Are Neurons Actually Collapsed? On the Fine-Grained Structure in Neural Representations

Yongyi Yang
University of Michigan
yongyi@umich.edu

Jacob Steinhardt
University of California, Berkeley
jsteinhardt@berkeley.edu

Wei Hu
University of Michigan
vvh@umich.edu

Abstract

Recent work has observed an intriguing "Neural Collapse" phenomenon in well-trained neural networks, where the last-layer representations of training samples with the same label collapse into each other. This suggests that the last-layer representations are completely determined by the labels, and do not depend on the intrinsic structure of input distribution. We provide evidence that this is not a complete description, and that the apparent collapse hides important fine-grained structure in the representations. Specifically, even when representations apparently collapse, the small amount of remaining variation can still faithfully and accurately captures the intrinsic structure of input distribution. As an example, if we train on CIFAR-10 using only 5 coarse-grained labels (by combining two classes into one super-class) until convergence, we can reconstruct the original 10-class labels from the learned representations via unsupervised clustering. The reconstructed labels achieve 93% accuracy on the CIFAR-10 test set, nearly matching the normal CIFAR-10 accuracy for the same architecture. Our findings show concretely how the structure of input data can play a significant role in determining the fine-grained structure of neural representations, going beyond what Neural Collapse predicts.

1 Introduction

Much of the success of deep neural networks has, arguably, been attributed to their ability to learn useful *representations*, or *features*, of the data (Rumelhart et al., 1985). Although neural networks are often trained to optimize a single objective function with no explicit requirements on the inner representations, there is ample evidence suggesting that these learned representations contain rich information about the input data (Levy & Goldberg, 2014; Olah et al., 2017). As a result, formally characterizing and understanding the structural properties of neural representations is of great theoretical and practical interest, and can provide insights on how deep learning works and how to make better use of these representations.

One intriguing phenomenon recently discovered by Papyan et al. (2020) is *Neural Collapse*, which identifies structural properties of last-layer representations during the terminal phase of training (i.e. after zero training error is reached). The simplest of these properties is that the last-layer representations for training samples with the same label collapse into a single point, which is referred to as "variability collapse (NC1)." This is surprising, since the collapsed structure is not necessary to achieve small training or test error, yet it arises consistently in standard architectures trained on standard classification datasets.

A series of recent papers were able to theoretically explain Neural Collapse under a simplified model called the *unconstrained feature model* or *layer-peeled model* (see Section 2 for a list of references).

I Can't Believe It's Not Better Workshop at NeurIPS 2022.

In this model, the last-layer representation of each training sample is treated as a free optimization variable and therefore the training loss essentially has the form of a matrix factorization. Under a variety of different setups, it was proved that the solution to this simplified problem should satisfy Neural Collapse. Although Neural Collapse is relatively well understood in this simplified model, this model completely ignores the role of the input data because the loss function is independent of the input data. Conceptually, this suggests that **Neural Collapse is only determined by the labels** and may happen regardless of the input data distribution. Zhu et al. (2021) provided further empirical support of this claim via a random labeling experiment.

On the other hand, it is conceivable that the **intrinsic structure of the input distribution *should* play a role** in determining the structure of neural net representations. For example, if a class contains a heterogeneous set of input data (such as different subclasses), it is possible that their heterogeneity is also respected in their feature representations (Sohoni et al., 2020). However, this appears to contradict Neural Collapse, because Neural Collapse would predict that all the representations collapse into the same point as long as they have the same class label. This dilemma motivates us to study the following main question in this paper:

How can we reconcile the roles of the ***intrinsic structure of input distribution*** *vs. the* ***explicit structure of the labels*** *in determining the last-layer representations in neural networks?*

Our methodology and findings. To study the above question, we design experiments to manually create a *mismatch* between the intrinsic structure of the input distribution and the explicit labels provided for training in standard classification datasets and measure how the last-layer representations behave in response to our interventions. This allows us to isolate the effect of the input distribution from the effect of labels. Our main findings are summarized below.

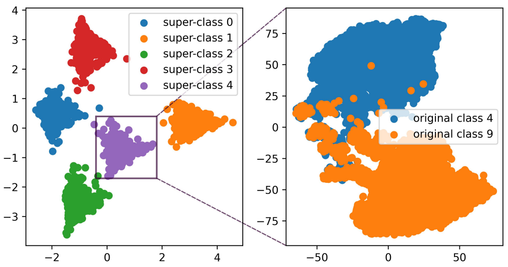

Figure 1: Fine-grained clustering structure of the last-layer representations of ResNet-18 trained on Coarse CIFAR-10 (5 super-classes). Left figure: PCA visualization for all training samples. Right figure: t-SNE visualization for all training samples in super-class 4 (which consists of original classes 4 and 9).

First, both the intrinsic structure of the input distribution and the explicit labels provided in training clearly affect the structure of the last-layer representations. The effect of input distribution emerges earlier in training, while the effect of labels appears at a later stage. For example, for both Coarse CIFAR-10 and Fine CIFAR-10, at some point the representations naturally form 10 clusters according to the original CIFAR-10 labels (which comes from the intrinsic input structure), even though 5 or 20 different labels are provided for training. Later in training (after 100% training accuracy is reached), the representations collapse into 5 or 20 clusters driven by the explicit labels provided, as predicted by Neural Collapse.

Second, even after Neural Collapse has occurred according to the explicit label information, the seemingly collapsed representations corresponding to each label can still exhibit *fine-grained structures* determined by the input distribution. As an illustration, Figure 1 visualizes the representations from the last epoch of training a ResNet-18 on Coarse CIFAR-10. While globally there are 5 separated clusters as predicted by Neural Collapse, if we zoom in on each cluster, it clearly consists of two subclusters which correspond to the original CIFAR-10 classes. We also find that this phenomenon persists even after a very long training period (e.g. 1,000 epochs), indicating that the effect of input distribution is not destroyed by that of the labels, at least not within a normal training budget.

To further validate our finding that significant input information is present in the last-layer representations despite Neural Collapse, we perform a simple *Cluster-and-Linear-Probe (CLP)* procedure on the representations from ResNet-18 trained on Coarse CIFAR-10, in which we use an unsupervised clustering method to reconstruct the original labels, and then train a linear classifier on top of these representations using the reconstructed labels. We find that CLP can achieve $> 93\%$ accuracy on the original CIFAR-10 test set, matching the standard accuracy of ResNet-18, even though only 5 coarse labels are provided the entire time.

Takeaway. While Neural Collapse is an intriguing phenomenon that consistently happens in well-trained neural networks, we provide concrete evidence showing that it is not the most comprehensive description of the behavior of last-layer representations in practice, as it fails to capture the possible fine-grained properties determined by the intrinsic structure of the input distribution.

2 Related Work

The Neural Collapse phenomenon was originally discovered by Papyan et al. (2020), and has led to a series of further investigations.

A number of papers Fang et al. (2021); Lu & Steinerberger (2020); Wojtowytsch et al. (2020); Mixon et al. (2022); Zhu et al. (2021); Ji et al. (2021); Han et al. (2021); Zhou et al. (2022); Tirer & Bruna (2022); Yaras et al. (2022) studied a simplified "unconstrained feature model", also known as "layer-peeled model", and showed that Neural Collapse provably happens under a variety of settings. This model treats the last-layer representations of all training samples as free optimization variables. By doing this, the loss function no longer depends on the input data, and therefore this line of work is unable to capture any effect of the input distribution on the structure of the representations. Ergen & Pilanci (2021); Tirer & Bruna (2022); Weinan & Wojtowytsch (2022) considered more complicated models but still did not incorporate the role of the input distribution.

Hui et al. (2022) studied the connection of Neural Collapse to generalization and concluded that Neural Collapse occurs only on the training set, not on the test set. Galanti et al. (2021) found that Neural Collapse does generalize to test samples as well as new classes, and used this observation to study transfer learning and few-shot learning.

Sohoni et al. (2020) observed that the last-layer representations of different subclasses within the same class are often separated into different clusters, and used this observation to design an algorithm for improving group robustness. However, it is unclear whether their observation happens in the Neural Collapse regime, while we find that fine-grained structure in representations can co-exist with Neural Collapse. Furthermore, Sohoni et al. (2020) looked at settings in which different subclasses have different accuracies and attributed the representation separability phenomenon to this performance difference. On the other hand, we find that representation separability happens even when there is no performance gap between different subclasses.

3 Preliminaries and Setup

This section introduces the notation used throughout this paper, as well as a more detailed explanation of Neural Collapse and a clarification of the experiment setup.

For a classification task dataset, we denote it by $\mathcal{D} = \{(\boldsymbol{x}_k, y_k)\}_{k=1}^n$, where $(\boldsymbol{x}_k, y_k) \in \mathbb{R}^{d'} \times [C]$ is a pair of input features and label, n is the number of samples, d' is the input dimension, and C is the number of classes.

For a given neural network, we denote the last-layer representation by $H \in \mathbb{R}^{n \times d}$, i.e. the hidden representation before the final linear transformation, where d is the last-layer dimensionality. For an original class $c \in [C]$, we denote the number of samples in class c by n_c, and the last-layer representation of k-th sample in class c by $\boldsymbol{h}_k^{(c)}$.

3.1 Preliminaries of Neural Collapse

Neural Collapse characterizes 4 phenomena, named NC1-NC4. Here we introduce the first two which concern the structure of the last-layer representations.

NC1, or variability collapse, predicts that the variance of last-layer representations of samples within the same class vanishes as training proceeds. Formally, it can be measured by $\mathsf{NC}_1 = \frac{1}{C}\mathrm{Tr}\left(\Sigma_W \Sigma_B^\dagger\right)$ (Papyan et al., 2020; Zhu et al., 2021), which should tend to 0. The Σ_W and Σ_B are defined as

$$\Sigma_W = \frac{1}{C}\sum_{c \in [C]} \frac{1}{n_c}\sum_{i=1}^{n_c} \left(\boldsymbol{h}_i^{(c)} - \boldsymbol{\mu}_c\right)\left(\boldsymbol{h}_i^{(c)} - \boldsymbol{\mu}_c\right)^\top \text{ and } \Sigma_B = \frac{1}{C}\sum_{c \in [C]} \left(\boldsymbol{\mu}_c - \boldsymbol{\mu}_G\right)\left(\boldsymbol{\mu}_c - \boldsymbol{\mu}_G\right)^\top, \tag{1}$$

where $\boldsymbol{\mu}_c = \frac{1}{n_c}\sum_{k=1}^{n_c} \boldsymbol{h}_k^{(c)}$ are the class means and $\boldsymbol{\mu}_G = \frac{1}{n}\sum_{k=1}^{n} \boldsymbol{h}_k$ is the global mean.

NC2 predicts that the class means converge to a special structure, i.e. their normalized covariance converges to the Simplex Equiangular Tight Frame (ETF). This can be characterize by

$$\mathsf{NC}_2 \overset{\text{def}}{=} \left\| \frac{MM^\top}{\|MM^\top\|_{\mathcal{F}}} - \frac{1}{\sqrt{C-1}} \left(I - \frac{1}{C}\mathbf{1}_C\mathbf{1}_C^\top \right) \right\|_{\mathcal{F}} \to 0 \tag{2}$$

during training, where M is the stack of centralized class-means, whose c-th row is $\boldsymbol{\mu}_c - \boldsymbol{\mu}_G$, and $\mathbb{1}_C \in \mathbb{R}^C$ is an all-one vector in length C, I is the identity matrix.

3.2 Experiment Setup

In our experiment, we explore the role of input distribution and labels through assigning coarser or finer labels to each sample, and then explore the structure of last-layer representation of a model trained on the dataset with coarse or fine labels and see to what extent the information of original labels are preserved.

The coarse labels are created in the following way. First choose a number $\tilde{C}$ which is divisible by C, and create coarse labels by $\tilde{y}_k = y_k \bmod \tilde{C}$, which merges the classes whose index have the same modulus w.r.t. $\tilde{C}$ and thus create $\tilde{C}$ super-classes. Since the original index of classes generally has no special meanings, this process should act similar to randomly merging classes[1]. We say the samples with the same coarse label belongs to the same super-class, and call the dataset $\tilde{\mathcal{D}} = \{(x_k, \tilde{y}_k)\}_{k=1}^{n}$ the coarse dataset, which is the dataset we used to train the model.

To create fine labels, we randomly split each class into two sub-classes. Specifically, the fine labels are created by $\hat{y}_k = y_k + \beta C$, where $\hat{y}_k$ is the fine label of sample k and C is the number of original classes and β is a Bernoulli Variable. This process result in a dataset $\hat{\mathcal{D}} = \{(x_k, \hat{y}_k)\}_{k=1}^{n}$ with $2C$ classes. Same as before, we call $\hat{\mathcal{D}}$ the fine dataset.

4 Exploring the Fine-Grained Representation Structure with Coarse CIFAR-10

In this section, we experiment with coarsely labeled datasets, using Coarse CIFAR-10 as an illustrative example, and defer the experiments with fine labels to appendix. Specifically, the model is trained on the training set of Coarse CIFAR-10 for a certain number of steps that is sufficient for the model to converge. We then take the last-layer representations of the model during training and explore their structure.

In order to make an exhaustive observation, the experiment is repeated using different learning rates and weight-decay rates. Specifically, we choose the learning rate of training in $\{10^{-1}, 10^{-2}, 10^{-3}\}$ and weight-decay rate in $\{5 \times 10^{-3}, 5 \times 10^{-4}, 5 \times 10^{-5}\}$ and permute all possible combinations of them. The experiment is also conducted on multiple datasets and network architectures. Due to space limit, in this section we focus on experiment of ResNet-18 on Coarse CIFAR-10, where the original $C = 10$ and the number of coarse labels is chosen as $\tilde{C} = 5$. For training hyper-parameters, we only report results for two representative hyper-parameter combinations, referred to as config #1 and config #2. The learning rate of config #1 is set to 0.01 and weight-decay rate is 5×10^{-4}, and for config #2 the learning rate is 0.01 and weight decay rate is 5×10^{-5}. Both settings reach 0 training error quickly during training. We defer complete experiment results to appendix.

As a preliminary result, we first show that under our experiment settings, Neural Collapse does happen, i.e. the representation does converge to 5 clusters and the class-means form a Simplex ETF structure. Specifically, we measure NC_1 and NC_2 defined in Section 3.1 for both config #1 and config #2, with C replaced by $\tilde{C}$ since we are calculating it on coarse dataset. The results are shown in Figure 2, which matches previous results in Papyan et al. (2020); Zhu et al. (2021), indicating that Neural Collapse does happen under both settings.

4.1 Class Distance

We first look at the average square Euclidian distance of last-layer representations between each two *original* classes. Formally, we calculate a class distance matrix $D \in \mathbb{R}^{C \times C}$, whose entries are $D_{i,j} = \frac{1}{n_i n_j} \sum_{u=1}^{n_i} \sum_{v=1}^{n_j} \left\| \boldsymbol{h}_u^{(i)} - \boldsymbol{h}_v^{(j)} \right\|_2^2$, for all $i, j \in [C]$, where $\boldsymbol{h}_k^{(c)}$ represents the last-layer representation of the k-th sample of super-class u.

[1] We adopt this determined process for simplicity and reproducibility. However, we do provide additional results with random merging in appendix.

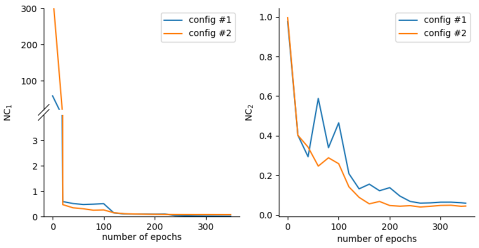

Figure 2: The value of NC_1 and NC_2 w.r.t. number of training epochs.

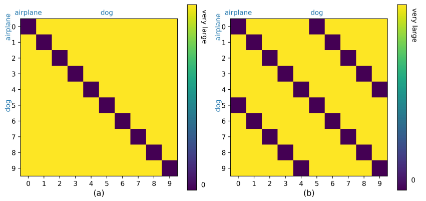

Figure 3: An illustration of predicted class distance matrix heatmaps of Coarse CIFAR-10, each row and column represents an original class. (a): If input distribution dominates the last-layer representations. (b): If Neural Collapse dominates the last-layer representations.

Since the model is trained on coarse dataset, Neural Collapse asserts that for every original class pair i, j in the same super-class (including the case of $i = j$), the class distance $D_{i,j}$ should be very small. In Coarse CIFAR-10, this will result in three darks lines (let darker color represents lower value) in the heatmap of D since each super class contains two original classes, as illustrated in Figure 3 (b). For example, in Coarse CIFAR-10 the original class "airplane" and "dog" both belong to the super class "airplane & dog", therefore per Neural Collapse's prediction, their last-layer representations would collapse to each other, making the average square distance extremely small compared to other entries. In contrast, if the last-layer representations perfectly reflects the distribution of input, i.e. original classes, the class distance matrix should be a diagonal matrix as shown in Figure 3 (a), because the last-layer representation of samples in each original class only collapse to the class-mean of this original class.

Figure 4 display the heatmap of class distance matrix D, arranged by number of training epochs. From the results there are two surprising observations can be made: Firstly, there are indeed three dark lines at the final stage, however, the three dark lines does not show up simultaneously, and the central line – represents the samples in the same original classes – shows up earlier. The second observation is, even in the final stage where the training error is zero, the three lines can be not of the same degree of darkness, with the central line shallower, and this is especially apparent in the second row of Figure 4.

Those observations suggests that, the actual behaviour of the last-layer representations is between the cases predicted in Figure 3 (a) and (b): the input distribution and training label both have an impact on the distribution of the last-layer representations, and although the time of them to show up is different, both of them can be present even after reaching zero training error for a long time. These observations suggest both the existence of Neural Collapse and the inadequacy of Neural Collapse to completely describe the behaviour of last-layer representations.

4.2 Visualization

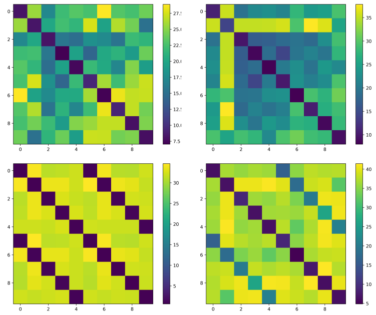

Figure 4: The heatmap of class distance matrix. Up: config #1. Down: config #2. Left: epoch number = 20. Right: epoch number = 350.

In this subsection, we take a closer look at the last-layer representations of the model at the end of training by reducing the dimensionality of the last-layer representations to 2 through t-SNE (Van der Maaten & Hinton, 2008) and visualize them. Specifically, we visualize each super-class separately, but color the samples whose original labels are different with different colors.

Due to space limiation, we only show the visualization results of config #1 here in Figure 5 and defer the complete result to appendix. It can be observed that, there is distinguishable difference in the distribution of original labels in both config #1. This suggests that, the input distribution information, i.e. the original label information, is well preserved in the last-layer representations under both settings, albeit for config #1 it seems it can not be distinguished through distance matrix in Section 4.1.

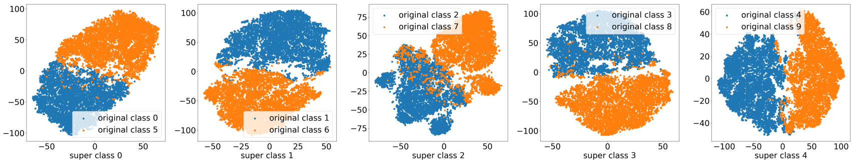

Figure 5: The t-SNE visualization of config #1. Each grid is a super-class.

4.3 Learning CIFAR-10 from 5 Coarse Labels

As the results in Section 4.2 suggest, even after the training accuracy has reached 100% for a long time, the samples within each super-class still exhibit a clear structure per their original class, and those structures act as clusters after reducing dimensionality. Inspired by this observation, we perform a Cluster-and-Linear-Probe (CLP) test to evaluate to what extent the original class information is preserved in the last-layer representations. In CLP, we use the representations learned on Coarse CIFAR-10 to reconstruct 10 labels and run a linear probe with the reconstructed labels. Specifically, we first use t-SNE to reduce the dimensionality to 2 and then use KMeans to find 2 clusters of the dimensionality-reduced representations within each super-class. We use the clusters as reconstructed labels to do a linear probe. In linear probe, we train a linear classifier on top of the previously learned representation H on the training set with reconstructed labels and evaluate the learned linear classifier on original test set. Notice that because we don't know the mapping of reconstructed classes to true original classes, we permute each possible mapping and report the highest performance. We also train a linear probe with original training labels as a comparison. The results of config #1 and config #2 are shown in Figures 6 and 7 respectively.

In most cases, the performance of CLP on original test set is comparable with linear probe trained on true original labels or even with models originally trained on CIFAR-10. Notice that the representation H is obtained through the model trained with coarse labels, and the label reconstruction only uses information of H and the number of original classes. This means we can achieve a very high performance on original test set even if we only have access to almost only the information of coarse labels. This result further confirms that the input distribution can play an important role in the last-layer representations.

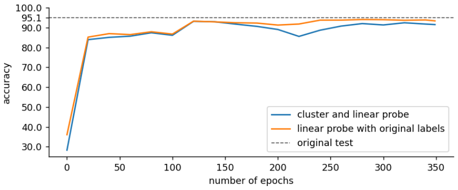

Figure 6: The CLP result of config #1. "original test" is the highest test set accuracy achieved by ResNet18 trained on original CIFAR-10 with the same training hyper-parameters of config #1.

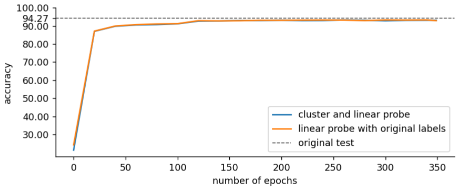

Figure 7: The CLP result of config #2. "original test" is the highest test set accuracy achieved by ResNet18 trained on original CIFAR-10 with the same training hyper-parameters of config #2.

5 Discussion

In this paper, we initiated the study of the role of the intrinsic structure of the input data distribution on the last-layer representations of neural networks, and in particular, how to reconcile it with the Neural Collapse phenomenon, which is only driven by the explicit labels provided in the training procedure. Through a series of experiments, we provide concrete evidence that the representations can exhibit clear fine-grained structure despite their apparent collapse. While Neural Collapse is an intriguing phenomenon and deserves further studies to understand its cause and consequences, our work calls for more scientific investigations of the structure of neural representations that go beyond Neural Collapse.

We note that the fine-grained representation structure we observed depends on the inductive biases of the network architecture and the training algorithm, and may not appear universally. In our experiments on Coarse CIFAR-10, we observe the fine-grained structure for ResNet and DenseNet, but not for VGG (see appendix for extended results). We also note that certain choices of learning rate and weight-decay rate lead to stronger fine-grained structure than others. We leave a thorough investigation of such subtlety for future work.

References

Tolga Ergen and Mert Pilanci. Revealing the structure of deep neural networks via convex duality. In *International Conference on Machine Learning*, pp. 3004–3014. PMLR, 2021.

Cong Fang, Hangfeng He, Qi Long, and Weijie J Su. Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training. *Proceedings of the National Academy of Sciences*, 118(43):e2103091118, 2021.

Tomer Galanti, András György, and Marcus Hutter. On the role of neural collapse in transfer learning. *arXiv preprint arXiv:2112.15121*, 2021.

XY Han, Vardan Papyan, and David L Donoho. Neural collapse under mse loss: Proximity to and dynamics on the central path. *arXiv preprint arXiv:2106.02073*, 2021.

Like Hui, Mikhail Belkin, and Preetum Nakkiran. Limitations of neural collapse for understanding generalization in deep learning. *arXiv preprint arXiv:2202.08384*, 2022.

Wenlong Ji, Yiping Lu, Yiliang Zhang, Zhun Deng, and Weijie J Su. An unconstrained layer-peeled perspective on neural collapse. *arXiv preprint arXiv:2110.02796*, 2021.

Omer Levy and Yoav Goldberg. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pp. 171–180, Ann Arbor, Michigan, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-1618. URL `https://aclanthology.org/W14-1618`.

Jianfeng Lu and Stefan Steinerberger. Neural collapse with cross-entropy loss. *arXiv preprint arXiv:2012.08465*, 2020.

Dustin G Mixon, Hans Parshall, and Jianzong Pi. Neural collapse with unconstrained features. *Sampling Theory, Signal Processing, and Data Analysis*, 20(2):1–13, 2022.

Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017. doi: 10.23915/distill.00007. https://distill.pub/2017/feature-visualization.

Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40): 24652–24663, 2020.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

Nimit Sohoni, Jared Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *Advances in Neural Information Processing Systems*, 33:19339–19352, 2020.

Tom Tirer and Joan Bruna. Extended unconstrained features model for exploring deep neural collapse. *arXiv preprint arXiv:2202.08087*, 2022.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

E Weinan and Stephan Wojtowytsch. On the emergence of simplex symmetry in the final and penultimate layers of neural network classifiers. In *Mathematical and Scientific Machine Learning*, pp. 270–290. PMLR, 2022.

Stephan Wojtowytsch et al. On the emergence of simplex symmetry in the final and penultimate layers of neural network classifiers. *arXiv preprint arXiv:2012.05420*, 2020.

Can Yaras, Peng Wang, Zhihui Zhu, Laura Balzano, and Qing Qu. Neural collapse with normalized features: A geometric analysis over the riemannian manifold. *arXiv preprint arXiv:2209.09211*, 2022.

Jinxin Zhou, Xiao Li, Tianyu Ding, Chong You, Qing Qu, and Zhihui Zhu. On the optimization landscape of neural collapse under mse loss: Global optimality with unconstrained features. *arXiv preprint arXiv:2203.01238*, 2022.

Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features. *Advances in Neural Information Processing Systems*, 34:29820–29834, 2021.

A How Does Semantic Similarity Affect the Fine-Grained Structure?

In our experiments with Coarse CIFAR-10, each coarse label is obtained by combining two classes regardless of the semantics. The fact that the neural network can separate the two classes in its representation space implies that the network recognizes these two classes as semantically different (even though they are given the same coarse label). In this section, we explore the following question: If the sub-classes in a super-class have semantic similarity, will the representations still exhibit a fine-grained structure to distinguish them? Intuitively, if the coarse label provided is "natural" and consists of semantically similar sub-classes, it is possible that the neural network will not distinguish between them and just produce truly collapsed representations.

We take an initial step towards this question by looking at ResNet-18 trained on CIFAR-100 using the official 20 super-classes (each super-class contains 5 sub-classes) as labels. Unlike randomly merging classes as we did in Section 4, the official super-classes of CIFAR-100 are natural, merging classes with similar semantics (for example, "beaver" and "dolphin" both belong to "aquatic mammals"). This offers a perfect testbed for our question.

We find that ResNet-18 indeed is not able to distinguish all sub-classes in its representation space, but can still produce separable representations if some sub-classes within a super-class are sufficiently different. Interestingly, the notion of semantic similarity of ResNet-18 turns out to agree well with that of humans. Figures 8 and 9 show the t-SNE visualizations of representations from two super-classes. From the visualizations, although there are not as clear clusters as for Coarse CIFAR-10, the representations do exhibit visible separations between certain sub-classes. In Figure 8, "bicycles" and "motorcycles" are entangled together, while they are separated from "bus", "pickup truck", and "train", which is human-interpretable. In Figure 9, "crab" and "lobster" are mixed together, which are both aquatic and belong to malacostraca, while the other three are not and have more differentiative representations.

These results confirm the intuition that the fine-grained structure in last-layer representations is affected by, or even based on, the semantic similarity between the inputs.

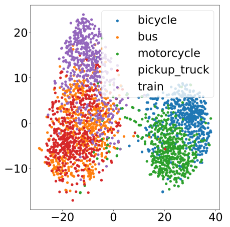

Figure 8: The t-SNE visualization of the last-layer representations of super-class "vehicles 1" of ResNet-18 trained on CIFAR-100 with original super-classes.

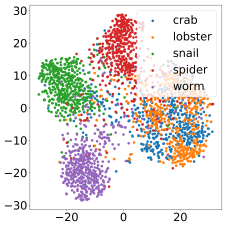

Figure 9: The t-SNE visualization of the last-layer representations of super-class "non-insect invertebrates" of ResNet-18 trained on CIFAR-100 with original super-classes.

B Complete Coarse CIFAR-10 Experiment Results

From this section, we provide extended experiment results. As mentioned in Section 3.2, we permute learning rate in $\{10^{-1}, 10^{-2}, 10^{-3}\}$ and weight decay rate in $\{5 \times 10^{-3}, 5 \times 10^{-4}, 5 \times 10^{-5}\}$. Generally, the results will be shown in a 3×3 table, of which each grid represents the result of one hyper-parameter combination, with each row has the same learning rate and each column has the same weight decay rate.

In this section, we repeat the experiments in Section 4 with all learning rate and weight-decay rate combinations. Firstly, we present the training statistics (accuracy, loss) of all hyper-parameters in Figures 10 and 11 as an reference. It can be observed that all hyper-parameter groups achieved very low training error except the first one (weight decay = 5×10^{-3}, learning rate = 10^{-1}). In fact, the last two hyper-parameter combinations (learning rate = 10^{-3}, weight decay $\in \{5 \times 10^{-4}, 5 \times 10^{-5}\}$) didn't achieve exactly 0 training error (their training error are $< 0.5\%$ but not exactly 0), and the other 6 hyper-parameter combinations all achieved exactly 0 training error.

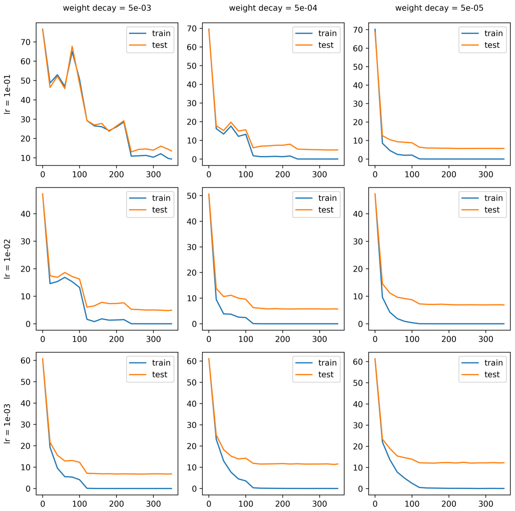

Figure 10: Training and test error during training.

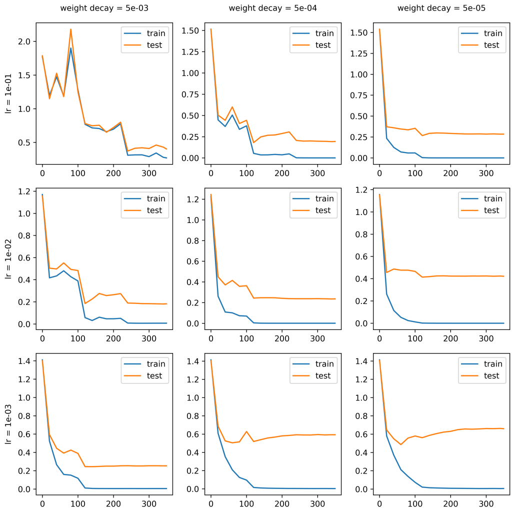

Figure 11: Training and test loss during training.

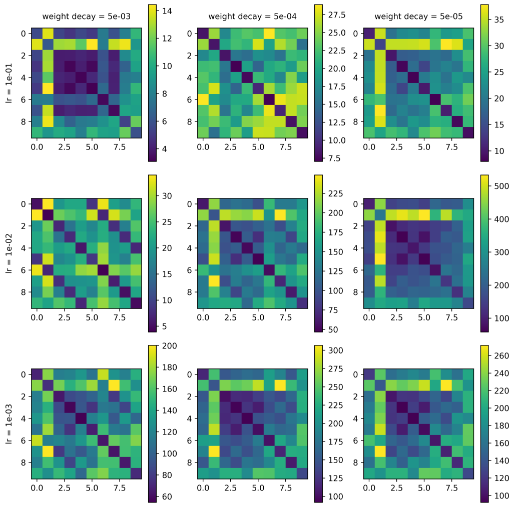

Figure 12: The heatmaps of class distance matrices of different hyper-parameter combinations at epoch 20.

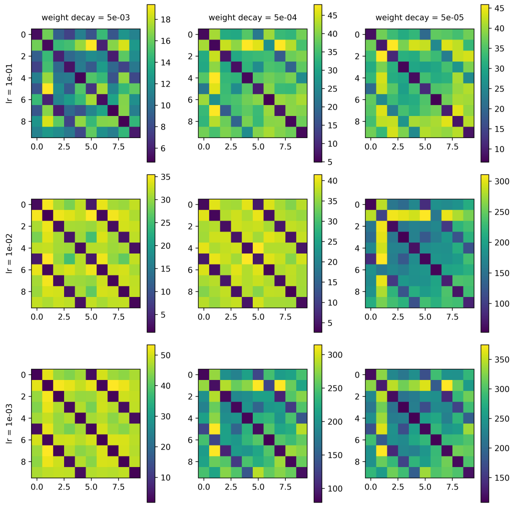

Figure 13: The heatmaps of class distance matrices of different hyper-parameter combinations at epoch 120.

B.1 Class Distance

Here we present the visualization of the heatmap of class distance matrix D which is defined in Section 4.1. We choose 4 epochs to show the trend during training. The results are presented in Figures 12 to 15, whose epoch numbers are $20, 120, 240$ and 349 respectively.

B.2 Visualization

In this section, we present the t-SNE visualization result of ResNet-18 on Coarse CIFAR-10 in Figures 16 to 18. The results are divided into three groups, each of which has the same learning rate and the format of each group is the same as Figure 5.

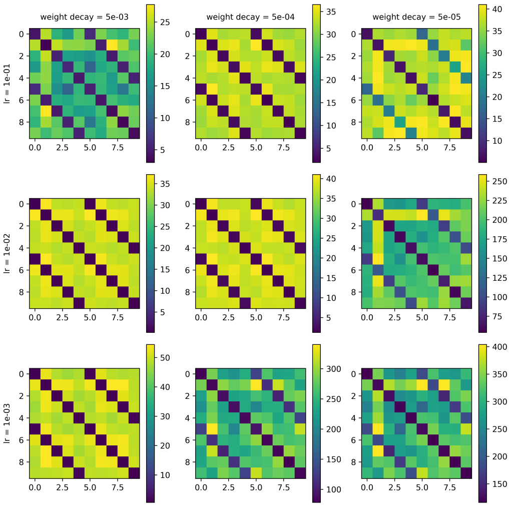

Figure 14: The heatmaps of class distance matrices of different hyper-parameter combinations at epoch 240.

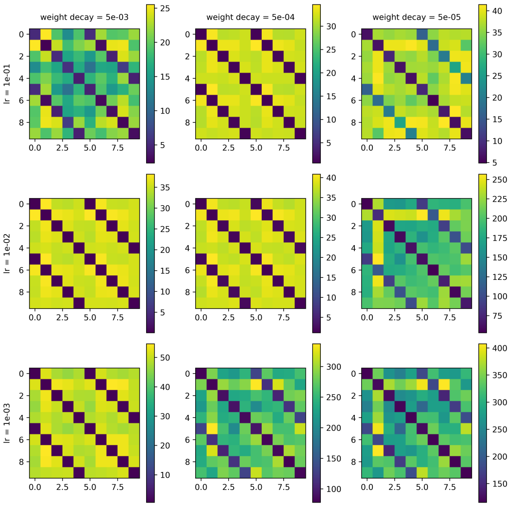

Figure 15: The heatmaps of class distance matrices of different hyper-parameter combinations at epoch 349.

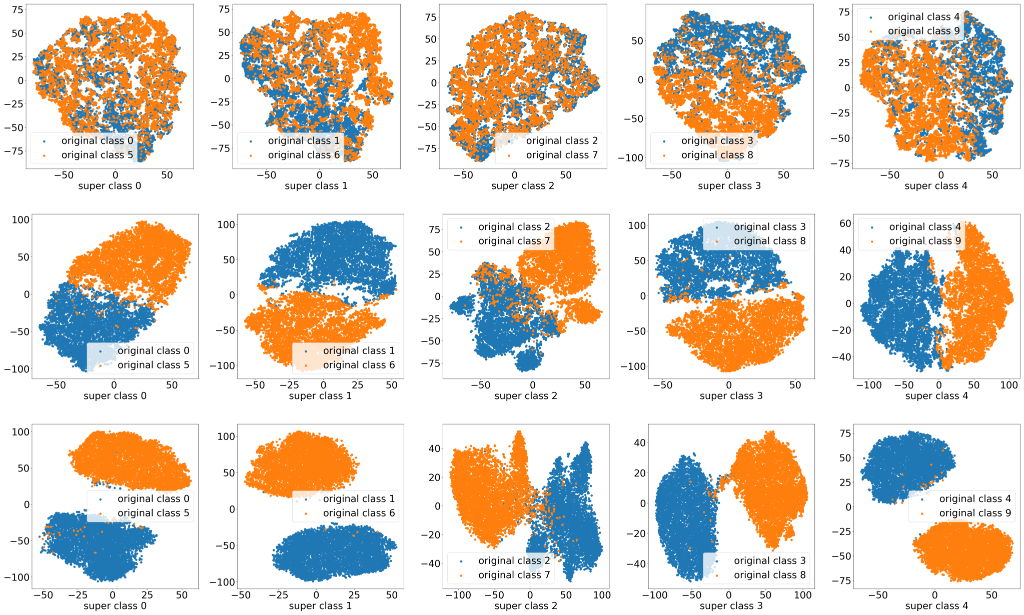

Figure 16: Visualization of last layer representations of ResNet-18 trained on Coarse CIFAR-10 with learning rate = 0.1. Each row represents a hyper-parameter combination and each column represents a super-class. The weight decay rates from top to bottom are $5 \times 10^{-3}, 5 \times 10^{-4}, 5 \times 10^{-5}$.

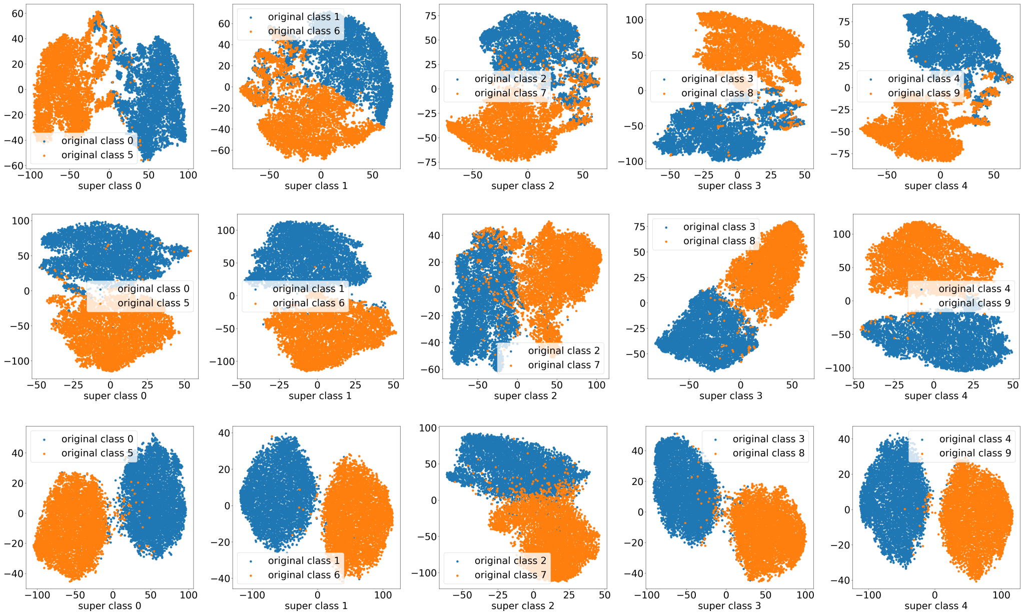

Figure 17: Visualization of last layer representations of ResNet-18 trained on Coarse CIFAR-10 with learning rate = 0.01. Each row represents a hyper-parameter combination and each column represents a super-class. The weight decay rates from top to bottom are $5 \times 10^{-3}, 5 \times 10^{-4}, 5 \times 10^{-5}$.

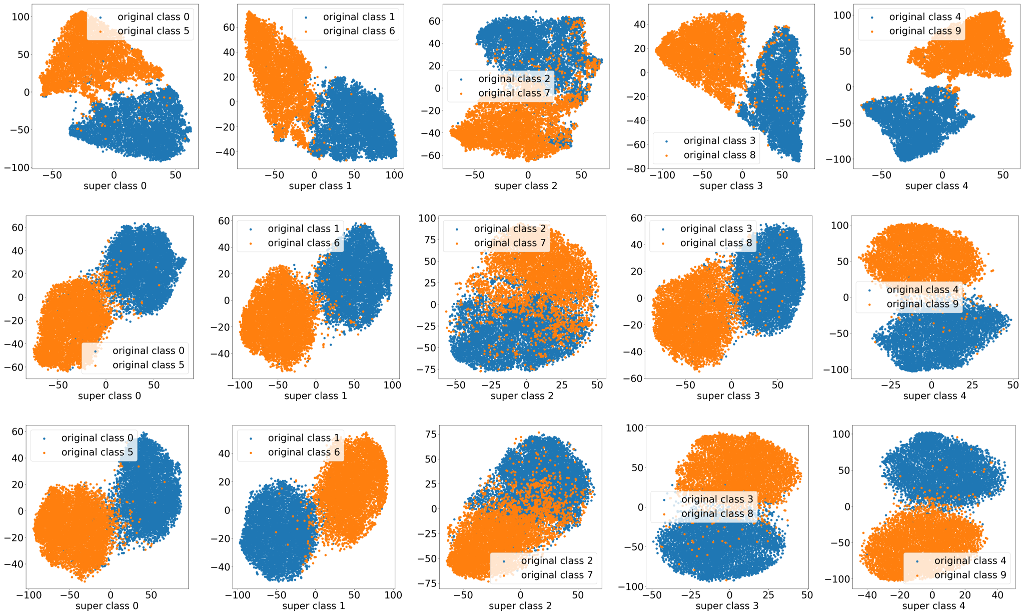

Figure 18: Visualization of last layer representations of ResNet-18 trained on Coarse CIFAR-10 with learning rate = 0.001. Each row represents a hyper-parameter combination and each column represents a super-class. The weight decay rates from top to bottom are $5 \times 10^{-3}, 5 \times 10^{-4}, 5 \times 10^{-5}$.

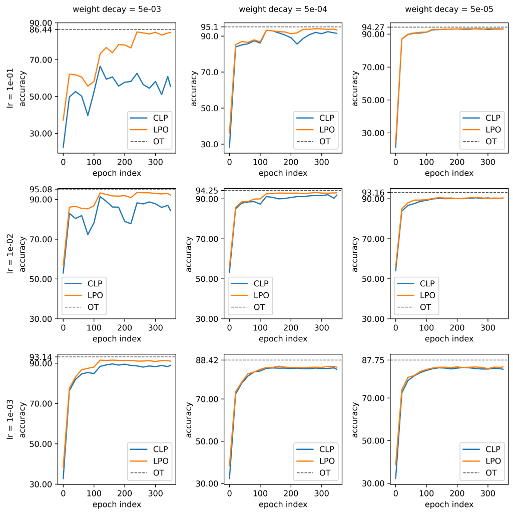

Figure 19: The result of Cluster-and-Linear-Probe test. In the figure“CLP” refers to Cluster-and-Linear-Probe, “LPO” refers to linear probe with original labels and “OT’ refers to the test set accuracy of model trained on original CIFAR-10.

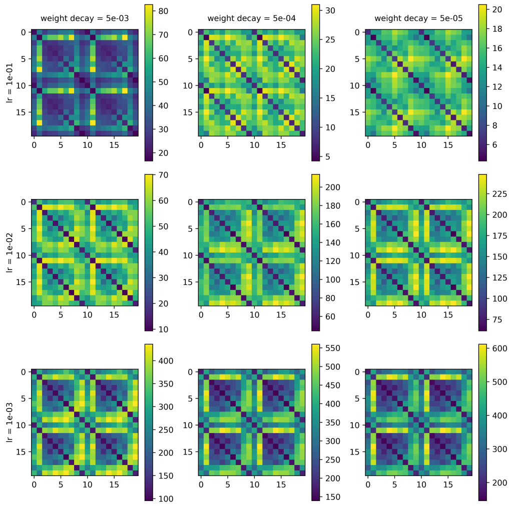

Figure 20: The heatmaps of class distance matrices of different hyper-parameter combinations on Fine CIFAR-10 at epoch 20.

B.3 Cluster-and-Linear-Probe

The Cluster-and-Linear-Probe test results of ResNet-18 trained on Coarse CIFAR-10 with all hyper-parameter combinations are presented in Figure 19.

C Class Distance Result of Fine CIFAR-10

In this section, we consider finely-labeled dataset. We construct a fine version of CIFAR-10 with the process described in Section 3.2, and call it Fine CIFAR-10. We provide the visualization of the class distance matrix of Fine CIFAR-10 with all hyper-parameter combinations. As before, multiple epochs during training are selected to display a evolutionary trend of the class distance matrices. The results are presented in Figures 20 to 22, whose epoch numbers are 20, 200 and 350 respectively.

D Experiment of ResNet-18 on Coarse CIFAR-100

In this section, we report extended experiment results on Coarse CIFAR-100. The same with the case of Coarse CIFAR-10, we construct CIFAR-100 through the label coarsening process described in Section 3.2 and choose $\tilde{C} = 20$, so that every 5 original classes are merged into one super-class. We repeat most of experiments in Section 4.

D.1 Class Distance

The heatmaps of distance matrices are presented in Figures 23 to 25, whose epoch number are 20, 200 and 350 respectively.

D.2 Cluster-and-Linear-Probe

Notice that we omit the visualization result of Coarse CIFAR-100 since there are too many figures. We present the Cluster-and-Linear-Probe results to reflect the clustering property of last-layer representations learned on Coarse CIFAR-100. The CLP results are presented in Figure 26.

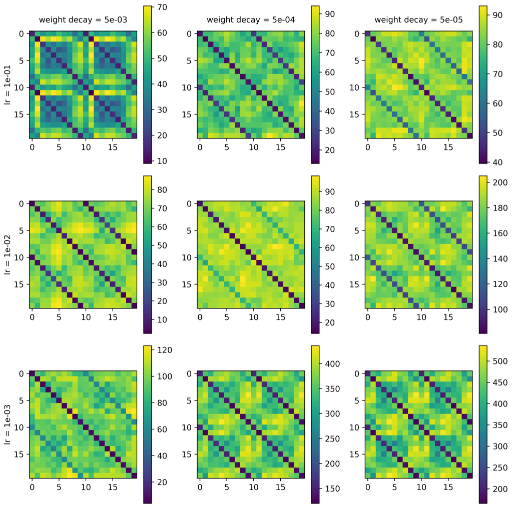

Figure 21: The heatmaps of class distance matrices of different hyper-parameter combinations on Fine CIFAR-10 at epoch 200.

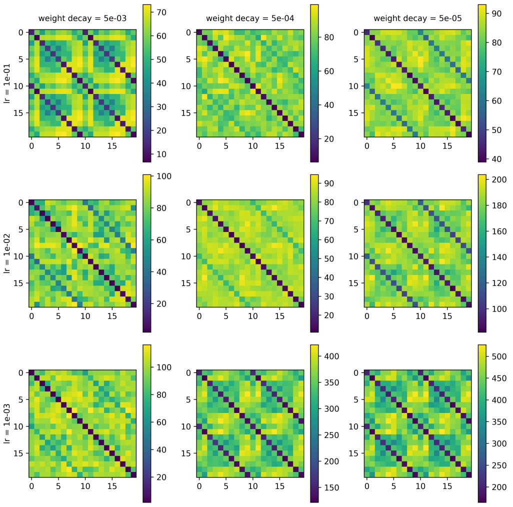

Figure 22: The heatmaps of class distance matrices of different hyper-parameter combinations on Fine CIFAR-10 at epoch 350.

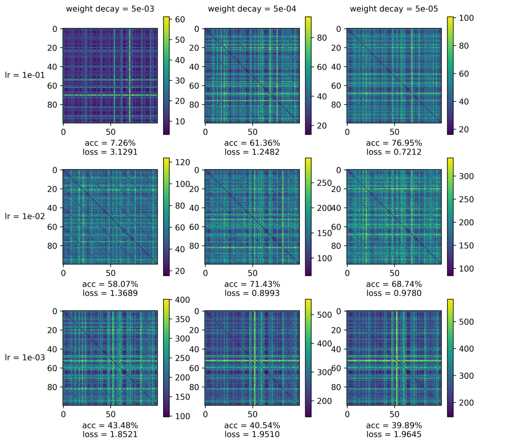

Figure 23: The heatmaps of class distance matrices of different hyper-parameter combinations on Coarse CIFAR-100 at epoch 20.

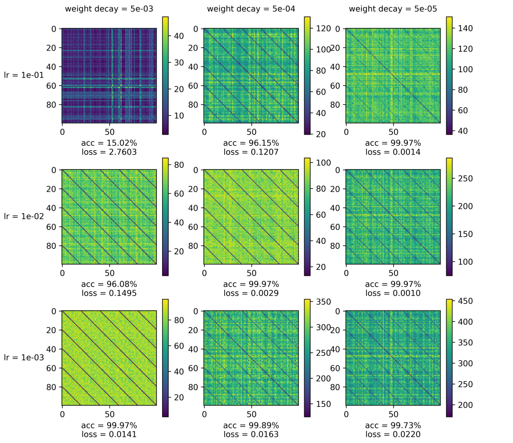

Figure 24: The heatmaps of class distance matrices of different hyper-parameter combinations on Coarse CIFAR-100 at epoch 200.

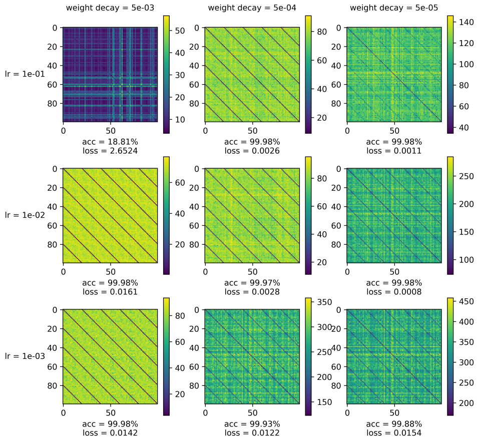

Figure 25: The heatmaps of class distance matrices of different hyper-parameter combinations on Coarse CIFAR-100 at epoch 350.

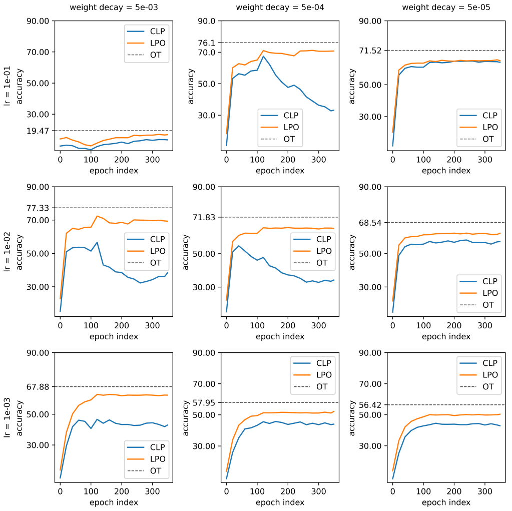

Figure 26: The result of Cluster-and-Linear-Probe test. In the figure“CLP” refers to Cluster-and-Linear-Probe, “LPO” refers to linear probe with original labels and “OT’ refers to the test set accuracy of model trained on original CIFAR-10.

E Random Coarse CIFAR-10

In this section, as mentioned in Section 3.2, we make our experiment more complete by performing a random combination of labels on CIFAR-10 rather than using a determined coarsening process as in the main paper. The dataset construction is almost the same as the process of assigning coarse labels described in Section 3.2, except here we randomly shuffle the class indices before coarsening them.

The class distance matrices of three difference epochs are shown in Figures 27 to 29. From the results we can see, although there are no longer three dark lines, for each row there are generally two dark blocks, represents the original classes belongs to the same super-class, and the same observations in Section 4 can still be made here.

F Experiment with DenseNet

We also perform our experiments with different neural network structures for completeness. In this section, we show the result with DenseNet-121 on Coarse CIFAR-10. The experiments with DenseNet is supportive to our observations in the main paper.

F.1 Class Distance

The class distance matrices of three epochs during training are presented in Figures 30 to 32.

F.2 Cluster-and-Linear-Probe

The Cluster-and-Linear-Probe test results are presented in Figure 33.

G Experiment with VGG

We also extend our experiments to VGG-18. Interestingly, VGG to some extent is a counter example of the observations made in the main paper: it only displays Neural Collapse, and can not distinguish different original classes within one super-class, even in an early stage of training. The reason why VGG is abnormal requires further exploration.

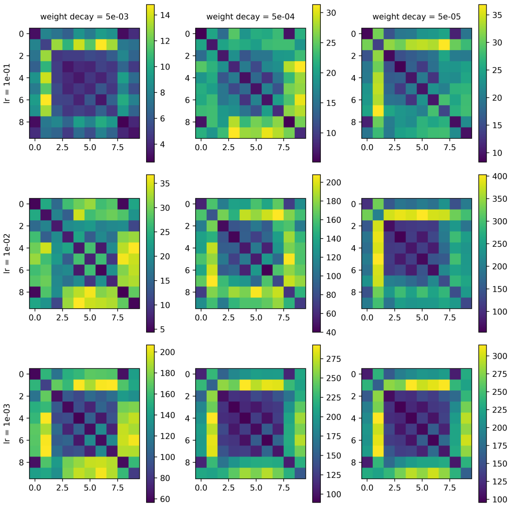

Figure 27: The heatmaps of class distance matrices of different hyper-parameter combinations on Random Coarse CIFAR-10 at epoch 20.

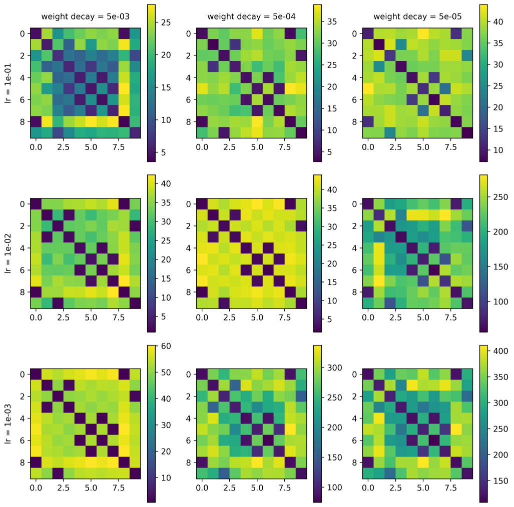

Figure 28: The heatmaps of class distance matrices of different hyper-parameter combinations on Random Coarse CIFAR-10 at epoch 200.

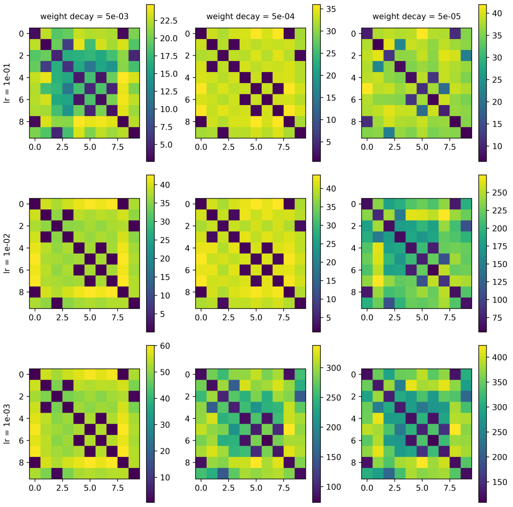

Figure 29: The heatmaps of class distance matrices of different hyper-parameter combinations on Random Coarse CIFAR-10 at epoch 350.

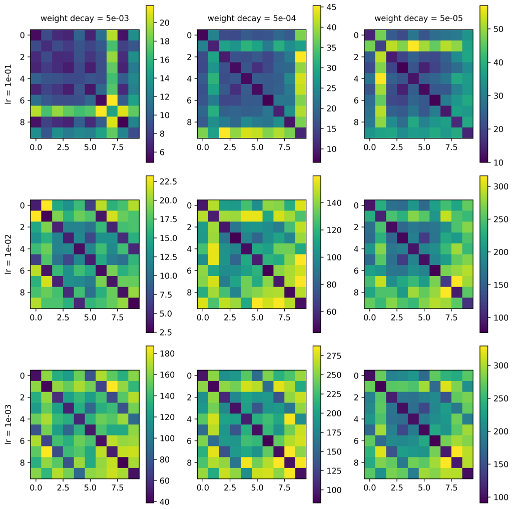

Figure 30: The heatmaps of class distance matrices of different hyper-parameter combinations with DenseNet-121 on Coarse CIFAR-10 at epoch 20.

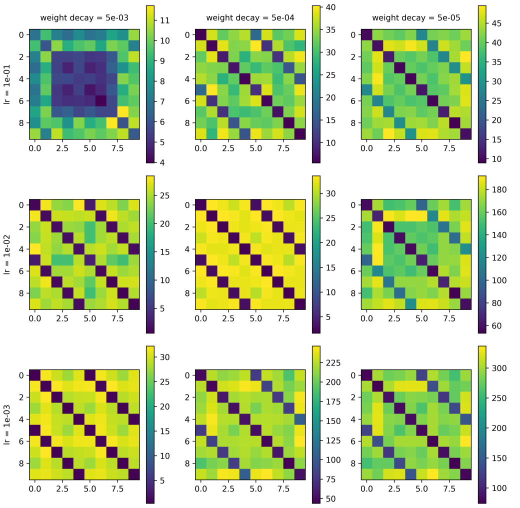

Figure 31: The heatmaps of class distance matrices of different hyper-parameter combinations with DenseNet-121 on Coarse CIFAR-10 at epoch 20.

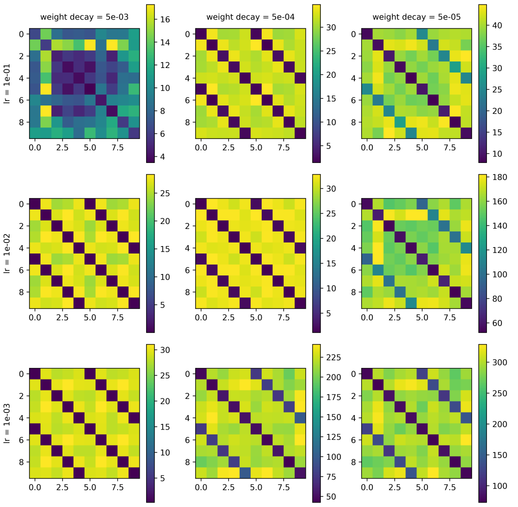

Figure 32: The heatmaps of class distance matrices of different hyper-parameter combinations with DenseNet-121 on Coarse CIFAR-10 at epoch 20.

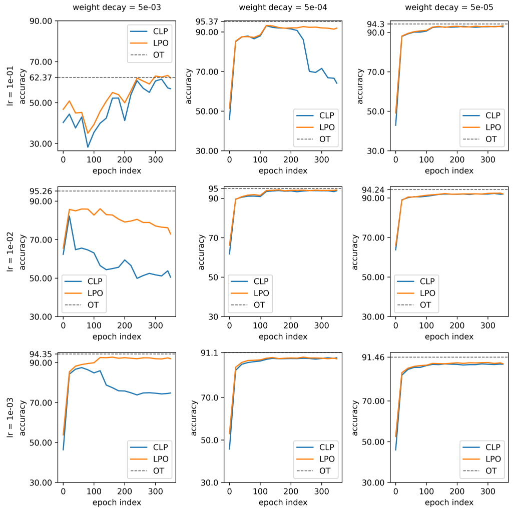

Figure 33: The result of Cluster-and-Linear-Probe test with DenseNet-121. In the figure“CLP” refers to Cluster-and-Linear-Probe, “LPO” refers to linear probe with original labels and “OT’ refers to the test set accuracy of model trained on original CIFAR-10.

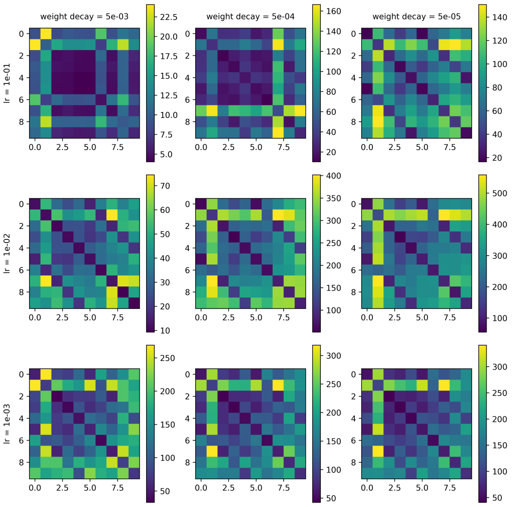

Figure 34: The heatmaps of class distance matrices of different hyper-parameter combinations with VGG-18 on Coarse CIFAR-10 at epoch 20.

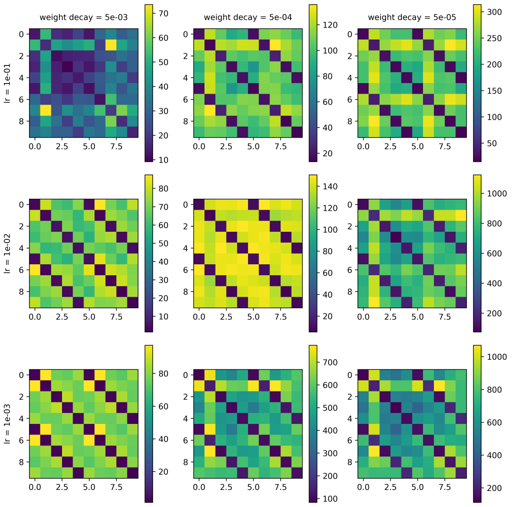

Figure 35: The heatmaps of class distance matrices of different hyper-parameter combinations with VGG-18 on Coarse CIFAR-10 at epoch 200.

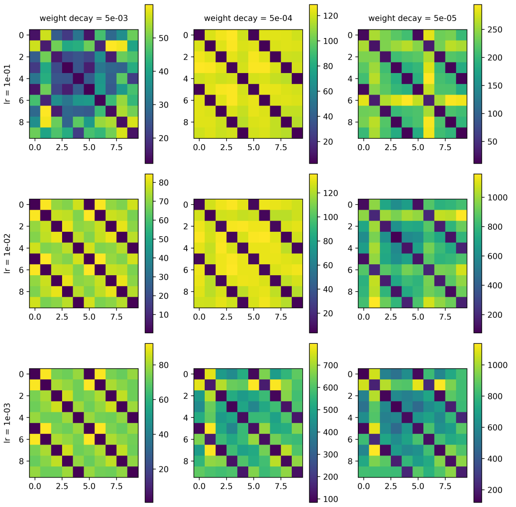

Figure 36: The heatmaps of class distance matrices of different hyper-parameter combinations with VGG-18 on Coarse CIFAR-10 at epoch 350.

The class distance matrices of three epochs with VGG during training are shown in Figures 34 to 36. It can be observed that the three dark lines appears almost at the same time and always be of nearly the same darkness. This represents the trend predicted by Neural Collapse (Figure 3 (a)), but rejects the prediction made by (Figure 3 (b)).