# RETHINKING UNIFORMITY IN SELF-SUPERVISED REPRESENTATION LEARNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Self-supervised representation learning has achieved great success in many machine learning tasks. While many research efforts focus on learning better representations by preventing the model from the *collapse* problem, less attention has been drawn to analyzing the collapse degrees of representations. In this paper, we present a formal study of collapse analysis via the *uniformity* metric, which measures how uniformly learned representations distribute on the surface of the unit hypersphere. We fundamentally find that *representation that obeys zero-mean isotropic Gaussian distribution is with the ideal uniformity* since its $l_2$-normalized form uniformly distributes on the surface of the unit hypersphere. Therefore, we propose to use the Wasserstein distance between the distribution of learned representations and the ideal distribution as a quantifiable metric of *uniformity*. Moreover, we design five desirable constraints for ideal uniformity metrics, based on which we find that the proposed uniformity metric satisfies all constraints while the existing one does not. Synthetic experiments also demonstrate the proposed uniformity metric is capable to deal with the dimensional collapse while the existing one is insensitive. Furthermore, we impose the proposed *uniformity* metric as an auxiliary loss term for various existing self-supervised methods, which consistently improves the downstream performance.

## 1 INTRODUCTION

Self-supervised representation learning has become increasingly popular in machine learning community (Chen et al., 2020; He et al., 2020; Caron et al., 2020; Grill et al., 2020; Chen & He, 2021; Zbontar et al., 2021), and achieved impressive results in various tasks such as object detection, segmentation, and text classification (Xie et al., 2021; Wang et al., 2021b; Yang et al., 2021; Zhao et al., 2021; Wang et al., 2021a; Gunel et al., 2021). Aiming at learning transferable features without label supervision, a common practice of self-supervised representation learning is to employ a Siamese network (Bromley et al., 1994), which could learn invariant representations under a series of different augmentations. However, a common issue with this approach is the existence of trivial constant solutions that all representations collapse to a constant point (Chen & He, 2021), as visualized in Fig. 1, known as the *collapse problem* (Jing et al., 2022).

Many efforts have been made to prevent the vanilla Siamese network from the collapse problem. The well-known solutions can be summarized into three types: contrastive learning (Chen et al., 2020; He et al., 2020; Caron et al., 2020), asymmetric model architecture (Grill et al., 2020; Chen & He, 2021), and redundancy reduction (Zbontar et al., 2021; Zhang et al., 2022b). While these solutions could avoid the complete constant collapse, they might still suffer from a *dimensional collapse* (Hua et al., 2021) in which representations occupy a lower-dimensional subspace instead of the entire available embedding space (Jing et al., 2022), as depicted in the Fig. 1. Therefore, to show the effectiveness of the aforementioned approaches, we need a quantifiable metric to measure the collapse degree of learned representations.

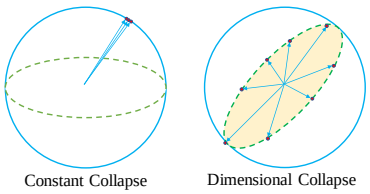



Figure 1: The left figure presents constant collapse, and the right figure visualizes dimensional collapse.

To gain a quantifiable analysis of collapse degree, recent works (Arora et al., 2019; Wang & Isola, 2020) propose to divide the loss function into *alignment* and *uniformity* terms. For instance, recent objective functions such as InfoNCE (van den Oord et al., 2018; Chen et al., 2020) and cross-correlation employed in Barlow Twins (Zbontar et al., 2021) could be divided into two terms. These *uniformity* terms could explain the degree of collapse to some extent, since they measure the variability of learned representations (Zbontar et al., 2021). However, the calculation of these *uniformity* terms relies on the choice of anchor-positive pair, making them hard to be used as general metrics. Wang et al (Wang & Isola, 2020) further propose a formal definition of *uniformity* metric via Radial Basis Function (RBF) kernel (Cohn & Kumar, 2007). Despite its usefulness, we theoretically and empirically demonstrate that the metric is insensitive to the dimensional collapse.

In this paper, we fundamentally find that *representation that obeys zero-mean isotropic Gaussian distribution is with the ideal uniformity* since its $l_2$-normalized form uniformly distributes on the surface of the unit hypersphere. Based on the above finding, we use the Wasserstein distance between the distribution of learned representations and the ideal distribution as the metric of *uniformity*, which can quantitatively analyze the dimensional collapse.

Furthermore, to facilitate comparison between our proposed *uniformity* metric and the existing one (Wang & Isola, 2020), we design five desirable constraints (desiderata) for ideal uniformity metrics. Synthetic experiments also demonstrate the proposed uniformity metric is capable to deal with the dimensional collapse while the existing one is insensitive. Lastly, we apply our proposed *uniformity* metric in the practical scenarios, namely, imposing it as an auxiliary loss term for various existing self-supervised methods, which consistently improves the downstream performance.

The contributions of this work are as below: (i) We fundamentally find that *representation that obeys zero-mean isotropic Gaussian distribution is with the ideal uniformity*, see Theorem 1; (ii) We design five desirable properties (Desiderata) for ideal uniformity; (iii) Based on Theorem 1, we propose an alternative uniformity metric, which is theoretically and empirically proved to be superior to the existing one (Wang & Isola, 2020); and (iv) It could consistently improve the performance by using the proposed uniformity metric as an auxiliary loss in various self-supervised methods.

## 2 BACKGROUND

### 2.1 SELF-SUPERVISED REPRESENTATION LEARNING

Self-supervised representation learning aims to learn representations that are invariant to a series of different augmentations. Towards this end, a typical choice is to employ a symmetric network architecture, also called Siamese network (Bromley et al., 1994). Specially, given the dataset $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, ..., \mathbf{x}_n\}$, Siamese network takes as input two randomly augmented views $\mathbf{x}_i^a$ and $\mathbf{x}_i^b$ from a input sample $\mathbf{x}_i$. Then the two views are processed by an encoder network $f(\cdot)$ consisting of a backbone (e.g., ResNet (He et al., 2016)) and a projection MLP head (Chen et al., 2020), denoted as $g(\cdot)$. Both the encoder network and projection MLP head share trainable weights $\theta$ for two views. In this way, representations of two views can be obtained as $\mathbf{z}_i^a = g_\theta(f_\theta(\mathbf{x}_i^a))$ and $\mathbf{z}_i^b = g_\theta(f_\theta(\mathbf{x}_i^b))$. To learn invariant representations, a natural solution is to maximize the similarity between representations of two views, and Mean Square Error (MSE) is a widely used loss function to align their $l_2$-normalized representations on the surface of the unit hypersphere.

$$\mathcal{L}_{align}^{\theta} = \| \frac{\mathbf{z}_i^a}{\|\mathbf{z}_i^a\|} - \frac{\mathbf{z}_i^b}{\|\mathbf{z}_i^b\|} \|_2^2 = 2 - 2 \cdot \frac{\langle \mathbf{z}_i^a, \mathbf{z}_i^b \rangle}{\|\mathbf{z}_i^a\| \cdot \|\mathbf{z}_i^b\|} \tag{1}$$

However, a common issue with this approach easily learns an undesired trivial solution that all representations collapse to a constant, as depicted in Fig. 1.

### 2.2 EXISTING SOLUTIONS TO CONSTANT COLLAPSE

To prevent the Siamese network from the constant collapse, existing well-known solutions can be summarized into three types: contrastive learning, asymmetric model architecture, and redundancy reduction. More details will be explained in this section.

**Contrastive Learning** Contrastive learning is one effective way to avoid constant collapse, and the core idea is to repulse negative pairs while attracting positive pairs. SimCLR (Chen et al., 2020) is

one of the most representative works, which first proposes an in-batch negative trick that employs samples in a batch as negative samples. However, its effectiveness heavily relies on the large batch size. To overcome the limitation, MoCo (He et al., 2020) proposes a memory bank to save more representations as negative samples. Besides instance-wise contrastive learning approaches, some recent works also propose clustering-based contrastive learning by bringing together a clustering objective with contrastive learning (Li et al., 2021; Caron et al., 2020).

**Asymmetric Model Architecture** Asymmetric model architecture is another approach to prevent constant collapse, the core idea is to break the symmetry of the Siamese network. A possible explanation is that asymmetric architecture could encourage encoding more information (Grill et al., 2020). To keep asymmetry, BYOL (Grill et al., 2020) proposes to use an extra predictor in one branch of the Siamese network, and use momentum update and stop-gradient operator in the other branch. Chen et al. propose SimSiam (Chen & He, 2021) by removing the momentum update from BYOL, its success demonstrates the momentum update is not the key to preventing collapse. Mirror-SimSiam (Zhang et al., 2022a) further swap the stop-gradient operator to the other branch, its failure refutes the claim in SimSiam (Chen & He, 2021), that the stop-gradient operator is the key component to preventing the model from collapse. Although empirical results demonstrate the effectiveness of these approaches, the nature of asymmetric model architecture is still unclear.

**Redundancy Reduction** The principle for redundancy reduction to prevent constant collapse is to maximize the information content of the representations. The core is to achieve decorrelation by making the matrix based on representations as close to the identity matrix as possible. Barlow Twins (Zbontar et al., 2021) tries to achieve this end on the cross-correlation matrix, while VICReg (Bardes et al., 2022) chooses on the covariance matrix. Instead of applying regularization to the matrix, W-MSE(Ermolov et al., 2021) employs a direct way to make the covariance matrix equal to the identity matrix via feature-wise whitening. Zero-CL (Zhang et al., 2022b) further proposes the hybrid of instance-wise and feature-wise whitening to achieve this end.

### 2.3 Collapse Analysis

While aforementioned solutions could effectively prevent model from constant collapse, they might still suffer from the dimensional collapse in which representations occupy a lower-dimensional subspace instead of the entire available embedding space, as depicted in the Fig. 1. The evidence of dimensional collapse was identified in contrastive learning by singular value spectrum of representations (Jing et al., 2022). However, the singular value spectrum is in the form of pictures, making it hard to conduct statistical comparisons among various approaches in terms of collapse analysis.

To gain a quantifiable analysis of collapse degree, Wang et al. propose a formal definition of *uniformity* metric in (Wang & Isola, 2020), via Radial Basis Function (RBF) kernel (Cohn & Kumar, 2007). More specially, given a set of representation vectors $\{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$ ($\mathbf{z}_i \in \mathbb{R}^m$), the *uniformity* metric is defined as follows:

$$\mathcal{L}_{\mathcal{U}} \triangleq \log \frac{1}{n(n-1)/2} \sum_{i=2}^{n} \sum_{j=1}^{i-1} e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2}, t > 0, \tag{2}$$

Where $t$ is a fixed parameter, and is generally set to 2. Despite its usefulness, we theoretically and empirically demonstrate this metric is insensitive to the dimensional collapse.

## 3 Methodology

In this section, to seek a better analysis of dimensional collapse, we propose a new *uniformity* metric based on the distribution of learned representations. Specially, we theoretically demonstrate *the maximum uniformity could be achieved when learned representations obey zero-mean isotropic gaussian distribution* in Section 3.1. To enforce the uniqueness of the ideal distribution, we adopt its $l_2$-normalized form, and we empirically verify it is an approximately Gaussian distribution in Section 3.2. Based on this principle, we propose to use Wasserstein distance between the distribution of learned representations and its ideal distribution as an *uniformity metric* in Section 3.3.

### 3.1 ZERO-MEAN ISOTROPIC GAUSSIAN DISTRIBUTION, THE MAXIMUM UNIFORMITY

As shown in Theorem 1, we provide a theorem that states maximum uniformity could be achieved if learned representations obey zero-mean isotropic gaussian distribution ($\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$).

**Theorem 1.** *For a random variable $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_m)$, and $\mathbf{Z} \in \mathbb{R}^m$, its $l_2$-normalized form $\mathbf{Y} = \mathbf{Z}/\|\mathbf{Z}\|_2$ will uniformly distribute on the surface of the unit hypersphere $\mathcal{S}^{m-1}$.*

*Proof.* To prove the theorem, we would like to first prove Lemma 1 for later use.

**Lemma 1.** *Suppose a random variable $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_m)$ and $\mathbf{U}$ is an arbitrary orthogonal matrix. We define $\mathbf{Y} = \mathbf{Z}/\|\mathbf{Z}\|_2$. Then, $\mathbf{Y}$ is **i**dentically **d**istributed operation as $\mathbf{UY}$, namely,*

$$\mathbf{Y} \overset{id}{\leftrightarrow} \mathbf{UY}$$

See detailed proof in Lemma 1 in Appendix A.

Therefore, $\mathbf{Y}$ is an identically distributed operation as $\mathbf{UY}$ after an arbitrary orthogonal transformation. To conclude that the random variable $\mathbf{Y}$ uniformly distributes on the on the surface of the unit hypersphere $\mathcal{S}^{m-1} = \{\mathbf{y} \in \mathbb{R}^m : \|\mathbf{y}\|_2 = 1\}$, here we use **the proof by contradiction**.

Let us assume the opposite of the above conclusion: $\mathbf{Y}$ does not uniformly distribute on the surface of the unit hypersphere $\mathcal{S}^{m-1}$. In other words, the density of each specified-sized area in $\mathbf{Y}$ is not identical for the unit hypersphere $\mathcal{S}^{m-1}$. The random variable $\mathbf{Y}$ has a continuous density $\rho$. Suppose that for $\mathbf{r}_1, \mathbf{r}_2 \in \mathcal{S}^{m-1}$, $\mathbf{r}_1 \neq \mathbf{r}_2$ and $\rho(\mathbf{r}_1) > \rho(\mathbf{r}_2)$, there exists a radius $\epsilon$ for any $l_2$-norm (also holds for other norms) such that on

$$\mathcal{D}_1 = \{\mathbf{r} \in \mathcal{S}^{m-1} : \|\mathbf{r} - \mathbf{r}_1\|_2 < \epsilon\},$$
$$\mathcal{D}_2 = \{\mathbf{r} \in \mathcal{S}^{m-1} : \|\mathbf{r} - \mathbf{r}_2\|_2 < \epsilon\},$$

we still have

$$\forall \mathbf{r} \in \mathcal{D}_1, \forall \mathbf{s} \in \mathcal{D}_2, \rho(\mathbf{r}) > \rho(\mathbf{s}).$$

Therefore, $P(\mathcal{D}_1) > P(\mathcal{D}_2)$. Since $\mathbf{Y} \overset{id}{\leftrightarrow} \mathbf{UY}$ according to Lemma 1, $\mathcal{D}_2$ can be obtained from $\mathcal{D}_1$ by a orthogonal transformation [1], which implies that $P(\mathcal{D}_1) = P(\mathcal{D}_2)$.

Contradiction! Hence $\rho(\mathbf{r}_1) = \rho(\mathbf{r}_2)$ for $\forall \mathbf{r}_1, \mathbf{r}_2 \in \mathcal{S}^{m-1}$ and $\mathbf{r}_1 \neq \mathbf{r}_2$. Therefore, $\mathbf{Y} = \mathbf{Z}/\|\mathbf{Z}\|_2$ uniformly distributes on the hypersphere $\mathcal{S}^{m-1}$. □

Based on Theorem 1, one could get Corollary 1, which states that a mixture of two independent random variables following zero-mean isotropic gaussian distribution also achieves the maximum uniformity.

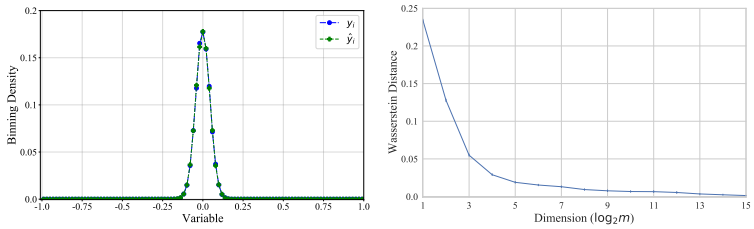
**Corollary 1.** *For a random variable $\mathbf{Z}_1, \mathbf{Z}_2 \in \mathbb{R}^m$ that both follow Gaussian distributions. Namely, $\mathbf{Z}_1 \sim \mathcal{N}(\mathbf{0}, \sigma_1^2\mathbf{I}_m)$, and $\mathbf{Z}_2 \sim \mathcal{N}(\mathbf{0}, \sigma_2^2\mathbf{I}_m)$. Let $\mathbf{Z}$ be a mixture distribution [2] derived from $\mathbf{Z}_1$ and $\mathbf{Z}_2$ with any binary selection probabilities. Its $l_2$-normalized form $\mathbf{Y} = \mathbf{Z}/\|\mathbf{Z}\|_2$ also uniformly distribute on the surface of the unit hypersphere $\mathcal{S}^{m-1}$.*

### 3.2 ON THE $l_2$-NORMALIZED GAUSSIAN DISTRIBUTION $\mathbf{Z}/\|\mathbf{Z}\|_2$

Due to the mixture form discussed in Corollary 1, the ideal distribution (i.e., the mixture of various zero-mean isotropic Gaussian distributions) is not unique. Since one could define different mixtures of zero-mean isotropic Gaussian distributions, each of which might have different norms encapsulated in

[1] Let $\boldsymbol{W}$ be a orthogonal transformation such that $\boldsymbol{W}\boldsymbol{r}_1 = \boldsymbol{r}_2$. $\mathcal{D}_2$ could be obtained by transforming every points from $\mathcal{D}_1$ using orthogonal transformation $\boldsymbol{W}$, namely $\mathcal{D}_2 = \{\boldsymbol{W}\boldsymbol{r} : \boldsymbol{r} \in \mathcal{D}_1\}$,

[2] A mixture distribution is the probability distribution of a random variable that is derived from a collection of other random variables. This could be implemented by first sampling a random variable based on a given probability distribution w.r.t a ratio of each random variable, and then sampling a value based on the selected random variable.




(a) Binning density of $y_i$ and $\hat{y}_i$

(b) Wasserstein distance between the binning density of $y_i$ and $\hat{y}_i$

Figure 2: (a) presents overlapped binning density between $y_i$ and $\hat{y}_i$, indicating $y_i$ and $\hat{y}_i$ have almost the same distribution. In (b), we use Wasserstein distance to measure the distribution distance between $y_i$ and $\hat{y}_i$, and find the distance is converged to zero with the large dimension. More details see in Appendix E.

$\sigma_1, \cdots, \sigma_k$. Therefore, we turn to investigate the **$l2$-normalized form of these zero-mean isotropic Gaussian distributions** [3].

$l2$-normalized form of these zero-mean isotropic Gaussian distributions, denoted as $\mathbf{Y}$, obeys Uniform distribution on the surface of the unit hypersphere $\mathbf{Y} \sim U(\mathcal{S}^{m-1})$. However, it is quite difficult to deal with this distribution in Euclidean space which we commonly use in a neural network. Fortunately, in this paper, we find a fundamental conclusion about $\mathbf{Y}$ in the Euclidean space, as stated in Theorem 2:

**Theorem 2.** *For a random variable $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_m)$, and $\mathbf{Z} \in \mathbb{R}^m$, for the $l_2$-normalized form $\mathbf{Y} = \mathbf{Z}/\|\mathbf{Z}\|_2$, its mean and covariance matrix can be formulated as follows.*

$$\boldsymbol{\mu} = \mathbf{0}, \quad \boldsymbol{\Sigma} = \frac{1}{m}\mathbf{I}_m,$$

*See the proof in Appendix B.*

Therefore, this inspires us to make an **assumption** that $l_2$-normalized zero-mean isotropic Gaussian distribution (denoted $\mathbf{Y}$) follows, or at least is close to, an approximated Gaussian distribution $\mathcal{N}(\mathbf{0}, \frac{1}{m}\mathbf{I}_m)$ ($m$ is the dimension of the variable). Note that $\mathcal{N}(\mathbf{0}, \frac{1}{m}\mathbf{I}_m)$ enjoys the merits of uniqueness (a proper distribution to design uniformity metric), and might be used as an approximated distribution for $\mathbf{Y}$.

Without losing any generality, we analyze an arbitrary dimension of $\mathbf{Y}$. The distribution of the $i$-th dimension of $\mathbf{Y}$, as denoted as a random variable $y_i$ is visualized in Fig. 2(a). Observe that the binning density of $y_i$ is almost overlapped with $\hat{y}_i \sim \mathcal{N}(0, \frac{1}{m}\sigma^2)$, and the Wasserstein Distance (as defined in Appendix E) between $y_i$ and $\hat{y}_i$ is converged to zero with the large dimension, indicating $y_i$ and $\hat{y}_i$ have almost the same distribution. We find the assumption empirically works well in this work, and we leave theoretical evidence for future work.

### 3.3 Wasserstein Distance, a New Metric for Uniformity

In this section, we propose to use the distance between the distribution of learned representations and its ideal distribution $\mathcal{N}(\mathbf{0}, \frac{1}{m}\mathbf{I}_m)$ as a uniformity metric. Specially, we collect a set of data vectors from learned representations, i.e., $\{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$. To keep consistency with the ideal distribution, we adopt $l_2$-normalized vectors, and calculate the mean and covariance matrix as follows:

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{z}_i/\|\mathbf{z}_i\|, \quad \boldsymbol{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{z}_i/\|\mathbf{z}_i\| - \boldsymbol{\mu})^T (\mathbf{z}_i/\|\mathbf{z}_i\| - \boldsymbol{\mu}), \tag{3}$$

[3] Most recent self-supervised representation learning approaches learn representations with a $l_2$ norm constraint (Zbontar et al., 2021; Wang & Isola, 2020; Chen & He, 2021; Grill et al., 2020; Chen et al., 2020), restricting the output representations to the surface of unit hypersphere, i.e., the $l2$-normalized representation ($\mathbf{Y} \stackrel{def}{=} \mathbf{Z}/\|\mathbf{Z}\|_2$) should be on the surface of the unit hypersphere $\mathcal{S}^{m-1}$. This suggests that directions of learned representation vectors (instead of the absolute amplitude of elements in the vectors) matter when capturing the semantic information of instances.

Where $\boldsymbol{\mu} \in \mathbb{R}^m$, $\boldsymbol{\Sigma} \in \mathbb{R}^{m \times m}$, and $m$ is the dimension size of vectors. To facilitate the calculation of distribution distance, we apply a Gaussian hypothesis to learned representations $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Based on this assumption, we employ Wasserstein distance [4], a well-known distribution distance, to calculate the distance between two distributions, which takes the minimum cost of turning one pile into the other when viewing each distribution as a unit amount of earth/soil. Despite its complexity, Wasserstein distance over Gaussian distributions is easy to calculate as illustrated in Theorem 3 in Appendix D. We instantiate Equation 11 with the distribution of learned representations and ideal distribution. Then, an *uniformity* metric via Wasserstein distance can be formulated as follows:

$$\mathcal{W}_2 \triangleq \sqrt{\|\boldsymbol{\mu}\|_2^2 + 1 + Tr(\boldsymbol{\Sigma}) - \frac{2}{\sqrt{m}} Tr(\boldsymbol{\Sigma}^{1/2})}, \tag{4}$$

The smaller $\mathcal{W}_2$, the more nearby the distribution of learned representations to the ideal distribution, indicating the larger *uniformity* of representations. Besides its usefulness in collapse analysis, our proposed uniformity metric can be also used as an additional loss for various existing self-supervised methods due to its differentiability during the backward pass. One difference from the *uniformity metric* is that the mean and covariance matrix in Equation 3 is calculated by batch data during the training phase.

## 4 On Uniformity Metrics

In this section, we first introduce the desirable properties (called 'Desiderata') of any well-defined uniformity metric in Sec. 4.1. In Sec.4.2 and Sec. 4.3, we compare the proposed uniformity metric $-\mathcal{W}_p$ with existing uniformity metric $-\mathcal{L}_\mathcal{U}$ theoretically and empirically respectively.

### 4.1 Desiderata of Uniformity

A uniformity metric is a function to map a set of learned representations (typically dense vectors) to a uniformity indicator (typically a real number).

$$\mathcal{U} : \{\mathbb{R}^m\}^n \to \mathbb{R}, \tag{5}$$

$\mathcal{D} \in \{\mathbb{R}^m\}^n$ is a set of learned vectors ($\mathcal{D} = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$), each vector is the feature representation of a instance, $\mathbf{z}_i \in \mathbb{R}^m$. In this section, we formally define five desiderata (i.e., desirable properties) for any uniformity metrics.

Intuitively, *uniformity* should be invariant to the permutation of instances, since it does not affect the distribution.

**Property 1.** ***Instance Permutation Constraint (IPC)***

$$\mathcal{U}(\pi(\mathcal{D})) = \mathcal{U}(\mathcal{D}), \tag{6}$$

$\pi$ is an instance permutation operator that changes the order of representations.

The *uniformity* should be invariant when all representations are re-scaled, since modern machine learning tends to use directions of learned representation vectors (instead of the absolute amplitude of elements in the vectors) to capture the semantic information of instances. For example, most recent self-supervised representation learning approaches learn representations with a $l_2$ norm constraint (Zbontar et al., 2021; Wang & Isola, 2020; Chen & He, 2021; Grill et al., 2020; Chen et al., 2020), restricting the output representations to the surface of unit hypersphere, i.e., $\mathcal{D}^s = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_n\}$, and $\mathbf{s}_i = \mathbf{z}_i / \|\mathbf{z}_i\|_2$ is on the surface of the unit hypersphere $\mathcal{S}^{m-1}$.

**Property 2.** ***Instance Scaling Constraint (ISC)***

$$\mathcal{U}(\{\lambda_1 \mathbf{z}_1, \lambda_2 \mathbf{z}_2, ..., \lambda_n \mathbf{z}_n\}) = \mathcal{U}(\mathcal{D}), \quad \forall \lambda_i \in \mathbb{R}^+, \tag{7}$$

The *uniformity* is invariant when instances are cloned, since the cloning operator cannot change the original distribution density.

[4] We also discuss using other distribution distances as uniformity metrics, such as Kullback-Leibler Divergence and Bhattacharyya Distance over Gaussian distribution. See more details in Appendix D

**Property 3.** ***Instance Cloning Constraint (ICC)***

$$\mathcal{U}(\mathcal{D} \cup \mathcal{D}) = \mathcal{U}(\mathcal{D}), \tag{8}$$

$\cup$ is the union of two sets that can achieve instance cloning, $\mathcal{D} \cup \mathcal{D} = \{\mathbf{z}_1, \cdots, \mathbf{z}_n, \mathbf{z}_1, \cdots, \mathbf{z}_n\}$.

The *uniformity* decreases when cloning features for each instance, since the feature-level clone will bring some redundancy, leading to dimension collapse (Zbontar et al., 2021; Bardes et al., 2022).

**Property 4.** ***Feature Cloning Constraint (FCC)***

$$\mathcal{U}(\mathcal{D} \oplus \mathcal{D}) \leq \mathcal{U}(\mathcal{D}), \tag{9}$$

$\oplus$ is an feature-level concatenation operator that can achieve feature cloning as $\mathcal{D} \oplus \mathcal{D} = \{\mathbf{z}_1 \oplus \mathbf{z}_1, \mathbf{z}_2 \oplus \mathbf{z}_2, ..., \mathbf{z}_n \oplus \mathbf{z}_n\}$, and where $\mathbf{z}_i \oplus \mathbf{z}_i = [z_{i1}, \cdots, z_{im}, z_{i1}, \cdots, z_{im}]^T \in \mathbb{R}^{2m}$. $\mathcal{U}(\mathcal{D} \oplus \mathcal{D}) = \mathcal{U}(\mathcal{D})$ if and only if $\mathbf{z}_1 = \mathbf{z}_2 = ... = \mathbf{z}_n = \mathbf{0}^m$.

The *uniformity* decreases when adding constant features for each instance, since it introduces uninformative features and results in some collapsed dimensions.

**Property 5.** ***Feature Baby Constraint (FBC)***

$$\mathcal{U}(\mathcal{D} \oplus \mathbf{0}^k) \leq \mathcal{U}(\mathcal{D}), \quad k \in \mathbb{N}^+, \tag{10}$$

$\mathcal{D} \oplus \mathbf{0}^k = \{\mathbf{z}_1 \oplus \mathbf{0}^k, \mathbf{z}_2 \oplus \mathbf{0}^k, ..., \mathbf{z}_n \oplus \mathbf{0}^k\}$, and $\mathbf{z}_i \oplus \mathbf{0}^k = [z_{i1}, z_{i2}, ..., z_{im}, 0, 0, ..., 0]^T \in \mathbb{R}^{m+k}$. $\mathcal{U}(\mathcal{D} \oplus \mathbf{0}^k) \leq \mathcal{U}(\mathcal{D})$ if and only if $\mathbf{z}_1 = \mathbf{z}_2 = ... = \mathbf{z}_n = \mathbf{0}^m$.

### 4.2 Examining desiderata of Uniformity

In this section, we employ our designed desiderata in Section 4 as criterion to conduct theoretical analysis for two metrics $-\mathcal{L}_\mathcal{U}$ in Equation 2 and $-\mathcal{W}_2$ in Equation 4. All theoretical analysis results are summarized in Table 1 [5]. See detailed proofs (Lemma 3, 4, 5, 6, 7, and 8) in Appendix C, it shows that the existing uniformity metric (Wang & Isola, 2020) ($-\mathcal{L}_\mathcal{U}$) violates all three constraints while our proposed uniformity metric ($-\mathcal{W}_2$) could satisfy all of them. This demonstrates that our proposed metric is a more reasonable metric to measure the *uniformity* from the theoretical perspective.

Table 1: Theoretical analysis results on the two metrics w.r.t the five desiderata constraints.

| Metric | *Property IPC* | *Property ISC* | *Property ICC* | *Property FCC* | *Property FBC* |
|---|---|---|---|---|---|
| $-\mathcal{L}_\mathcal{U}$ | ✔ | ✔ | ✗[6] | ✗ | ✗ |
| $-\mathcal{W}_2$ | ✔ | ✔ | ✔ | ✔ | ✔ |

Interestingly, proposed metric $-\mathcal{W}_2$ satisfies *FBC Property* while the baseline metric $-\mathcal{L}_\mathcal{U}$ does not. This opens a new angle to explain the advantage of our proposed metric $-\mathcal{W}_2$ from the dimensional collapse perspective. Specially, the larger $k$ would bring the more serious dimensional collapse for $\mathcal{D} \oplus \mathbf{0}^k$ than $\mathcal{D}$. However, $-\mathcal{L}_\mathcal{U}$ fails to identify the more serious dimensional collapse due to $-\mathcal{L}_\mathcal{U}(\mathcal{D} \oplus \mathbf{0}^k) = -\mathcal{L}_\mathcal{U}(\mathcal{D})$ according to Lemma 5. On the contrary, our proposed metric is sensitive to the dimensional collapse as $-\mathcal{W}_2(\mathcal{D} \oplus \mathbf{0}^k) < -\mathcal{W}_2(\mathcal{D})$ based on Lemma 7.

### 4.3 Empirical Analysis via Synthetic Data

**Correlation between $\mathcal{L}_\mathcal{U}$ and $\mathcal{W}_2$.** We employ synthetic experiments to study uniformity metrics. In detail, we manually sample 50000 data vectors from different distributions, such as standard Gaussian distribution $\mathcal{N}(\mathbf{0}, I)$, uniform Distribution $U(\mathbf{0}, \mathbf{1})$, the mixture of Gaussian, etc. Based on these data vectors, we estimate the uniformity of different distributions by two metrics. As shown in Fig. 3, standard Gaussian distribution achieves the minimum values by both $\mathcal{W}_2$ and $\mathcal{L}_\mathcal{U}$, which indicates that standard Gaussian distribution could achieve larger uniformity than other distributions. This empirical result is consistent with Theorem 1 that standard Gaussian distribution achieves the maximum uniformity.

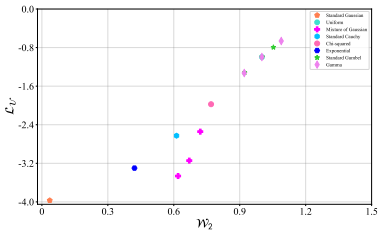



Figure 3: Uniformity analysis on distributions via two metrics.

[5] To simplify analysis, we exclude the special case $\mathbf{z}_1 = \mathbf{z}_2 = ... = \mathbf{z}_n$ as this case is a trivial or undesired solution for the self-supervised representation learning.

[6] $-\mathcal{L}_\mathcal{U}$ could satisfy *Property ICC* if and only if $\mathbf{z}_1 = \mathbf{z}_2 = ... = \mathbf{z}_n$

**On the Dimensional Collapse.** To synthesize data with various specified degrees of dimensional collapse, we concatenate the *zero vectors* (i.e., they are full dimensional collapse) with *sampled data vectors* from the standard Gaussian distribution (i.e., ideal uniformity without collapse). The percentage of zero-value dimensions of the concatenated vectors is $\eta$ while that of non-zero vectors is $1 - \eta$. As shown in Fig. 5(a) and Fig. 5(b) in Appendix F, $\mathcal{W}_2$ is capable of capturing salient sensitivity to collapse level, while $\mathcal{L}_\mathcal{U}$ keeps almost no change even in $80\%$ collapse level, indicating $\mathcal{L}_\mathcal{U}$ is insensitive to the dimensional collapse.

Interestingly, as visualized in Fig. 6 in Appendix F, $\mathcal{L}_\mathcal{U}$ becomes indistinguishable with different degrees of dimension collapse ($\eta = 25\%, 50\%,$ and$75\%$) when the dimension $m$ becomes large (e.g., $m \geq 2^8$). On the contrary, our proposed $\mathcal{W}_2$ is constant to the dimension number under a specific degree of dimension collapse; $\mathcal{W}_2$ only depends on the degree of dimension collapse and is independent of the dimension number. In summary, our proposed metric $\mathcal{W}_2$ is a more reasonable metric to measure the *uniformity* than the existing one $\mathcal{L}_\mathcal{U}$ from an empirical perspective.

## 5 EXPERIMENTS

In this section, we impose the proposed uniformity metric as an auxiliary loss term for various existing self-supervised methods, and conduct experiments on CIFAR-10 and CIFAR-100 datasets to demonstrate its effectiveness. Codes implemented in Pytorch will be released.

**Models** We conduct experiments on a series of self-supervised representation learning models: (i) AlignUniform (Wang & Isola, 2020), whose loss objective consists of an alignment objective and a uniform objective. (ii) three contrastive methods, i.e., SimCLR (Chen et al., 2020), MoCo (He et al., 2020), and NNCLR (Dwibedi et al., 2021). (iii) two asymmetric models, i.e., BYOL (Grill et al., 2020) and SimSiam (Chen & He, 2021). (iv) two methods via redundancy reduction, i.e., BarlowTwins (Zbontar et al., 2021) and Zero-CL (Zhang et al., 2022b). To study the behavior of proposed Wasserstein distance in the self-supervised representation learning, we impose it as an auxiliary loss term to the following models: MoCo v2, BYOL, BarlowTwins, and Zero-CL. To facilitate better use of Wasserstein distance, we also design a linear decay for weighting Wasserstein distance during the training phase, i.e., $\alpha_t = \alpha_{max} - t * (\alpha_{max} - \alpha_{min})/T$, where $t$, $T$, $\alpha_{max}$ , $\alpha_{min}$, $\alpha_t$ are current epoch, maximum epochs, maximum weight, minimum weight, and current weight, respectively. More detailed parameter setting see in Table 3 in Appendix G.

**Setting** To make a fair comparison, we conduct all experiments on a single 1080 GPU. Also, we adopt the same network architecture for all models, i.e., ResNet-18 (He et al., 2016) as the encoder, a three-layer MLP as the projector, and a three-layer MLP as the projector, respectively. Besides, We use LARS optimizer (You et al., 2017) with a base learning rate $0.2$, along with a cosine decay learning rate schedule (Loshchilov & Hutter, 2017) for all models. We evaluate all models under a linear evaluation protocol. In specific, models are pre-trained for 500 epochs and evaluated by adding a linear classifier and training the classifier for 100 epochs while keeping the learned representations unchanged. We also deploy the same augmentation strategy for all models, which is the composition of a series of data augmentation operations, such as color distortion, rotation, and cutout. Following (da Costa et al., 2022), we set temperature $t = 0.2$ for all contrastive methods. As for MoCo (He et al., 2020) and NNCLR (Dwibedi et al., 2021) that require an extra queue to save negative samples, we set the queue size to $2^{12}$.

**Metrics** We evaluate the above methods from two perspectives: one is linear evaluation accuracy measured by Top-1 accuracy (Acc@1) and Top-5 accuracy (Acc@5); another is representation capacity. According to (Arora et al., 2019; Wang & Isola, 2020), alignment and uniformity are the two most important properties to evaluate self-supervised representation learning. We use two metrics $\mathcal{L}_\mathcal{U}$ and $\mathcal{W}_2$ to measure the uniformity, and a metric $\mathcal{A}$ to measure the alignment between the positive pairs (Wang & Isola, 2020). More details about the alignment metric see in Appendix H.

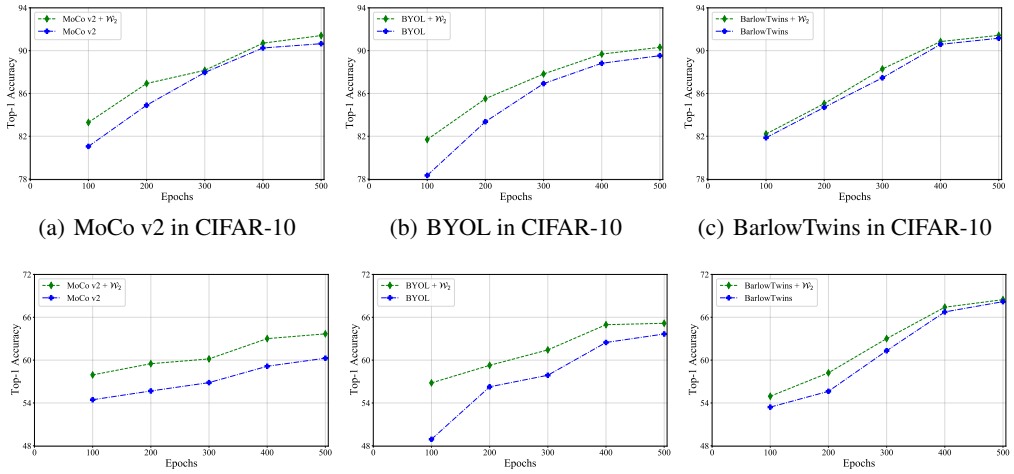
**Main Results** As shown in Tab. 2 We could observe that by imposing $\mathcal{W}_2$ as an additional loss it consistently improves the performance than that without the loss. Interestingly, although it slightly harms alignment, it usually results in improvement in uniformity and finally leads to better accuracy. This demonstrates the effectiveness of $\mathcal{W}_2$ as a uniformity metric. Note imposing an additional loss

Table 2: Main comparison on CIFAR-10 and CIFAR-100 datasets. Proj. and Pred. are the hidden dimension in projector and predictor. $\uparrow$ and $\downarrow$ mean gains and losses, respectively.

| Methods | Proj. | Pred. | CIFAR-10 | | | | | CIFAR-100 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Acc@1↑ | Acc@5↑ | $\mathcal{W}_2 \downarrow$ | $\mathcal{L}_{\mathcal{U}} \downarrow$ | $\mathcal{A} \downarrow$ | Acc@1↑ | Acc@5↑ | $\mathcal{W}_2 \downarrow$ | $\mathcal{L}_{\mathcal{U}} \downarrow$ | $\mathcal{A} \downarrow$ |
| SimCLR | 256 | ✗ | 89.85 | 99.78 | 1.04 | -3.75 | 0.47 | 63.43 | 88.97 | 1.05 | -3.75 | 0.50 |
| NNCLR | 256 | 256 | 87.46 | 99.63 | 1.23 | -3.12 | 0.38 | 54.90 | 83.81 | 1.23 | -3.18 | 0.43 |
| SimSiam | 256 | 256 | 86.71 | 99.67 | 1.19 | -3.33 | 0.39 | 56.10 | 84.34 | 1.21 | -3.29 | 0.42 |
| AlignUniform | 256 | ✗ | 90.37 | 99.76 | 0.94 | -3.82 | 0.51 | 65.08 | 90.15 | 0.95 | -3.82 | 0.53 |
| MoCo v2 | 256 | ✗ | 90.65 | 99.81 | 1.06 | -3.75 | 0.51 | 60.27 | 86.29 | 1.07 | -3.60 | 0.46 |
| MoCo v2 + $\mathcal{W}_2$ | 256 | ✗ | 91.41 $\uparrow_{0.76}$ | 99.68 | 0.33 $\uparrow_{0.73}$ | -3.84 | 0.63 $\downarrow_{0.12}$ | 63.68 $\uparrow_{3.41}$ | 88.48 | 0.28 $\uparrow_{0.79}$ | -3.86 | 0.66 $\downarrow_{0.20}$ |
| BYOL | 256 | 256 | 89.53 | 99.71 | 1.21 | -2.99 | **0.31** | 63.66 | 88.81 | 1.20 | -2.87 | **0.33** |
| BYOL + $\mathcal{W}_2$ | 256 | 256 | 90.31 $\uparrow_{0.78}$ | 99.77 | 0.38 $\uparrow_{0.83}$ | -3.90 | 0.65 $\downarrow_{0.34}$ | 65.16 $\uparrow_{1.50}$ | 89.25 | 0.36 $\uparrow_{0.84}$ | -3.91 | 0.69 $\downarrow_{0.36}$ |
| BarlowTwins | 256 | ✗ | 91.16 | 99.80 | 0.22 | -3.91 | 0.75 | 68.19 | 90.64 | 0.23 | -3.91 | 0.75 |
| BarlowTwins + $\mathcal{W}_2$ | 256 | ✗ | **91.43** $\uparrow_{0.27}$ | 99.78 | 0.19 $\uparrow_{0.03}$ | -3.92 | 0.76 $\downarrow_{0.01}$ | 68.47 $\uparrow_{0.28}$ | 90.64 | 0.19 $\uparrow_{0.04}$ | -3.91 | 0.79 $\downarrow_{0.04}$ |
| Zero-CL | 256 | ✗ | 91.35 | 99.74 | 0.15 | **-3.94** | 0.70 | 68.50 | 90.97 | 0.15 | -3.93 | 0.75 |
| Zero-CL + $\mathcal{W}_2$ | 256 | ✗ | 91.42 $\uparrow_{0.07}$ | **99.82** | **0.14** $\uparrow_{0.01}$ | **-3.94** | 0.71 $\downarrow_{0.01}$ | **68.55** $\uparrow_{0.05}$ | **91.02** | **0.14** $\uparrow_{0.01}$ | **-3.94** | 0.76 $\downarrow_{0.01}$ |

during training does not affect the training or inference efficiency; therefore, adding $\mathcal{W}_2$ as loss is beneficial without any tangible costs.

**Convergence Analysis** We test the Top-1 accuracy of these models on CIFAR-10 and CIFAR-100 via linear evaluation protocol (as described in Section 5) when training them in different epochs. As shown in Fig. 4. By imposing $\mathcal{W}_2$ as an additional loss for these models, it converges faster than the raw models, especially for MoCo v2 and BYOL with serious *collapse problem*.




(a) MoCo v2 in CIFAR-10 (b) BYOL in CIFAR-10 (c) BarlowTwins in CIFAR-10

(d) MoCo v2 in CIFAR-100 (e) BYOL in CIFAR-100 (f) BarlowTwins in CIFAR-100

Figure 4: Visualization on Convergence.

## 6 Conclusion

In this paper, to better understand the *collapse* problem in self-supervised representation learning, we present a systematical analysis of the collapse degree of representations. To achieve a quantifiable analysis of the dimensional collapse, we propose to use the Wasserstein distance between the distribution of learned representations and the ideal distribution as the metric of *uniformity*. Then, we design five desirable constraints for ideal uniformity metrics to make theoretical comparisons between our proposed metric and the existing one. Besides, we conduct synthetic experiments which demonstrate the proposed uniformity metric is capable to deal with the dimensional collapse while the existing one is insensitive. Finally, we apply our proposed metric in the practical scenarios, and impose the proposed *uniformity* metric as an auxiliary loss term for various existing self-supervised methods, which consistently improves the downstream performance. In future work, the proposed metric could also be used to conduct collapse analysis beyond self-supervised representation learning, e.g., sentence representations learned from BERT in natural language processing.

## REFERENCES


Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. In *ICML*, 2019.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022.

A. Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*, 1943.

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a siamese time delay neural network. In *NeurIPS*, 1994.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021.

Henry Cohn and Abhinav Kumar. Universally optimal distribution of points on spheres. *Journal of the American Mathematical Society*, 2007.

Victor Guilherme Turrisi da Costa, Enrico Fini, Moin Nabi, N. Sebe, and Elisa Ricci. Solo-learn: A library of self-supervised methods for visual representation learning. *JMLR*, 2022.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *ICCV*, 2021.

Aleksandr Ermolov, Aliaksandr Siarohin, E. Sangineto, and N. Sebe. Whitening for self-supervised representation learning. In *ICML*, 2021.

Jean-Bastien Grill, Florian Strub, Florent Altch'e, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. Supervised contrastive learning for pre-trained language model fine-tuning. In *ICLR*, 2021.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

Tianyu Hua, Wenxiao Wang, Zihui Xue, Sucheng Ren, Yue Wang, and Hang Zhao. On feature decorrelation in self-supervised learning. In *ICCV*, 2021.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. In *ICLR*, 2022.

Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven C. H. Hoi. Prototypical contrastive learning of unsupervised representations. In *ICLR*, 2021.

David Lindley and Solomon Kullback. Information theory and statistics. *Journal of the American Statistical Association*, 54:825, 1959.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2017.

Ingram Olkin and Friedrich Pukelsheim. The distance between two random vectors with given dispersion matrices. *Linear Algebra and its Applications*, 48:257–263, 1982.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *ArXiv*, 2018.

Dong Wang, Ning Ding, Pijian Li, and Haitao Zheng. Cline: Contrastive learning with semantic negative examples for natural language understanding. In *ACL*, 2021a.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, 2020.

Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *CVPR*, 2021b.

Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Zhenguo Li, and Ping Luo. Detco: Unsupervised contrastive learning for object detection. In *ICCV*, 2021.

Ceyuan Yang, Zhirong Wu, Bolei Zhou, and Stephen Lin. Instance localization for self-supervised detection pretraining. In *CVPR*, 2021.

Yang You, Igor Gitman, and Boris Ginsburg. Scaling sgd batch size to 32k for imagenet training. *ArXiv*, 2017.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *ICML*, 2021.

Chaoning Zhang, Kang Zhang, Chenshuang Zhang, Trung X. Pham, Chang D. Yoo, and In So Kweon. How does simsiam avoid collapse without negative samples? a unified understanding with self-supervised contrastive learning. In *ICLR*, 2022a.

Shaofeng Zhang, Feng Zhu, Junchi Yan, Rui Zhao, and Xiaokang Yang. Zero-CL: Instance and feature decorrelation for negative-free symmetric contrastive learning. In *ICLR*, 2022b.

Xiangyu Zhao, Raviteja Vemulapalli, P. A. Mansfield, Boqing Gong, Bradley Green, Lior Shapira, and Ying Wu. Contrastive learning for label efficient semantic segmentation. In *ICCV*, 2021.

## A PROOF OF LEMMA 1

To prove Lemma 1, we have to first prove Lemma 2 shown as below:

**Lemma 2.** *Suppose a random variable* $\mathbf{Z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ *(*$\mathbf{Z} \in \mathbb{R}^m$*) and* $\mathbf{U} \in \mathbb{R}^{m \times m}$ *is an arbitrary orthogonal matrix. We define* $\hat{\mathbf{Z}} = \mathbf{U}\mathbf{Z}$*. Then,* $\hat{\mathbf{Z}} \sim \mathcal{N}(\mathbf{U}\boldsymbol{\mu}, \mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^T)$.

*Proof.* $\mathbf{Z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, and its probability density function (pdf) can be written as:

$$f_{\mathbf{Z}}(\mathbf{z}) = \frac{1}{(2\pi)^{m/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\{-\frac{1}{2}(\mathbf{z}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{z}-\boldsymbol{\mu})\}$$

where $|\boldsymbol{\Sigma}| = \det \boldsymbol{\Sigma}$ is the determinant of the covariance $\boldsymbol{\Sigma}$. As $\hat{\mathbf{Z}} = \mathbf{U}\mathbf{Z}$, then the pdf of the $\hat{\mathbf{Z}}$ can be formulated as:

$$\begin{aligned} f_{\hat{\mathbf{Z}}}(\hat{\mathbf{z}}) &= \frac{1}{|\mathbf{U}|} f_{\mathbf{Z}}(\mathbf{U}^{-1}\hat{\mathbf{z}}) \\ &= \frac{1}{(2\pi)^{m/2}|\mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^T|^{1/2}} \exp\{-\frac{1}{2}(\mathbf{U}^{-1}\hat{\mathbf{z}}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{U}^{-1}\hat{\mathbf{z}}-\boldsymbol{\mu})\} \\ &= \frac{1}{(2\pi)^{m/2}|\mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^T|^{1/2}} \exp\{-\frac{1}{2}(\hat{\mathbf{z}}-\mathbf{U}\boldsymbol{\mu})^T(\mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^T)^{-1}(\hat{\mathbf{z}}-\mathbf{U}\boldsymbol{\mu})\} \end{aligned}$$

therefore, $\hat{\mathbf{Z}} \sim \mathcal{N}(\mathbf{U}\boldsymbol{\mu}, \mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^T)$. □

Now we start to prove Lemma 1, which states 'the $l2$-normalized distribution of a random variable $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_m)$ is invariant to arbitrary orthogonal transformation ($\mathbf{U}$). In other words, let $\mathbf{Y} = \mathbf{Z}/\|\mathbf{Z}\|_2$. $\mathbf{Y}$ is **i**dentically **d**istributed operation as $\mathbf{U}\mathbf{Y}$'

*Proof.* As $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_m)$, for any orthogonal matrix $\mathbf{U}$:

$$\begin{aligned} \hat{\mathbf{Z}} = \mathbf{U}\mathbf{Z} &\sim \mathcal{N}(\mathbf{U}\mathbf{0}, \sigma^2\mathbf{U}\mathbf{I}_m\mathbf{U}^T), && (\text{ by Lemma 2}) \\ &\sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_m) && (\mathbf{U}\mathbf{0} = \mathbf{0}, \mathbf{U}\mathbf{I}_m\mathbf{U}^T = \mathbf{U}\mathbf{U}^T = \mathbf{I}_m), \end{aligned}$$

Since the random variable $\hat{\mathbf{Z}} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_m)$, $\hat{\mathbf{Z}}$ is identically distributed with the random variable $\mathbf{Z}$. We denote **i**dentically **d**istributed operation as $\hat{\mathbf{Z}} \overset{id}{\leftrightarrow} \mathbf{Z}$. For the $l_2$-normalized variables:

$$\mathbf{Y} = \mathbf{Z}/\|\mathbf{Z}\|_2, \quad \hat{\mathbf{Y}} = \hat{\mathbf{Z}}/\|\hat{\mathbf{Z}}\|_2, \quad \hat{\mathbf{Y}} \overset{id}{\leftrightarrow} \mathbf{Y}.$$

Since $\|\mathbf{U}\mathbf{Z}\|_2 = \sqrt{(\mathbf{U}\mathbf{Z})^T(\mathbf{U}\mathbf{Z})} = \sqrt{\mathbf{Z}^T\mathbf{U}^T\mathbf{U}\mathbf{Z}} = \sqrt{\mathbf{Z}^T\mathbf{Z}} = \|\mathbf{Z}\|_2$,

$$\hat{\mathbf{Y}} = \frac{\mathbf{U}\mathbf{Z}}{\|\mathbf{U}\mathbf{Z}\|_2} = \frac{\mathbf{U}\mathbf{Z}}{\|\mathbf{Z}\|_2} = \mathbf{U}\mathbf{Y},$$

Therefore, $\mathbf{Y} \overset{id}{\leftrightarrow} \mathbf{U}\mathbf{Y}$ for any orthogonal transformation $\mathbf{U}$. Finally, we have Lemma 1 proved. □

## B PROOF OF THE THEOREM 2

*Proof.* $\mathbf{Z} = [z_1, z_2, \cdots, z_m] \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_m)$, and its probability density function (pdf) can be written as:

$$f_{\mathbf{Z}}(\mathbf{z}) = \frac{1}{(2\pi)^{m/2}|\sigma^2\mathbf{I}_m|^{1/2}} \exp\{-\frac{1}{2}\mathbf{z}^T(\sigma^2\mathbf{I}_m)^{-1}\mathbf{z}\} = \frac{1}{(2\pi\sigma^2)^{m/2}} \exp\{-\frac{1}{2}\sum_{i=1}^{m} z_i^2/\sigma^2\},$$

We denote $\mathbf{Y} = [y_1, y_2, \cdots, y_m]$. Then the mean of $i$-th variable $y_i$ can be written as below:

$$\mathbb{E}[y_i] = \int_{z_1}\int_{z_2}\int_{\dots}\int_{z_m} \frac{z_i}{\sqrt{\sum_i^m z_i^2}} \frac{1}{(2\pi\sigma^2)^{m/2}} \exp\{-\frac{1}{2}\sum_{i=1}^{m} z_i^2/\sigma^2\} dz_1 dz_2 \cdots dz_m,$$

As $\frac{z_i}{\sqrt{\sum_i^m z_i^2}}$ is an odd function, $\mathbb{E}[y_i] = 0$, and we further conclude $\boldsymbol{\mu} = \mathbb{E}[\mathbf{Y}] = \mathbf{0}$. We also derive the covariance matrix of $\mathbf{Y}$ according to its definition as below:

$$\boldsymbol{\Sigma} = \mathbb{E}[(\mathbf{Y} - \mathbb{E}[\mathbf{Y}])(\mathbf{Y} - \mathbb{E}[\mathbf{Y}])^T] = \begin{pmatrix} \mathbb{E}[y_1^2] & \mathbb{E}[y_1 y_2] & \cdots & \mathbb{E}[y_1 y_m] \\ \mathbb{E}[y_2 y_1] & \mathbb{E}[y_2^2] & \cdots & \mathbb{E}[y_2 y_m] \\ \cdots & \cdots & \cdots & \cdots \\ \mathbb{E}[y_m y_1] & \mathbb{E}[y_m y_2] & \cdots & \mathbb{E}[y_m^2] \end{pmatrix}$$

Then $\mathbb{E}[y_i y_j]$ ($\forall i \neq j$) can be formulated as follows:

$$\mathbb{E}[y_i y_j] = \int_{z_1} \int_{z_2} \int_{...} \int_{z_m} \frac{z_i}{\sqrt{\sum_i^m z_i^2}} \frac{z_j}{\sqrt{\sum_i^m z_i^2}} \frac{1}{(2\pi\sigma^2)^{m/2}} \exp\{-\frac{1}{2}\sum_{i=1}^m z_i^2/\sigma^2\} dz_1 dz_2 \cdots dz_m,$$

As $\frac{z_i}{\sqrt{\sum_i^m z_i^2}} \frac{z_j}{\sqrt{\sum_i^m z_i^2}}$ is an odd function, $\mathbb{E}[y_i y_j] = 0$ ($\forall i \neq j$). In terms of diagonal elements in $\boldsymbol{\Sigma}$, we employ the symmetry to conclude $\mathbb{E}[y_1^2] = \mathbb{E}[y_2^2] = \cdots = \mathbb{E}[y_m^2]$. Based on this principle, we conclude $\mathbb{E}[y_i^2] = \frac{1}{m}$ via below equations:

$$\mathbb{E}[\sum_i^m y_i^2] = m\mathbb{E}[y_i^2], \quad \mathbb{E}[\sum_i^m y_i^2] = \mathbb{E}[\frac{\sum_i^m z_i^2}{\sum_i^m z_i^2}] = 1,$$

Therefore, $\boldsymbol{\Sigma} = \frac{1}{m}\mathbf{I}_m$. □

## C Proof of the Desiderata

To make theoretical comparison between our proposed uniformity metric $-\mathcal{W}_2$ and baseline metric $-\mathcal{L}_\mathcal{U}$, we provide mathematical proof to verify whether these metrics satisfy designed desiderata in Section 4.1. In terms of *Property IPC* and *Property ISC*, we can directly use their definition to demonstrate both two metrics satisfy the two constraints. Therefore, we focus more attention on other three constraints. Detailed proof can be seen in Lemma 3, Lemma 4, Lemma 5, Lemma 6, Lemma 7 and Lemma 8.

**Lemma 3.** *Suppose a set of random variables $\mathcal{D} = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$ ($\mathbf{z}_i \in \mathbb{R}^m$), and a function as follows:*

$$\mathcal{L}_\mathcal{U}(\mathcal{D}) \triangleq \log \frac{1}{n(n-1)/2} \sum_{i=2}^n \sum_{j=1}^{i-1} e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2},$$

*Then $\mathcal{L}_\mathcal{U}(\mathcal{D} \cup \mathcal{D}) \geq \mathcal{L}_\mathcal{U}(\mathcal{D})$. $\mathcal{L}_\mathcal{U}(\mathcal{D} \cup \mathcal{D}) = \mathcal{L}_\mathcal{U}(\mathcal{D})$ if and only if $\mathbf{z}_1 = \mathbf{z}_2 = ... = \mathbf{z}_n$.*

*Proof.*

$$\begin{aligned} \mathcal{L}_\mathcal{U}(\mathcal{D} \cup \mathcal{D}) &\triangleq \log \frac{1}{2n(2n-1)/2} (4\sum_{i=2}^n \sum_{j=1}^{i-1} e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2} + \sum_{i=1}^n e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_i}{\|\mathbf{z}_i\|}\|_2^2}) \\ &= \log \frac{1}{2n(2n-1)/2} (4\sum_{i=2}^n \sum_{j=1}^{i-1} e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2} + n), \end{aligned}$$

We set $G = \sum_{i=2}^n \sum_{j=1}^{i-1} e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2}$, and then we have:

$$G = \sum_{i=2}^n \sum_{j=1}^{i-1} e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2} \leq \sum_{i=2}^n \sum_{j=1}^{i-1} e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_i}{\|\mathbf{z}_i\|}\|_2^2} = n(n-1)/2$$

$G = n(n-1)/2$ if and only if $\mathbf{z}_1 = \mathbf{z}_2 = ... = \mathbf{z}_n$.

$$\begin{aligned} \mathcal{L}_\mathcal{U}(\mathcal{D} \cup \mathcal{D}) - \mathcal{L}_\mathcal{U}(\mathcal{D}) &= \log \frac{4G+n}{2n(2n-1)/2} - \log \frac{G}{n(n-1)/2} \\ &= \log \frac{(4G+n)n(n-1)/2}{2nG(2n-1)/2} = \log \frac{(4G+n)(n-1)}{4nG-2G} \\ &= \log \frac{4nG - 4G + n^2 - n}{4nG - 2G} \geq \log 1 = 0. \end{aligned}$$

$\mathcal{L}_{\mathcal{U}}(\mathcal{D} \cup \mathcal{D}) = \mathcal{L}_{\mathcal{U}}(\mathcal{D})$ if and only if $G = n(n-1)/2$, which requires $\mathbf{z}_1 = \mathbf{z}_2 = ... = \mathbf{z}_n$. □

**Lemma 4.** *Suppose a set of random variables $\mathcal{D} = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$ ($\mathbf{z}_i \in \mathbb{R}^m$), and a function as follows:*

$$\mathcal{L}_{\mathcal{U}}(\mathcal{D}) \triangleq \log \frac{1}{n(n-1)/2} \sum_{i=2}^{n} \sum_{j=1}^{i-1} e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2},$$

*Then $\mathcal{L}_{\mathcal{U}}(\mathcal{D} \oplus \mathcal{D}) = \mathcal{L}_{\mathcal{U}}(\mathcal{D})$.*

*Proof.* Given $\mathbf{z}_i = [z_{i1}, z_{i2}, ..., z_{im}]^T$, and $\mathbf{z}_j = [z_{j1}, z_{j2}, ..., z_{jm}]^T$, and we set $\hat{\mathbf{z}}_i = \mathbf{z}_i \oplus \mathbf{z}_i$ and $\hat{\mathbf{z}}_j = \mathbf{z}_j \oplus \mathbf{z}_j$, Then:

$$\mathcal{L}_{\mathcal{U}}(\mathcal{D} \oplus \mathcal{D}) \triangleq \log \frac{1}{n(n-1)/2} \sum_{i=2}^{n} \sum_{j=1}^{i-1} e^{-t\|\frac{\hat{\mathbf{z}}_i}{\|\hat{\mathbf{z}}_i\|} - \frac{\hat{\mathbf{z}}_j}{\|\hat{\mathbf{z}}_j\|}\|_2^2},$$

As $\hat{\mathbf{z}}_i = [z_{i1}, z_{i2}, ..., z_{im}, z_{i1}, z_{i2}, ..., z_{im}]^T$ and $\hat{\mathbf{z}}_j = [z_{j1}, z_{j2}, ..., z_{jm}, z_{j1}, z_{j2}, ..., z_{jm}]^T$, then $\|\hat{\mathbf{z}}_i\| = \sqrt{2}\|\mathbf{z}_i\|$, $\|\hat{\mathbf{z}}_j\| = \sqrt{2}\|\mathbf{z}_j\|$, and $\langle \hat{\mathbf{z}}_i, \hat{\mathbf{z}}_j \rangle = 2\langle \mathbf{z}_i, \mathbf{z}_j \rangle$, therefore:

$$\|\frac{\hat{\mathbf{z}}_i}{\|\hat{\mathbf{z}}_i\|} - \frac{\hat{\mathbf{z}}_j}{\|\hat{\mathbf{z}}_j\|}\|_2^2 = 2 - 2\frac{\langle \hat{\mathbf{z}}_i, \hat{\mathbf{z}}_j \rangle}{\|\hat{\mathbf{z}}_i\|\|\hat{\mathbf{z}}_j\|} = 2 - 2\frac{2\langle \mathbf{z}_i, \mathbf{z}_j \rangle}{\sqrt{2}\|\mathbf{z}_i\|\sqrt{2}\|\mathbf{z}_j\|} = \|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2,$$

Therefore, $\mathcal{L}_{\mathcal{U}}(\mathcal{D} \oplus \mathcal{D}) = \mathcal{L}_{\mathcal{U}}(\mathcal{D})$. □

**Lemma 5.** *Suppose a set of random variables $\mathcal{D} = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$ ($\mathbf{z}_i \in \mathbb{R}^m$), and a function as follows:*

$$\mathcal{L}_{\mathcal{U}}(\mathcal{D}) \triangleq \log \frac{1}{n(n-1)/2} \sum_{i=2}^{n} \sum_{j=1}^{i-1} e^{-t\|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2},$$

*Then $\mathcal{L}_{\mathcal{U}}(\mathcal{D} \oplus \mathbf{0}^k) = \mathcal{L}_{\mathcal{U}}(\mathcal{D})$.*

*Proof.* Given $\mathbf{z}_i = [z_{i1}, z_{i2}, ..., z_{im}]^T$, and $\mathbf{z}_j = [z_{j1}, z_{j2}, ..., z_{jm}]^T$, and we set $\hat{\mathbf{z}}_i = \mathbf{z}_i \oplus \mathbf{0}^k$ and $\hat{\mathbf{z}}_j = \mathbf{z}_j \oplus \mathbf{0}^k$, Then:

$$\mathcal{L}_{\mathcal{U}}(\mathcal{D} \oplus \mathbf{0}^k) \triangleq \log \frac{1}{n(n-1)/2} \sum_{i=2}^{n} \sum_{j=1}^{i-1} e^{-t\|\frac{\hat{\mathbf{z}}_i}{\|\hat{\mathbf{z}}_i\|} - \frac{\hat{\mathbf{z}}_j}{\|\hat{\mathbf{z}}_j\|}\|_2^2},$$

As $\hat{\mathbf{z}}_i = [z_{i1}, z_{i2}, ..., z_{im}, 0, 0, ..., 0]^T$, and $\hat{\mathbf{z}}_j = [z_{j1}, z_{j2}, ..., z_{jm}, 0, 0, ..., 0]^T$, then $\|\hat{\mathbf{z}}_i\| = \|\mathbf{z}_i\|$, $\|\hat{\mathbf{z}}_j\| = \|\mathbf{z}_j\|$, and $\langle \hat{\mathbf{z}}_i, \hat{\mathbf{z}}_j \rangle = \langle \mathbf{z}_i, \mathbf{z}_j \rangle$, therefore:

$$\|\frac{\hat{\mathbf{z}}_i}{\|\hat{\mathbf{z}}_i\|} - \frac{\hat{\mathbf{z}}_j}{\|\hat{\mathbf{z}}_j\|}\|_2^2 = 2 - 2\frac{\langle \hat{\mathbf{z}}_i, \hat{\mathbf{z}}_j \rangle}{\|\hat{\mathbf{z}}_i\|\|\hat{\mathbf{z}}_j\|} = 2 - 2\frac{\langle \mathbf{z}_i, \mathbf{z}_j \rangle}{\|\mathbf{z}_i\|\|\mathbf{z}_j\|} = \|\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} - \frac{\mathbf{z}_j}{\|\mathbf{z}_j\|}\|_2^2,$$

Therefore, $\mathcal{L}_{\mathcal{U}}(\mathcal{D} \oplus \mathbf{0}^k) = \mathcal{L}_{\mathcal{U}}(\mathcal{D})$. □

**Lemma 6.** *Suppose a set of random variables $\mathcal{D} = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$ ($\mathbf{z}_i \in \mathbb{R}^m$), and corresponding mean vector and covariance matrix are as follows:*

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{z}_i/\|\mathbf{z}_i\|, \quad \boldsymbol{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{z}_i/\|\mathbf{z}_i\| - \boldsymbol{\mu})^T(\mathbf{z}_i/\|\mathbf{z}_i\| - \boldsymbol{\mu}),$$

*For a function as follows:*

$$\mathcal{W}_2 \triangleq \sqrt{\|\boldsymbol{\mu}\|_2^2 + 1 + Tr(\boldsymbol{\Sigma}) - \frac{2}{\sqrt{m}} Tr(\boldsymbol{\Sigma}^{1/2})},$$

*Then $\mathcal{W}_2(\mathcal{D} \cup \mathcal{D}) = \mathcal{W}_2(\mathcal{D})$.*

*Proof.* As $\mathcal{D} \cup \mathcal{D} = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n, \mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$, then its mean vector and covariance matrix can be formulated as follows:

$$\hat{\boldsymbol{\mu}} = \frac{1}{2n}\sum_{i=1}^{n} 2\mathbf{z}_i/\|\mathbf{z}_i\| = \boldsymbol{\mu}, \quad \hat{\boldsymbol{\Sigma}} = \frac{1}{2n}\sum_{i=1}^{n} 2(\mathbf{z}_i/\|\mathbf{z}_i\| - \hat{\boldsymbol{\mu}})^T(\mathbf{z}_i/\|\mathbf{z}_i\| - \hat{\boldsymbol{\mu}}) = \boldsymbol{\Sigma},$$

Therefore:

$$\mathcal{W}_2(\mathcal{D} \cup \mathcal{D}) \triangleq \sqrt{\|\hat{\boldsymbol{\mu}}\|_2^2 + 1 + Tr(\hat{\boldsymbol{\Sigma}}) - \frac{2}{\sqrt{m}} Tr(\hat{\boldsymbol{\Sigma}}^{1/2})} = \mathcal{W}_2(\mathcal{D}).$$

□

**Lemma 7.** *Suppose a set of random variables $\mathcal{D} = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$ ($\mathbf{z}_i \in \mathbb{R}^m$), and corresponding mean vector and covariance matrix are as follows:*

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{z}_i/\|\mathbf{z}_i\|, \quad \boldsymbol{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{z}_i/\|\mathbf{z}_i\| - \boldsymbol{\mu})^T(\mathbf{z}_i/\|\mathbf{z}_i\| - \boldsymbol{\mu}),$$

*For a function as follows:*

$$\mathcal{W}_2 \triangleq \sqrt{\|\boldsymbol{\mu}\|_2^2 + 1 + Tr(\boldsymbol{\Sigma}) - \frac{2}{\sqrt{m}} Tr(\boldsymbol{\Sigma}^{1/2})},$$

*Then $\mathcal{W}_2(\mathcal{D} \oplus \mathbf{0}^k) > \mathcal{W}_2(\mathcal{D})$.*

*Proof.* Given $\mathbf{z}_i = [z_{i1}, z_{i2}, ..., z_{im}]^T$, and $\hat{\mathbf{z}}_i = \mathbf{z}_i \oplus \mathbf{0}^k = [z_{i1}, z_{i2}, ..., z_{im}, 0, 0, ..., 0]^T \in \mathbb{R}^{m+k}$, for the set: $\mathcal{D} \oplus \mathbf{0}^k$, its mean vector and covariance matrix can be formulated as follows:

$$\hat{\boldsymbol{\mu}} = \begin{pmatrix} \boldsymbol{\mu} \\ \mathbf{0}^k \end{pmatrix}, \quad \hat{\boldsymbol{\Sigma}} = \begin{pmatrix} \boldsymbol{\Sigma} & \mathbf{0}^{m\times k} \\ \mathbf{0}^{k\times m} & \mathbf{0}^{k\times k} \end{pmatrix}$$

Therefore, $Tr(\hat{\boldsymbol{\Sigma}}) = Tr(\boldsymbol{\Sigma})$, and $Tr(\hat{\boldsymbol{\Sigma}}^{1/2}) = Tr(\boldsymbol{\Sigma}^{1/2})$:

$$\begin{aligned}
\mathcal{W}_2(\mathcal{D} \oplus \mathbf{0}^k) &\triangleq \sqrt{\|\hat{\boldsymbol{\mu}}\|_2^2 + 1 + Tr(\hat{\boldsymbol{\Sigma}}) - \frac{2}{\sqrt{m+k}} Tr(\hat{\boldsymbol{\Sigma}}^{1/2})} \\
&= \sqrt{\|\boldsymbol{\mu}\|_2^2 + 1 + Tr(\boldsymbol{\Sigma}) - \frac{2}{\sqrt{m+k}} Tr(\boldsymbol{\Sigma}^{1/2})} \\
&> \sqrt{\|\boldsymbol{\mu}\|_2^2 + 1 + Tr(\boldsymbol{\Sigma}) - \frac{2}{\sqrt{m}} Tr(\boldsymbol{\Sigma}^{1/2})} = \mathcal{W}_2(\mathcal{D})
\end{aligned}$$

Therefore, $\mathcal{W}_2(\mathcal{D} \oplus \mathbf{0}^k) > \mathcal{W}_2(\mathcal{D})$. □

**Lemma 8.** *Suppose a set of random variables $\mathcal{D} = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$ ($\mathbf{z}_i \in \mathbb{R}^m$), and corresponding mean vector and covariance matrix are as follows:*

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{z}_i/\|\mathbf{z}_i\|, \quad \boldsymbol{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{z}_i/\|\mathbf{z}_i\| - \boldsymbol{\mu})^T(\mathbf{z}_i/\|\mathbf{z}_i\| - \boldsymbol{\mu}),$$

*For a function as follows:*

$$\mathcal{W}_2 \triangleq \sqrt{\|\boldsymbol{\mu}\|_2^2 + 1 + Tr(\boldsymbol{\Sigma}) - \frac{2}{\sqrt{m}} Tr(\boldsymbol{\Sigma}^{1/2})},$$

*Then $\mathcal{W}_2(\mathcal{D} \oplus \mathcal{D}) > \mathcal{W}_2(\mathcal{D})$.*

*Proof.* Given $\mathbf{z}_i = [z_{i1}, z_{i2}, ..., z_{im}]^T$, and $\hat{\mathbf{z}}_i = \mathbf{z}_i \oplus \mathbf{z}_i = [z_{i1}, z_{i2}, ..., z_{im}, z_{i1}, z_{i2}, ..., z_{im}]^T \in \mathbb{R}^{2m}$, for the set: $\mathcal{D} \oplus \mathcal{D}$, its mean vector and covariance matrix can be formulated as follows:

$$\hat{\boldsymbol{\mu}} = \begin{pmatrix} \boldsymbol{\mu}/\sqrt{2} \\ \boldsymbol{\mu}/\sqrt{2} \end{pmatrix}, \quad \hat{\boldsymbol{\Sigma}} = \begin{pmatrix} \boldsymbol{\Sigma}/2 & \boldsymbol{\Sigma}/2 \\ \boldsymbol{\Sigma}/2 & \boldsymbol{\Sigma}/2 \end{pmatrix}$$

As $\hat{\boldsymbol{\Sigma}}^{1/2} = \begin{pmatrix} \boldsymbol{\Sigma}^{1/2}/2 & \boldsymbol{\Sigma}^{1/2}/2 \\ \boldsymbol{\Sigma}^{1/2}/2 & \boldsymbol{\Sigma}^{1/2}/2 \end{pmatrix}$, $Tr(\hat{\boldsymbol{\Sigma}}) = Tr(\boldsymbol{\Sigma})$ and $Tr(\hat{\boldsymbol{\Sigma}}^{1/2}) = Tr(\boldsymbol{\Sigma}^{1/2})$, Therefore,

$$\begin{aligned} \mathcal{W}_2(\mathcal{D} \oplus \mathcal{D}) &\triangleq \sqrt{\|\hat{\boldsymbol{\mu}}\|_2^2 + 1 + Tr(\hat{\boldsymbol{\Sigma}}) - \frac{2}{\sqrt{2m}} Tr(\hat{\boldsymbol{\Sigma}}^{1/2})} \\ &= \sqrt{\|\boldsymbol{\mu}\|_2^2 + 1 + Tr(\boldsymbol{\Sigma}) - \frac{2}{\sqrt{2m}} Tr(\boldsymbol{\Sigma}^{1/2})}, \\ &> \sqrt{\|\boldsymbol{\mu}\|_2^2 + 1 + Tr(\boldsymbol{\Sigma}) - \frac{2}{\sqrt{m}} Tr(\boldsymbol{\Sigma}^{1/2})} = \mathcal{W}_2(\mathcal{D}), \end{aligned}$$

Therefore, $\mathcal{W}_2(\mathcal{D} \oplus \mathcal{D}) > \mathcal{W}_2(\mathcal{D})$. □

## D Distribution Distances over Gaussian Distribution

In this section, besides Wasserstein distance over Gaussian distribution, as shown in Theorem 3, we also discuss using other distribution distances as *uniformity* metrics, and make comparisons with Wasserstein distance. As provided Kullback-Leibler Divergence and Bhattacharyya Distance over Gaussian distribution in Theorem 4 and in Theorem 5, both calculations require the covariance matrix is a full rank matrix, making them hard to be used to conduct dimensional collapse analysis. On the contrary, our proposed *uniformity* metric via Wasserstein distance is free from such requirement on the covariance matrix, making it easier to be widely used in practical scenarios.

**Theorem 3.** ***Wasserstein Distance*** *(Olkin & Pukelsheim (1982)) Suppose two random variables $\mathbf{Z}_1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $\mathbf{Z}_2 \sim \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ obey multivariate normal distributions, then $l_2$-Wasserstein distance between $\mathbf{Z}_1$ and $\mathbf{Z}_2$ is:*

$$\mathcal{W}_2(\mathbf{Z}_1, \mathbf{Z}_2) = \sqrt{\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|_2^2 + Tr(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2 - 2(\boldsymbol{\Sigma}_2^{1/2}\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_2^{1/2})^{1/2})}, \tag{11}$$

**Theorem 4.** ***Kullback-Leibler Divergence*** *(Lindley & Kullback (1959)) Suppose two random variables $\mathbf{Z}_1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $\mathbf{Z}_2 \sim \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ obey multivariate normal distributions, then Kullback-Leibler divergence between $\mathbf{Z}1$ and $\mathbf{Z}_2$ is:*

$$\mathcal{D}_{KL}(\mathbf{Z}_1, \mathbf{Z}_2) = \frac{1}{2}((\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}_2^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) + Tr(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1 - \mathbf{I}) + \ln \frac{\det \boldsymbol{\Sigma}_2}{\det \boldsymbol{\Sigma}_1}),$$

**Theorem 5.** ***Bhattacharyya Distance*** *(Bhattacharyya (1943)) Suppose two random variables $\mathbf{Z}_1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $\mathbf{Z}_2 \sim \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ obey multivariate normal distributions, $\boldsymbol{\Sigma} = \frac{1}{2}(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2)$, then bhattacharyya distance between $\mathbf{Z}1$ and $\mathbf{Z}_2$ is:*

$$\mathcal{D}_B(\mathbf{Z}_1, \mathbf{Z}_2) = \frac{1}{8}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) + \frac{1}{2} \ln \frac{\det \boldsymbol{\Sigma}}{\sqrt{\det \boldsymbol{\Sigma}_1 \det \boldsymbol{\Sigma}_2}},$$

## E The definition of Wasserstein Distance

**Definition 1. Wasserstein Distance or *Earth-Mover* Distance** with $p$ norm is defined as below:

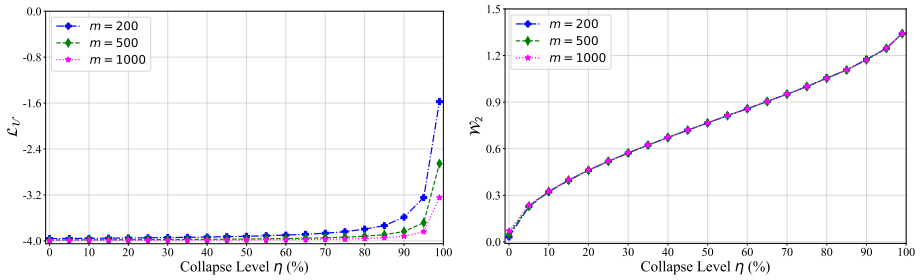
$$W_p(\mathbb{P}_r, \mathbb{P}_g) = (\inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}\left[\|x - y\|^p\right])^{1/p}, \tag{12}$$

where $\Pi(\mathbb{P}_r, \mathbb{P}_g)$ denotes the set of all joint distributions $\gamma(x, y)$ whose marginals are respectively $\mathbb{P}_r$ and $\mathbb{P}_g$. Intuitively, when viewing each distribution as a unit amount of earth/soil, Wasserstein Distance or *Earth-Mover* Distance takes the minimum cost of transporting "mass" from $x$ to $y$ in order to transform the distribution $\mathbb{P}_r$ into the distribution $\mathbb{P}_g$.

As visualized in Fig. 2(a), we use 50000 samples to calculate the binning density for $y_i$ and $\hat{y}_i$ ($m = 2^9$ and the number of bins is 101). To measure the distribution distance between $y_i$ and $\hat{y}_i$, we first instantiate $\mathbb{P}_r$ and $\mathbb{P}_g$ with the binning density of $y_i$ and $\hat{y}_i$, respectively. Then we use $W_1(\mathbb{P}_r, \mathbb{P}_g)$ as the distribution distance, as visualized in Fig. 2(b). In terms of the samples used in the calculation of binning density, we sample them from $y_i$ and $\hat{y}_i$ ten times with different seeds.

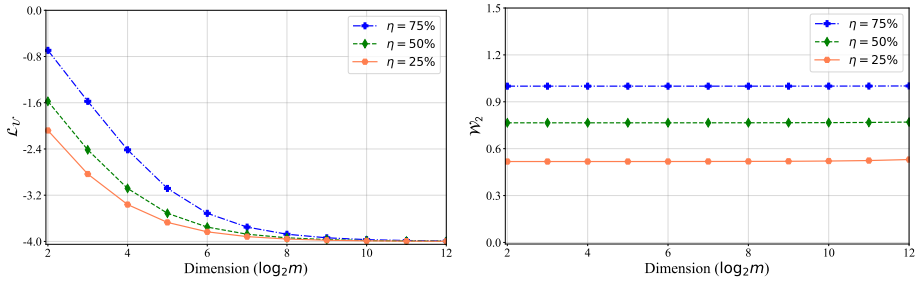
## F Empirical Analysis on the Dimensional Collapse

In this section, we use synthetic data with various specified degrees of dimensional collapse, to check whether the two metrics ($\mathcal{L}_\mathcal{U}$ and $\mathcal{W}_2$) are capable to identify dimensional collapse. As shown in Fig. 5(a) and Fig. 5(b), $\mathcal{W}_2$ has the good ability to capture salient sensitivity to collapse level, while $\mathcal{L}_\mathcal{U}$ fails. We also find $\mathcal{L}_\mathcal{U}$ becomes indistinguishable with different degrees of dimension collapse ($\eta = 25\%, 50\%,$ and$75\%$) when the dimension $m$ becomes large (e.g., $m \geq 2^8$), as visualized in Fig. 6(a). On the contrary, our proposed $\mathcal{W}_2$ is constant to the dimension number under a specific degree of dimension collapse; $\mathcal{W}_2$ only depends on the degree of dimension collapse and is independent of the dimension number, as shown in Fig. 6(b).




(a) Collapse analysis via $\mathcal{L}_\mathcal{U}$ (b) Collapse analysis via $\mathcal{W}_2$

Figure 5: Empirical analysis on different degrees of dimension collapse. $\mathcal{W}_2$ well captures salient sensitivity to collapse level, while $\mathcal{L}_\mathcal{U}$ fails.




(a) Collapse analysis via $\mathcal{L}_\mathcal{U}$ (b) Collapse analysis via $\mathcal{W}_2$

Figure 6: Dimensional collapse analysis w.r.t various dimensions. $\mathcal{L}_\mathcal{U}$ fails to identify the dimensional collapse with a large dimension, while the ability of $\mathcal{W}_2$ to identify the dimensional collapse is free from the influence of dimensions.

## G Parameter Setting in The Experiments

In Section 5, we impose the proposed uniformity metric, i.e., Wassserstein distance in Equation 4 as an additional loss for various existing self-supervised methods. To adapt different models, we adopt a linear decay for weighting Wasserstein distance, and specific parameter setting is shown in Table 3.

Table 3: Parameter setting for various models in experiments.

| Models | MoCo v2 | BYOL | BarlowTwins | Zero-CL |
|---|---|---|---|---|
| $\alpha_{max}$ | 1.0 | 0.2 | 30.0 | 30.0 |
| $\alpha_{min}$ | 1.0 | 0.2 | 0 | 30.0 |

## H ALIGNMENT METRIC FOR SELF-SUPERVISED REPRESENTATION LEARNING

As one of the important indicators to evaluate representation capacity, the alignment metric measures the distance among semantically similar samples in the representation space, and smaller alignment generally brings better representation capacity. Wang et al (Wang & Isola, 2020) propose a simpler approach by calculating the average distance between the positive pairs as alignment, and it can be formulated as follows:

$$\mathcal{A} \triangleq \mathbb{E}_{(\mathbf{z}_i^a, \mathbf{z}_i^b) \sim p_{\mathbf{z}}^{pos}} [\| \frac{\mathbf{z}_i^a}{\|\mathbf{z}_i^a\|} - \frac{\mathbf{z}_i^b}{\|\mathbf{z}_i^b\|} \|_2^{\beta}] \tag{13}$$

Where $(\mathbf{z}_i^a, \mathbf{z}_i^b)$ is a positive pair as discussed in Section 2.1. We set $\beta = 2$ in the experiments.