-of-the-art, and Tschannen et al. (2020) note that "the performance of these methods depends strongly on the bias that is encoded not only in the encoders, but also on the actual form of the used MI estimators". In our opinion, their results strongly motivates further study

[1]Division of Information Science and Engineering (ISE), KTH Royal Institute of Technology, Stockholm, Sweden [2]Apple. Correspondence to: Borja Rodríguez-Gálvez <borjarg@kth.se>, Luca Zappella <lzappella@apple.com>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

of the mechanisms by which, and to what extent, the MI maximization takes place in representation learning.

In this work, we center our analysis of MVSSL methods around the MI between the learned representations of different views Z_1, Z_2. The MI lower bound we focus on consists of an *entropy* and a *reconstruction* term (Gallager, 1968):

$$I(Z_1; Z_2) \geq \underbrace{H(Z_2)}_{\text{Entropy}} + \underbrace{\mathbb{E}[\log q_{Z_2|Z_1}(Z_2)]}_{\text{Reconstruction term}} := I_{\texttt{ER}}(Z_1; Z_2),$$

where the $\log q_{Z_2|Z_1}$ corresponds to a choice of a similarity function between representations used in MVSSL, e.g., a cosine similarity. We refer to this bound as `ER`, referring to the *Entropy* and *Reconstruction* terms. Focusing on this bound, rather than the `InfoNCE`, allows us to analyze a wide range of MVSSL methods through the lens of MI.

The work closest in spirit to ours is (Wang & Isola, 2020), which analyzes the contrastive MVSSL methods through the lens of *alignment* and *uniformity*, two metrics which they derive through formulating desiderata for the learned representations. While their motivation was, in the light of the results of Tschannen et al. (2020), to offer an alternative interpretation of `InfoNCE`, different than as a lower bound on MI, we show the metrics they define coincide with a specific instantiation of the `ER` MI bound we consider. We generalize their results through the use of the `ER` bound which allows us to also analyze the clustering- and distillation-based MVSSL methods.

Our contributions in this work are the following:

- We review how, and to what extent, the major families of MVSSL methods (contrastive, clustering, and distillation-based) maximize MI via the use of the `ER` bound on MI. Specifically, we show that the clustering-based methods `SwAV` (Caron et al., 2020) and `DeepCluster` (Caron et al., 2018) maximize the `ER` bound and therefore the MI between representations of different views.
- We empirically show that simply substituting the loss function and instead optimizing `ER` in `SimCLR` (Chen et al., 2020a), `BYOL` (Grill et al., 2020), and `DINO` (Caron et al., 2021) results in similar performance while improving resiliency with respect to training with smaller batch sizes or exponential moving average (EMA) coefficients. This is especially important for distillation methods such as `BYOL` or `DINO`, as they become resilient to batch size changes without any need for hyperparameter changes or gradient accumulation.
- Finally, we show that it is not necessary for distillation methods like `BYOL` to maximize entropy to achieve competitive results, although mechanisms such as the softmax centering in `DINO` and other related architectural constraints prevent the entropy collapse.

2. Background

Here, we introduce some notation, the multi-view self-supervised learning setting, and the relevant bounds on MI.

Notation X represents a random variable (RV) with probability mass function or density p_X, and x is its realization. Expectations are denoted as $\mathbb{E}[f(X)] = \mathbb{E}_{x\sim p_X}[f(x)]$. The conditional density for a fixed realization x is denoted as $p_{Y|X=x}$. The density $q_{Y|X}$ is not the real conditional density of X given Y, but an an auxiliary one that serves, e.g., as an optimization target. The mutual information is denoted as $I(X;Y)$, the Shannon and the differential entropy are both denoted as $H(X)$, and the Kullback-Leibler divergence between densities p and q is denoted as $D_{\text{KL}}(p\|q)$. A sub-sequence of elements from a to b in a sequence x is denoted as $x^{(a:b)}$, and all elements except $x^{(i)}$ as $x^{(\neq i)}$.

Multi-view self-supervised learning In MVSSL, for each data sample X, we generate two (or more) views V_b. These views are commonly obtained by using augmentations (Bachman et al., 2019; Tian et al., 2020b; Chen et al., 2020a; Caron et al., 2020; Zbontar et al., 2021), by leveraging multiple modalities (Radford et al., 2021), or natural views of data (Tian et al., 2020a), e.g., multiple camera views of the same scene. Views V_b are chosen or engineered such that most of the semantic information remains unchanged with respect to the original data sample X and shared between the views (Tian et al., 2020b). Each view is then passed through a neural network encoder $f_\theta(\cdot)$ to produce representations R_b which are in turn projected via $\pi_\theta(\cdot)$, usually a small MLP, into a lower dimensional space to yield Z_b, where θ are the learnable parameters. Typically, the intermediate representations R_b are used for downstream tasks and transfer learning, as that yields better performance than using Z_b (Chen et al., 2020a; Bordes et al., 2023). The parameters θ are learned by optimizing an objective which encourages the projections Z_b to be predictive of the other branches' outputs $Z_{(\neq b)}$. This is commonly achieved by optimizing a *similarity* score, such as the L2 distance. Most of the methods use two views and we will focus on this setting, without loss of generality.[1] Since the processing of each view takes place separately and for some methods differs between views, we refer to those separate computation paths as *branches*. See Figure 1 for an illustrative diagram.

The three families of MVSSL considered in this work are *contrastive*, *clustering*- and *distillation*-based methods. Contrastive methods work by comparing the projections of the two views of the same datum (or *positive pairs*), with a set of projections of different data (or *negative pairs*). The different methods in this category are usually distinguished by

[1] When more than two views are considered, the objective decomposes into a sum of independent sub-objectives based on view pairs, see e.g., Tian et al. (2020a) or Caron et al. (2018).

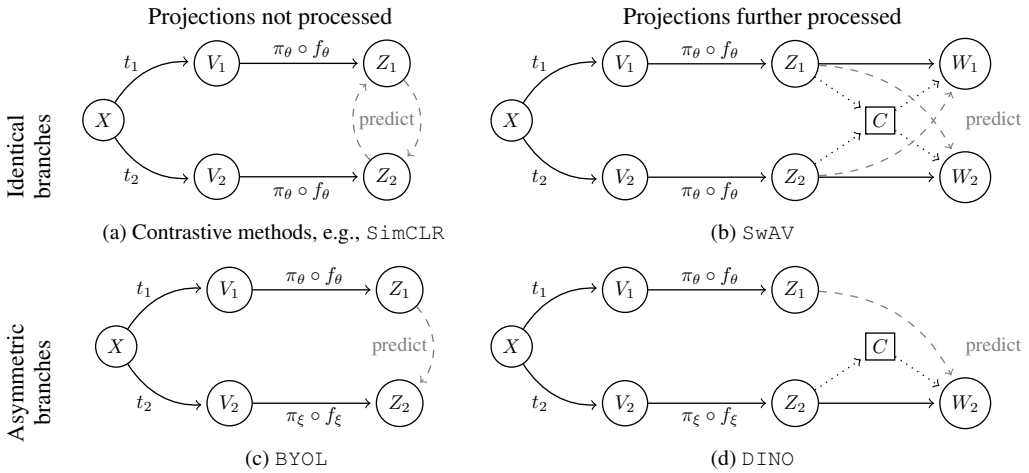

Figure 1. The MVSSL prototypes. An image X is transformed with augmentations t to generate two views V and projections Z. Dashed and dotted lines indicate loss functions and optional relationships between variables respectively. **Top:** Identical branches: Parameters θ are identical across branches and the loss is symmetric. **Bottom:** Asymmetric branches: Parameters θ, ξ across branches are different and the loss is asymmetric. **Left:** The projections Z are not further processed. **Right:** The projections Z are processed into auxiliary discrete variables W, potentially using another variable C. Parameters θ, ξ are optimized such that Z are predictive of the other branch's W.

how they define the negative pairs. Most of these methods are derived either from the metric learning literature (Sohn, 2016) or the InfoNCE objective (van den Oord et al., 2018), which is a lower bound on the mutual information between the projections $I(Z_1; Z_2)$. We discuss these methods in detail in Section **3.1**. Clustering methods cluster the projections from one branch and use the resulting discrete cluster assignments as targets for the other branch by optimizing a cross-entropy loss (Caron et al., 2018; 2020; Asano et al., 2019). Distillation-based methods design the two branches asymmetrically, using one branch's projections as targets for the other (Grill et al., 2020; Chen & He, 2021; Caron et al., 2021). The two branches, referred to as *teacher* and *student*, differ. Common differences include gradients being computed only by the student (stop-grad), teacher's parameters being set via an EMA of the student's, and an additional predictor network for the student.

Mutual information lower bounds Estimating MI is fundamentally difficult (McAllester & Stratos, 2020) and for gradient-based representation learning, it is common to rely on the gradients of a lower bound on MI without estimating MI directly (Poole et al., 2019). In this work, the core quantity of interest is the MI between MVSSL projections $I(Z_1; Z_2)$. Two MI lower bounds that can be used to optimize this quantity are InfoNCE and ER.

InfoNCE (van den Oord et al., 2018; Poole et al., 2019) is a lower bound on MI. In MVSSL, the MI is between the projections Z_1, Z_2. It is estimated from a sequence of i.i.d. samples of pairs $(Z_1^{(1:k)}, Z_2^{(1:k)})$ from the joint density p_{Z_1,Z_2}:

$$I_{\text{NCE}}(Z_1; Z_2) := \frac{1}{k} \sum_{i=1}^{k} \mathbb{E}\left[\log \frac{e^{f(Z_1^{(i)}, Z_2^{(i)})}}{\frac{1}{k}\sum_{j=1}^{k} e^{f(Z_1^{(i)}, Z_2^{(j)})}}\right], \quad (1)$$

where $f(\cdot,\cdot)$ is a function scoring similarity between vectors, e.g., cosine similarity. Many contrastive methods use it as a loss function in the original or slightly different forms depending on negative sample choice. We discuss the MI maximization in this class of methods in detail in Section 3.1.

The ER bound is a long standing result in information theory (Gallager, 1968). It can be derived by considering a tractable *reconstruction density* $q_{Z_2|Z_1}$ that for MVSSL corresponds to a choice of a similarity function:

$$I(Z_1;Z_2) = \mathbb{E}\left[\log \frac{q_{Z_2|Z_1}(Z_2)}{p_{Z_2}(Z_2)}\right] + \mathbb{E}[\overbrace{D_{\text{KL}}(p_{Z_2|Z_1} \| q_{Z_2|Z_1})}^{\geq 0}]$$
$$\geq H(Z_2) + \mathbb{E}[\log q_{Z_2|Z_1}(Z_2)] := I_{\text{ER}}(Z_1;Z_2). \quad (2)$$

In the MVSSL setting, $q_{Z_2|Z_1}$ is a design choice and we are interested in optimizing the parameters of $\pi_\theta \circ f_\theta$ such that the resulting density p_{Z_1,Z_2} maximizes $I_{\text{ER}}(Z_1; Z_2)$. The density p_{Z_1,Z_2} implicitly results from sampling inputs X, possibly transforming them via stochastic transformations t, and then deterministically transforming them through the encoder $\pi_\theta \circ f_\theta$ to form Z. The term $\mathbb{E}[D_{\text{KL}}(p_{Z_2|Z_1} \| q_{Z_2|Z_1})]$ determines the magnitude of the gap of the I_{ER} bound.

The term *reconstruction* originates from information theory. It is often concerned with reconstructing a signal from a compressed code and is equal to $-H(Z_2|\hat{Z}_2)$, where $\hat{Z}_2$ is a RV such that $Z_2 - Z_1 - \hat{Z}_2$ is a Markov chain. We find it also more appropriate to reason about MVSSL such as the right column of Figure 1, where Z_1 and W_2 belong to different spaces, and hence the term *similarity* seems less accurate.

Intuitively, the *entropy* and *reconstruction* terms in the ER bound (2) play different roles in MVSSL. The entropy term determines how much information from one projection *can*

be learnt, while the reconstruction term determines how much of this available information *is learnt*. For instance, let the projections lay on the sphere: the more spread out (higher entropy) the projections of different data are, the more revealing (higher mutual information) it is if projections from different views of the same datum are close (lower reconstruction error). Conversely, if one branch projects all data to the same point (lowest entropy, also known as *collapse*), the projections from the other branch can't reveal any information about them.

MVSSL for small batch sizes Small batch sizes degrade the performance of MVSSL methods, especially contrastive ones (Chen et al., 2020a; Grill et al., 2020; Caron et al., 2021). Potentially, this is due to the fact that most methods maximize the entropy either explicitly or implicitly, as shown in this paper, and the entropy estimation is limited to $\log k$ bits for a batch size of k (McAllester & Stratos, 2020). Some works (HaoChen et al., 2021; Chen et al., 2021; Yuan et al., 2022) addressed this issue and modified existing methods to perform well under the small batch size regime.

3. MVSSL and MI optimization

In this section, we reflect on the relationship between different MVSSL methods and the MI. First, we review the known connection between contrastive methods and MI maximization through the `InfoNCE` bound, as well as the lack thereof. Also, we show that none of the existing methods formally maximize the `ER` bound, while all of them are a good proxy for it. Next, we show for the first time that the clustering-based methods `DeepCluster` (Caron et al., 2018) and `SwAV` (Caron et al., 2020) also optimize the MI through the `ER` bound. Finally, we interpret the techniques used in distillation-based methods such as EMA (Grill et al., 2020) and softmax centering (Caron et al., 2021) as mechanisms to prevent the entropy collapse. The results of this section are summarized in Table **1**.

3.1. Contrastive methods

Contrastive learning (CL) methods are the family of MVSSL methods that have been most closely connected to MI maximization in the existing literature and, as such, a good starting point for our analysis. Here, we first give a review of the connections established through the `InfoNCE` bound and otherwise, before exhibiting the relationship to the `ER` bound. Summarizing, generally CL algorithms cannot be formally shown to maximize the `InfoNCE` nor the `ER` bound due to the violation of the i.i.d. assumption. This is not the case for `CMC` those methods derived from it, nor for methods using a memory bank like Instance Discrimination (Wu et al., 2018, `IR`) or `MoCo` (He et al., 2020; Chen et al., 2020b) under particular circumstances, which do maximize the `InfoNCE`. Nevertheless, as also concluded by Wang & Isola (2020), CL is a good proxy for entropy maximization, and therefore, for MI maximization.

Given the projection of a view of datum i, e.g., $Z_2^{(i)}$, contrastive learning algorithms aim to maximize its similarity with the projection of another view of the same datum, e.g., $Z_1^{(i)}$ (*positive sample*), while making it as different as possible from the projections of a set of *negative samples* $\mathcal{S}_{\text{neg}}(Z_2^{(i)})$. This is achieved by minimizing a cross entropy loss based on a similarity score. Given a batch of k samples a generic contrastive loss for the second branch is

$$\mathcal{L}_{\text{contr,2}} := -\frac{1}{k}\sum_{i=1}^{k}\log\frac{e^{f(Z_2^{(i)},Z_1^{(i)})}}{\sum_{Z'\in\mathcal{S}_{\text{neg}}(Z_2^{(i)})} e^{f(Z_2^{(i)},Z')}} \tag{3}$$

and the full loss is $\mathcal{L}_{\text{contr}} := (\mathcal{L}_{\text{contr,1}} + \mathcal{L}_{\text{contr,2}})/2$, where usually $f = \text{sim}(\cdot)/\tau$, $\text{sim}(\cdot)$ is the cosine similarity, and τ is a temperature parameter. Then, different CL methods are distinguished by how the set of negative samples for a particular sample $Z_2^{(i)}$ is constructed. Note that the negatives might include samples from the other branches.

In `CMC` (Tian et al., 2020a), the negative samples set is composed of all the other projections from the opposite branch, i.e., $\mathcal{S}_{\text{neg}}(Z_2^{(i)}) = Z_1^{(1:k)}$. Comparing (1) and (3) with these negative samples we see that `CMC` maximizes the `InfoNCE` bound and $\mathbb{E}[-\mathcal{L}_{\text{CMC}}] \leq I(Z_1; Z_2) - \log k$.

The maximization of the `InfoNCE` bound can be similarly shown for methods that can be derived from the basic `CMC`, like the full `CMC`, where more than two views are considered; (Bachman et al., 2019), which adapts `DIM` (Hjelm et al., 2019) to the basic `CMC`; and (Tian et al., 2020b), which attempts to learn the augmentations that best suit the information maximization.

For `SimCLR` (Chen et al., 2020a), on the other hand, the negative samples are all the projections other than $Z_2^{(i)}$, i.e., $\mathcal{S}_{\text{neg}}(Z_2^{(i)}) = Z_2^{(\neq i)} \cup Z_1^{(1:k)}$. Given such a definition of the negative set, even if all negative samples were identically distributed, the negative samples are not independent as $Z_1^{(j)}$ and $Z_2^{(j)}$ are derived from the same datum j, for all js. As shown in (Tschannen et al., 2020), `InfoNCE` is not maximized when violating the independence assumption. Hence, `SimCLR` does not maximize the `InfoNCE` bound. This also holds true for methods that are derived from `SimCLR` such as (Ramapuram et al., 2021).

Finally, methods like `IR` or `MoCo` use representations from a memory bank as negative samples, i.e., $\mathcal{S}_{\text{neg}}(Z_2^{(i)}) = Z_{\text{bank}}^{(1:m)}$. In these cases the negative samples can be dependent and are not identically distributed with respect to $Z_2^{(i)}$. However, Wu et al. (2020) showed that under certain mild conditions on the distribution of these samples the contrastive loss used in these methods is a lower bound on the

InfoNCE, and thus optimizing it also maximizes MI.

Relationship with the ER bound None of the contrastive methods above directly translates to an optimization of the ER bound, even if it may appear so. In the context of (3), if we consider a density s.t. $q_{Z_2|Z_1=z_1}(z_2) \propto \exp f(z_2, z_1)$, the expected value of the first term corresponds to the reconstruction error in (2), and when $f(\cdot,\cdot)$ is the cosine similarity with temperature τ, the density $q_{Z_2|Z_1=z_1}$ corresponds to a von Mises–Fisher density with mean direction z_1 and concentration parameter $1/\tau$. However, as shown above, in all methods analyzed, the negative samples are either not independent between themselves (as in SimCLR), or not identically distributed with respect to the positive sample (as in MoCo), or the set contains the positive pair itself (as in CMC). Therefore, the log-denominator in (3) is not an unbiased kernel density estimator (KDE, Joe (1989)) of the entropy and therefore its expectation is not necessarily the entropy $H(Z_2)$ from (2).

Nonetheless, all these methods force the projections to be maximally separated from the negative samples in a convex set (usually the hypersphere). Moreover, the highest entropy distribution on a convex set is precisely the uniform distribution on that volume. Hence, the contrastive loss, even with non-i.i.d. negative samples, is a good proxy for entropy maximization, and therefore, for MI maximization. Wang & Isola (2020) make a similar observation and conclude that maximizing the uniformity of the samples in the projections' space is required for good performance.

Caveats As seen above, most current analyses for CL methods require the i.i.d. assumption, which is not usually met due to the use of batch normalization. The breaking of the independence assumption is important as it can break the InfoNCE results (Tschannen et al., 2020; Wu et al., 2020). Nonetheless, it does not discredit that the result of the KDE is a good proxy to maximize the entropy.

3.2. Clustering-based methods

In this section, we show that both DeepCluster (Caron et al., 2018; Asano et al., 2019) and SwAV (Caron et al., 2020) maximize the ER lower bound on the MI between the projections of different views of the data $I_{\text{ER}}(Z_1; Z_2)$.

The key observation underlying the results in this section is that DeepCluster and SwAV generate a discrete surrogate of the projections, e.g., for the second branch $W_2 = \phi(Z_2)$, and that they maximize the ER bound on $I(Z_1; W_2) \leq I(Z_1; Z_2)$, where the inequality holds by the data processing inequality. For the rest of the section, let $\mathcal{Z} \subseteq \mathbb{R}^d$ and $\mathcal{W} = \{1, \dots, m\}$.

DeepCluster has an asymmetric setting with $\xi = \theta$ (Figure 1d). First, the cluster assignments $W_2^{(i)} = \phi(Z_2^{(i)})$ of all the n data points are obtained solving the problem

$$C^\star \in \operatorname*{arg\,inf}_{C \in \mathbb{R}^{d\times m}} \frac{1}{n}\sum_{i=1}^{n} \|Z_2^{(i)} - C p_2^{(i)}\|^2,$$

with $p_2^{(i)} \in \{0,1\}^m$ and $\|p_2^{(i)}\|_0 = 1$, where $C^\star$ represent the m centroids of the clusters in $\mathcal{Z}$ and $p_2^{(i)}$ is the p.m.f. of $W_2^{(i)}$ given $Z_2^{(i)}$.[2] Then, the parameters θ are optimized by minimizing the cross entropy

$$\mathcal{L}_{\text{DeepCluster}} := -\frac{1}{k}\sum_{i=1}^{k} \left(p_2^{(i)}\right)^\intercal \log\left(\mathsf{s} \circ g_\theta(Z_1^{(i)})\right),$$

where $g_\theta : \mathcal{Z} \to \mathbb{R}^m$ is a small predictor network, and s is the softmax operator. Note that Z also depends on θ via $Z = \pi_\theta \circ f_\theta(V)$, see Figure 1. With $q_{W_2|Z_1=z_1} = \mathsf{s} \circ g_\theta(z_1)$, *this optimization precisely amounts to maximizing the reconstruction term in the ER bound for* $I(Z_1; W_2)$. Furthermore, to prevent degenerate solutions, Caron et al. (2018) sample the images of each batch based on a uniform distribution over cluster assignments, i.e. for each batch $p_{W_2} \approx \frac{1}{k}\sum_{i=1}^{k} p_2^{(i)}$ is almost uniform. Through this, *the entropy* $H(W_2)$ *is approximately maximized*. Combined with the maximization of the reconstruction term via $\mathcal{L}_{\text{DeepCluster}}$, this implies DeepCluster *maximizes the ER MI bound*.

Now, let us turn to SwAV. SwAV has a symmetric setting (Figure 1b). We focus on branch $b = 2$, as the analysis is analogous for the other branch. Here, the cluster assignments $W_2^{(i)} = \phi(Z_2^{(i)})$ are obtained solving the following optimization problem

$$P_2 = \operatorname*{arg\,max}_{P \in \mathcal{P}} \left\{ \operatorname{Tr}\left(Z_2^{(1:k)} C^\intercal P^\intercal\right) + \epsilon H(P) \right\},$$

where $Z_2^{(1:k)} \in \mathbb{R}^{k\times d}$, $C \in \mathbb{R}^{m\times d}$ are the m centroids (or prototypes) in $\mathbb{R}^d$, $\mathcal{P} = \{P \in \mathbb{R}_+^{k\times m} : P^\intercal \mathbf{1}_k = \mathbf{1}_m/m \text{ and } P\mathbf{1}_m = \mathbf{1}_k/k\}$ is the transportation polytope, and $\mathbf{1}_k$ is the all ones vector in $\mathbb{R}^k$. Let $C^{(i)}$ and $P_2^{(i)}$ denote the i-th row of C and P_2, respectively. In SwAV, both the projections and the prototypes lay in the unit hypersphere, i.e., $Z^{(i)}, C^{(i)} \in \mathbb{S}^{d-1}$, and thus maximizing the dot product is equivalent to minimizing the squared ℓ_2 norm distance (Grill et al., 2020). Moreover, to aid the optimization calculations, an entropic regularization is included to approximately solve it using the Sinkhorn-Knopp algorithm (Sinkhorn, 1974; Cuturi, 2013), where $H(P_2) := -\sum_{i=1}^{k} \left(P_2^{(i)}\right)^\intercal \log P_2^{(i)}$.

The l-th element of $P_2^{(i)}$ can be understood as the probability of assigning $Z_2^{(i)}$ to the cluster $W_2^{(i)} = l$. The optimization aims to have $P_2 \in \mathcal{P}$ and therefore $P_2^\intercal \mathbf{1}_k \approx \mathbf{1}_m/m$, which

[2] Asano et al. (2019) obtain the clusters solving an optimal transport problem similar to SwAV.

by this interpretation would mean that $p_{W_2|Z_2} \approx \mathbf{1}_m/m$ is approximately uniform, thus maximizing the entropy $H(W_2|Z_2)$. As $H(W_2|Z_2) \leq H(W_2)$, this construction *maximizes the desired entropy $H(W_2)$ in the* ER *bound.*

For SwAV, similarly to DeepCluster, *the reconstruction term is maximized* by minimizing the loss function

$$\mathcal{L}_{\text{SwAV},2} \coloneqq -\frac{1}{k}\sum_{i=1}^{k}\left(p_2^{(i)}\right)^{\intercal}\log\left(\mathsf{s}\big(CZ_1^{(i)}\big)\right),$$

where $p_2^{(i)} = P_2^{(i)}/(\mathbf{1}_m^{\intercal}P_2^{(i)})$ and $q_{W_2|Z_1=z_1} = \mathsf{s}(Cz_1)$, hence maximizing the mutual information $I(Z_1;W_2)$. An analogous analysis for the branch $b=1$ reveals that minimizing $\mathcal{L}_{\text{SwAV},1}$ with the entropic regularisation assignment maximizes the mutual information $I(W_1;Z_2)$. In SwAV, the prototypes are treated as parameters of the network (i.e., $C \in \theta$) and are updated using stochastic gradient descent to minimize $\mathcal{L}_{\text{SwAV}}$. This implies *SwAV also maximizes ER.*

3.3. Distillation methods

Distillation methods naturally optimize the reconstruction term of the ER bound since the projection of one branch is optimized to predict the projection of the other branch. However, it is more challenging to understand if and how they might maximize the entropy term of ER, hence, we cannot yet claim they are maximizing the MI. There are some tools, such as EMA or centering, that distillation methods employ that could have an effect on the entropy. In fact, such tools are key to prevent the phenomenon known as collapse (Grill et al., 2020; Caron et al., 2021). Our analysis of their role below does not yield definitive, formal statements. However, it should still shed some light on this question.

First, let us detail how each method maximizes the reconstruction term of the ER bound. We start by analyzing the reconstruction term for the BYOL loss, which is the ℓ_2 normalised mean squared error

$$\mathcal{L}_{\text{BYOL}} \coloneqq \frac{1}{k}\sum_{i=1}^{k}\left\|\overline{g_\theta(Z_1^{(i)})} - \overline{Z_2^{(i)}}\right\|^2, \tag{4}$$

where $\overline{x} \coloneqq x/\|x\|$. Since $\|\overline{x}-\overline{y}\|^2 = 2(1-\text{sim}(x,y))$, optimizing (4) is equivalent to maximizing the reconstruction term in the ER bound with a von Mises–Fisher reconstruction density with mean direction $\overline{g_\theta(Z_1^{(i)})}$ and concentration parameter 1. For DINO, the loss is similar to the one used by the clustering-based methods, namely

$$\mathcal{L}_{\text{DINO}} \coloneqq -\frac{1}{k}\sum_{i=1}^{k}\mathsf{s}\big((Z_2^{(i)}-C)/\tau_2\big)^{\intercal}\log\left(\mathsf{s}\big(Z_1^{(i)}/\tau_1\big)\right), \tag{5}$$

where C is a centering variable, and τ_1, τ_2 are temperature hyperparameters. Letting $p_{W_2|Z_2=z_2} = \mathsf{s}\big((z_2-C)/\tau_2\big)$ and $q_{W_2|Z_1=z_1} = \mathsf{s}(z_1/\tau_1)$ shows that optimizing (5) is equivalent to maximizing the reconstruction term in the ER bound of $I(Z_1;W_2) \leq I(Z_1;Z_2)$.

Let us now analyze the potential effect of the stabilizing algorithms used by distillation methods on the entropy of the projections to understand if distillation methods also maximize the entropy term of the ER bound. We focus on the role of EMA and centering.

EMA introduces an asymmetry between the teacher and the student in distillation methods (Figure 1b and d). Specifically, the teacher's parameters ξ track the student's parameters θ during the optimization with the use of EMA: $\xi \leftarrow \lambda\xi + (1-\lambda)\theta$ for some $\lambda \in (0,1)$ close to 1. The hypothesis is two-fold: on the one hand, while ξ does depend on θ, the dependence is weak enough so that $H(Z_2)$ or $H(W_2)$ is not degrading to values yielding trivial bounds. This would happen in the extreme case of $\xi = \theta$, for which minimizing the respective losses will have an optimal solution $\theta^\star$ that would be highly concentrated or degenerate around one point, under which $H(Z_2) \to -\infty$ or $H(W_2) = 0$, which clearly would not maximize the MI. On the other hand, the dependence of ξ on θ, while weak, ensures that the projections Z_2 capture information about the data. If this was not the case, e.g., by fixing ξ to random values, the then random projections Z_2 would contain very little information about X. In this case, despite maximising $I(Z_1;Z_2)$ via minimising the respective losses and simultaneously ensuring constant entropy $H(Z_2)$ (due to the random projections), the information learned would still be little as by the data processing inequality $I(Z_1;Z_2) \leq I(X;Z_2)$. BYOL and DINO balance this trade-off between not maximizing MI due to minimal entropy and maximizing MI to a small achievable minimum with constant entropy with their choice of λ, but the resulting effect on entropy and MI maximization is hard to estimate.

Beyond EMA, DINO also promotes a high conditional entropy $H(W_2|Z_2)$ through the centering before the softmax operation. Like in SwAV, this avoids collapse as it controls the entropy $H(W_2)$ via $H(W_2|Z_2) \leq H(W_2)$. To be precise, the center C in (5) is updated with an EMA of the previous projections, that is, $C \leftarrow \mu C + \frac{1-\mu}{k}\sum_{i=1}^{k} Z_2^{(i)}$ for some $\mu \in (0,1)$. Then, the right balance between this EMA and the temperature parameters τ_1 and τ_2 adjusts how uniform the conditional density $p_{W_2|Z_2}$ is. This promotes a high conditional entropy $H(W_2|Z_2)$. However, having a completely uniform conditional density means that $p_{W_2|Z_2} = p_{W_2}$ and thus no information of Z_2 is in W_2. For this reason, Caron et al. (2021) need to also include a sharpening of the conditional density via the temperature τ_2. Therefore, the degree of maximization of $H(W_2)$ is hard to quantify as it depends on the chosen values of the parameters μ, τ_1, and τ_2.

Table 1. The relation between existing MVSSL methods and the maximization of MI via the `InfoNCE` *and* `ER` *lower bounds.* ✓: formally shown, (✓): approximately or empirically, ×: no formal or empirical evidence, *: previously known (Section **3.1**).

Model	InfoNCE	ER	Violation
CMC	✓*	(✓)	-
SimCLR	×	(✓)	negatives not i.i.d.
IR, MoCo	(✓)*	(✓)	negatives not i.i.d.
DeepCluster	×	✓	-
SwAV	×	✓	-
BYOL	×	(✓)	not max. entropy
DINO	×	(✓)	not max. entropy

To summarize, the use of both EMA and centering is crucial for distillation methods to work, and they do affect the entropy term of the ER bound. However, it is not yet possible to quantify these effects exactly, hence, one cannot make any statement that distillation methods maximize MI, despite clearly maximizing the reconstruction term of the ER bound.

4. Optimizing the ER bound in practice

In this section, we describe different ways to maximize the ER bound regardless of the MVSSL prototype (see Figure **1**). That is, we will describe how to estimate the entropy and the reconstruction term in (2) when the projections are not processed (Figure **1**a and c). The case when discrete surrogates are generated (Figure **1**b and d) is discussed in Appendix **A.2**. Then, the objective resulting from such an estimation is maximized. Later, in Section **5**, we use these approaches on top of the architectures of current contrastive and distillation-based methods and observe that their performance is on par (or slightly better) than their original formulation, and that they become more resilient to the choice of the batch size and EMA coefficient without the need for neither adjusted hyper-parameters nor accumulated gradients.

4.1. Maximizing MI between projections

We consider an estimation of the ER bound of the MI between the projections $I_{\texttt{ER}}(Z_1; Z_2)$. Let $f(z_2, z_1)$ be a function measuring the similarity between z_1 and z_2. Choosing the reconstruction density $q_{Z_2|Z_1=z_1}(z_2) \propto \exp f(z_2, z_1)$, an unbiased estimate of the reconstruction term is given by

$$\widehat{\text{Rec}}_{\text{cont}} := \frac{1}{k} \sum\nolimits_{i=1}^{k} f(Z_2^{(i)}, Z_1^{(i)}), \tag{6}$$

where the term associated with the normalizing constant of the density is discarded as it does not affect the optimization. To estimate the entropy term, one may consider different variants of KDEs. For example, both the KDE of Joe (1989)

$$\hat{H}(Z_2)_{\text{KDE,Joe}} := -\frac{1}{k} \sum_{i=1}^{k} \log \hat{p}_{Z_2}(Z_2^{(i)}) \tag{7}$$

or the plug-in estimator (Krishnamurthy & Wang, 2015)

$$\hat{H}(Z_2)_{\text{KDE,plug-in}} := -\sum_{i=1}^{k} \hat{p}_{Z_2}(Z_2^{(i)}) \log \hat{p}_{Z_2}(Z_2^{(i)}) \tag{8}$$

can be used (both give similar results in practice, see Appendix **D**). Here, $\hat{p}_{Z_2}(z)$ is Joe (1989)'s KDE of p_{Z_2}:

$$\hat{p}_{Z_2}(z) := \frac{1}{kh^d} \sum_{j=1}^{k} q\left(\frac{z - Z_2^{(j)}}{h}\right), \tag{9}$$

with kernel $q(\cdot)$ and bandwidth $h \in \mathbb{R}_+$. Both the reconstruction and the entropy estimators are (asymptotically) unbiased and converge in mean squared error (MSE) with an appropriate choice of the bandwidth (see Appendix **A**). The selection of an optimal kernel bandwidth can be seen as a limitation of ER. While minimizing the number of hyper-parameters would be desirable, the bandwidth plays a similar role to the temperature term typically tuned in other SSL methods, e.g. (Chen et al., 2020a). So much so, that we adopted as bandwidth the same temperature parameter specified by the SSL methods on top of which we incorporate ER.

Connection to CL When the chosen kernel q is such that $q(z_2 - z_1) = f(z_2, z_1)$, then maximizing the ER bound with estimators (6, 7) is *equivalent to contrastive learning* with the negative samples being $\mathcal{S}_{\text{neg}}(Z_2^{(i)}) = Z_2^{(\neq i)}$, up to constants independent of the optimization parameters.

Connection to Uniformity and Alignment The *alignment* and *uniformity* objective of Wang & Isola (2020) is a relaxation of the ER objective with estimators (6, 7). Let $f(z_2, z_1) = \|z_2 - z_1\|_2^{\alpha}$, then the estimator (6) recovers their alignment term. Consider also a kernel $q(z_2 - z_1) \propto \exp\big(-t\|z_2 - z_1\|_2^2\big)$, then Joe (1989)'s KDE (7) recovers their alignment term after applying Jensen's inequality.[3] Hence, our analysis can be considered a natural extension of their analysis to other MVSSL families.

Connection to Identifiability Under certain assumptions, MVSSL partitions the latent representations into a content component, invariant to augmentations, and a style component, which can change with augmentations (Von Kügelgen et al., 2021). The ER objective recovers their main theorem (Theorem 4.4) with a reconstruction density $q_{Z_2|Z_1=z_1}(z_2) \propto \exp\big(-\|z_2 - z_1\|_2^2\big)$. Moreover, CL methods implicitly invert the underlying generative model of the observed data, again under certain assumptions (Zimmermann et al., 2021). We show that the same is true for methods maximising the ER bound, revealing that the main reason for this inversion is not the contrastive nature of the methods, but that they maximize the mutual information (see Appendix **B**).

[3] The application of Jensen's inequality makes Wang & Isola (2020)'s objective a looser MI lower bound than the ER bound.

4.2. Dealing with an EMA

The maximization of the ER bound is compatible with an asymmetric structure (Figure 1c, d) where the teacher's parameters ξ are updated with an EMA of the student's parameters θ. The objective is equivalent to the maximization of the symmetric bound with an additional stop_gradient operator on the teacher's projections. The optimization from the reconstruction of the teacher from the student is unaffected. Then, since the entropy of the student's projections Z (or surrogates W) is maximized, it will also be maximized for the teacher, which is only updated through the EMA. This is confirmed empirically in Section 5.

5. Experiments

In this section, we show that replacing the objective of common MVSSL methods with the ER bound results in competitive performance while being more robust to the changes in batch size and EMA coefficient without changing any other hyperparameters. Further experiments are included in Appendices **E** and **G** and the code is available at https://github.com/apple/ml-entropy-reconstruction.

Experimental Setup For all experiments, we pre-train a resnet50 (He et al., 2016) on the ImageNet (Deng et al., 2009) training set. We train for 400 epochs and following Chen et al. (2020b) we use a batch size of 4096 with the LARS optimizer (You et al., 2017) with linear warmup, a single cycle cosine annealed learning rate schedule, and a base learning rate of 0.3 (Goyal et al., 2017) . We chose BYOL, DINO, and SimCLR as baseline methods, with CMC results presented in Appendix **E**. For each model except DINO, we substitute their objective function by the continuous estimate of the ER bound from Section **4**,[4] while keeping the original set of augmentations and their original projection heads. For DINO we estimate the entropy as the average of the discrete plug-in entropy among replicas. CMC shares augmentations and projection head with SimCLR.

Training with ER yields competitive accuracy We train a linear classifier on top of the ImageNet pre-trained features and report the test accuracy in Table **2**. For all models, we kept their original hyperparameters. For SimCLR, adding ER increases test accuracy ($+0.72$) while for BYOL and DINO it decreases slightly (-1.5 and -1.65, respectively).

ER further improves distillation method's stability with small batch size and small EMA coefficients The right column in Table **2** shows the performance degradation when training with batch size 512 and EMA coefficient of 0.8 instead of 0.99 (we observe similar results with a batch size 1024 or an EMA coefficient of 0.6). The original version of BYOL and DINO exhibit the largest degradation of all algorithms. This can also be observed in Figure **2**. Note that Grill et al. (2020) provided recipes to train BYOL with smaller batch sizes by retuning hyperparemeters or by gradient accumulation. They also observed that the batch size had a strong influence on the optimal EMA coefficient. Here, we limit our observation to what happens when nothing else is changed in the optimization. Interestingly, we observe that ER significantly improves the resilience towards the change in batch size for all methods tested, especially for BYOL where the degradation is reduced from -20.32 to -0.21. Regarding the EMA coefficient, we observe a degradation of -8.25 for DINO and -2.62 for BYOL which are reduced to -0.92 and -0.41 respectively with ER.

[4]We use the plug-in estimator instead of Joe (1989)'s, but we observe both to perform almost identically (Appendix **D**).

Table 2. Training with ER yields competitive performance while improving stability with small batch size and EMA coefficients. **Model**: set of augmentations, loss, and projection head. *Our implementation. **ER**: the original loss has been substituted by the ER bound (2). **MI**: known to maximize MI. (✓): no formal proof (Section **4.2**). $\mathbf{\Delta 512}$: accuracy drop with batch size 512. $\mathbf{\Delta EMA}_{0.8}$: accuracy drop with EMA coefficient of 0.8.

Model	MI	Acc (↑)	Δ512(↓)	$\Delta\text{EMA}_{0.8}$(↓)
DINO	?	75.59	6.76	8.25
DINO + ER	(✓)	73.39	2.35	0.92
BYOL	?	73.42	23.65	2.63
BYOL + ER	(✓)	71.94	2.35	0.41
SimCLR	×	70.23	2.17	-
SimCLR + ER	✓	70.86	1.01	-

In fact, we find that training with ER outperforms recent literature on small-batch SSL training (HaoChen et al., 2021; Chen et al., 2021; Yuan et al., 2022). For example, for SimCLR with batch size 512, we report an accuracy of 69.85 (Table **2**) while the most recent of these works reports an accuracy of 68.8 (Yuan et al., 2022).

BYOL does not maximize entropy Figure **2** shows the evolution of entropy and reconstruction during training (top and middle) and the ImageNet accuracy (bottom) (see Appendix **F** for clustering methods like DeepCluster and SwAV). We observe that methods trained with ER clearly maximize entropy while others such as BYOL with batch size 4096 display a slight decrease in entropy while still achieving high accuracy. This might provide an empirical answer to the question left in Section **3.3** and indicate that BYOL does not maximize entropy. The EMA was introduced to avoid representation collapse in the absence of negative samples. When properly tuned, the effect seems sufficient to maintain a high entropy and create discriminative representations. Nevertheless, one could argue that it does not take full advantage of the overall space (or we would observe higher entropy) and that the accuracy is very sensitive to its tunning (see Table **2** and Figure **2**). In addition to the EMA, DINO introduces a softmax centering

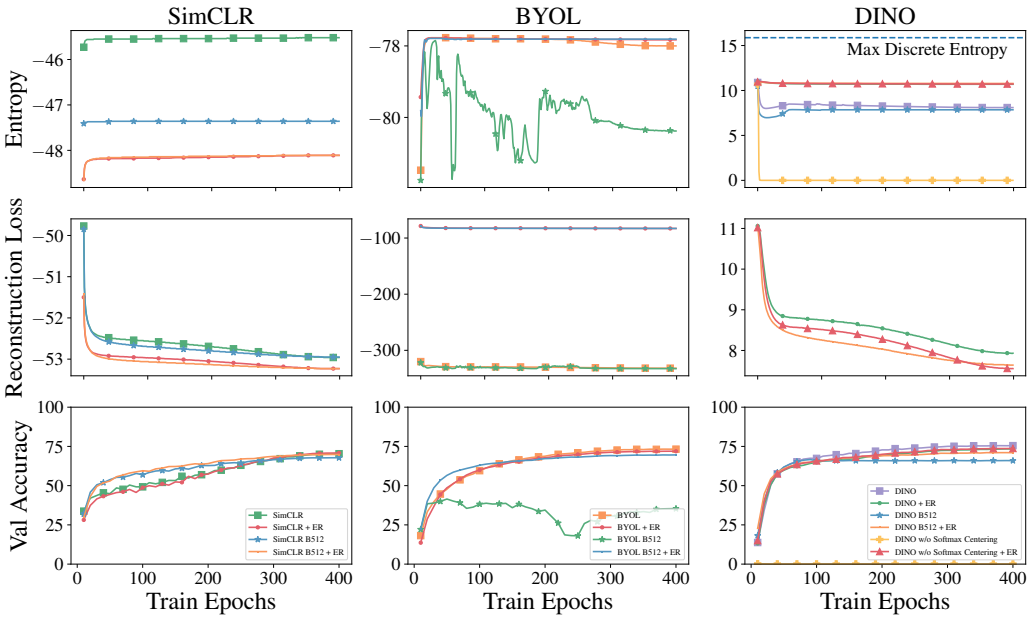

Figure 2. ER maximizes entropy during training (top) while it is unclear for distillation methods. ER allows training DINO w/o softmax centering. Top: Entropy dynamics while training SimCLR, BYOL, DINO w/ and w/o ER, and DINO w/ and w/o softmax centering for 400 epochs. Middle: Reconstruction loss dynamics. Bottom: top-1 accuracy on the ImageNet test set (linear probe trained online).

procedure to keep the output probabilities in a certain range. In Figure **2**, we observe that DINO's entropy and accuracy become extremely low when softmax centering is deactivated. Notably, *adding ER makes it possible to train DINO without softmax centering*, which confirms that softmax centering plays a role in keeping the entropy high (Section **3.3**).

ER is not sensitive to the entropy estimator All ER models except DINO used a KDE-based entropy estimator. To gain more insight into the effect of the estimator, we train a continuous KDE-based version of DINO + ER and compare it with the one reported in Table **2**, which uses an exact discrete estimator. We find no significant differences between their performances (see Appendix **E**).

6. Discussion

We showed to what extent different MVSSL methods maximize MI through the ER bound on the MI. First, we revisited previous knowledge about the maximization of MI in contrastive methods and reinterpreted it in the context of ER. Second, we showed that two clustering-based methods, DeepCluster and SwAV, maximize the ER bound. Third, we interpreted two distillation-based methods, BYOL and DINO, as maintaining a stable level of entropy while maximizing the reconstruction term of the ER bound.

We explained how ER can be optimized in most MVSLL frameworks, and we showed empirically that SimCLR, BYOL and DINO, when optimizing the ER bound result in a performance which is competitive with that of the respective original versions. We also showed that it is not necessary for distillation methods like BYOL to maximize entropy to achieve competitive results. This is an interesting observation in the context of (Wang & Isola, 2020) who conclude both alignment and uniformity are required for contrastive methods to work well, we showed that at least for distillation methods, maximizing uniformity is not necessary. Uniformity (or high entropy), however, seems to be correlated with resilience as all methods became more resilient to smaller batch size and/or EMA coefficient when maximizing ER, with a particularly pronounced effect for distillation methods. Understanding the exact mechanism for these behaviors remains an exciting subject of future work.

Finally, our theoretical analysis in Section **4.1** and Appendix **B** indicates that methods that explicitly maximize the ER bound should yield desirable identifiability properties. We believe that exploring this result in practice is an exciting avenue for future research.

Acknowldegments

The authors thank the reviewers for their valuable feedback, which resulted in new experiments and clarifications that strengthened the paper, as well as the colleagues at Apple for productive discussions that helped shape and fortify the paper, especially Effrosyni Simou, Michal Klein, Tatiana Likhomanenko, and R. Devon Hjelm.

Borja Rodríguez-Gálvez was funded, in part, by the Swedish research council under contract 2019-03606.

References

Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. Deep variational information bottleneck. In *International Conference on Learning Representations (ICLR)*, 2017.

Antos, A. and Kontoyiannis, I. Convergence properties of functional estimates for discrete distributions. *Random Structures & Algorithms*, 19(3-4):163–193, 2001.

Asano, Y., Rupprecht, C., and Vedaldi, A. Self-labelling via simultaneous clustering and representation learning. In *International Conference on Learning Representations (ICLR)*, 2019.

Bachman, P., Hjelm, R. D., and Buchwalter, W. Learning representations by maximizing mutual information across views. In *Neural Information Processing Systems (NeurIPS)*, 2019.

Belghazi, M. I., Baratin, A., Rajeshwar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, D. Mutual information neural estimation. In *International Conference on Machine Learning (ICML)*, 2018.

Bordes, F., Balestriero, R., Garrido, Q., Bardes, A., and Vincent, P. Guillotine regularization: Why removing layers is needed to improve generalization in self-supervised learning. *Transactions of Machine Learning Research (TMLR)*, 2023.

Caron, M., Bojanowski, P., Joulin, A., and Douze, M. Deep clustering for unsupervised learning of visual features. In *European Conference on Computer Vision (ECCV)*, 2018.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. In *Neural Information Processing Systems (NeurIPS)*, 2020.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. In *International Conference on Computer Vision (ICCV)*, 2021.

Chen, J., Gan, Z., Li, X., Guo, Q., Chen, L., Gao, S., Chung, T., Xu, Y., Zeng, B., Lu, W., et al. Simpler, faster, stronger: Breaking the log-k curse on contrastive learners with flatnce. *arXiv preprint arXiv:2107.01152*, 2021.

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)*, 2020a.

Chen, X. and He, K. Exploring simple siamese representation learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020b.

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. In *Neural Information Processing Systems (NeurIPS)*, 2013.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Gallager, R. G. *Information theory and reliable communication*, volume 588. Springer, 1968.

Girsanov, I. On a property of non-degenerate diffusion processes. *Theory of Probability & Its Applications*, 4(3): 329–333, 1959.

Goldfeld, Z. and Polyanskiy, Y. The information bottleneck problem and its applications in machine learning. *IEEE Journal on Selected Areas in Information Theory*, 1(1): 19–38, 2020.

Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., and He, K. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al. Bootstrap your own latent: A new approach to self-supervised learning. In *Neural Information Processing Systems (NeurIPS)*, 2020.

HaoChen, J. Z., Wei, C., Gaidon, A., and Ma, T. Provable guarantees for self-supervised deep learning with spectral contrastive loss. In *Neural Information Processing Systems (NeurIPS)*, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., and Bengio, Y. Learning deep representations by mutual information estimation and maximization. In *International Conference for Learning Representations (ICLR)*, 2019.

Joe, H. Estimation of entropy and other functionals of a multivariate density. *Annals of the Institute of Statistical Mathematics*, 41(4):683–697, 1989.

Krishnamurthy, A. and Wang, Y. Lectures on information processing and learning. *UC Berkeley lecture notes*, 2015.

Linsker, R. Self-organization in a perceptual network. *Computer*, 21(3):105–117, 1988.

McAllester, D. and Stratos, K. Formal limitations on the measurement of mutual information. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.

Poole, B., Ozair, S., van den Oord, A., Alemi, A., and Tucker, G. On variational bounds of mutual information. In *International Conference on Machine Learning (ICML)*, 2019.

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)*, 2021.

Ramapuram, J., Busbridge, D., Suau, X., and Webb, R. Stochastic contrastive learning. *arXiv preprint arXiv:2110.00552*, 2021.

Rodríguez Gálvez, B., Thobaben, R., and Skoglund, M. The convex information bottleneck lagrangian. *Entropy*, 22 (1):98, 2020.

Sinkhorn, R. Diagonal equivalence to matrices with prescribed row and column sums. *Proceedings of the American Mathematical Society*, 45(2):195–198, 1974.

Sohn, K. Improved deep metric learning with multi-class n-pair loss objective. In *Neural Information Processing Systems (NeurIPS)*, 2016.

Tian, Y., Krishnan, D., and Isola, P. Contrastive multiview coding. In *European Conference on Computer Vision (ECCV)*, 2020a.

Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. What makes for good views for contrastive learning? In *Neural Information Processing Systems (NeurIPS)*, 2020b.

Tishby, N., Pereira, F. C., and Bialek, W. The information bottleneck method. In *Proceedings of the Annual Allerton Conference on Communication, Control and Computing*, 1999.

Tsai, Y.-H. H., Wu, Y., Salakhutdinov, R., and Morency, L.-P. Self-supervised learning from a multi-view perspective. In *International Conference on Learning Representations (ICLR)*, 2020.

Tschannen, M., Djolonga, J., Rubenstein, P. K., Gelly, S., and Lucic, M. On mutual information maximization for representation learning. In *International Conference on Learning Representations (ICLR)*, 2020.

van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. In *Neural Information Processing Systems (NeurIPS)*, 2018.

Velickovic, P., Fedus, W., Hamilton, W. L., Liò, P., Bengio, Y., and Hjelm, R. D. Deep graph infomax. *International Conference on Learning Representations (ICLR)*, 2019.

Von Kügelgen, J., Sharma, Y., Gresele, L., Brendel, W., Schölkopf, B., Besserve, M., and Locatello, F. Self-supervised learning with data augmentations provably isolates content from style. In *Neural Information Processing Systems (NeurIPS)*, 2021.

Wang, T. and Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning (ICML)*, 2020.

Wu, M., Mosse, M., Zhuang, C., Yamins, D., and Goodman, N. Conditional negative sampling for contrastive learning of visual representations. In *International Conference on Learning Representations (ICLR)*, 2020.

Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

You, Y., Gitman, I., and Ginsburg, B. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

Yuan, Z., Wu, Y., Qiu, Z.-H., Du, X., Zhang, L., Zhou, D., and Yang, T. Provable stochastic optimization for global contrastive learning: Small batch does not harm performance. In *International Conference on Machine Learning (ICML)*, 2022.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning (ICML)*, 2021.

Zimmermann, R. S., Sharma, Y., Schneider, S., Bethge, M., and Brendel, W. Contrastive learning inverts the data generating process. In *International Conference on Machine Learning (ICML)*, 2021.

Appendices

A. Entropy and reconstruction estimators MSE convergence

In this section, we describe the MSE behaviour of the entropy and reconstruction estimators of Section 4.

A.1. Estimators on a "continuous space"

A.1.1. Entropy estimation and selection of the bandwidth parameter

The bias and variance of Joe (1989)'s KDE estimator $\hat{H}_{\text{KDE,Joe}}$ of the entropy $H(Z_2)$ are (Joe, 1989, Section 4, page 695)

$$\mathbb{B}[\hat{H}_{\text{KDE,Joe}}] \in \mathcal{O}(k^{-1}h^{4-d}) + \mathcal{O}(k^{-2}h^{-2d}) + \mathcal{O}(h^4) \text{ and}$$
$$\mathbb{V}[\hat{H}_{\text{KDE, Joe}}] \in \mathcal{O}(k^{-1}) + \mathcal{O}(k^{-2}h^{8-d}) + \mathcal{O}(k^{-2}h^{-d}) + \mathcal{O}(k^{-1}h^{8-d}) + \mathcal{O}(k^{-2}h^{4-2d}) + \mathcal{O}(h^8).$$

Hence, as long as $h \in \mathcal{O}(k^{-1/(d+\varepsilon)})$ for some small $\varepsilon > 0$ both the bias and the variance vanish, and the estimator convergences in MSE, even if it does so at a slow rate. Then, a sensible choice of the bandwidth is $h \approx 1$ since $k^{-1/(d+\varepsilon)} \to 1$ as d increases.

Under the mild assumption that the distribution of Z_b is β-smooth (i.e., it belongs to the Hölder or Sobolev classes) then the bias and variance of both KDE estimators $\hat{H}_{\text{KDE}}$ are (Krishnamurthy & Wang, 2015)

$$\mathbb{B}[\hat{H}_{\text{KDE}}] \in \mathcal{O}(h^{\beta}) \text{ and}$$
$$\mathbb{V}[\hat{H}_{\text{KDE}}] \in \mathcal{O}(k^{-1}h^{-d}).$$

As previously, the bias and the variance of the estimator only vanish if $h \in \mathcal{O}(k^{-1/(d+\varepsilon)})$ for some small $\varepsilon > 0$, with the optimal choice $h = k^{-1/(d+2\beta)}$. Nonetheless, having a bias term independent of the parameter of the optimisation is not harmful in itself. Hence, when the KDE estimator is employed only for optimisation purposes both $h \in \mathcal{O}(k^{-1/(d+\varepsilon)})$ and $h \in \mathcal{O}(1)$ may work. For instance, for the experiments using the von Mises–Fisher distribution we set $h = 0.1$ to match the temperature employed by (Tian et al., 2020a, `CMC`) and (Chen et al., 2020a, `SimCLR`).

A.1.2. Reconstruction estimation

Note that $\log q_{Z_2|Z_1^{(i)}}(Z_2^{(i)})$ are independent and identically distributed random variables with expectation $\mathbb{E}[\log q_{Z_2|Z_1}(Z_2)]$. Hence, the empirical estimator is unbiased. Similarly, the variance of the estimator is $\mathbb{V}[\frac{1}{k}\sum_{i=1}^{k} \log q_{Z_2|Z_1^{(i)}}(Z_2^{(i)})] = \sigma_q^2/k$, where the individual variance is $\sigma_q^2 = \mathbb{V}[\log q_{Z_2|Z_1}(Z_2)]$.

Consider now that a reconstruction density is of the form $q_{Z_2|Z_1=z_1}(z_2) = Ce^{-\rho(z_2,z_1)}$ and that the projections lay in a convex body $\mathcal{Z} \in \mathbb{R}^d$. Then, we know that $\log q_{Z_2|Z_1=z_2}(z_1) \in [\log C - \rho(\mathcal{Z}), \log C]$, where $\rho(\mathcal{Z})$ is the diameter of $\mathcal{Z}$ with respect to ρ. Therefore, by the Popoviciu's inequality on variances we have that $\sigma_q^2 \leq \rho(\mathcal{Z})^2/4$, which implies that for $\rho(\mathcal{Z}) < \infty$ the estimator converges in MSE. This holds for the two cases considered in this paper:

- Von Mises–Fisher distribution in $\mathcal{Z} = \mathbb{S}^{d-1}$: Here the diameter with respect to $\rho(z_1, z_2) = \kappa \text{sim}(z_1, z_2)$ is $\rho(\mathcal{Z}) = \kappa^2$ and hence the estimator converges in MSE at a $\kappa^2/(4k)$ rate.

- Gaussian distribution in $\mathcal{Z} = [-1, 1]^d$: Here the diameter with respect to $\rho(z_1, z_2) = \|z_1 - z_2\|^2/(2\sigma^2)$ is $\rho(\mathcal{Z}) = 2d/\sigma^2$ and hence the estimator converges in MSE at a $d/(2k\sigma^2)$ rate.

Remark A.1. Consider, without loss of generality from the setting in Section 4, a reconstruction density of the form $q_{Z_2|Z_1=z_1}(z_2) = C(z_1)e^{-\rho(z_2,z_1)}$. Then, to be precise, the reconstruction estimate in (6) is biased with bias $\mathbb{E}[\log C(Z_1)]$. However, since this term does not affect the optimization, we say the estimator is unbiased, meaning that is unbiased to the terms required for the optimization.

To see why the term does not affect the optimization, assume first that Z_1 is fixed to z_1. Then, $C(z_1)$ is a constant and clearly optimizing $\mathbb{E}[f(Z_2, Z_1)|Z_1 = z_1] + \log C(z_1)$ is equivalent to optimizing just $\mathbb{E}[f(Z_2, Z_1)]$. Then, note that $\mathbb{E}[f(Z_2, Z_1) + \log C(Z_1)] = \mathbb{E}_{z_1 \sim p_{Z_1}}[\mathbb{E}[f(Z_2, Z_1)|Z_1 = z_1] + \log C(z_1)]$, and hence optimizing $\mathbb{E}[f(Z_2, Z_1)]$ is sufficient.

A.2. Estimators in a discrete space

A.2.1. Maximizing MI between projections and surrogates

Now, we consider an estimation of the `ER` bound for the MI between the projections and the discrete surrogates $I(Z_1; W_2)$. We will assume that the discrete surrogates are in $[d]$. As previously, we may consider the following unbiased estimate of the reconstruction term

$$\widehat{\text{Rec}}_{\text{disc}} := \frac{1}{k}\sum_{i=1}^{k} \log q_{W_2|Z_1=Z_1^{(i)}}(W_2^{(i)}), \tag{10}$$

where the reconstruction density could simply be, for instance, $q_{W_2|Z_1=z} = \mathsf{s}(z)$. Since W_2 is a discrete RV, one may consider the empirical estimate of the marginal, i.e., $\hat{p}_{W_2} := \frac{1}{k}\sum_{i=1}^{k} p_{W_2|Z_2=Z_2^{(i)}}$, and use the unbiased plug-in

estimate (Girsanov, 1959) of the entropy:

$$\hat{H}(W_2)_{\text{plug-in}} := -\frac{1}{k}\sum_{i=1}^{k} \hat{p}_{W_2}(W_2^{(i)}) \log \hat{p}_{W_2}(W_2^{(i)}). \tag{11}$$

Both the reconstruction and entropy estimators are unbiased and converge in MSE, see the following sections.

Remark A.2. In practice, if the product of the batch size k and the discrete dimension d is large, one may instead consider an average of plug-in estimates. For simplicity, assume that the batch size is a multiple of $r \in \mathbb{N}$, then this estimate is

$$\hat{H}(W_2)_{\text{plug-in-avg}} := \frac{1}{r}\sum_{j=1}^{r} \hat{H}(W_2)^{(j)}_{\text{plug-in}} \tag{12}$$

where $\hat{H}(W_2)^{(j)}_{\text{plug-in}}$ is the plug-in entropy estimate of the j-th chunk of the batch size data. Usually, each entropy estimation is done in a different machine, or *replica*.

A.2.2. Entropy estimation

The plug-in estimator $\hat{H}_{\text{PI}}$ of the entropy $H(W_2)$ is known to have the following bias and variance terms (see e.g. (Girsanov, 1959, Equations (3) and (4)) or (Antos & Kontoyiannis, 2001, Introduction)):

$$\mathbb{B}[\hat{H}_{\text{plug-in}}] \in \mathcal{O}\Big(\frac{d-1}{2k}\Big) + \mathcal{O}\Big(\frac{1}{k^2}\Big) \text{ and}$$
$$\mathbb{V}[\hat{H}_{\text{plug-in}}] \in \mathcal{O}\Big(\frac{\sigma_p^2}{k}\Big) + \mathcal{O}\Big(\frac{1}{k^2}\Big),$$

where $\sigma_p^2 = \mathbb{V}[-\log p_{W_2}(W_2)]$. The bias and the variance vanish as long as d is fixed and $\sigma_p^2 < \infty$, meaning that the estimator converges in MSE.

Note that $p_{W_2} = \mathbb{E}[\mathsf{s}(Z_2)]$, where s is the softmax operator. Hence, we have that

$$\begin{aligned}\mathbb{V}[-\log p_{W_2}(W_2)] &\le \mathbb{E}[\log^2 p_{W_2}(W_2)] \\ &\le \mathbb{E}\Big[\Big(Z_{2,W_2} - \log\Big(\sum_{l=1}^{d} e^{Z_{2,l}}\Big)\Big)^2\Big] \\ &\le \mathbb{E}\big[(\log d + Z_{2,\max} - Z_{2,\min})^2\big],\end{aligned}$$

where the first inequality follows from the fact that $\mathbb{V}[X] \le \mathbb{E}[X^2]$; the second from Jensen's inequality and the formula of the softmax; and the last one from the log-sum-exp trick. Here, $Z_{2,l}$ denotes the l-th element of the random vector Z_2.

In the particular case where the projections lie in the sphere $\mathbb{S}^{d-1}$ we have that $\sigma_p^2 \le (\log d + 1)^2$. Similarly, if they lay in the cube $[-1,1]^d$, we have that $\sigma_p^2 \le (\log d + 2)^2$. Therefore, under these standard conditions the variance vanishes at a rate in $\mathcal{O}(\log^2(d)/k) + \mathcal{O}(1/k^2)$.

Remark A.3. If the average of plug-in estimators is used instead, we may note that this estimator will have a drop in bias and variance proportional to the number of replicas r, where the variance is not quadratically affected due to the variance reduction effect of the averaging. More precisely,

$$\mathbb{B}[\hat{H}_{\text{plug-in-avg}}] \in \mathcal{O}\Big(\frac{r(d-1)}{2k}\Big) + \mathcal{O}\Big(\frac{r}{k^2}\Big) \text{ and}$$
$$\mathbb{V}[\hat{H}_{\text{plug-in-avg}}] \in \mathcal{O}\Big(\frac{\sigma_p^2}{k}\Big) + \mathcal{O}\Big(\frac{r}{k^2}\Big).$$

A.2.3. Reconstruction estimation

As in Appendix **A.1.2**, note that $\log q_{W_2|Z_1^{(i)}}(W_2^{(i)})$ are independent and identically distributed random variables with expectation $\mathbb{E}[\log q_{W_2|Z_1}(W_2)]$. Hence, the empirical estimator is unbiased. Similarly, the variance of the estimator is $\mathbb{V}[\frac{1}{k}\sum_{i=1}^{k} \log q_{W_2|Z_1^{(i)}}(W_2^{(i)})] = \sigma_q^2/k$, where $\sigma_q^2 = \mathbb{V}[\log q_{W_2|Z_1}(W_2)]$. Hence, the variance vanishes as long as $\sigma_q^2 < \infty$, meaning that the estimator converges in MSE.

As for the entropy estimation, note that $q_{W_2|Z_1} = \mathsf{s}(Z_1)$. Hence, repeating the analysis above in Appendix **A.2.2** we obtain that $\sigma_q^2 \le \mathbb{E}\big[(\log d + Z_{1,\max} - Z_{1,\min})^2\big]$ and therefore for projections in the sphere $\mathbb{S}^{d-1}$ or the cube $[-1,1]^d$ the variance vanishes at a rate in $\mathcal{O}(\log^2(d)/k) + \mathcal{O}(1/k^2)$.

B. Properties of maximizing MI via BA

This section formalises and contextualises statements in Section **4**.

B.1. Recovering the true latent variables

Let us consider the standard assumption in independent components analysis (ICA), namely that the data X is generated by a nonlinear, invertible generative process $X = g(\tilde{Z})$ from some original latent variables $\tilde{Z}$. Assume further that the different views from the image can be understood as $V_1 = g(\tilde{Z}_1)$ and $V_2 = g(\tilde{Z}_2)$, where there is some joint density of the latent variables $p_{\tilde{Z}_1,\tilde{Z}_2}$. The next theorem shows how Zimmermann et al. (2021) theory can be adapted to prove that mutli-view SSL methods that maximize the mutual information between their projections $I(Z_1; Z_2)$ can obtain projections equivalent to the true latent variables up to affine transformations.

Theorem B.1. *Assume that the latent variables and the network's projections lay on a convex body $\mathcal{Z} \in \mathbb{R}^d$. Further assume that the latent variables' marginal distribution is uniform and that the conditional density is described by a semi-metric ρ as $p_{\tilde{Z}_2|\tilde{Z}_1=\tilde{z}_1}(\tilde{z}_2) = C(\tilde{z}_1)e^{-\rho(\tilde{z}_1,\tilde{z}_2)}$. Now let the reconstruction density match the conditional density up to a constant scaling factor $q_{Z_2|Z_1=z_1}(z_2) = C_h(\tilde{z}_1)e^{-\alpha\rho(z_1,z_2)}$. If the generative process g and the pa-*

rameterised network functions $\pi \circ f$ are invertible and differentiable, and the parameters θ maximize the lower bound (2) *of the mutual information $I(Z_1; Z_2)$, then the projections are equivalent to the true latent variables up to affine transformations.*

Proof. As in (Zimmermann et al., 2021), let $h = \pi \circ f \circ g$ be a helper function that brings the true latent variables to the projections so that $Z_1 = h(\tilde{Z}_1)$ and $Z_2 = h(\tilde{Z}_2)$.

Disregard for a moment the entropy term $H(Z_2)$. From (Zimmermann et al., 2021, Proposition 4) we know that if the reconstruction term is maximized (the cross entropy is minimised) then $\rho(\tilde{z}_1, \tilde{z}_2) = \alpha\rho(h(\tilde{z}_1), h(\tilde{z}_2))$ and $C(\tilde{z}_1) = C_h(\tilde{z}_1)$. Moreover, from (Zimmermann et al., 2021, Theorem 4) we further know that h is an invertible affine transformation; i.e. $Z_2 = A\tilde{Z}_2 + b$ for some $A \in \mathbb{R}^{d\times d}$ and some $b \in \mathbb{R}^d$.

Now note that

$$\begin{aligned} H(Z_2) &= -\mathbb{E}\Big[\log \mathbb{E}\Big[C_h(\tilde{Z}_1)e^{-\alpha\rho(h(\tilde{Z}_1),h(\tilde{Z}_2)}\Big]\Big] \\ &= -\mathbb{E}\Big[\log \mathbb{E}\Big[C(\tilde{Z}_1)e^{-\rho(\tilde{Z}_1,\tilde{Z}_2)}\Big]\Big] = H(\tilde{Z}_2). \end{aligned}$$

Then, since the latent variables' are uniformly distributed, their entropy is maximal $H(\tilde{Z}) = \log|\mathcal{Z}|$.

Therefore, the unique family of maximizers of the reconstruction term recover the latent variables up to affine transformations are maximizers of the entropy, and hence are the unique family of maximizers of the mutual information. Indeed, take some other maximizer of the entropy, if it is not from this family, it is not a maximizer of the reconstruction and therefore the resulting mutual information is lower. □

Remark B.2. Following the same reasoning and supporting on Zimmermann et al. (2021)'s theory, we may note that in the particular case that the semi-metric ρ is described by an L^p norm, then the projections are equivalent to the true latent variables up to generalised permutations; that is, $Z = A\tilde{Z}$ for some $A \in \mathbb{R}^{d\times d}$ such that $(Az)_i = \alpha\beta_i z_{\sigma(i)}$, where $\alpha \in \mathbb{R}$, $\beta_i \in \{1, -1\}$, and σ is a permutation. Similarly, in the more restrictive case that the projections are in the sphere $\mathcal{Z} = \mathbb{S}^{d-1}$ and the conditional densities are von Mises–Fisher densities, then the projections are equivalent to the true latent variables up to linear transformations; that is, $Z = A\tilde{Z}$ for some $A \in \mathbb{R}^{d\times d}$ such that $A^\intercal A = \alpha I$ for some $\alpha \in \mathbb{R}$.

B.2. Empirical evaluation of identifiability properties

In order to empirically study the consequences of the theoretical results of the previous section, we replicate the results of the experiments reported in (Zimmermann et al., 2021, Tables 1 and 2) and study the impact of using the `ER` objective on the identifiability properties.

The results are reported in Tables **3** and **4**. The results of Zimmermann et al. (2021) are reported in the column `InfoNCE` as they train their unsupervised models with the `InfoNCE` objective. In these experiments, we used Joe (1989)'s entropy estimator (7) and we set the KDE bandwidth to $h = 1$ or $h = 10$ as per (9) (we report both results).

At a high level, our results indicate that using the `ER` objective yields qualitatively similar identifiability properties as using the `InfoNCE` objective, as predicted by theory. One discrepancy of note is the difference for $h = 10$ in the first row of Table 3 since that is the setting for which the generative process and the model match the assumptions of Theorem B.1. However, for $h = 1$, the performance is close to the once achieved by `InfoNCE`. Other settings, including those which break the theoretical assumptions, do not exhibit significant differences between `ER` and `InfoNCE`.

B.3. Isolating semantic from irrelevant information

Similarly to Appendix **B.1**, let us consider that the data X is generated by a nonlinear, invertible generative process $X = g(\tilde{Z})$ from some original latent variables $\mathcal{Z}$ and that the different views can be understood as $V_1 = g(\tilde{Z}_1)$ and $V_2 = g(\tilde{Z}_2)$, where there is some joint density of the latent variables $p_{\tilde{Z}_1,\tilde{Z}_2}$.

Assume that the latent variables can be written as $\tilde{Z} = [S, U]$, where $S \in \mathbb{R}^d$ is some semantic (or content) variable, $U \in \mathbb{R}^{d_u}$ is some irrelevant (or style) variable, and $[\cdot]$ denotes the concatentation operation. Furthermore, let us adopt the assumptions from Von Kügelgen et al. (2021) for the content-preserving conditional density $p_{\tilde{Z}_2|\tilde{Z}_1}$.

Assumption B.3 (Content-invariance)**.** The conditional density $p_{\tilde{Z}_2|\tilde{Z}_1}$ of the latent variables of different views has the form

$$p_{\tilde{Z}_2|\tilde{Z}_1=\tilde{z}_1}(\tilde{z}_2) = \delta(s_2 - s_1)p_{U_2|U_1=u_1}(u_2),$$

for all $\tilde{z}_1 = [s_1, u_1]$ and $\tilde{z}_2 = [s_2, u_2]$ in $\mathcal{Z}$ and where $p_{U_2|U_1=u_1}$ is continuous for all $u_1 \in \mathbb{R}^{d_u}$.

Assumption B.4 (Style changes)**.** Let $\mathcal{A}$ be the set of subsets of irrelevant variables $A \subseteq \{1, \dots, d_u\}$ and let p_A be a density on $\mathcal{A}$. Then, the conditional density $p_{U_2|U_1}$ is obtained via sampling $a \sim p_A$ and letting

$$p_{U_2|U_1,A=u_1,a}(u_2) = \delta(u_{2,a^c}, u_{1,a^c})p_{U_{2,a}|U_{1,a}=u_{1,a}}(u_{2,a}),$$

where $p_{U_{2,a}|U_{1,a}=u_{1,a}}$ is a continuous density for all $u_{1,a} \in \mathbb{R}^{|a|}$, and where $u_{2,a}$ is an abuse of notation to refer to the elements of u_2 indexed by a, and analogously for u_1 and for a^c, which is a shortcut for $\mathcal{A} \setminus a$.

Then, the next theorem shows how Von Kügelgen et al. (2021) can be adapted to prove that multi-view SSL methods that maximize the mutual information between the pro-

Table 3. Identifiability up to affine transformations, replicated from Table 1 of Zimmermann et al. (2021) and updated with results trained using the ER objective, rather than InfoNCE. The numbers for the InfoNCE column are taken from Zimmermann et al. (2021) with the exception of the last two rows marked with *, where we were not able to replicate their results. The ER configurations for $h = 1$ marked with † exhibited training instability which is why their results are considerably lower.

Generative process g			Model f		R^2 Score [%]		
Space	$p(\cdot)$	$p(\cdot\|\cdot)$	Space	$q_h(\cdot\|\cdot)$	InfoNCE	$\text{ER}_{h=10}$	$\text{ER}_{h=1}$
Sphere	Uniform	vMF(κ=1)	Sphere	vMF(κ=1)	99.42	89.31	98.94
Sphere	Uniform	vMF(κ=10)	Sphere	vMF(κ=1)	99.86	99.87	99.80
Sphere	Uniform	Laplace(λ=0.05)	Sphere	vMF(κ=1)	99.91	99.88	99.62
Sphere	Uniform	Normal(σ=0.05)	Sphere	vMF(κ=1)	99.86	99.82	99.31
Box	Uniform	Normal(σ=0.05)	Unbounded	Normal	99.60	99.53	$71.47^\dagger$
Box	Uniform	Laplace(λ=0.05)	Unbounded	Normal	99.64	99.57	$17.30^\dagger$
Box	Uniform	Laplace(λ=0.05)	Unbounded	GenNorm(β=3)	99.70	99.76	99.73
Box	Uniform	Normal(σ=0.05)	Unbounded	GenNorm(β=3)	99.69	99.72	99.67
Sphere	Normal(σ=1)	Laplace(λ=0.05)	Sphere	vMF(κ=1)	99.02	99.09	98.77
Sphere	Normal(σ=1)	Normal(σ=0.05)	Sphere	vMF(κ=1)	99.02	98.96	98.43
Unbounded	Laplace(λ=1)	Normal(σ=1)	Unbounded	Normal	89.85^*	88.86	88.69
Unbounded	Normal(σ=1)	Normal(σ=1)	Unbounded	Normal	95.26^*	89.89	89.74

Table 4. Identifiability up to generalized permutations, replicated from Table 2 of Zimmermann et al. (2021) and updated with results trained using the ER objective rather than InfoNCE. The numbers for the InfoNCE column are taken from Zimmermann et al. (2021) with the exception of the rows marked with *, where we were not able to replicate their results.

Generative process g			Model f		MCC Score [%]		
Space	$p(\cdot)$	$p(\cdot\|\cdot)$	Space	$q_h(\cdot\|\cdot)$	InfoNCE	$\text{ER}_{h=10}$	$\text{ER}_{h=1}$
Box	Uniform	Laplace(λ=0.05)	Box	Laplace	98.62	98.49	97.49
Box	Uniform	GenNorm(β=3; λ=0.05)	Box	GenNorm(β=3)	99.90	99.90	95.51
Box	Uniform	Normal(σ=0.05)	Box	Normal	99.77	99.74	96.22
Box	Uniform	Laplace(λ=0.05)	Box	Normal	99.76	99.76	99.74
Box	Uniform	GenNorm(β=3; λ=0.05)	Box	Laplace	98.80	98.77	98.65
Box	Uniform	Laplace(λ=0.05)	Unbounded	Laplace	98.57	98.57	98.53
Box	Uniform	GenNorm(β=3; λ=0.05)	Unbounded	GenNorm(β=3)	60.54^*	61.23	51.44
Box	Uniform	Normal(σ=0.05)	Unbounded	Normal	58.26	56.52	53.14
Box	Uniform	Laplace(λ=0.05)	Unbounded	Normal	59.67	56.32	31.10
Box	Uniform	Normal(σ=0.05)	Unbounded	GenNorm(β=3)	54.59^*	54.58	39.01

jections $I(Z_1; Z_2)$ can obtain projections that capture and isolate the semantic information of the true latent variables.

Theorem B.5. *Consider Assumption **B.3** and Assumption **B.4** and further assume that*

1. *the generative process g is smooth, invertible and with a smooth inverse (i.e., a diffeomorphism);*
2. *$p_{\tilde{Z}}$ is a smooth, continuous density on $\mathcal{Z}$ with $p_{\tilde{Z}} > 0$ a.e.; and*
3. *for any $j \in \{1, \dots, n_u\}$, there is an $a \subseteq \{1, \dots, n_u\}$ such that $j \in a$, $p_A(a) > 0$, $p_{U_{2,a}|U_{1,a}=u_{1,a}}(u_{2,a})$ is smooth with respect to both $u_{1,a}$ and $u_{2,a}$, and for any $u_{1,a}$ it holds that $p_{U_{2,a}|U_{1,a}=u_{1,a}}(u_{2,a}) > 0$ for all $u_{2,a}$ in some open, non-empty subset containing $u_{1,a}$.*

If the parameterised network function $\pi \circ f$ is smooth, the projections space is $(0,1)^d \subseteq \mathbb{R}^d$, and the parameters θ are found to maximize the mutual information $I(Z_1; Z_2)$ lower bound (2) with the reconstruction density $q_{Z_2|Z_1=z_1}(z_2) = C_{\text{gauss}}(1)e^{-\|z_2 - z_2\|_2^2}$, then there is a bijection between the projections Z and the true semantic variables S.

Proof. The proof follows directly by Von Kügelgen et al. (2021, Theorem 4.4) by noting that the maximizing of the mutual information lower bound (2) with the reconstruction density $q_{Z_2|Z_1=z_1}(z_2) = C_{\text{gauss}}(1)e^{-\|z_2 - z_2\|_2^2}$ coincides with the minimisation of their theorem. □

C. Algorithm

Algorithm **1** describes the main algorithm to maximise the ER bound. The algorithm includes the possibility of considering the projections in the standard projection space $\mathcal{Z}$, which usually is the d-shpere, or to further generate discrete surrogates in $\{1, 2, \dots, d\}$. In case the projections are

not further processed, it allows to use either Joe (1989)'s or the plug-in KDE estimators for the entropy. Finally, it also includes an option to add the ER bound into distillation methods. The algorithm does not spell out how to extend it to more than two views, but that is done in the usual way, see e.g. (Caron et al., 2021).

D. Performance of different KDE estimators

In Section **4.1**, we show that the entropy term of ER can be approximated with Joe (1989)'s KDE or the plug-in estimator (Krishnamurthy & Wang, 2015). In Figure **3** we empirically show that using both estimators to optimize the ER bound on SimCLR (SimCLR + ER) leads to the same performance in terms of ImageNet top-1 accuracy.

E. Extended results

Table **5** extends Table **2** with CMC and EMA ablation for SimCLR + ER and DINO + ER. As it can be seen the EMA is essential for distillation methods to achieve good performance. Thus, we hypothesize that the function of EMA is not only limited to keeping the entropy high. We observe that all contrastive methods, including CMC, are more stable when reducing the batch size than distillation methods.

For Table **5**, we used a continuous entropy KDE-estimator for every method. This includes DINO, contrary to the exact discrete estimator used in the main text (Table **2**), showing that ER is resilient to the entropy estimator used. To make it clearer, we run an ablation with the two estimators and observe no significant difference in their results (see Table **6**).

Table 5. Training w/ ER yields comparable performance while it improves robustness to changes in batch size. **Model**: the set of augmentations, loss, and projection head. *Our re-implementation of the original. **ER**: the original loss has been substituted by the ER bound (2). **EMA**: the model uses EMA. **MI**: known to maximize MI. Parentheses: no formal proof provided (Section **4.2**). **Δ512**: accuracy drop when reducing batch size to 512.

Model	EMA	MI	Acc (↑)	Δ512 (↓)
DINO*	✓	?	75.28	8.63
DINO + ER	×	✓	67.30	5.45
DINO + ER	✓	(✓)	73.63	2.67
BYOL*	✓	?	73.42	23.65
BYOL + ER	×	✓	71.70	3.20
BYOL + ER	✓	(✓)	71.94	2.35
CMC*	×	✓	69.95	3.06
SimCLR*	×	×	70.23	2.17
SimCLR + ER	×	✓	70.86	1.01

Table 6. There is no significant difference between the discrete and the continuous entropy estimators.

Model	Discrete	Acc (↑)	Δ512 (↓)
DINO*	✓	75.28	8.63
DINO + ER	✓	73.39	2.35
DINO + ER	×	73.63	2.67

F. Entropy minimization in discrete MVSSL algorithms

Figure **4** shows the evolution of the entropy for discrete methods such as SwAV, DeepCluster, and DINO. The entropy is normalized by $\log |\mathcal{W}|$, which is the maximum possible entropy for these methods. We observe that while the normalized entropy does not increase over time, its value is kept close to the maximum. This is expected for SwAV and DeepCluster, since we have shown that they maximize the ER (Section **3.2**). For DINO, we showed that softmax centering and sharpening could be used to maintain a high entropy value (Section **3.3** and Figure **4**).

The behavior of the entropies was to be expected, as we discuss below.

In SwAV, the Sinkhorn-Knopp in the first iteration almost completely accomplishes that the conditional entropy $p_{W_2|Z_2} = \mathbf{1}_m/m$, as it has full liberty to do so. Therefore, the marginal entropy p_{W_2} is uniform and the entropy is maximal. As the iterations continue, the tension between the cross entropy related to the reconstruction term and the entropy maximization, together with the fact that Caron et al. (2020) only perform three iterations of the Sinkhorn-Knopp algorithm, push the conditional entropy $p_{W_2|Z_2}$ slightly away from uniformity, thus decreasing the entropy.

In DeepCluster, the images of each batch are sampled based on a uniform distribution over cluster assignments, i.e. for each batch $p_{W_2} \approx \frac{1}{k}\sum_{i=1}^{k} p_2^{(i)}$ is almost uniform. This way, the entropy $H(W_2)$ is approximately maximized.

In DINO, the centering step at the first iteration, before the weights have been updated, completely accomplishes a uniform conditional distribution, as then $\xi = \theta$ and thus the sharpening has no effect. In the second iteration, the weights already are different $\xi \neq \theta$, and the low value of the temperature pushes the conditional entropy $p_{W_2|Z_2}$ away from uniformity. This is compensated by the centering, which avoids that $p_{W_2|Z_2}$ becomes completely degenerate. Therefore, the result is an entropy that is lower than the maximal (sharpening) but not minimal (centering). The tension between these to mechanisms evolves towards higher entropies as the temperature is scheduled to increase over time, and thus the entropy does too.

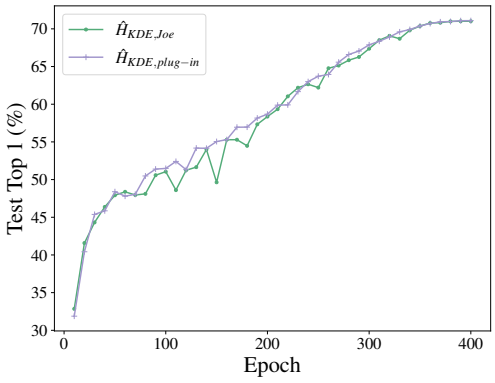

Figure 3. Joe's and the plug-in KDE achieve the same performance when used to optimize the `ER` *bound.* `SimCLR` + `ER` top-1 test accuracy on the ImageNet with different KDE to estimate the entropy term.

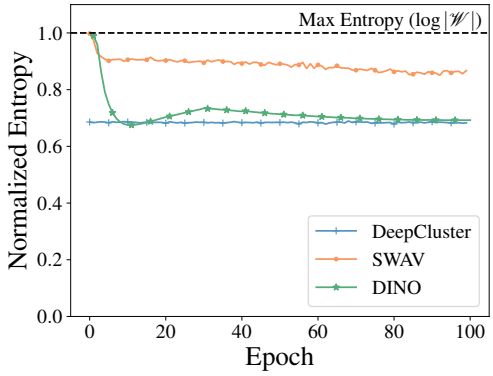

Figure 4. `DINO`*,* `SwAV`*, and* `DeepCluster` *keep a high entropy in their embedding spaces.*

Algorithm 1 MVSSL algorithm maximising the `ER` bound

Input: Dataset $\mathcal{D} = x^{(1:n)}$, batch size k, flag for discrete surrogate `is_discrete`, reconstruction density q_{rec}, kernel density q_{KDE}, flag for Joe (1989)'s or plug-in estimators `is_Joe`, encoder and projector networks f_θ and π_θ, flag for distillation or not distillation `is_distillation`, EMA parameter $\lambda \in (0, 1)$, learning rate η, learning rate schedule, augmentation set $\mathcal{T}$, and number of iterations `niter`.

Set `iter` = 1.
while `iter` $\leq$ `niter` **do**
  Draw a batch $x^{(1:k)}$ uniformly at random from the dataset $\mathcal{D}$.
  Draw two sets of augmentations $t^{(1:k)}$ and $t^{(1:k)}$ uniformly at random from $\mathcal{T}$.
  for all $i \in \{1, \ldots, k\}$ **do**
    Retrieve the projections $z_1^{(i)} = \pi_\theta \circ f_\theta \circ t_1^{(i)}(x^{(i)})$ and $z_2^{(i)} = \pi_\theta \circ f_\theta \circ t_2^{(i)}(x^{(i)})$.
  end for
  if `is_distillation` **then**
    Apply a stop gradient to the teacher projections: $z_2^{(i)} \leftarrow \texttt{stopgrad}(z_2^{(i)})$.
  end if
  if `is_discrete` **then**
    Calculate the empirical p.m.f.s $\hat{p}_1 = \frac{1}{k}\sum_{i=1}^{k} \mathsf{s}(z_1^{(i)})$ and $\hat{p}_2 = \frac{1}{k}\sum_{i=1}^{k} \mathsf{s}(z_2^{(i)})$.
    Calculate the entropy estimators $\hat{H}_1$ and $\hat{H}_2$ using the empirical p.m.f.s according to (11)
    Calculate the reconstruction estimates $\widehat{\text{Rec}}_1$ and $\widehat{\text{Rec}}_2$ according to (10).
  else
    Calculate the densities' KDE $\hat{p}_1$ and $\hat{p}_2$ according to (9).
    if `is_Joe` **then**
      Calculate Joe (1989)'s KDE of the entropy $\hat{H}_1$ and $\hat{H}_2$ using the densities' KDE according to (7).
    else
      Calculate the plug-in KDE of the entropy $\hat{H}_2$ and $\hat{H}_2$ using the densities' KDE according to (8).
    end if
    Calculate the reconstruction estimates $\widehat{\text{Rec}}_1$ and $\widehat{\text{Rec}}_2$ according to (6).
  end if
  Calculate the loss $\mathcal{L} = -0.5\big(\hat{H}_1 + \widehat{\text{Rec}}_1 + \hat{H}_2 + \widehat{\text{Rec}}_2\big)$.
  Update the network weights using gradient descent: $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}$.
  if `is_distillation` **then**
    Update teacher's weights according to $\xi \leftarrow \lambda\xi + (1 - \lambda)\theta$.
  end if
end while

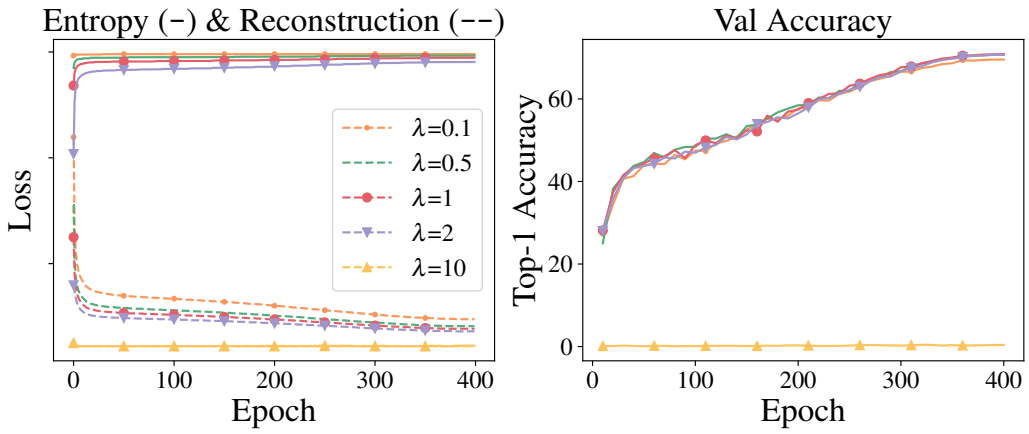

Figure 5. Increasing the weight (λ) of the reconstruction term leads to poor performance. Left: evolution of the entropy (−) and reconstruction (−−) of ER embedding spaces. Right: top-1 validation accuracy (%) on ImageNet.

G. Additional experiments with different weights for the reconstruction objective

Recall the ER objective from (2). This objective is theoretically justified as a lower bound on the mutual information between the projections of different branches. Moreover, Section **2** gives an intuition of the role of both the entropy and the reconstruction terms. Empirically, it may be interesting to consider giving different weights to each of the terms and see if this may lead to better performance. More precisely, it may be interesting to consider the objective

$$I_{\text{ER},\lambda} \coloneqq H(Z_2) + \lambda\, \mathbb{E}[\log q_{Z_2|Z_1}(Z_2)]$$

for different values of $\lambda > 0$ as we show in Figure **5**.

Intuitively, for projections in $\mathbb{R}^d$ that are not further projected into a discrete surrogate, the reconstruction is given by (6) choosing the reconstruction density $q_{Z_2|Z_1=z_1}(z_2) \propto \exp f(z_2, z_1)$, where $f(z_2, z_1)$ is a function measuring the similarity of z_1 and z_2. Therefore, the objective $I_{\text{ER},\lambda}$ is equivalent to consider a density $q_{Z_2|Z_1=z_1}(z_2) \propto \exp \lambda f(z_2, z_1)$. Under this interpretation, we may understand better the results obtained with the different weights since lower values of λ lead to flatter densities and high values of λ to very spikier densities:

- If the density is very flat, then the reconstruction term does not vary much even if Z_1 and Z_2 are close. This happens as the reconstruction density makes Z_1 and Z_2 only loosely dependent. Therefore, the network has a hard time learning and the performance decreases.
- If the density is very spiky (almost a delta), then it means that $Z_1 \approx Z_2$, so the network collapses.

When the projections are projected into a discrete surrogate, then the λ parameter should only be interpreted as a temperature parameter similar to those in other MVSSL methods. However, the understanding and intuition provided above still apply.