# UNDERSTANDING DIMENSIONAL COLLAPSE IN CONTRASTIVE SELF-SUPERVISED LEARNING

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

## ABSTRACT

Self-supervised visual representation learning aims to learn useful representations without relying on human annotations. Joint embedding approach bases on maximizing the agreement between embedding vectors from different views of the same image. Various methods have been proposed to solve the collapsing problem where all embedding vectors collapse to a trivial constant solution. Among these methods, contrastive learning prevents collapse via negative sample pairs. It has been shown that non-contrastive methods suffer from a lesser collapse problem of a different nature: dimensional collapse, whereby the embedding vectors end up spanning a lower-dimensional subspace instead of the entire available embedding space. Here, we show that dimensional collapse also happens in contrastive learning. In this paper, we shed light on the dynamics at play in contrastive learning that leads to dimensional collapse. Inspired by our theory, we propose a novel contrastive learning method, called *DirectCLR*, which directly optimizes the representation space without relying on a trainable projector. Experiments show that *DirectCLR* outperforms SimCLR with a trainable linear projector on ImageNet.

## 1 INTRODUCTION

Self-supervised learning aims to learn useful representations of the input data without relying on human annotations. Recent advances in self-supervised visual representation learning based on joint embedding methods (Misra & Maaten, 2020b; He et al., 2020; Chen et al., 2020a; Chen & He, 2020; Grill et al., 2020; Zbontar et al., 2021; Bardes et al., 2021; Chen et al., 2020b; Dwibedi et al., 2021; Li et al., 2021; Misra & Maaten, 2020a; HaoChen et al., 2021; Assran et al., 2021; Caron et al., 2021) show that self-supervised representations have competitive performances compared with supervised ones. These methods generally aim to learn representations invariant to data augmentations by maximizing the agreement between embedding vectors from different distortions of the same images.

As there are trivial solutions where the model maps all input to the same constant vector, known as the collapsing problem, various methods have been proposed to solve this problem that rely on different mechanisms. Contrastive methods like Chen et al. (2020a) and He et al. (2016) define 'positive' and 'negative' sample pairs which are treated differently in the loss function. Non-contrastive methbods like Grill et al. (2020) and Chen & He (2020) use stop-gradient, and an extra predictor to prevent collapse without negative pairs; Caron et al. (2018; 2020) use an additional clustering step; and Zbontar et al. (2021) minimize the redundant information between two branches.

These self-supervised learning methods are successful in preventing complete collapse whereby all representation vectors shrink into a single point. However, it has been observed empirically in non-contrastive learning methods (Hua et al., 2021; Tian et al., 2021) that while embedding vectors do not completely collapse; they collapse along certain dimensions. This is known as *dimensional collapse* (Hua et al., 2021), whereby the embedding vectors only span a lower-dimensional subspace.

In contrastive methods, the repulsive effect of negative examples looks like it should prevent this kind of dimensional collapse. But in this work, we show that contrastive learning methods still suffer from dimensional collapse. We theoretically study the dynamics behind this phenomenon. In short, the gradient from the contrastive loss drives the singular vectors of the adjacent weight matrices aligned. And small singular values of these weight matrices experience a proportionally smaller gradient and hence grow significantly slower. As a result, these singular values remain a significantly lower value compared to others, indicating collapsed dimensions.

Inspired by our theory, we propose a novel contrastive learning method, called *DirectCLR*, which directly optimizes the representation space without relying on a trainable projector. *DirectCLR* outperforms SimCLR with a linear trainable projector on ImageNet.

We summarize our contributions as follows:

- We empirically show that contrastive self-supervised learning suffers from dimensional collapse whereby all the embedding vectors fall into a lower-dimensional subspace instead of the entire available embedding space.
- We provide a theoretical understanding of the dimensional collapse phenomenon. We show that under contrastive loss, the gradient will drive the singular vectors in the weight matrices to be aligned. And small singular values will see proportionally small gradient and grow significantly slower, resulting in collapsed dimensions.
- We propose *DirectCLR*, a novel contrastive learning method that directly optimizes the representation space without relying on a trainable projector. *DirectCLR* outperforms SimCLR with a linear trainable projector.

## 2 Related Works

### 2.1 Self-supervised Learning Methods

Joint embedding methods are a promising approach in self-supervised learning, whose principle is to match the embedding vectors of augmented views of a training instance. Contrastive methods (Chen et al., 2020a; He et al., 2016) directly compare training samples by effectively viewing each sample as its own class, typically based on the InfoNCE contrastive loss (van den Oord et al., 2018) which encourages representations from positive pairs of examples to be close in the embedding space while representations from negative pairs are pushed away from each other. In practice, contrastive methods are known to require a large number of negative samples. Non-contrastive methods do not directly rely on explicit negative samples. These include clustering-based methods (Caron et al., 2018; 2020), redundancy reduction methods (Zbontar et al., 2021; Bardes et al., 2021) and methods using special architecture design (Grill et al., 2020; Chen & He, 2020).

### 2.2 Theoretical Understanding of Self-supervised Learning

Although self-supervised learning models have shown success in learning useful representations and have outperformed their supervised counterpart in several downstream transfer learning benchmarks (Chen et al., 2020a), the underlying dynamics of these methods remains somewhat mysterious and poorly understood. Several theoretical works have attempted to understand it. Arora et al. (2019b); Lee et al. (2020); Tosh et al. (2021) theoretically proved that the learned representations via contrastive learning are useful for downstream tasks. Tian et al. (2021) explained why non-contrastive learning methods like BYOL (Grill et al., 2020) and SimSiam (Chen & He, 2020) work: the dynamics of the alignment of eigenspaces between the predictor and its input correlation matrix play a key role in preventing complete collapse.

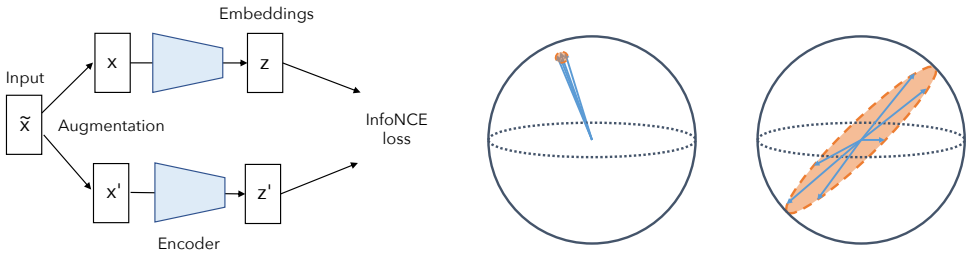
### 2.3 Implicit Regularization

It has been theoretically explained that gradient descent will drive adjacent matrices aligned in a linear neural network setting (Ji & Telgarsky, 2019). Under the aligned matrix assumption, Gunasekar et al. (2018) prove that gradient descent can derive minimal nuclear norm solution. Arora et al. (2019a) extend this concept to the deep linear network case by theoretically and empirically demonstrating that a deep linear network can derive low-rank solutions. In general, over-parametrized neural networks tend to find flatter local minima (Saxe et al., 2019; Neyshabur et al., 2019; Soudry et al., 2018; Barrett & Dherin, 2021).

## 3 Dimensional Collapse

Self-supervised learning methods learn useful representation by minimizing the distances between embedding vectors from augmented images (Figure 1a). On its own, this would result in a collapsed

solution where the produced representation becomes constant (Figure 1b). Contrastive methods prevent complete collapse via the negative term that pushes embedding vectors of different input images away from each other. In this section, we show that while they prevent complete collapse, contrastive methods still experience a dimensional collapse in which the embedding vectors occupy a lower-dimensional subspace than their dimension (Figure 1c).




(a) embedding space (b) complete collapse (c) dimensional collapse

Figure 1: Illustration of the collapsing problem. For complete collapse, the embedding vectors collapse to same point. For dimensional collapse, the embedding vectors only span a lower dimensional space.

We train a SimCLR model (Chen et al. (2020a)) with a two-layer MLP projector. We followed the standard recipe and trained the model on ImageNet for 100 epoch. We evaluate the dimensionality by collecting the embedding vectors on the validation set. We compute the covariance matrix ($C = \sum_i (\mathbf{z} - \bar{\mathbf{z}})(\mathbf{z} - \bar{\mathbf{z}})^T / N$) and apply singular value decomposition on this matrix ($C = USV^T$, $S = diag(\sigma_k)$). The spectrum is made of the singular values in sorted order and logarithmic scale ($\{\log(\sigma_k)\}$). See Figure 7. We observe that a number of singular values collapse to zero, thus representing collapsed dimensions.

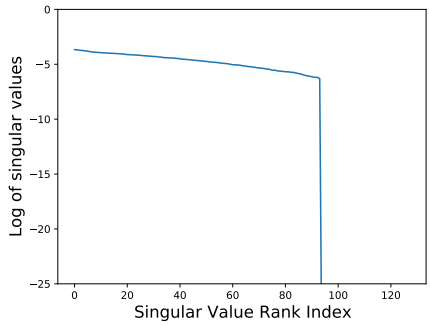



Figure 2: Singular value spectrum of the embedding space. The embedding vectors are computed from a pretrained SimCLR model on the validation set of ImageNet. Each vector has a size of 128. The spectrum contains the singular values of the covariance matrix of these embedding vectors in sorted order and logarithmic scale. Here we observe about 25 singular values drop to zero, indicating those dimensions in the embedding space have collapsed.

## 4 Dynamics

### 4.1 Two-layer linear model

To understand the dynamics of dimensional collapse in contrastive self-supervised learning, we start with the simplest over-parametrized setting by choosing the network as a two-layer linear MLP without bias. The weight matrices of these two layers are denoted by $W_1 \in \mathbb{R}^{n \times n}$ and $W_2 \in \mathbb{R}^{n \times n}$. For each branch, the input vector is $\mathbf{x} \in \mathbb{R}^n$. The embedding vector from each branch is $\mathbf{z} = W_2 W_1 \mathbf{x}$, hence $\mathbf{z} \in \mathbb{R}^n$. We do not normalize $\mathbf{z}$. See Figure. 3

We focus on a typical contrastive loss, InfoNCE (van den Oord et al., 2018):

$$L = \sum_i \log \frac{\exp(-|\mathbf{z}_i - \mathbf{z}_i'|^2/2)}{\sum_j \exp(-|\mathbf{z}_i - \mathbf{z}_j|^2/2)} \quad (1)$$

where $\mathbf{z}_i$ and $\mathbf{z}_i'$ are pairs of representation vectors from two branches. $j$ indicates the index of samples within the minibatch.

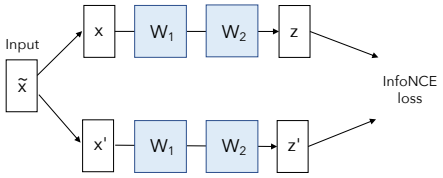



Figure 3: Two-layer Linear Model

In our demonstrative experiments, the input data is sampled by $\tilde{\mathbf{x}} \sim \mathcal{N}(0, I)$ and the augmentation is an isotropic Gaussian noise $\epsilon \sim \mathcal{N}(0, \epsilon_0 I)$. We assume a small augmentation, i.e. $\epsilon_0 \ll 1$. We use a basic stochastic gradient descent without momentum or weight decay.

We summarize the sketch of our theory here:

1. The singular vectors of the weight matrices $W_1$ and $W_2$ corresponding to the same ranking singular values align during training.
2. The gradient on each singular value is proportional to itself and the paired singular value with the same ranking from the other matrix. Hence, the small singular values of each weight matrix grow significantly slower.
3. The small singular values of these two matrices evolve to be significantly smaller compared to others. The corresponding dimensions will collapse in the embedding space.

### 4.2 Gradient Flow Dynamics

We study the approximate dynamics via gradient flow, i.e., gradient descent with an infinitesimally small learning rate. We first derive the gradient flow on matrices $W_1$ and $W_2$ without specifying the exact form of the loss function.

**Theorem 1.** *(Weight matrices dynamics) The weight matrices of the projector in the two layer linear contrastive self-supervised learning model evolves by:*

$$\dot{W}_1 = -W_2^T G \quad (2)$$

$$\dot{W}_2 = -G W_1^T \quad (3)$$

*where $G = \sum_i (\boldsymbol{g}_{z_i} \boldsymbol{x}_i^T + \boldsymbol{g}_{z_i'} \boldsymbol{x}_i'^T)$, and $\boldsymbol{g}_{z_i}$, $\boldsymbol{g}_{z_i'}$ represents the gradient on the embedding vector $\boldsymbol{z}_i$, $\boldsymbol{z}_i'$, respectively.*

See proof in Appendix. B.1.

For InfoNCE loss defined in Eqn. 1, the gradient of the embedding vector for each branch can be written as

$$\mathbf{g}_{\mathbf{z}_i} = -2\mathbf{z}_i' + \sum_j \alpha_{ij} \mathbf{z}_j + \sum_j \alpha_{ji} (\mathbf{z}_j - \mathbf{z}_i) + \mathbf{z}_i \quad (4)$$

where $\{\alpha_{ij}\}$ are the softmax of similarity of between $\mathbf{z}_i$ and $\{\mathbf{z}_j\}$. Since $\mathbf{z}_i = W_2 W_1 \mathbf{x}_i$, we have

$$G = -W_2 W_1 X \quad (5)$$

where $X = 2\sum_i (\mathbf{x}_i' \mathbf{x}_i^T) - \sum_i \sum_j (\alpha_{ij} \mathbf{x}_j \mathbf{x}_i^T + \alpha_{ji} \mathbf{x}_j \mathbf{x}_i^T) + \sum_i (1 - \sum_j \alpha_{ji}) \mathbf{x}_i \mathbf{x}_i^T$. Here, we omitted the term contributed by $\mathbf{g}_{\mathbf{z}_i'} \mathbf{x}_i'^T$ by including all $\mathbf{x}_i'$ term in $\sum_i$.

**Lemma 1.** *Under InfoNCE loss, the expectation of $X$ is a positive definite matrix.*

See proof in Appendix B.2.

We apply singular value decomposition on both matrices $W_1$ and $W_2$, i.e., $W_1 = U_1 S_1 V_1^T$, $W_2 = U_2 S_2 V_2^T$ and $S_1 = diag(\sigma_1)$, $S_2 = diag(\sigma_2)$. We are interested in the interaction between the two matrices $W_1$ and $W_2$, which is governed by the interaction between the adjacent orthogonal matrix $V_2$ and $U_1$. We define the alignment matrix $A = V_2^T U_1$, whose $(k, k')$-entry represents the alignment between the $k$-th right singular vector of $W_2$ and the $k'$-th left singular vector of $W_1$. We derive the dynamics of the alignment matrix.

**Corollary 1.** *(Weight Matrices align) Assuming there is no degeneration in $S_1$ and $S_2$, then the alignment matrix $A = V_2^T U_1$ will converge to identity matrix.*

See proof in Appendix. B.3. Here, we also empirically demonstrate that under InfoNCE loss, the absolute value of the alignment matrix $A$ approximately converges to an identity matrix. See Figure. 4.

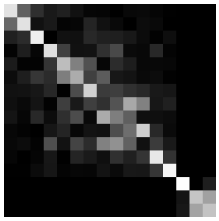


Figure 4: Visualization of the alignment matrix $A = V_2^T U_1$. The model is a 2-layer linear toy model with each weight matrix of the size of 16x16. The alignment matrix approximately converges to an identity matrix.

The alignment effect has been studied in other scenarios (Ji & Telgarsky, 2019; Radhakrishnan et al., 2020). In the real case, when some of our assumptions are not satisfied, e.g., there are degenerate singular values in weight matrices, we will not observe a perfect alignment. This can be easily understood by the fact that the singular decomposition is no longer unique given degenerate singular values. In our toy experiment, we specifically initialize the weight matrices to have non-degenerate singular values. In real scenario, when weight matrices are randomly initialized, we will only observe the alignment matrix to converge to a block-diagonal matrix, with each block representing a group of degenerate singular values.

Given the fact that singular vectors corresponding to the same singular value approximately align, we can now study the dynamics of the singular values of each matrix based on the specific form of the gradient.

**Theorem 2.** *(Singular value dynamics) The singular values of the weight matrices $W_1$ and $W_2$ evolve by:*

$$\dot{\sigma_1^k} = -\sum_{k'} (\boldsymbol{v}_2^{k'T} \boldsymbol{u}_1^k) \sigma_2^{k'} (\boldsymbol{u}_2^{k'T} G \boldsymbol{v}_1^k) \tag{6}$$

$$\dot{\sigma_2^k} = -\sum_{k'} (\boldsymbol{u}_1^{k'T} \boldsymbol{v}_2^k) \sigma_1^{k'} (\boldsymbol{u}_2^{kT} G \boldsymbol{v}_1^{k'}) \tag{7}$$

See proof in Appendix. B.4.

According to Corollary 1, for $\sigma_1^k$ and $\sigma_2^k$ with same index, the corresponding singular vector pairs $\mathbf{v}_2^k$ and $\mathbf{u}_1^k$ will get aligned, i.e., $\mathbf{v}_2^{k'T} \mathbf{u}_1^k \to \delta_{i,j}$. Therefore, Eqn. 6 and Eqn. 7 can be simplified to

$$\dot{\sigma_1^k} \to -\sigma_2^k (\mathbf{u}_2^{kT} G \mathbf{v}_1^k) \tag{8}$$

$$\dot{\sigma_2^k} \to -\sigma_1^k (\mathbf{u}_2^{kT} G \mathbf{v}_1^k) \tag{9}$$

Here, we show that the low-rank nature of the gradient from the contrastive loss will drive the small singular values to be significantly smaller. Insert this Eqn, 5 to Eqn. 8 and 9, and considering the alignment, we derive

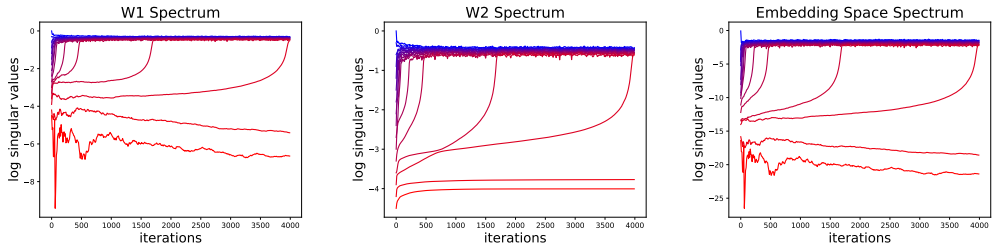
**Corollary 2.** *The singular values of the weight matrices $W_1$ and $W_2$ under InfoNCE loss evolve by:*

$$\dot{\sigma}_1^k \to \sigma_1^k \sigma_2^{k2} \boldsymbol{v}_1^{kT} X \boldsymbol{v}_1^k \tag{10}$$

$$\dot{\sigma}_2^k \to \sigma_2^k \sigma_1^{k2} \boldsymbol{v}_1^{kT} X \boldsymbol{v}_1^k \tag{11}$$

Since $X$ is a positive definite matrix, the term $\mathbf{v}_1^{kT} X \mathbf{v}_1^k$ is always non-negative. This shows that each singular value has a gradient proportional to itself and the paired ones (singular values with same

ranking from the other matrix). Since all singular values are sorted, this explains why we observe that the smallest group of singular values grow significantly slower. See demonstrative experiment results in Figure 5a and 5b.




(a) $W_1$ (b) $W_2$ (c) Embedding Space

Figure 5: Evolution of the singular values of the weight matrices and the embedding space covariance matrix. The model is a 2-layer linear toy model with each weight matrix of the size of 16x16. The lowest few singular values of each weight matrix remain significantly smaller values.

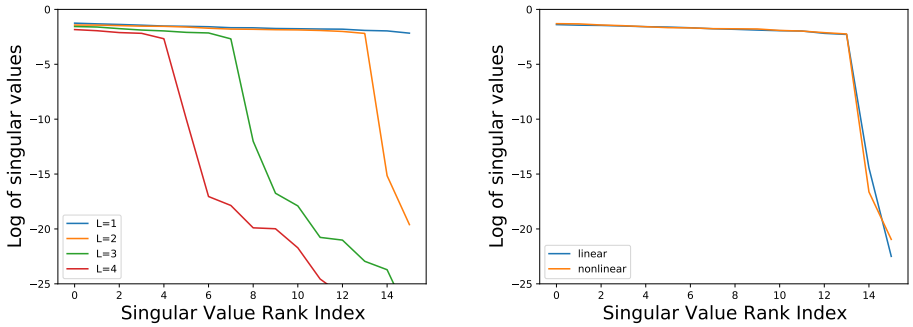
**Corollary 3.** *Under InfoNCE loss, the small singular values of the embedding space covariance matrix remains significantly smaller than others.*

The embedding space is identified by the singular value spectrum of the covariance matrix on the embedding vectors, $C = \sum(\mathbf{z}-\bar{\mathbf{z}})(\mathbf{z}-\bar{\mathbf{z}})^T/N = \sum(\mathbf{x}-\bar{\mathbf{x}})W_1^T W_2^T W_2 W_1(\mathbf{x}-\bar{\mathbf{x}})^T/N$. As alignment matrix $A$ converges to identity matrix, we have $C \to \sum(\mathbf{x}-\bar{\mathbf{x}})V_1 S_1^2 S_2^2 V_1^T(\mathbf{x}-\bar{\mathbf{x}})^T/N$. Therefore, the small singular value pairs in $S_1$ and $S_2$ will be multiplied. As a result, $C$ is approximately low-rank, indicating collapsed dimensions. See Figure. 5c for experimental verification.

### 4.3 Effect of More Layers and Nonlinearity

In our toy model, we focused on a two-layer linear MLP setting. Here, we empirically show that our theory extends to multilayer and nonlinear cases.

Stronger over-parametrization leads to a stronger collapsing effect, which has been shown theoretically (Arora et al., 2019a; Barrett & Dherin, 2021) and empirically (Jing et al., 2020). See Figure. 6a. This can be explained by the fact that more adjacent matrices getting aligned, and the collapsing in the product matrix gets amplified. Note that for a single-layer case, $L = 1$, there is no dimensional collapse in the embedding space.




(a) multiple layers (b) nonlinear

Figure 6: Embedding space singular value spectrums with (a) different layers; (b) nonlinearity. The models have a weight matrix size of 16x16. Adding more linear layers in the network leads to more collapsed dimensions. Adding nonlinearity leads to similar the collapsing effect.

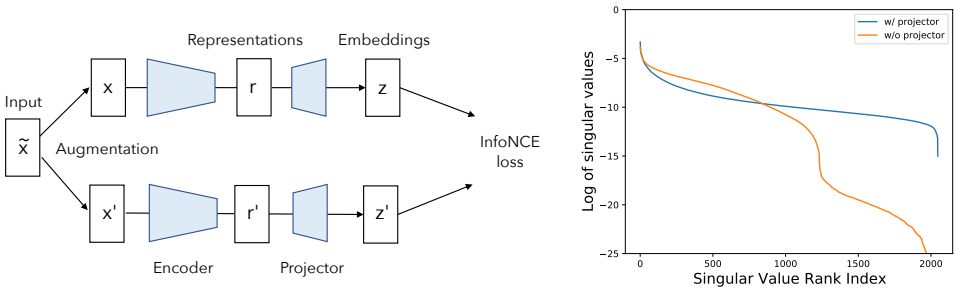
We empirically show that the collapsing effect also applies to the nonlinear scenario. We insert ReLU between linear layers and observe a similar singular value collapse compared to the linear case. See Figure. 6b.

# 5 DIRECTCLR

## 5.1 MOTIVATION

In typical contrastive learning methods, the output vectors from the backbone encoder used for downstream tasks are not directly optimized. These vectors are defined as representation vectors. Instead, a common practice is to add an MLP projector on top of the encoder and apply the loss function to the projector's output. The output vectors from the final projector are defined as embedding vectors. See Figure 7a.

We train two SimCLR models (Chen et al., 2020a) with and without a projector. The representation space spectrum are shown in Figure 7b. The dimensional collapse in representation space happens when the model is trained without a projector. Thus, the projector prevents the collapse in the representation space.




(a) representation and embedding

(b) Representation space spectrum

Figure 7: (a) Definition of representation space and the embedding space; (b) Singular value spectrums of the representation space of pretrained contrastive learning models (pretrained with or without a projector). The representation vectors are the output from the ResNet50 encoder and directly used for downstream tasks. Each vector has a size of 2048. Without a projector, SimCLR suffers from dimensional collapse in the representation space.

Our theory does not directly explain why there's no dimensional collapse in the representation space when there is a projector for contrastive learning models. Indeed, if the whole network is a simple MLP, we would observe progressive dimensional collapse across layers, as the rank of a hidden layer decreases when it moves close to the output (similar to Figure 6). However, in the real case, we observe almost no collapsed dimensions in the representation space, see Figure 7b. We suspect that the specific building block in the backbone encoder, such as residual connection, prevents the collapse. To theoretically understand why a projector prevents collapse in the representation space remains our future work.

## 5.2 MAIN IDEA

The projector in contrastive learning is essential to prevent dimensional collapse in representation space. Based on our theory, we claim the following propositions regarding the role of projector in contrastive learning models.

First, the gradient will drive the projector weight matrix aligned with the last layer of the backbone. Hence, the projector weight matrix can be replaced by a diagonal matrix with the same singular values. Second, the projector only applies a gradient to a subspace to the representations. Hence, the weight matrix in the projector is effectively low-rank.

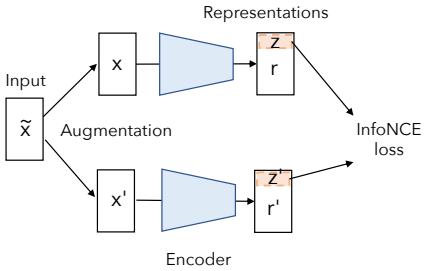



Figure 8: *DirectCLR*: no trainable projector, simply apply InfoNCE loss on the a fixed sub-vector of the representations

Therefore, we propose to directly implement such a function by only send a sub-vector of the representation vector to the loss function. We call our method *DirectCLR*. In contrast to all recent state-of-the-art self-supervised learning methods, our method directly optimizes the representation space.

See Figure 8, *DirectCLR* picks a subvector of the representation $\mathbf{z} = \mathbf{r}[0 : d_0]$, where $d_0$ is a hyperparameter. Then, it applies a standard InfoNCE loss on this normalized subvector $\hat{\mathbf{z}} = \mathbf{z}/|\mathbf{z}|$, $L = \sum_i \log \frac{\exp(\hat{\mathbf{z}}_i \cdot \hat{\mathbf{z}}'_i)}{\sum_j \exp(\hat{\mathbf{z}}_i \cdot \hat{\mathbf{z}}_j)}$.

We train *DirectCLR* with a standard recipe of SimCLR for 100 epochs on ImageNet. The backbone encoder is a ResNet50. *DirectCLR* demonstrates better performance compared to SimCLR with a trainable linear projector on ImageNet. The linear probe accuracies for each model are listed in Table 1.

| Loss function | Projector | Top-1 Accuracy |
|---|---|---|
| SimCLR | 2-layer nonlinear projector | 66.5 |
| SimCLR | 1-layer linear projector | 61.1 |
| SimCLR | no projector | 51.5 |
| *DirectCLR* | no projector | 62.7 |

Table 1: Linear probe accuracy on ImageNet. *DirectCLR* outperforms SimCLR with 1-layer linear projector.

We visualize the learnt representation space spectrum in Figure. 9. *DirectCLR* prevents dimensional collapse in the representation space similar to the functionality of a trainable projector in SimCLR.

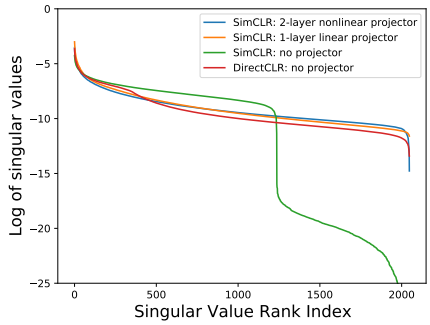



Figure 9: Representaiton space spectrum of *DirectCLR* compared to SimCLR (a) with a 2-layer nonlinear projector (b) with a 1-layer linear projector (c) without projector. The spectrums are computed based on the output from the backbone, using ImgaeNet validation set. Similar to SimCLR with projectors, *DirectCLR* is able to prevent dimensional collapse in the representation space.

One may suspect that the contrastive loss in *DirectCLR* does not apply a gradient on the rest part of the representation vector $\mathbf{r}[d_0 :]$, then why these dimensions would contain useful information? In fact, the entire representation vector $\mathbf{r}$ is trained and contains useful information. The rest part of the representation vector is copied from the layer before the last residual block via residual connection. This part of the representation does not experience a gradient directly from the loss function but is updated by the gradient through the convolution block. Thus, *DirectCLR* takes advantage of the residual connection in the ResNet backbone. The last block of the backbone performs effectively as a projector for $\mathbf{r}[d_0 :]$. See Figure. 10.

### 5.3 Ablation Study

**Fixed Low-rank projector** *DirectCLR* is effectively a fixed linear projector with a diagonal matrix. We observe that it performs similarly on ImageNet when using a fixed low-rank matrix as a projector (62.3% linear probe accuracy), whose singular values are set to have $d_0$ number of 1 and 0 for the

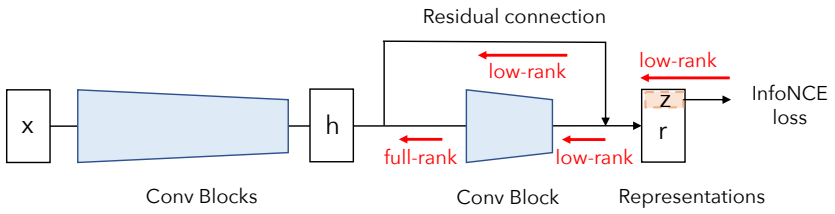



Figure 10: Why *DirectCLR* works. The rest part the representation vector does not directly see gradient from the loss function. But it is copied from the previous layer via residual connection, which experiences full rank gradient passed through the last convolution block.

rest. Thus, their only difference is that this fixed projector has an extra fixed orthogonal matrix in between. This proves that the representation space (or equivalently the previous layer) has evolved to be aligned to this orthogonal matrix during training. This also matches the prediction of our theory that the alignment happens in the projector MLP.

**Trainable Diagonal projector** We trained a SimCLR model with a trainable projector that is constrained only to have diagonal terms. The model achieves $60.2\%$ linear probe accuracy on ImageNet, which is comparable to a SimCLR with a 1-layer linear projector. This can also be explained by the alignment phenomenon that the projector gets aligned with the previous layer in the backbone.

## 6 Conclusions

In this work, we showed that contrastive self-supervised learning suffers from dimensional collapse, where the embedding vectors only span a lower-dimensional subspace. We provided the theoretical understanding of this phenomenon and showed that it is caused by the nature of contrastive loss. Inspired by our theory, we proposed a novel contrastive self-supervised learning method *DirectCLR* that directly optimizes the representation space without relying on a trainable projector. *DirectCLR* outperforms SimCLR with a linear projector on ImageNet.

## References

Sanjeev Arora, Nadav Cohen, W. Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. In *NeurIPS*, 2019a.

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

Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Armand Joulin, Nicolas Ballas, and Michael G. Rabbat. Semi-supervised learning of visual features by non-parametrically predicting view assignments with support samples. *ArXiv*, abs/2104.13963, 2021.

Adrien Bardes, J. Ponce, and Y. LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *ArXiv*, abs/2105.04906, 2021.

D. Barrett and B. Dherin. Implicit gradient regularization. *ArXiv*, abs/2009.11162, 2021.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

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.

Mathilde Caron, Hugo Touvron, Ishan Misra, Herv'e J'egou, J. Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *ArXiv*, abs/2104.14294, 2021.

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

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2020.

Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *ArXiv*, abs/2003.04297, 2020b.

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. *ArXiv*, abs/2104.14548, 2021.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila 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.

Suriya Gunasekar, Blake E. Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Implicit regularization in matrix factorization. *2018 Information Theory and Applications Workshop (ITA)*, pp. 1–10, 2018.

Jeff Z. HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *ArXiv*, abs/2106.04156, 2021.

Kaiming He, Xiangyu 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. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9726–9735, 2020.

Tianyu Hua, Wenxiao Wang, Zihui Xue, Yue Wang, Sucheng Ren, and Hang Zhao. On feature decorrelation in self-supervised learning. *ArXiv*, abs/2105.00470, 2021.

Ziwei Ji and Matus Telgarsky. Gradient descent aligns the layers of deep linear networks. *ArXiv*, abs/1810.02032, 2019.

L. Jing, J. Zbontar, and Y. LeCun. Implicit rank-minimizing autoencoder. *ArXiv*, abs/2010.00679, 2020.

J. Lee, Qi Lei, Nikunj Saunshi, and Jiacheng Zhuo. Predicting what you already know helps: Provable self-supervised learning. *ArXiv*, abs/2008.01064, 2020.

Junnan Li, Pan Zhou, Caiming Xiong, R. Socher, and S. Hoi. Prototypical contrastive learning of unsupervised representations. *ArXiv*, abs/2005.04966, 2021.

Ishan Misra and L. V. D. Maaten. Self-supervised learning of pretext-invariant representations. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6706–6716, 2020a.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020b.

Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Y. LeCun, and Nathan Srebro. Towards understanding the role of over-parametrization in generalization of neural networks. *ArXiv*, abs/1805.12076, 2019.

Adityanarayanan Radhakrishnan, Eshaan Nichani, D. Bernstein, and Caroline Uhler. On alignment in deep linear neural networks. *arXiv: Learning*, 2020.

Andrew M. Saxe, James L. McClelland, and S. Ganguli. A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences*, 116:11537 – 11546, 2019.

Daniel Soudry, E. Hoffer, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *ArXiv*, abs/1710.10345, 2018.

Yuandong Tian, Xinlei Chen, and S. Ganguli. Understanding self-supervised learning dynamics without contrastive pairs. *ArXiv*, abs/2102.06810, 2021.

Christopher Tosh, A. Krishnamurthy, and Daniel J. Hsu. Contrastive learning, multi-view redundancy, and linear models. *ArXiv*, abs/2008.10150, 2021.

Aäron van den Oord, Y. Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *ArXiv*, abs/1807.03748, 2018.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arxiv:2103.03230*, 2021.

## A Useful Lemmas

We adapt two useful lemmas from Arora et al. (2019a).

**Lemma 2.** *Given a matrix $W$ and the dynamics that $W$ evolves by $\dot{W}$, the singular values of this matrix evolve by:*

$$\dot{\sigma^k} = \boldsymbol{u}^{k^T} \dot{W} \boldsymbol{v}^k \tag{12}$$

*where $\boldsymbol{u}^k$ and $\boldsymbol{v}^k$ are singular value $\sigma^k$'s corresponding left and right singular vectors. i.e. the $k$-th column of matrices $U$ and $V$ respectively.*

*Proof.* Given a matrix $W$ and its singular value decomposition $W = USV^T$. We have the dynamics of the matrix

$$\dot{W} = \dot{U}SV^T + U\dot{S}V^T + US\dot{V}^T$$

Multiplying $U^T$ from the left and multiplying $V$ from the right, considering $U$ and $V$ are orthogonal matrices, we have

$$U^T\dot{W}V = U^T\dot{U}S + \dot{S} + S\dot{V}^TV$$

Since $S = diag(\sigma^k)$ is a diagonal matrix, we have

$$\dot{\sigma^k} = \mathbf{u}^{k^T}\dot{W}\mathbf{v}^k - \mathbf{u}^{k^T}\dot{\mathbf{u}^k}\sigma^k - \sigma^k\dot{\mathbf{v}^k}^T\mathbf{v}^k$$

Again, considering $\mathbf{u}^k$ and $\mathbf{v}^k$ have unit-norm, we have $\mathbf{u}^{k^T}\dot{\mathbf{u}^k} = 0$ and $\dot{\mathbf{v}^k}^T\mathbf{v}^k = 0$. Therefore, we derive

$$\dot{\sigma^k} = \mathbf{u}^{k^T}\dot{W}\mathbf{v}^k$$

□

**Lemma 3.** *Given a matrix $W$ and the dynamics that $W$ evolves by $\dot{W}$, the singular vectors of this matrix evolve by:*

$$\dot{U} = U(H \odot (U^T\dot{W}VS + SV^T\dot{W}^TU)) \tag{13}$$

$$\dot{V} = V(H \odot (V^T\dot{W}^TUS + SU^T\dot{W}V)) \tag{14}$$

*where $\odot$ represents Hadamard element-wise multiplication. $H$ is a skew-symmetric matrix*

$$H^{k,k'} = \begin{cases} 1/(\sigma^{k^2} - \sigma^{k'^2}) & \text{if } k \neq k' \\ 0 & \text{if } k = k' \end{cases} \tag{15}$$

*Proof.* Same as proof for Lemma 2, we start from the following equation

$$U^T\dot{W}V = U^T\dot{U}S + \dot{S} + S\dot{V}^TV$$

Considering the fact that $U^T\dot{U}$ and $\dot{V}^TV$ are skew-symmetric matrices, whose diagonal terms are all zero, we Hadamard-multiply $\bar{I}$ to both sides of the equation. Here, $\bar{I}$ has all diagonal values equal zeros and all off-diagonal values equal to one, we have

$$\bar{I} \odot U^T\dot{W}V = U^T\dot{U}S + S\dot{V}^TV \tag{16}$$

Taking transpose, we have

$$\bar{I} \odot V^T \dot{W} U = -S U^T \dot{U} - \dot{V}^T V S \tag{17}$$

Right-multiplying $S$ to Eqn 16 and left-multiplying $S$ to Eqn 17, then adding them up, we have

$$U^T \dot{U} S^2 - S^2 U^T \dot{U} = \bar{I} \odot (U^T \dot{W} V S + S V^T \dot{W} U)$$

Therefore, we have

$$\dot{U} = U(H \odot (U^T \dot{W} V S + S V^T \dot{W}^T U))$$

where

$$H^{k,k'} = \begin{cases} 1/({\sigma^k}^2 - {\sigma^{k'}}^2) & \text{if } k \neq k' \\ 0 & \text{if } k = k' \end{cases}$$

Similar proof applies to Eqn. 14. □

**Lemma 4.** *(Alignment matrix dynamics) The alignment matrix $A$, defined by $A = V_2^T U_1$, evolves by:*

$$\dot{A} = -A(H_1 \odot (A^T F + F^T A)) + (H_2 \odot (A F^T + F A^T))A \tag{18}$$

*where $\odot$ represents Hadamard (element-wise) multiplication. $H_l$ is a skew-symmetric matrix, whose $(k, k')$-entry is given by*

$$H_l^{k,k'} = \begin{cases} 1/({\sigma_l^k}^2 - {\sigma_l^{k'}}^2) & \textit{if } k \neq k' \\ 0 & \textit{if } k = k' \end{cases} \tag{19}$$

*and $F$ is defined by*

$$F = S_2 U_2^T G V_1 S_1 \tag{20}$$

*Proof.* According to Lemma. 3, we have

$$\dot{U}_1 = U_1(H_1 \odot (U_1^T \dot{W}_1 V_1 S_1 + S_1 V_1^T \dot{W}_1^T U_1))$$
$$\dot{V}_2 = V_2(H_2 \odot (V_2^T \dot{W}_2^T U_2 S_2 + S_2 U_2^T \dot{W}_2 V_2))$$

Plugging the above two equations and Eqn. 2, Eqn. 3, the dynamics of the alignment matrix $A = V_2^T U_1$ can be written as

$$\begin{aligned}
\dot{A} &= V_2^T \dot{U}_1 + \dot{V}_2^T U_1 \\
&= V_2^T U_1 (H_1 \odot (U_1^T \dot{W}_1 V_1 S_1 + S_1 V_1^T \dot{W}_1^T U_1)) + (H_2 \odot (V_2^T \dot{W}_2^T U_2 S_2 + S_2 U_2^T \dot{W}_2 V_2))^T V_2^T U_1 \\
&= -A(H_1 \odot (U_1^T W_2^T G V_1 S_1 + S_1 V_1^T G^T W_2 U_1)) + (H_2 \odot (S_2 U_2^T G W_1^T V_2 + V_2^T W_1 G^T U_2 S_2))A \\
&= -A(H_1 \odot (U_1^T V_2 S_2 U_2^T G V_1 S_1 + S_1 V_1^T G^T U_2 S_2 V_2^T U_1)) \\
&\quad +(H_2 \odot (S_2 U_2^T G V_1 S_1 U_1^T V_2 + V_2^T U_1 S_1 V_1^T G^T U_2 S_2))A \\
&= -A(H_1 \odot (A^T S_2 U_2^T G V_1 S_1 + S_1 V_1^T G^T U_2 S_2 A) \\
&\quad +(H_2 \odot (S_2 U_2^T G V_1 S_1 A^T + A S_1 V_1^T G^T U_2 S_2))A \\
&= -A(H_1 \odot (A^T F + F^T A)) + (H_2 \odot (A F^T + F A^T))A
\end{aligned}$$

where

$$F = S_2 U_2^T G V_1 S_1$$

□

# B Proof of Theorems

## B.1 Proof of Theorem 1

*Proof.* Eqn. 2 and 3 can be derived simply from chain rule.

The gradient on matrix $W_2$ is

$$\frac{dL}{dW_2} = \sum_i \left(\frac{\partial L}{\partial \mathbf{z}_i} \frac{\partial \mathbf{z}_i}{\partial W_2} + \frac{\partial L}{\partial \mathbf{z}_i'} \frac{\partial \mathbf{z}_i'}{\partial W_2}\right) \tag{21}$$

We denote the gradient on $\mathbf{z}_i$ and $\mathbf{z}'_i$ as $\mathbf{g}_{\mathbf{z}_i}$ and $\mathbf{g}_{\mathbf{z}'_i}$. Since $\frac{\partial \mathbf{z}_i}{\partial W_2} = W_1\mathbf{x}_i$, $\frac{\partial \mathbf{z}'_i}{\partial W_2} = W_1\mathbf{x}'_i$, we get

$$\dot{W}_2 = -(\frac{dL}{dW_2})^T = -\sum_i (\mathbf{g}_{\mathbf{z}_i}{\mathbf{x}_i}^T + \mathbf{g}_{\mathbf{z}'_i}{\mathbf{x}'_i}^T)W_1^T \tag{22}$$

Similar proof applies to $W_1$. □

### B.2 Proof of Lemma 1

*Proof.* $X$ matrix is defined by $X = 2\sum_i(\mathbf{x}'_i\mathbf{x}_i^T) - \sum_i\sum_j(\alpha_{ij}\mathbf{x}_j\mathbf{x}_i^T + \alpha_{ji}\mathbf{x}_j\mathbf{x}_i^T) + \sum_i(1 - \sum_j \alpha_{ji})\mathbf{x}_i\mathbf{x}_i^T$.

We have two assumptions regarding the augmentation.

First, we assume that the augmentation is an isotropic additive noise. Therefore we have $\mathbb{E}[\mathbf{x}'_i\mathbf{x}_i^T] = \mathbb{E}[\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T]$ and $\mathbb{E}[\mathbf{x}_j\mathbf{x}_i^T] = \mathbb{E}[\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T]$, where $\tilde{\mathbf{x}}$ is the original input data without augmentation.

Second, we assume that the augmentation has a significant small value, we can also replace the last term into $\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T$. Now, we omit the $\sim$ and simply denote the original data as $\mathbf{x}$. Therefore, the expectation of $X$ can be written as

$$\begin{aligned}
\mathbb{E}(X) &= \mathbb{E}((3 - \sum_j \alpha_{ji})\mathbf{x}_i\mathbf{x}_i^T - \sum_j(\alpha_{ji} + \alpha_{ij})\mathbf{x}_j\mathbf{x}_i^T) \\
&= \mathbb{E}(\mathbf{x}_i\mathbf{x}_i^T - \sum_j \alpha_{ij}\mathbf{x}_j\mathbf{x}_i^T) + \mathbb{E}(\mathbf{x}_i\mathbf{x}_i^T - \sum_j \alpha_{ji}\mathbf{x}_j\mathbf{x}_i^T) + \mathbb{E}(1 - \sum_j \alpha_{ji})\mathbf{x}_i\mathbf{x}_i^T)
\end{aligned}$$

For the first term $\mathbb{E}_1 = \mathbb{E}(\mathbf{x}_i\mathbf{x}_i^T - \sum_j \alpha_{ij}\mathbf{x}_j\mathbf{x}_i^T)$, we decompose each $\mathbf{x}_j$ into $\mathbf{x}_j^{\|}$ and $\mathbf{x}_j^{\perp}$, where $\mathbf{x}_j^{\|}$ is parallel to $\mathbf{x}_i$ and $\mathbf{x}_j^{\perp}$ is perpendicular to $\mathbf{x}_i$. Therefore, we have

$$\begin{aligned}
\mathbb{E}_1 &= \mathbb{E}(\mathbf{x}_i\mathbf{x}_i^T - \sum_j \alpha_{ij}\mathbf{x}_j^{\|}\mathbf{x}_i^T - \sum_j \alpha_{ij}\mathbf{x}_j^{\perp}\mathbf{x}_i^T) \\
&= \mathbb{E}(\mathbf{x}_i\mathbf{x}_i^T - \sum_j \alpha_{ij}\frac{|\mathbf{x}_j^{\|}|}{|\mathbf{x}_i|}\mathbf{x}_i\mathbf{x}_i^T - \sum_j \alpha_{ij}\mathbf{x}_j^{\perp}\mathbf{x}_i^T) \\
&= \mathbb{E}(1 - \sum_j \alpha_{ij}\frac{|\mathbf{x}_j^{\|}|}{|\mathbf{x}_i|})\mathbf{x}_i\mathbf{x}_i^T - \mathbb{E}\sum_j \alpha_{ij}\mathbf{x}_j^{\perp}\mathbf{x}_i^T \\
&= \mathbb{E}(1 - \sum_j \alpha_{ij}\frac{\mathbf{x}_j \cdot \mathbf{x}_i}{|\mathbf{x}_i|^2})\mathbf{x}_i\mathbf{x}_i^T - \mathbb{E}\sum_j \alpha_{ij}\mathbf{x}_j^{\perp}\mathbf{x}_i^T
\end{aligned}$$

Because $\mathbf{x}_j$ is isotropic, we know $\mathbf{x}_j^{\perp}$ is symmetric given $\mathbf{x}_i$, i.e., for every minibatch of $\{\mathbf{x}_j\}$, there exist another minibatch $\{\bar{\mathbf{x}}_j\}$, such that $\mathbf{x}_j^{\|} = \bar{\mathbf{x}}_j^{\|}$ and $\mathbf{x}_j^{\perp} = -\bar{\mathbf{x}}_j^{\perp}$. Therefore, $\alpha_{ij} = \bar{\alpha}_{ij}$ and $\mathbb{E}[\sum_j \alpha_{ij}\mathbf{x}_j^{\perp}\mathbf{x}_i^T] = 0$, we have

$$\mathbb{E}_1 = \mathbb{E}(1 - \sum_j \alpha_{ij}\frac{\mathbf{x}_j \cdot \mathbf{x}_i}{|\mathbf{x}_i|^2})\mathbf{x}_i\mathbf{x}_i^T$$

For simplicity, we assume there's only one negative sample per minibatch, $\mathbb{E}_1 = \mathbb{E}\alpha_{ij}(1 - \frac{\mathbf{x}_j \cdot \mathbf{x}_i}{|\mathbf{x}_i|^2})\mathbf{x}_i\mathbf{x}_i^T$.

For $\mathbf{x}_i$, $\mathbf{x}_j$ pairs with $|\mathbf{x}_i| = |\mathbf{x}_j|$, the coefficient $\sum_i \alpha_{ij}(1 - \frac{\mathbf{x}_j \cdot \mathbf{x}_i}{|\mathbf{x}_i|^2}) > 0$.

For $\mathbf{x}_i$, $\mathbf{x}_j$ pairs with $|\mathbf{x}_i| \neq |\mathbf{x}_j|$, we leverage the fact that for any pair of $\mathbf{x}_i$ and $\mathbf{x}_j$, there exists another pair $\bar{\mathbf{x}}_i$ and $\bar{\mathbf{x}}_j$ with same directions but swapped amplitudes: $\mathbf{x}_i \parallel \bar{\mathbf{x}}_i$, $\mathbf{x}_j \parallel \bar{\mathbf{x}}_j$ and $|\mathbf{x}_i| = |\bar{\mathbf{x}}_j|$, $|\mathbf{x}_j| = |\bar{\mathbf{x}}_i|$. Hence, the corresponding $\alpha_{ij} = \bar{\alpha}_{ij}$. Combining their contribution to $\mathbb{E}_1$, we have

$\mathbb{E}_1 = \mathbb{E}\alpha_{ij}((1 - \frac{\mathbf{x}_j \cdot \mathbf{x}_i}{|\mathbf{x}_i|^2})\mathbf{x}_i\mathbf{x}_i^T + (1 - \frac{\bar{\mathbf{x}}_j \cdot \bar{\mathbf{x}}_i}{|\bar{\mathbf{x}}_i|^2})\bar{\mathbf{x}}_i\bar{\mathbf{x}}_i^T) = \mathbb{E}\alpha_{ij}(1 - 2\frac{\mathbf{x}_j \cdot \mathbf{x}_i}{|\mathbf{x}_i|^2} + \frac{|\mathbf{x}_j|^2}{|\mathbf{x}_i|^2})\mathbf{x}_i\mathbf{x}_i^T$ Therefore, the coefficient $(1 - 2\frac{\mathbf{x}_j \cdot \mathbf{x}_i}{|\mathbf{x}_i|^2} + \frac{|\mathbf{x}_j|^2}{|\mathbf{x}_i|^2}) \geq (1 - \frac{|\mathbf{x}_j|}{|\mathbf{x}_i|})^2 \geq 0$.

Since $\alpha_{ij} > 0$ and not all coefficient equal to zero, given the fact that $\mathbf{x}_i\mathbf{x}_i^T$ is a positive definite matrix, we have $\mathbb{E}_1$ is positive definite.

Similar proof applies to the second term $\mathbb{E}(\mathbf{x}_i\mathbf{x}_i^T - \sum_j \alpha_{ji}\mathbf{x}_j\mathbf{x}_i^T)$.

For the last term $\mathbb{E}(1 - \sum_j \alpha_{ji})\mathbf{x}_i\mathbf{x}_i^T) = \mathbb{E}_{\mathbf{x}_i}(1 - \sum_j \alpha_{ji})\mathbb{E}\mathbf{x}_i\mathbf{x}_i^T$, because $\alpha_{ji}$ averages over all possible minibatch $\{j\}$ in the dataset, we have $\mathbb{E}_{\mathbf{x}_i}(1 - \sum_j \alpha_{ji}) = \mathbb{E}_{\mathbf{x}_i}(1 - \sum_j \alpha_{ij}) = 0$.

Adding up the above three terms, we have $\mathbb{E}[X]$ is a positive definite matrix. □

### B.3 Proof of Corollary 1

Here, we prove that under the assumption that singular values are non-degenerate, the alignment matrix $A = V_2^T U_1$ converges to identity matrix.

*Proof.* According to Theorem 1, we have

$$\frac{d}{dt}(W_1W_1^T) = -W_1G^TW_2 - W_2^TGW_1^T$$
$$\frac{d}{dt}(W_2W_2^T) = -W_1G^TW_2 - W_2^TGW_1^T$$

therefore,

$$\frac{d}{dt}(W_1W_1^T - W_2^TW_2) = 0$$

or

$$W_1W_1^T - W_2^TW_2 = C$$

Next, we show that the Frobenius norm of each weight matrix grow to infinitely.

$$\frac{d}{dt}||W_1||_F^2 = \frac{d}{dt}tr(W_1W_1^T) = -tr(W_2^TGW_1^T) - tr(W_1G_1^TW_2)$$

According to Eqn. 5, $G = -W_2W_1X$, we have

$$\begin{aligned} -tr(W_2^TGW_1^T) &= tr(W_2^TW_2W_1XW_1^T) \\ &= tr(W_2W_1XW_1^TW_2^T) \end{aligned}$$

Because $X$ is a positive definite matrix (see Section B.2), we know $W_2W_1XW_1^TW_2^T$ is also positive definite, and $tr(W_2W_1XW_1^TW_2^T) > 0$. (Here we assume $W_2W_1$ not equal to 0). Therefore, we know $||W_1||_F^2$ (similarly $||W_2||_F^2$) will keep growing, and we have long time approximation

$$W_1W_1^T = W_2^TW_2$$

Plug in the singular value decomposition of $W_1$ and $W_2$, we have $U_1S_1^2U_1^T = V_2S_2^2V_2^T$. Assuming $W_1$ and $W_2$ have non-degenerate singular values, due to the uniqueness of eigen-decomposition, we have

$$U_1 = V_2$$

therefore,

$$V_2^TU_1 = I$$

□

Note that when the non-degenerate singular value assumption does not hold, the corresponding singular vectors are not unique and we will not observe the corresponding dimensions becoming aligned.

### B.4 PROOF OF THEOREM 2

*Proof.* According to Lemma. 2,

$$\dot{\sigma}_1^r = \mathbf{u}_1^{r\,T} \dot{W}_1 \mathbf{v}_1^r$$

Plugging in Eqn. 2, we have

$$\begin{aligned}
\dot{\sigma}_1^k &= -\mathbf{u}_1^{k\,T} W_2^T G \mathbf{v}_1^k \\
&= -\mathbf{u}_1^{k\,T} V_2 S_2 U_2^T G \mathbf{v}_1^k \\
&= -\sum_{k'} (\mathbf{v}_2^{k'\,T} \mathbf{u}_1^k) \sigma_2^{k'} (\mathbf{u}_2^{k'\,T} G \mathbf{v}_1^k)
\end{aligned}$$

Similar proof applies to Eqn. 7.

□