Appendix for “Learning Contrastive Embedding in Low-Dimensional Space”

Shuo Chen[†], Chen Gong[§], Jun Li[§], Jian Yang[§], Gang Niu[†], Masashi Sugiyama[‡]

Abstract

This supplementary document contains additional experiments and all technical proofs for **Theorem 2** and **Theorem 3** in the *NeurIPS’22* paper entitled “Learning Contrastive Embedding in Low-Dimensional Space”. It is indeed the appendix section of the paper. Source code is available at https://github.com/functioncs/CLLR.

A. Additional Experiments

A.1. Parametric Sensitivity

Here we investigate the parametric sensitivities of λ and α in our method. Specifically, we change λ and α in $[0.01, 5]$ and $[1, 20]$, respectively, and we record the classification accuracy of our method on *STL-10* dataset (batch size=256, epochs=100). Tab. 0.1 clearly shows that the accuracy variation of our method is smaller than 1.5.

Similar experiments are conducted on *CIFAR-10* dataset, where we can observe that the accuracy variation of our method is smaller than 2.0. These results clearly demonstrate that the two regularization parameters λ and α are very stable within a given range. It implies that the hyper-parameters of our method can be easily tuned in practice use.

Table 0.1: Parametric sensitivities of λ and α on *STL-10* dataset. Here λ and α are changed in $[0.01, 5]$ and $[1, 20]$, respectively.

λ \ α	**1**	**5**	**10**	**15**	**20**
0.01	78.4	79.3	79.2	78.2	78.0
0.1	78.2	79.1	79.2	78.8	77.9
0.5	77.8	78.6	79.2	**79.4**	79.2
5	78.9	78.9	78.9	78.6	**79.4**

[†] S. Chen and G. Niu are with RIKEN Center for Advanced Intelligence Project (AIP), Japan (E-mail: {shuo.chen.ya@riken.jp, gang.niu.ml@gmail.com}).

[§] C. Gong, J. Li, and J. Yang are with the PCA Lab, Key Lab of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, and Jiangsu Key Lab of Image and Video Understanding for Social Security, School of Computer Science and Engineering, Nanjing University of Science and Technology, China (E-mail: {junli, chen.gong, csjyang}@njust.edu.cn).

[‡] M. Sugiyama is with RIKEN Center for Advanced Intelligence Project (AIP), Japan; and also with the Graduate School of Frontier Sciences, The University of Tokyo, Japan (E-mail: sugi@k.u-tokyo.ac.jp).

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

Table 0.2: Parametric sensitivities of λ and α on *CIFAR-10* dataset. Here λ and α are changed in $[0.01, 5]$ and $[1, 20]$, respectively.

α λ	**1**	**5**	**10**	**15**	**20**
0.01	93.8	94.3	94.2	**95.2**	94.1
0.1	94.2	95.1	**95.2**	**95.2**	94.2
0.5	93.8	93.6	93.2	93.4	94.2
5	93.9	94.9	94.9	94.6	94.4

Table 0.3: Training time of the baseline methods and our proposed method (100 epochs, in hours).

Method	*CIFAR-10*		*ImageNet-100*	
	512	1024	512	1024
SimCLR [3]	2.3	1.3	10.9	5.5
DCL [5]	2.5	1.4	11.2	5.7
CLLR(SimCLR+$\ell_{2,1}$-norm)	2.3	1.4	10.9	5.6
CLLR(SimCLR+nuclear-norm)	2.4	1.5	11.2	5.8
CLLR(DCL+$\ell_{2,1}$-norm)	2.5	1.6	11.3	5.8
CLLR(DCL+nuclear-norm)	2.6	1.6	11.5	5.9

A.2. Running Time Comparsion

As we described in the manuscript, we adopt the sub-gradients of $\ell_{2,1}$-norm and nuclear-norm as the stochastic gradients during the iteration. However, the iteration of nuclear-norm may be time-consuming which involves the s*ingular value decomposition* (SVD) operation [1]. Therefore, here we further provide experiments to record the training time of our method as well as the corresponding baseline method. Specifically,we use four NVIDIA TeslaV100 GPUs to train our method based on SimCLR and DCL with 100 epochs, where the batch size is set to 512 and 1024.

In Tab. 0.3, we can find that the proposed regularizer only brings in little additional time consumption. This is because the gradient calculations of $\ell_{2,1}$-norm $\|\boldsymbol{L}\|_{2,1}$ and nuclear-norm $\|\boldsymbol{L}\|_*$ are independent to the size of training data, so the training time is still acceptable in practice use.

A.3. Comparison with Distillation-Based Contrastive Learning

We may notice that the distillation method can also reduce the dimensionality of contrastive embeddings. However, in the distillation-based CL, the distilled student model is usually supervised by the original teacher model, so the distillation-based CL may naturally inherit improper similarities learned by the original CL. In comparison, our CLLR directly reduces the feature dimensionality of the original CL to avoid / alleviate the improper similarity measure. Therefore, it is worth pointing out that our method is completely different from the distillation-based CL methods.

Here we further provide experiments in Tab. 0.4 to compare our method with the distillation-based CL methods. We select the recent works *wasserstein contrastive representation distillation* (WCoRD) [2] and *complementary relation contrastive distillation* (CRCD) [8]for comparsions, where the output dimensionlaities of their student networks are set to 256-dimension and 512-dimension. We can find that most distilled student models have the close or slightly lower classification accuracy compared with the corresponding baseline teacher models (as reported in their original paper). In comparison, our method can consistently improve the baseline method on all three datasets. Meanwhile, we observe that our method significantly outperforms the distillation-based methods in both 256-dimension and 512-dimension settings.

A.4. Experiments on Negative-Free Contrastive Learning

Although we implement our method on CL models that use both positive and negative samples, our proposed CLLR can also work with negative-free models. We follow the reviewer's suggestion

Table 0.4: Classification accuracy (%, Top5) of the distillation-based methods and our proposed method on *STL-10*, *CIFAR-10*, and *ImageNet-100* datasets (batch size = 512 / 1024, epochs = 500).

Method	*STL-10*		*CIFAR-10*		*ImageNet-100*	
	512	1024	512	1024	512	1024
SimCLR (Teacher)	81.3	82.3	91.3	93.3	77.9	80.5
WCoRD(256-dimension) [2]	80.2	81.3	90.3	90.3	76.9	75.5
WCoRD(512-dimension) [2]	81.2	81.4	92.5	91.4	77.2	79.7
CRCD(256-dimension) [8]	79.4	80.3	89.3	91.3	74.9	81.5
CRCD(512-dimension) [8]	81.4	82.4	92.0	90.4	78.2	79.7
CLLR(SimCLR+nuclear-norm, 256-dimension)	**85.2**	86.4	**93.7**	96.4	**81.2**	84.6
CLLR(SimCLR+nuclear-norm, 512-dimension)	84.4	**87.1**	93.3	**96.5**	80.9	**84.8**

Table 0.5: Classification accuracy (%, Top1 and Top5) of combining our proposed method with negative-free contrastive learning methods on *ImageNet-100* dataset (batch size = 1024 / 4096, epochs = 500).

Method	1024		4096	
	Top1	Top5	Top1	Top5
BYOL [6]	61.3	91.8	74.9	91.9
SimSiam [4]	70.9	91.9	73.6	92.8
CLLR(BYOL+nuclear-norm)	63.1	92.7	**76.5**	93.0
CLLR(SimSiam+nuclear-norm)	**72.2**	**92.9**	75.8	**93.8**

to conduct experiments on negative-free CL baselines (BYOL [6] and SimSiam [4], merely using positive pairs) to validate the effectiveness of our proposed method. As shown in Tab. 0.5, our method can consistently improve the compared methods upon themselves (Top1 and Top5 accuracy on *ImageNet-100* with 500 training epochs and batch size = 1024 / 4096).

A.5. Training Models via Other Optimizers

Since our proposed reconstruction loss and regularizer are differentiable almost everywhere, we can employ some other optimizers such as Adam to minimize the learning objective of our CLLR. Specifically, here use the Adam optimizer to training our model on *CIFAR-10* dataset (batch size = 256), and we record the corresponding training / test errors (%) after 100, 200, and 400 epochs. In Tab. 0.6, we observe that both SGD (learning rate = 5×10^{-3}) and Adam can converge well after 400 epochs. Therefore, our proposed method has good compatibility with existing (stochastic) optimizers.

Table 0.6: Training / test errors (%, Top5) of our method by using SGD and Adam on *CIFAR-10* dataset.

Optimizer	100 epochs	200 epochs	300 epochs	400 epochs
SGD	30.2±5.3 / 35.8±4.3	10.8±2.1 / 15.8±2.3	3.3±1.8 / 10.4±2.3	2.1±1.1 / 6.9±1.3
Adam [7]	20.2±4.3 / 25.4±3.3	14.1±1.9 / 18.8±4.1	3.4±1.5 / 10.5±3.4	2.4±1.2 / 7.2±2.1

B. Proofs

B.1. Derivation for Eq. (3)

According to the definition of gamma function, we have that

$$\begin{aligned}&\lim_{H\to\infty}(\pi^{H/2}/(H\cdot\Gamma(H/2)))/2^{H-1}\\&=\lim_{H\to\infty}(\pi^{H/2}/(H\cdot\int_0^\infty t^{H/2-1}\mathrm{e}^{-t}dt))/2^{H-1}\\&\leq\lim_{H\to\infty}(\pi^{H/2}/(H\cdot\int_1^2 t^{H/2-1}\mathrm{e}^{-t}dt))/2^{H-1}.\end{aligned}\tag{0.1}$$

By further using the *mean-value theorem*, we have

$$\begin{aligned}&\lim_{H\to\infty}(\pi^{H/2}/(H\cdot\int_1^2 t^{H/2-1}\mathrm{e}^{-t}dt))/2^{H-1}\\&\leq\lim_{H\to\infty}(\pi^{H/2}/(H\cdot\mathrm{e}^{-2}))/2^{H-1}\\&\leq\lim_{H\to\infty}\pi^{(H-1)/2}/2^{H-1}.\end{aligned}\tag{0.2}$$

Finally, it is easy to obtain that

$$\lim_{H\to\infty}\pi^{(H-1)/2}/2^{H-1}=\lim_{H\to\infty}(\pi/4)^{(H-1)/2}=0,\tag{0.3}$$

which is Eq. (3) in our manuscript.

B.2. Proof for Theorem 2

Theorem 2. *If the function $\mathcal{F}(\boldsymbol{\Phi},\boldsymbol{L})$ has δ-bounded gradient (i.e., $\|\nabla\mathcal{F}(\boldsymbol{\Phi},\boldsymbol{L})\|_2<\delta$), then we let $\eta=\sqrt{2(\mathcal{F}(\boldsymbol{\Phi}_{(0)},\boldsymbol{L}_{(0)})-\mathcal{F}(\boldsymbol{\Phi}^*,\boldsymbol{L}^*))/(S\delta^2T)}$, and for the iterations in Algorithm 1 we have that*

$$\begin{aligned}&\min_{0\leq t\leq T-1}\mathbb{E}[\,\|\nabla\mathcal{F}(\boldsymbol{\Phi}_{(t)},\boldsymbol{L}_{(t)})\|_2]\\&\leq\sqrt{2S(\mathcal{F}(\boldsymbol{\Phi}_{(0)},\boldsymbol{L}_{(0)})-\mathcal{F}(\boldsymbol{\Phi}^*,\boldsymbol{L}^*))/T)}\delta,\end{aligned}\tag{0.4}$$

where $S>0$ is the lipschitz constant such that $\|\nabla\mathcal{F}(\boldsymbol{\Phi},\boldsymbol{L})-\nabla\mathcal{F}(\boldsymbol{\Phi}',\boldsymbol{L}')\|_2\leq S\|[\boldsymbol{\Phi},\boldsymbol{L}]-[\boldsymbol{\Phi}',\boldsymbol{L}']\|_2$.

Proof. Firstly, by using the lipschitz continuity of $\mathcal{F}(\boldsymbol{\Phi},\boldsymbol{L})$ we have that

$$\begin{aligned}&\mathbb{E}[\,\mathcal{F}(\boldsymbol{\Phi}_{(t+1)},\boldsymbol{L}_{(t+1)})]-\mathbb{E}[\,\mathcal{F}(\boldsymbol{\Phi}_{(t)},\boldsymbol{L}_{(t)})]\\&\leq\mathbb{E}[\,\|\nabla\mathcal{F}(\boldsymbol{\Phi}_{(t+1)},\boldsymbol{L}_{(t+1)})-([\boldsymbol{\Phi}_{(t+1)},\boldsymbol{L}_{(t+1)}]-[\boldsymbol{\Phi}_{(t)},\boldsymbol{L}_{(t)}])\|_2^2\\&+S/2\|[\boldsymbol{\Phi}_{(t+1)},\boldsymbol{L}_{(t+1)}]-[\boldsymbol{\Phi}_{(t)},\boldsymbol{L}_{(t)}]\|_2^2]\\&\leq-\eta_t\mathbb{E}[\nabla\mathcal{F}(\boldsymbol{\Phi}_{(t)},\boldsymbol{L}_{(t)})\|_2^2]+(S\eta_t^2/2)\mathbb{E}[\|\nabla\mathcal{F}_{b_i}(\boldsymbol{\Phi}_{(t)},\boldsymbol{L}_{(t)})\|_2^2]\\&\leq-\eta_t\mathbb{E}[\nabla\mathcal{F}(\boldsymbol{\Phi}_{(t)},\boldsymbol{L}_{(t)})\|_2^2]+(S\eta_t^2/2)\delta^2,\end{aligned}\tag{0.5}$$

where the second inequality follows from the fact that $[\boldsymbol{\Phi}_{(t+1)},\boldsymbol{L}_{(t+1)}]$ is updated by Algorithm 1. Then, we have that

$$\mathbb{E}[\nabla\mathcal{F}(\boldsymbol{\Phi}_{(t)},\boldsymbol{L}_{(t)})\|_2^2]\leq(1/\eta_t)\mathbb{E}[\mathcal{F}(\boldsymbol{\Phi}_{(t)},\boldsymbol{L}_{(t)})-\mathcal{F}(\boldsymbol{\Phi}_{(t+1)},\boldsymbol{L}_{(t+1)})]+(L\eta_t/2)\delta^2,\tag{0.6}$$

and thus

$$\begin{cases}\mathbb{E}[\nabla\mathcal{F}(\boldsymbol{\Phi}_{(0)},\boldsymbol{L}_{(0)})\|_2^2]\leq(1/\eta_0)\mathbb{E}[\mathcal{F}(\boldsymbol{\Phi}_{(0)},\boldsymbol{L}_{(0)})-\mathcal{F}(\boldsymbol{\Phi}_{(1)},\boldsymbol{L}_{(1)})]+(S\eta_0/2)\delta^2,\\\mathbb{E}[\nabla\mathcal{F}(\boldsymbol{\Phi}_{(1)},\boldsymbol{L}_{(1)})\|_2^2]\leq(1/\eta_1)\mathbb{E}[\mathcal{F}(\boldsymbol{\Phi}_{(1)},\boldsymbol{L}_{(1)})-\mathcal{F}(\boldsymbol{\Phi}_{(2)},\boldsymbol{L}_{(2)})]+(S\eta_1/2)\delta^2,\\\dots\\\mathbb{E}[\nabla\mathcal{F}(\boldsymbol{\Phi}_{(T-1)},\boldsymbol{L}_{(T-1)})\|_2^2]\leq\frac{1}{\eta_{T-1}}\mathbb{E}[\mathcal{F}(\boldsymbol{\Phi}_{(T-1)},\boldsymbol{L}_{(T-1)})-\mathcal{F}(\boldsymbol{\Phi}_{(T)},\boldsymbol{L}_{(T)})]+\frac{S\eta_{T-1}}{2}\delta^2.\end{cases}\tag{0.7}$$

Finally, we sum all inequalities in the above Eq. (0.7) and letting $\eta_0 = \eta_1 = \cdots = \eta_{T-1} = \eta$. Then we have

$$\begin{aligned}
&\min_{0 \leq t \leq T-1} \mathbb{E}[\, \|\nabla \mathcal{F}(\boldsymbol{\Phi}_{(t)}, \boldsymbol{L}_{(t)})\|_2] \\
&\leq \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[\, \|\nabla \mathcal{F}(\boldsymbol{\Phi}_{(t)}, \boldsymbol{L}_{(t)})\|_2] + (S\eta/2)\delta^2 \\
&\leq \frac{1}{T\eta} \mathbb{E}[\mathcal{F}(\boldsymbol{\Phi}_{(0)}, \boldsymbol{L}_{(t)}) - \mathcal{F}(\boldsymbol{\Phi}_{(t)}, \boldsymbol{L}_{(t)})] + (S\eta/2)\delta^2 \\
&\leq \frac{1}{T\eta} (\mathcal{F}(\boldsymbol{\Phi}_{(0)}, \boldsymbol{L}_{(t)}) - \mathcal{F}(\boldsymbol{\Phi}^*, \boldsymbol{L}^*)) + (S\eta/2)\delta^2 \\
&\leq \frac{1}{\sqrt{T}} ((\mathcal{F}(\boldsymbol{\Phi}_{(0)}, \boldsymbol{L}_{(t)}) - \mathcal{F}(\boldsymbol{\Phi}^*, \boldsymbol{L}^*))/c + (Sc/2)\delta^2),
\end{aligned} \tag{0.8}$$

where $c = \eta\sqrt{T}$. We set $c = \sqrt{2(\mathcal{F}(\boldsymbol{\Phi}_{(0)}, \boldsymbol{L}_{(0)}) - \mathcal{F}(\boldsymbol{\Phi}^*, \boldsymbol{L}^*))/(S\delta^2)}$, and we have

$$\min_{0 \leq t \leq T-1} \mathbb{E}[\, \|\nabla \mathcal{F}(\boldsymbol{\Phi}_{(t)}, \boldsymbol{L}_{(t)})\|_2] \leq \sqrt{2S(\mathcal{F}(\boldsymbol{\Phi}_{(0)}, \boldsymbol{L}_{(0)}) - \mathcal{F}(\boldsymbol{\Phi}^*, \boldsymbol{L}^*))/T}\,\delta, \tag{0.9}$$

which completes the proof. □

B.3. Proof for Theorem 3

Theorem 3. *For any given* $n+1$ *i.i.d. random data points* $\boldsymbol{x},\, \boldsymbol{x}_1,\, \boldsymbol{x}_2,\, \ldots,\, \boldsymbol{x}_n \in \mathbb{R}^m$*, we denote that* $\mathcal{D}^{\max}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}} = \max\{\mathcal{D}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}}(\boldsymbol{x},\, \boldsymbol{x}_i) | i = 1, 2, \ldots, n\}$ *and* $\mathcal{D}^{\min}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}} = \min\{\mathcal{D}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}}(\boldsymbol{x},\, \boldsymbol{x}_i) | i = 1, 2, \ldots, n\}$*, and we have that*

$$\mathcal{P}\left\{(\mathcal{D}^{\max}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}} - \mathcal{D}^{\min}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}})/\mathcal{D}^{\min}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}} \geq \alpha\lambda C(\mathcal{X})\right\} = 1, \tag{0.10}$$

where $\mathcal{D}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}}(\boldsymbol{x},\, \boldsymbol{x}_i) = \|\widehat{\boldsymbol{L}}\widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}\widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i)\|_2/\mathrm{rank}(\widehat{\boldsymbol{L}})$*, and parameters* $\widehat{\boldsymbol{\Phi}}$ *and* $\widehat{\boldsymbol{L}}$ *are learned from Eq. (13).*

Proof. As $\widehat{\boldsymbol{\Phi}}$ and $\widehat{\boldsymbol{L}}$ are iterated by the optimization algorithm, we have

$$\begin{aligned}
&\mathcal{L}_{\text{NCE}}(\widehat{\boldsymbol{\Phi}}) + \lambda \mathbb{E}_{\boldsymbol{x} \in \mathcal{X}}[\, \|\widehat{\boldsymbol{L}}^\top \widehat{\boldsymbol{L}} \cdot \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{\Phi}}(\boldsymbol{x})\|_2^2\,] + \alpha\lambda \mathcal{R}(\widehat{\boldsymbol{\Phi}}, \widehat{\boldsymbol{L}}) \\
&\leq \mathcal{L}_{\text{NCE}}(\boldsymbol{\Phi}_{(0)}) + \lambda \mathbb{E}_{\boldsymbol{x} \in \mathcal{X}}[\, \|\boldsymbol{L}_{(0)}^\top \boldsymbol{L}_{(0)} \cdot \boldsymbol{\Phi}_{(0)}(\boldsymbol{x}) - \boldsymbol{\Phi}_{(0)}(\boldsymbol{x})\|_2^2\,] + \alpha\lambda \mathcal{R}(\boldsymbol{\Phi}_{(0)}, \boldsymbol{L}_{(0)}),
\end{aligned} \tag{0.11}$$

which implies that

$$\begin{aligned}
\mathcal{R}(\widehat{\boldsymbol{\Phi}}, \widehat{\boldsymbol{L}}) &\leq \frac{1}{\alpha\lambda} \left(\mathcal{F}(\boldsymbol{\Phi}_{(0)}, \boldsymbol{L}_{(0)}) - \mathcal{L}_{\text{NCE}}(\widehat{\boldsymbol{\Phi}}) - \lambda \mathbb{E}_{\boldsymbol{x} \in \mathcal{X}}[\, \|\widehat{\boldsymbol{L}}^\top \widehat{\boldsymbol{L}} \cdot \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{\Phi}}(\boldsymbol{x})\|_2^2\,] \right) \\
&= \frac{1}{\alpha\lambda} c_1 - \frac{1}{\alpha} c_2 + c_3 \\
&= \frac{1}{\alpha} \left(\frac{1}{\lambda} c_1 - c_2 \right) + c_3,
\end{aligned} \tag{0.12}$$

where

$$\begin{cases}
c_1 = \mathcal{L}_{\text{NCE}}(\boldsymbol{\Phi}_{(0)}) - \mathcal{L}_{\text{NCE}}(\widehat{\boldsymbol{\Phi}}), \\
c_2 = \mathbb{E}_{\boldsymbol{x} \in \mathcal{X}}[\, \|\boldsymbol{L}_{(0)}^\top \boldsymbol{L}_{(0)} \cdot \boldsymbol{\Phi}_{(0)}(\boldsymbol{x}) - \boldsymbol{\Phi}_{(0)}(\boldsymbol{x})\|_2^2 - \mathbb{E}_{\boldsymbol{x} \in \mathcal{X}}[\, \|\widehat{\boldsymbol{L}}^\top \widehat{\boldsymbol{L}} \cdot \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{\Phi}}(\boldsymbol{x})\|_2^2, \\
c_3 = \mathcal{R}(\boldsymbol{\Phi}_{(0)}, \boldsymbol{L}_{(0)}).
\end{cases} \tag{0.13}$$

Then we have that $\|\widehat{\boldsymbol{L}}\|_{2,1} \leq \frac{1}{\alpha}\left(\frac{1}{\lambda}c_1 - c_2\right) + c_3$ and $\|\widehat{\boldsymbol{L}}\|_* \leq \frac{1}{\alpha}\left(\frac{1}{\lambda}c_1 - c_2\right) + c_3$, respectively. Therefore, we have $\|\widehat{\boldsymbol{L}}\|_{2,0} \leq k_1\left(\frac{1}{\alpha}\left(\frac{1}{\lambda}c_1 - c_2\right) + c_3\right)$ and $\mathrm{rank}(\widehat{\boldsymbol{L}}) \leq k_2\left(\frac{1}{\alpha}\left(\frac{1}{\lambda}c_1 - c_2\right) + c_3\right)$. It implies that the pairwise distance $\mathcal{D}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}}(\boldsymbol{x},\, \boldsymbol{x}_i)$ satisfies that

$$
\begin{aligned}
&\frac{\mathcal{D}^{\max}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}} - \mathcal{D}^{\min}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}}}{\mathcal{D}^{\min}_{\widehat{\boldsymbol{\Phi}},\widehat{\boldsymbol{L}}}} \\
&= \frac{\max_{i=1,\ldots,n} \sqrt{\sum_{j=1}^{H} (\widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i))}/\mathrm{rank}(\widehat{\boldsymbol{L}}) - \min_{i=1,\ldots,n} \sqrt{\sum_{j=1}^{H} (\widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i))}/\mathrm{rank}(\widehat{\boldsymbol{L}})}{\min_{i=1,2,\ldots,n} \sqrt{\sum_{j=1}^{H} (\widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i))}/\mathrm{rank}(\widehat{\boldsymbol{L}})} \\
&= \frac{\max_{i=1,2,\ldots,n} \sqrt{\sum_{j=1}^{H} (\widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i))}/\mathrm{rank}(\widehat{\boldsymbol{L}})}{\min_{i=1,2,\ldots,n} \sqrt{\sum_{j=1}^{H} (\widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i))}/\mathrm{rank}(\widehat{\boldsymbol{L}})} - 1 \\
&\geq \frac{\max_{i=1,2,\ldots,n} \sqrt{\sum_{j=1}^{H} (\widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i))}/k_2 \left(\frac{1}{\alpha}\left(\frac{1}{\lambda}c_1 - c_2\right) + c_3\right)}{\min_{i=1,2,\ldots,n} \sqrt{\sum_{j=1}^{H} (\widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i))}/\mathrm{rank}(\widehat{\boldsymbol{L}})} \\
&= \frac{Q}{k_2 \left(\frac{1}{\alpha}\left(\frac{1}{\lambda}c_1 - c_2\right) + c_3\right)} \\
&\geq \frac{\alpha\lambda Q}{k_2 c_1},
\end{aligned}
\tag{0.14}
$$

where

$$
Q = \frac{\max_{i=1,2,\ldots,n} \sqrt{\sum_{j=1}^{H} (\widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i))}}{\min_{i=1,2,\ldots,n} \sqrt{\sum_{j=1}^{H} (\widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}) - \widehat{\boldsymbol{L}}^{(j)} \widehat{\boldsymbol{\Phi}}(\boldsymbol{x}_i))}/\mathrm{rank}(\widehat{\boldsymbol{L}})}. \tag{0.15}
$$

Finally, we let $C(\mathcal{X}) = Q/(k_2 c_1)$ and complete the proof. □

References

[1] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004. (document)

[2] Liqun Chen, Dong Wang, Zhe Gan, Jingjing Liu, Ricardo Henao, and Lawrence Carin. Wasserstein contrastive representation distillation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16296–16305, 2021. (document), 0.4

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, pages 1597–1607, 2020. 0.3

[4] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15750–15758, 2021. 0.5, (document)

[5] Ching-Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 33, 2020. 0.3

[6] Jean-Bastien Grill, Florian Strub, Florent Altche, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:21271–21284, 2020. 0.5, (document)

[7] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations (ICLR)*, 2018. 0.6

[8] Jinguo Zhu, Shixiang Tang, Dapeng Chen, Shijie Yu, Yakun Liu, Mingzhe Rong, Aijun Yang, and Xiaohua Wang. Complementary relation contrastive distillation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9260–9269, 2021. (document), 0.4