Exploring Deep Neural Networks via Layer-Peeled Model: Minority Collapse in Imbalanced Training

Cong Fang[a], **Hangfeng He**[b], **Qi Long**[c], and **Weijie J. Su**[d,1]

[a]Department of Key Laboratory of Machine Perception (MOE), Peking University; [b]Department of Computer and Information Science, University of Pennsylvania; [c]Department of Biostatistics, Epidemiology and Informatics, University of Pennsylvania; [d]Department of Statistics and Data Science, University of Pennsylvania

This manuscript was compiled on September 8, 2021

In this paper, we introduce the *Layer-Peeled Model*, a nonconvex yet analytically tractable optimization program, in a quest to better understand deep neural networks that are trained for a sufficiently long time. As the name suggests, this new model is derived by isolating the topmost layer from the remainder of the neural network, followed by imposing certain constraints separately on the two parts of the network. We demonstrate that the Layer-Peeled Model, albeit simple, inherits many characteristics of well-trained neural networks, thereby offering an effective tool for explaining and predicting common empirical patterns of deep learning training. First, when working on class-balanced datasets, we prove that any solution to this model forms a simplex equiangular tight frame, which in part explains the recently discovered phenomenon of neural collapse (1). More importantly, when moving to the imbalanced case, our analysis of the Layer-Peeled Model reveals a hitherto unknown phenomenon that we term *Minority Collapse*, which fundamentally limits the performance of deep learning models on the minority classes. In addition, we use the Layer-Peeled Model to gain insights into how to mitigate Minority Collapse. Interestingly, this phenomenon is first predicted by the Layer-Peeled Model before being confirmed by our computational experiments.

deep learning | surrogate model | optimization | class imbalance

1. Introduction

In the past decade, deep learning has achieved remarkable performance across a range of scientific and engineering domains (2–4). Interestingly, these impressive accomplishments were mostly achieved by heuristics and tricks, though often plausible, without much principled guidance from a theoretical perspective. On the flip side, however, this reality suggests the great potential a theory could have for advancing the development of deep learning methodologies in the coming decade.

Unfortunately, it is not easy to develop a theoretical foundation for deep learning. Perhaps the most difficult hurdle lies in the nonconvexity of the optimization problem for training neural networks, which, loosely speaking, stems from the interaction between different layers of neural networks. To be more precise, consider a neural network for K-class classification (in logits), which in its simplest form reads[*]

$$\boldsymbol{f}(\boldsymbol{x}; \boldsymbol{W}_{\text{full}}) = \boldsymbol{b}_L + \boldsymbol{W}_L \sigma\left(\boldsymbol{b}_{L-1} + \boldsymbol{W}_{L-1}\sigma(\cdots\sigma(\boldsymbol{b}_1 + \boldsymbol{W}_1\boldsymbol{x})\cdots)\right).$$

Here, $\boldsymbol{W}_{\text{full}} := \{\boldsymbol{W}_1, \boldsymbol{W}_2, \ldots, \boldsymbol{W}_L\}$ denotes the weights of the L layers, $\{\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_L\}$ denotes the biases, and $\sigma(\cdot)$ is a nonlinear activation function such as the ReLU. Owing to the complex and nonlinear interaction between the L layers,

when applying stochastic gradient descent to the optimization problem

$$\min_{\boldsymbol{W}_{\text{full}}} \quad \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}), \boldsymbol{y}_k) + \frac{\lambda}{2}\|\boldsymbol{W}_{\text{full}}\|^2 \quad [1]$$

with a loss function $\mathcal{L}$ for training the neural network, it becomes very difficult to pinpoint how a given layer influences the output $\boldsymbol{f}$ (above, $\{\boldsymbol{x}_{k,i}\}_{i=1}^{n_k}$ denotes the training examples in the k-th class, with label $\boldsymbol{y}_k$, $N = n_1 + \cdots + n_K$ is the total number of training examples, $\lambda > 0$ is the weight decay parameter, and $\|\cdot\|$ throughout the paper is the ℓ_2 norm). Worse, this difficulty in analyzing deep learning models is compounded by an ever growing number of layers.

Therefore, any attempt to develop a tractable and comprehensive theory for demystifying deep learning would presumably first need to simplify the interaction between a large number of layers. Following this intuition, in this paper we introduce the following optimization program as a *surrogate* model for Eq. (1) with the goal of unveiling quantitative patterns of deep neural networks:

$$\min_{\boldsymbol{W}_L, \boldsymbol{H}} \quad \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\mathcal{L}(\boldsymbol{W}_L\boldsymbol{h}_{k,i}, \boldsymbol{y}_k)$$

$$\text{s.t.} \quad \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 \leq E_W, \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i=1}^{n_k}\|\boldsymbol{h}_{k,i}\|^2 \leq E_H, \quad [2]$$

> ### Significance Statement
>
> The remarkable development of deep learning over the past decade relies heavily on sophisticated heuristics and tricks. To better exploit its potential in the coming decade, perhaps a rigorous framework for reasoning about deep learning is needed, which however is not easy to build due to the intricate details of neural networks. For near-term purposes, a practical alternative is to develop a mathematically tractable surrogate model yet maintaining many characteristics of neural networks. This paper proposes a model of this kind that we term the Layer-Peeled Model. The effectiveness of this model is evidenced by, among others, its ability to reproduce a known empirical pattern and to predict a hitherto unknown phenomenon when training deep learning models on imbalanced datasets.

[*]The softmax step is implicitly included in the loss function and we omit other operations such as max-pooling for simplicity.

where $\boldsymbol{W}_L = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K]^\top \in \mathbb{R}^{K \times p}$ is, as in Eq. (1), comprised of K linear classifiers in the last layer, $\boldsymbol{H} = [\boldsymbol{h}_{k,i} : 1 \leq k \leq K, 1 \leq i \leq n_k] \in \mathbb{R}^{p \times N}$ corresponds to the p-dimensional last-layer activations/features of all N training examples, and E_H and E_W are two positive scalars. Note that the bias terms are omitted for simplicity. Although still nonconvex, this new optimization program is presumably much more amenable to analysis than the old one Eq. (1) as the interaction now is only between two layers.
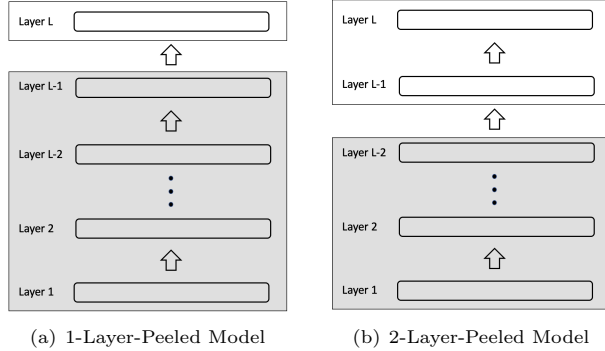

(a) 1-Layer-Peeled Model (b) 2-Layer-Peeled Model

Fig. 1. Illustration of Layer-Peeled Models. The right panel represents the 2-Layer-Peeled Model, which is discussed in Section 6. For each panel, we preserve the details of the white (top) box, whereas the gray (bottom) box is modeled by a simple decision variable for every training example.

In relating Eq. (2) to Eq. (1), a first simple observation is that $\boldsymbol{f}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}) = \boldsymbol{W}_L \sigma(\boldsymbol{W}_{L-1} \sigma(\cdots \sigma(\boldsymbol{W}_1 \boldsymbol{x}_{k,i}) \cdots))$ in Eq. (1) is replaced by $\boldsymbol{W}_L \boldsymbol{h}_{k,i}$ in Eq. (2). Put differently, the black-box nature of the last-layer features, namely $\sigma(\boldsymbol{W}_{L-1} \sigma(\cdots \sigma(\boldsymbol{W}_1 \boldsymbol{x}_{k,i}) \cdots))$, is now modeled by a simple decision variable $\boldsymbol{h}_{k,i}$ for each training example, with an overall constraint on their ℓ_2 norm. Intuitively speaking, this simplification is done by *peeling* off the topmost layer from the neural network. Thus, we call the optimization program (2) the *1-Layer-Peeled Model*, or simply the *Layer-Peeled Model*.

At a high level, the Layer-Peeled Model takes a *top-down* approach to the analysis of deep neural networks. As illustrated in Figure 1, the essence of the modeling strategy is to break down the neural network from top to bottom, specifically singling out the topmost layer and modeling all bottom layers collectively as a single variable. In fact, the top-down perspective that we took in the development of the Layer-Peeled Model was inspired by a recent breakthrough made by Papyan, Han, and Donoho (1), who discovered a mathematically elegant and pervasive phenomenon termed neural collapse in deep learning training. This top-down approach was also taken in (5–9) to investigate various aspects of deep learning models.

A. Two Applications. Despite its plausibility, the ultimate test of the Layer-Peeled Model lies in its ability to faithfully approximate deep learning models through explaining empirical observations and, better, predicting new phenomena. In what follows, we provide convincing evidence that the Layer-Peeled Model is up to this task by presenting two findings. To be concrete, we remark that the results below are concerned with well-trained deep learning models, which correspond to, in rough terms, (near) optimal solutions of Eq. (1).

Balanced Data. Roughly speaking, neural collapse (1) refers to the emergence of certain geometric patterns of the last-layer features $\sigma(\boldsymbol{W}_{L-1} \sigma(\cdots \sigma(\boldsymbol{W}_1 \boldsymbol{x}_{k,i}) \cdots))$ and the last-layer classifiers $\boldsymbol{W}_L$, when the neural network for *balanced* classification problems is well-trained in the sense that it is toward not only zero misclassification error but also negligible[†] cross-entropy loss. Specifically, the authors observed the following properties in their massive experiments: the last-layer features from the same class tend to be very close to their class mean; these K class means centered at the global-mean have the same length and form the maximally possible equal-sized angles between any pair; moreover, the last-layer classifiers become dual to the class means in the sense that they are equal to each other for each class up to a scaling factor. See a more precise description in Section B.

While it seems hopeless to rigorously prove neural collapse for multiple-layer neural networks Eq. (1) at the moment, alternatively, we seek to show that this phenomenon emerges in the surrogate model Eq. (2). More precisely, when the size of each class $n_k = n$ for all k, is it true that any global minimizer $\boldsymbol{W}_L^\star = [\boldsymbol{w}_1^\star, \ldots, \boldsymbol{w}_K^\star]^\top, \boldsymbol{H}^\star = [\boldsymbol{h}_{k,i}^\star : 1 \leq k \leq K, 1 \leq i \leq n]$ of Eq. (2) exhibits neural collapse? The following result answers this question in the affirmative:

Finding 1. *Neural collapse occurs in the Layer-Peeled Model.*

A formal statement of this result and a detailed discussion are given in Section 3.

This result applies to a family of loss functions $\mathcal{L}$, particularly including the cross-entropy loss and the contrastive loss (see, e.g., (10)). As an immediate implication, this result provides evidence of the Layer-Peeled Model's ability to characterize well-trained deep learning models.

Imbalanced Data. While a surrogate model would be satisfactory if it explains some already observed phenomenon, we set a *higher* standard for the model, asking whether it can predict a *new* common empirical pattern. Encouragingly, the Layer-Peeled Model happens to meet this standard. Specifically, we consider training deep learning models on imbalanced datasets, where some classes contain many more training examples than others. Despite the pervasiveness of imbalanced classification in many practical applications (11), the literature remains scarce on its impact on the trained neural networks from a theoretical standpoint. Here we provide mathematical insights into this problem by using the Layer-Peeled Model. In the following result, we consider optimal solutions to the Layer-Peeled Model on a dataset with two different class sizes: the first K_A majority classes each contain n_A training examples ($n_1 = n_2 = \cdots = n_{K_A} = n_A$), and the remaining $K_B := K - K_A$ minority classes each contain n_B examples ($n_{K_A+1} = n_{K_A+2} = \cdots = n_K = n_B$). We call $R := n_A/n_B > 1$ the imbalance ratio.

Finding 2. *In the Layer-Peeled Model, the last-layer classifiers corresponding to the minority classes, namely $\boldsymbol{w}_{K_A+1}^\star, \boldsymbol{w}_{K_A+2}^\star, \ldots, \boldsymbol{w}_K^\star$, collapse to a single vector when R is sufficiently large.*

This result is elaborated on in Section 4. The derivation involves some novel elements to tackle the nonconvexity of the Layer-Peeled Model Eq. (2) and the asymmetry due to the imbalance in class sizes.

[†] Strictly speaking, in the presence of an ℓ_2 regularization term, which is equivalent to weight decay, the cross-entropy loss evaluated at any global minimizer of Eq. (1) is bounded away from 0.
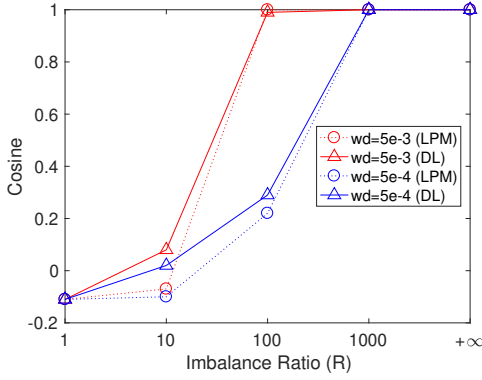
Fig. 2. Minority Collapse predicted by the Layer-Peeled Model (LPM, in dotted lines) and empirically observed in deep learning (DL, in solid lines) on imbalanced datasets with $K_A = 7$ and $K_B = 3$. The y-axis denotes the average cosine of the angles between any pair of the minority classifier $\boldsymbol{w}_{K_A+1}^{\star}, \ldots, \boldsymbol{w}_K^{\star}$ for both LPM and DL. The datasets we use are subsets of the CIFAR10 datasets (12) and the size of the majority classes is fixed to 5000. The experiments use VGG13 (13) as the deep learning architecture, with weight decay (wd) $\lambda = 5 \times 10^{-3}, 5 \times 10^{-4}$. The prediction is especially accurate in capturing the phase transition point where the cosine becomes 1 or, equivalently, the minority classifiers become parallel to each other. More details can be found in Section C.

In slightly more detail, we identify a phase transition as the imbalance ratio R increases: when R is below a threshold, the minority classes are distinguishable in terms of their last-layer classifiers; when R is above the threshold, they become indistinguishable. While this phenomenon is merely predicted by the simple Layer-Peeled Model Eq. (2), it appears in our computational experiments on deep neural networks. More surprisingly, our prediction of the phase transition point is in excellent agreement with the experiments, as shown in Figure 2.

This phenomenon, which we refer to as *Minority Collapse*, reveals the fundamental difficulty in using deep learning for classification when the dataset is widely imbalanced, even in terms of optimization, not to mention generalization. This is not a priori evident given that neural networks have a large approximation capacity (see, e.g., (14)). Importantly, Minority Collapse emerges at a finite value of the imbalance ratio rather than at infinity. Moreover, even below the phase transition point of this ratio, we find that the angles between any pair of the minority classifiers are already smaller than those of the majority classes, both theoretically and empirically.

B. Related Work. There is a venerable line of work attempting to gain insights into deep learning from a theoretical point of view (15–29). See also the reviews (30–33) and references therein.

The work of neural collapse by (1) in this body of work is particularly noticeable with its mathematically elegant and convincing insights. In brief, (1) observed the following four properties of the last-layer features and classifiers in deep learning training on balanced datasets:[‡]

> (NC1) Variability collapse: the within-class variation of the last-layer features becomes 0, which means that these features collapse to their class means.

> (NC2) The class means centered at their global mean collapse to the vertices of a simplex equiangular tight

---

[‡]See the mathematical description of neural collapse in Theorem 1.

frame (ETF) up to scaling.

> (NC3) Up to scaling, the last-layer classifiers each collapse to the corresponding class means.

> (NC4) The network's decision collapses to simply choosing the class with the closest Euclidean distance between its class mean and the activations of the test example.

Now we give the formal definition of ETF (1, 34).

Definition 1. *A K-simplex ETF is a collection of points in $\mathbb{R}^p$ specified by the columns of the matrix*

$$\boldsymbol{M}^{\star} = \sqrt{\frac{K}{K-1}} \boldsymbol{P} \left(\boldsymbol{I}_K - \frac{1}{K} \boldsymbol{1}_K \boldsymbol{1}_K^{\top} \right),$$

where $\boldsymbol{I}_K \in \mathbb{R}^{K \times K}$ is the identity matrix, $\boldsymbol{1}_K$ is the ones vector, and $\boldsymbol{P} \in \mathbb{R}^{p \times K}$ $(p \geq K)$[§] is a partial orthogonal matrix such that $\boldsymbol{P}^{\top} \boldsymbol{P} = \boldsymbol{I}_K$.

A common setup of the experiments for validating neural collapse is the use of the cross-entropy loss with ℓ_2 regularization, which corresponds to weight decay in stochastic gradient descent. Based on convincing arguments and numerical evidence, (1) demonstrated that the symmetry and stability of neural collapse improve deep learning training in terms of generalization, robustness, and interpretability. Notably, these improvements occur with the benign overfitting phenomenon (see (35–39)) during the terminal phase of training—when the trained model interpolates the in-sample training data.

In passing, we remark that concurrent works (40–43) produced neural collapse using different surrogate models. In slightly more detail, (40–42) obtained their models by peeling off the topmost layer. The difference, however, is that (41, 42) considered models that impose a norm constraint for each class, as opposed to an overall constraint as employed in the Layer-Peeled Model. Moreover, (40) analyzed gradient flow with an unconstrained features model using the squared loss instead of the cross-entropy loss. The work (43) provided an insightful perspective for the analysis of neural networks using convex duality. Relying on a convex formulation that is in the same spirit as our semidefinite programming relaxation, the authors of (43) observed neural collapse in their ReLU-based model by leveraging strong duality under certain conditions.

2. Derivation

In this section, we heuristically derive the Layer-Peeled Model as an analytical surrogate for well-trained neural networks. Although our derivation lacks rigor, the goal is to reduce the complexity of the optimization problem Eq. (1) while roughly preserving its structure. Notably, the penalty $\frac{\lambda}{2} \|\boldsymbol{W}_{\text{full}}\|^2$ corresponds to weight decay used in training deep learning models, which is necessary for preventing this optimization program from attaining its minimum at infinity when $\mathcal{L}$ is the cross-entropy loss. For simplicity, we omit the biases in the neural network $\boldsymbol{f}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}})$.

Taking a top-down standpoint, our modeling strategy starts by singling out the weights $\boldsymbol{W}_L$ of the topmost layer and

---

[§]To be complete, we only require $p \geq K - 1$. When $p = K - 1$, we can choose $\boldsymbol{P}$ such that $\left[\boldsymbol{P}^{\top}, \boldsymbol{1}_K \right]$ is an orthogonal matrix.

rewriting Eq. (1) as

$$\min_{\boldsymbol{W}_L, \boldsymbol{H}} \quad \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}(\boldsymbol{W}_L \boldsymbol{h}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{-L}), \boldsymbol{y}_k)$$
$$+ \frac{\lambda}{2} \|\boldsymbol{W}_L\|^2 + \frac{\lambda}{2} \|\boldsymbol{W}_{-L}\|^2, \qquad [3]$$

where the last-layer feature function $\boldsymbol{h}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{-L}) := \sigma(\boldsymbol{W}_{L-1}\sigma(\cdots\sigma(\boldsymbol{W}_1 \boldsymbol{x}_{k,i})\cdots))$ and $\boldsymbol{W}_{-L}$ denotes the weights from all layers but the last layer. From the Lagrangian dual viewpoint, a minimum of the optimization program above is also an optimal solution to

$$\min_{\boldsymbol{W}_L, \boldsymbol{W}_{-L}} \quad \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}(\boldsymbol{W}_L \boldsymbol{h}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{-L}), \boldsymbol{y}_k)$$
$$\text{s.t.} \quad \|\boldsymbol{W}_L\|^2 \leq C_1, \|\boldsymbol{W}_{-L}\|^2 \leq C_2, \qquad [4]$$

for some positive numbers C_1 and C_2.¶ To clear up any confusion, note that due to its nonconvexity, Eq. (3) may admit multiple global minima and each in general corresponds to different values of C_1, C_2. Next, we can equivalently write Eq. (4) as

$$\min_{\boldsymbol{W}_L, \boldsymbol{H}} \quad \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}(\boldsymbol{W}_L \boldsymbol{h}_{k,i}, \boldsymbol{y}_k)$$
$$\text{s.t.} \quad \|\boldsymbol{W}_L\|^2 \leq C_1, \qquad [5]$$
$$\boldsymbol{H} \in \left\{ \boldsymbol{H}(\boldsymbol{W}_{-L}) : \|\boldsymbol{W}_{-L}\|^2 \leq C_2 \right\},$$

where $\boldsymbol{H} = [\boldsymbol{h}_{k,i} : 1 \leq k \leq K, 1 \leq i \leq n_k]$ denotes a decision variable and the function $\boldsymbol{H}(\boldsymbol{W}_{-L})$ is defined as $\boldsymbol{H}(\boldsymbol{W}_{-L}) := [\boldsymbol{h}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{-L}) : 1 \leq k \leq K, 1 \leq i \leq n_k]$ for any $\boldsymbol{W}_{-L}$.

To simplify Eq. (5), we make the *ansatz* that the range of $\boldsymbol{h}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{-L})$ under the constraint $\|\boldsymbol{W}_{-L}\|^2 \leq C_2$ is approximately an ellipse in the sense that

$$\left\{ \boldsymbol{H}(\boldsymbol{W}_{-L}) : \|\boldsymbol{W}_{-L}\|^2 \leq C_2 \right\}$$
$$\approx \left\{ \boldsymbol{H} : \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \|\boldsymbol{h}_{k,i}\|^2 \leq C_2' \right\} \qquad [6]$$

for some $C_2' > 0$. Loosely speaking, this ansatz asserts that $\boldsymbol{H}$ should be regarded as a variable in an ℓ_2 space. To shed light on the rationale behind the ansatz, note that $\boldsymbol{h}_{k,i}$ intuitively lives in the dual space of $\boldsymbol{W}$ in view of the appearance of the product $\boldsymbol{W} \boldsymbol{h}_{k,i}$ in the objective. Furthermore, $\boldsymbol{W}$ is in an ℓ_2 space for the ℓ_2 constraint on it. Last, note that ℓ_2 spaces are self-dual.

Inserting this approximation into Eq. (5), we obtain the following optimization program, which we call the Layer-Peeled Model:

$$\min_{\boldsymbol{W}, \boldsymbol{H}} \quad \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}(\boldsymbol{W} \boldsymbol{h}_{k,i}, \boldsymbol{y}_k)$$
$$\text{s.t.} \quad \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 \leq E_W, \qquad [7]$$
$$\frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \|\boldsymbol{h}_{k,i}\|^2 \leq E_H.$$

¶ Denoting by $(\boldsymbol{W}_L^\star, \boldsymbol{W}_{-L}^\star)$ an optimal solution to Eq. (3), then we can take $C_1 = \|\boldsymbol{W}_L^\star\|^2$ and $C_2 = \|\boldsymbol{W}_{-L}^\star\|^2$.

For simplicity, above and henceforth we write $\boldsymbol{W} := \boldsymbol{W}_L \equiv [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K]^\top$ for the last-layer classifiers/weights and the thresholds $E_W = C_1/K$ and $E_H = C_2'/K$.

This optimization program is nonconvex but, as we will show soon, is generally mathematically tractable for analysis. On the surface, the Layer-Peeled Model has no dependence on the data $\{\boldsymbol{x}_{k,i}\}$, which however is not the correct picture, since the dependence has been implicitly incorporated into the threshold E_H.

In passing, we remark that neural collapse does *not* emerge if the second constraint of Eq. (7) uses the ℓ_q norm for any $q \neq 2$ (strictly speaking, ℓ_q is not a norm when $q < 1$), in place of the ℓ_2 norm. This fact in turn justifies in part the ansatz Eq. (6). This result is formally stated in Proposition 2 in Section 6.

3. Layer-Peeled Model for Explaining Neural Collapse

In this section, we consider training deep neural networks on a balanced dataset—that is, $n_k = n$ for all classes $1 \leq k \leq K$. Our main finding is that the Layer-Peeled Model displays the neural collapse phenomenon, just as in deep learning training (1). The proofs are all deferred to SI Appendix. Throughout this section, we assume $p \geq K-1$ unless otherwise specified. This assumption is satisfied in many popular network architectures, where p is usually tens or hundreds of times of K.

A. Cross-Entropy Loss. The cross-entropy loss is perhaps the most popular loss used in training deep learning models for classification tasks. This loss function takes the form

$$\mathcal{L}(\boldsymbol{z}, \boldsymbol{y}_k) = -\log\left(\frac{\exp(\boldsymbol{z}(k))}{\sum_{k'=1}^{K} \exp(\boldsymbol{z}(k'))} \right), \qquad [8]$$

where $\boldsymbol{z}(k')$ denotes the k'-th entry of the logit $\boldsymbol{z}$. Recall that $\boldsymbol{y}_k$ is the label of the k-th class and the feature $\boldsymbol{z}$ is set to $\boldsymbol{W} \boldsymbol{h}_{k,i}$ in the Layer-Peeled Model Eq. (7). In contrast to the complex deep neural networks, which are often considered a black-box, the Layer-Peeled Model is much easier to deal with. As an exemplary use case, the following result shows that any minimizer of the Layer-Peeled Model Eq. (7) with the cross-entropy loss admits an almost closed-form expression.

Theorem 1. *In the balanced case, any global minimizer* $\boldsymbol{W}^\star \equiv [\boldsymbol{w}_1^\star, \ldots, \boldsymbol{w}_K^\star]^\top, \boldsymbol{H}^\star \equiv [\boldsymbol{h}_{k,i}^\star : 1 \leq k \leq K, 1 \leq i \leq n]$ *of* Eq. (7) *with the cross-entropy loss obeys*

$$\boldsymbol{h}_{k,i}^\star = C\boldsymbol{w}_k^\star = C'\boldsymbol{m}_k^\star \qquad [9]$$

for all $1 \leq i \leq n, 1 \leq k \leq K$, *where the constants* $C = \sqrt{E_H/E_W}, C' = \sqrt{E_H}$, *and the matrix* $[\boldsymbol{m}_1^\star, \ldots, \boldsymbol{m}_K^\star]$ *forms a K-simplex ETF specified in Definition 1.*

Remark 2. Note that the minimizers $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$'s are equivalent to each other up to rotation. This is because of the rational invariance of simplex ETFs (see $\boldsymbol{P}$ in Definition 1).

This theorem demonstrates the highly symmetric geometry of the last-layer features and weights of the Layer-Peeled Model, which is precisely the phenomenon of neural collapse. Explicitly, Eq. (9) says that all within-class (last-layer) features are the same: $\boldsymbol{h}_{k,i}^\star = \boldsymbol{h}_{k,i'}^\star$ for all $1 \leq i, i' \leq n$; next, it also says that the K class-mean features $\boldsymbol{h}_k^\star := \boldsymbol{h}_{k,i}^\star$ together exhibit

a K-simplex ETF up to scaling, from which we immediately conclude that

$$\cos \angle(\boldsymbol{h}_k^\star, \boldsymbol{h}_{k'}^\star) = -\frac{1}{K-1} \qquad [10]$$

for any $k \neq k'$ by Definition 1;[||] in addition, Eq. (9) also displays the precise duality between the last-layer classifiers and features. Taken together, these facts indicate that the minimizer $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ satisfies exactly (NC1)–(NC3). Last, Property (NC4) is also satisfied by recognizing that, for any given last-layer features $\boldsymbol{h}$, the predicted class is $\arg\max_k \boldsymbol{w}_k^\star \cdot \boldsymbol{h}$, where $\boldsymbol{a} \cdot \boldsymbol{b}$ denotes the inner product of the two vectors. Note that the prediction satisfies

$$\arg\max_k \boldsymbol{w}_k^\star \cdot \boldsymbol{h} = \arg\max_k \boldsymbol{h}_k^\star \cdot \boldsymbol{h} = \arg\min_k \|\boldsymbol{h}_k^\star - \boldsymbol{h}\|^2.$$

Conversely, the presence of neural collapse in the Layer-Peeled Model offers evidence of the effectiveness of our model as a tool for analyzing neural networks. To be complete, we remark that other models were very recently proposed to justify the neural collapse phenomenon (40–42) (see also (44)).

B. Extensions to Other Loss Functions. In the modern practice of deep learning, various loss functions are employed to take into account the problem characteristics. Here we show that the Layer-Peeled Model continues to exhibit the phenomenon of neural collapse for some popular loss functions.

Contrastive Loss. Contrastive losses have been extensively used recently in both supervised and unsupervised deep learning (10, 45–47). These losses pull similar training examples together in their embedding space while pushing apart dissimilar examples. Here we consider the supervised contrastive loss (48), which (in the balanced case) is defined through the last-layer features by introducing $\mathcal{L}_c$ as

$$\frac{1}{n}\sum_{j=1}^{n} -\log\left(\frac{\exp(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^{K}\sum_{\ell=1}^{n}\exp(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k',\ell}/\tau)}\right), \qquad [11]$$

where $\tau > 0$ is a parameter. Note that this loss function uses the label information implicitly. As the loss does not involve the last-layer classifiers explicitly, the Layer-Peeled Model in this case takes the form[**]

$$\min_{\boldsymbol{H}} \quad \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n}\mathcal{L}_c(\boldsymbol{h}_{k,i}, \boldsymbol{y}_k)$$
$$\text{s.t.} \quad \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{h}_{k,i}\|^2 \leq E_H. \qquad [12]$$

We show that this Layer-Peeled Model also exhibits neural collapse in its last-layer features, even though the label information is not explicitly explored in the loss.

Theorem 3. *Any global minimizer of* Eq. (12) *satisfies*

$$\boldsymbol{h}_{k,i}^\star = \sqrt{E_H}\boldsymbol{m}_k^\star \qquad [13]$$

for all $1 \leq k \leq K$ and $1 \leq i \leq n$, where $[\boldsymbol{m}_1^\star, \ldots, \boldsymbol{m}_K^\star]$ forms a K-simplex ETF.

[||] Note that the cosine value $-\frac{1}{K-1}$ corresponds to the largest possible angle for any K points that have an equal ℓ_2 norm and equal-sized angles between any pair. As pointed out in (1), the largest angle implies a large-margin solution (6).

[**] In Eq. (11), $\boldsymbol{h}_{k,i} \equiv \boldsymbol{h}(\boldsymbol{x}_{k,i}, \boldsymbol{W}_{-L})$ depends on the data, whereas in Eq. (12) $\boldsymbol{h}_{k,i}$'s form the decision variable $\boldsymbol{H}$.

Theorem 3 shows that the contrastive loss in the associated Layer-Peeled Model does a perfect job in pulling together training examples from the same class. Moreover, as seen from the denominator in Eq. (11), minimizing this loss would intuitively render the between-class inner products of last-layer features as small as possible, thereby pushing the features to form the vertices of a K-simplex ETF up to scaling.

Softmax-Based Loss. The cross-entropy loss can be thought of as a softmax-based loss. To see this, define the softmax transform as

$$\boldsymbol{S}(\boldsymbol{z}) = \left[\frac{\exp(\boldsymbol{z}(1))}{\sum_{k=1}^{K}\exp(\boldsymbol{z}(k))}, \ldots, \frac{\exp(\boldsymbol{z}(K))}{\sum_{k=1}^{K}\exp(\boldsymbol{z}(k))}\right]^\top$$

for $\boldsymbol{z} \in \mathbb{R}^K$. Let g_1 be any nonincreasing convex function and g_2 be any nondecreasing convex function, both defined on $(0, 1)$. We consider a softmax-based loss function that takes the form

$$\mathcal{L}(\boldsymbol{z}, \boldsymbol{y}_k) = g_1\left(\boldsymbol{S}(\boldsymbol{z})(k)\right) + \sum_{k'=1,\ k'\neq k}^{K} g_2\left(\boldsymbol{S}(\boldsymbol{z})(k')\right). \qquad [14]$$

Here, $\boldsymbol{S}(\boldsymbol{z})(k)$ denotes the k-th element of $\boldsymbol{S}(\boldsymbol{z})$. Taking $g_1(x) = -\log x$ and $g_2 \equiv 0$, we recover the cross-entropy loss. Another example is to take $g_1(x) = (1-x)^q$ and $g_2(x) = x^q$ for $q > 1$, which can be implemented in most deep learning libraries such as PyTorch (49).

We have the following theorem regarding the softmax-based loss functions in the balanced case.

Theorem 4. *Assume $\sqrt{E_H E_W} > \frac{K-1}{K}\log\left(K^2\sqrt{E_H E_W} + (2K-1)(K-1)\right)$. For any loss function defined in* Eq. (14)*, $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ given by* Eq. (9) *is a global minimizer of* Eq. (7)*. Moreover, if g_2 is strictly convex and at least one of g_1, g_2 is strictly monotone, then any global minimizer must be given by* Eq. (9)*.*

In other words, neural collapse continues to emerge with softmax-based losses under mild regularity conditions. The first part of this theorem does not preclude the possibility that the Layer-Peeled Model admits solutions other than Eq. (9). When applied to the cross-entropy loss, it is worth pointing out that this theorem is a weak version of Theorem 1, albeit more general. Regarding the first assumption in Theorem 4, note that E_H and E_W would be arbitrarily large if the weight decay λ in Eq. (1) is sufficiently small, thereby meeting the assumption concerning $\sqrt{E_H E_W}$ in this theorem.

We remark that Theorem 4 does not require the convexity of the loss $\mathcal{L}$. To circumvent the hurdle of nonconvexity, our proof in SI Appendix presents several novel elements.

In passing, we leave the experimental confirmation of neural collapse with these loss functions for future work.

4. Layer-Peeled Model for Predicting Minority Collapse

Deep learning models are often trained on datasets where there is a disproportionate ratio of observations in each class (50–52). For example, in the Places2 challenge dataset (53), the number of images in its majority scene categories is about eight times that in its minority classes. Another example is the Ontonotes dataset for part-of-speech tagging (54), where the number of words in its majority classes can be more

than one hundred times that in its minority classes. While empirically the imbalance in class sizes often leads to inferior model performance of deep learning (see, e.g., (11)), there remains a lack of a solid theoretical footing for understanding its effect, perhaps due to the complex details of deep learning training.

In this section, we use the Layer-Peeled Model to seek a fine-grained characterization of how class imbalance impacts neural networks that are trained for a sufficiently long time. In particular, neural collapse no longer emerges in the presence of class imbalance (see numerical evidence in Figure S2 in SI Appendix). Instead, our analysis predicts a phenomenon we term *Minority Collapse*, which fundamentally limits the performance of deep learning especially on the minority classes, both theoretically and empirically. All omitted proofs are relegated to SI Appendix.

A. Technique: Convex Relaxation. When it comes to imbalanced datasets, the Layer-Peeled Model no longer admits a simple expression for its minimizers as in the balanced case, due to the lack of symmetry between classes. This fact results in, among others, an added burden on numerically computing the solutions of the Layer-Peeled Model.

To overcome this difficulty, we introduce a convex optimization program as a relaxation of the nonconvex Layer-Peeled Model Eq. (7), relying on the well-known result for relaxing a quadratically constrained quadratic program as a semidefinite program (see, e.g., (55)). To begin with, defining $\boldsymbol{h}_k$ as the feature mean of the k-th class (i.e., $\boldsymbol{h}_k := \frac{1}{n_k} \sum_{i=1}^{n_k} \boldsymbol{h}_{k,i}$), we introduce a new decision variable $\boldsymbol{X} := \left[\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_K, \boldsymbol{W}^\top\right]^\top \left[\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_K, \boldsymbol{W}^\top\right] \in \mathbb{R}^{2K \times 2K}$. By definition, $\boldsymbol{X}$ is positive semidefinite and satisfies

$$\frac{1}{K} \sum_{k=1}^{K} \boldsymbol{X}(k,k) = \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{h}_k\|^2 \overset{a}{\leq} \frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \|\boldsymbol{h}_{k,i}\|^2 \leq E_H$$

and

$$\frac{1}{K} \sum_{k=K+1}^{2K} \boldsymbol{X}(k,k) = \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 \leq E_W,$$

where $\overset{a}{\leq}$ follows from the Cauchy–Schwarz inequality. Thus, we consider the following semidefinite programming problem:[††]

$$
\begin{aligned}
\min_{\boldsymbol{X} \in \mathbb{R}^{2K \times 2K}} \quad & \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}(\boldsymbol{z}_k, \boldsymbol{y}_k) \\
\text{s.t.} \quad & \boldsymbol{X} \succeq 0, \\
& \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{X}(k,k) \leq E_H, \quad \frac{1}{K} \sum_{k=K+1}^{2K} \boldsymbol{X}(k,k) \leq E_W, \\
& \text{for all } 1 \leq k \leq K, \\
& \boldsymbol{z}_k = \left[\boldsymbol{X}(k,K+1), \boldsymbol{X}(k,K+2), \ldots, \boldsymbol{X}(k,2K)\right]^\top.
\end{aligned}
\tag{15}
$$

Lemma 1 below relates the solutions of Eq. (15) to that of Eq. (7).

Lemma 1. *Assume $p \geq 2K$ and the loss function $\mathcal{L}$ is convex in its first argument. Let $\boldsymbol{X}^\star$ be a minimizer of the convex program (15). Define $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ as*

$$
\begin{aligned}
\left[\boldsymbol{h}_1^\star, \boldsymbol{h}_2^\star, \ldots, \boldsymbol{h}_K^\star, (\boldsymbol{W}^\star)^\top\right] &= \boldsymbol{P}(\boldsymbol{X}^\star)^{1/2}, \\
\boldsymbol{h}_{k,i}^\star &= \boldsymbol{h}_k^\star, \quad \text{for all } 1 \leq i \leq n, 1 \leq k \leq K,
\end{aligned}
\tag{16}
$$

where $(\boldsymbol{X}^\star)^{1/2}$ denotes the positive square root of $\boldsymbol{X}^\star$ and $\boldsymbol{P} \in \mathbb{R}^{p \times 2K}$ is any partial orthogonal matrix such that $\boldsymbol{P}^\top \boldsymbol{P} = \boldsymbol{I}_{2K}$. Then $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a minimizer of Eq. (7). Moreover, if all $\boldsymbol{X}^\star$'s satisfy $\frac{1}{K} \sum_{k=1}^{K} \boldsymbol{X}^\star(k,k) = E_H$, then all the solutions of Eq. (7) are in the form of Eq. (16).

This lemma in effect says that the relaxation does *not* lead to any loss of information when we study the Layer-Peeled Model through a convex program, thereby offering a computationally efficient tool for gaining insights into the terminal phase of training deep neural networks on imbalanced datasets. An appealing feature is that the size of the program (15) is independent of the number of training examples. Besides, this lemma predicts that even in the imbalanced case the last-layer features collapse to their class means under mild conditions. Therefore, Property (NC1) is satisfied (see more discussion about the condition in SI Appendix).

The assumption of the convexity of $\mathcal{L}$ in the first argument is satisfied by a large class of loss functions. The condition that the first K diagonal elements of any $\boldsymbol{X}^\star$ make the associated constraint saturated is also not restrictive. For example, we prove in SI Appendix that this condition is satisfied for the cross-entropy loss. We also remark that Eq. (15) is not the unique convex relaxation. An alternative is to relax Eq. (7) via a nuclear norm-constrained convex program (56, 57) (see more details in SI Appendix).

B. Minority Collapse. With the technique of convex relaxation in place, now we numerically solve the Layer-Peeled Model on imbalanced datasets, with the goal of identifying possible non-trivial patterns. As a worthwhile starting point, we consider a dataset that has K_A majority classes each containing n_A training examples and K_B minority classes each containing n_B training examples. That is, assume $n_1 = n_2 = \cdots = n_{K_A} = n_A$ and $n_{K_A+1} = n_{K_A+2} = \cdots = n_K = n_B$. For convenience, call $R := n_A / n_B > 1$ the imbalance ratio. Note that the case $R = 1$ reduces to the balanced setting.

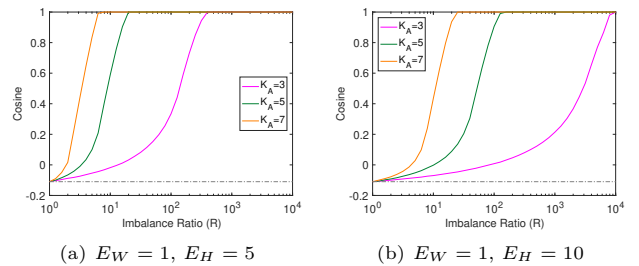(a) $E_W = 1$, $E_H = 5$ (b) $E_W = 1$, $E_H = 10$

Fig. 3. The average cosine of the angles between any pair of the minority classifier solved from the Layer-Peeled Model. The average cosine reaches 1 once R is above some threshold. The total number of classes $K_A + K_B$ is fixed to 10. The gray dash-dotted line indicates the value of $-\frac{1}{K-1}$, which is given by Eq. (10). The between-majority-class angles can still be large even when Minority Collapse emerges. Notably, our simulation suggests that the minority classifiers exhibit an equiangular frame and so do the majority classifiers.

[††] Although Eq. (15) involves a semidefinite constraint, it is not a semidefinite program in the strict sense because a semidefinite program uses a linear objective function.

An important question is to understand how the K_B last-layer minority classifiers behave as the imbalance ratio R increases, as this is directly related to the model performance on the minority classes. To address this question, we show that the average cosine of the angles between any pair of the K_B minority classifiers in Figure 3 by solving the simple convex program (15). This figure reveals a two-phase behavior of the minority classifiers $\boldsymbol{w}_{K_A+1}^\star, \boldsymbol{w}_{K_A+2}^\star, \ldots, \boldsymbol{w}_K^\star$ as R increases:

(1) When $R < R_0$ for some $R_0 > 0$, the average between-minority-class angle becomes smaller as R increases.

(2) Once $R \geq R_0$, the average between-minority-class angle become zero and, in addition, the minority classifiers have about the same length. This implies that all the minority classifiers collapse to a single vector.

Above, the phase transition point R_0 depends on the class sizes K_A, K_B and the thresholds E_H, E_W. This value becomes smaller when E_W, E_H, or the numer of minority classes K_B is smaller while fixing the other parameters (see more numerical examples in Figure S2 in SI Appendix).

We refer to the phenomenon that appears in the second phase as Minority Collapse. While it can be expected that the minority classifiers become closer to each other as the level of imbalance increases, surprisingly, these classifiers become completely indistinguishable once R hits a *finite* value. Once Minority Collapse takes place, the neural network would predict equal probabilities for all the minority classes regardless of the input. As such, its predictive ability is by no means better than a coin toss when conditioned on the minority classes. This situation would only get worse in the presence of adversarial perturbations. This phenomenon is especially detrimental when the minority classes are more frequent in the application domains than in the training data. Even outside the regime of Minority Collapse, the classification might still be unreliable if the imbalance ratio is large as the softmax predictions for the minority classes can be close to each other.

To put the observations in Figure 3 on a firm footing, we prove in the theorem below that Minority Collapse indeed emerges in the Layer-Peeled Model as R tends to infinity.

Theorem 5. *Assume $p \geq K$ and $n_A/n_B \to \infty$, and fix K_A and K_B. Let $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ be any global minimizer of the Layer-Peeled Model* Eq. (7) *with the cross-entropy loss. As $R \equiv n_A/n_B \to \infty$, we have*

$$\lim \boldsymbol{w}_k^\star - \boldsymbol{w}_{k'}^\star = \boldsymbol{0}_p, \quad \text{for all } K_A < k < k' \leq K.$$

To intuitively see why Minority Collapse occurs, first note that the majority classes become the predominant part of the risk function as the level of imbalance increases. The minimization of the objective, therefore, pays too much emphasis on the majority classifiers, encouraging the between-majority-class angles to grow and meanwhile shrinking the between-minority-class angles to zero. As an aside, an interesting question for future work is to prove that $\boldsymbol{w}_k^\star$ and $\boldsymbol{w}_{k'}^\star$ are exactly equal for sufficiently large R.

C. Experiments. At the moment, Minority Collapse is merely a prediction of the Layer-Peeled Model. An immediate question thus is: does this phenomenon really occur in real-world neural networks? At first glance, it does not necessarily have to be the case since the Layer-Peeled Model is a dramatic simplification of deep neural networks.

To address this question, we resort to computational experiments.[‡‡] Explicitly, we consider training two network architectures, VGG and ResNet (58), on the FashionMNIST (59) and CIFAR10 datasets, and in particular, replace the dropout layers in VGG with batch normalization (60). As both datasets have 10 classes, we use three combinations of $(K_A, K_B) = (3, 7), (5, 5), (7, 3)$ to split the data into majority classes and minority classes. In the case of FashionMNIST (CIFAR10), we let the K_A majority classes each contain all the $n_A = 6000$ ($n_A = 5000$) training examples from the corresponding class of FashionMNIST (CIFAR10), and the K_B minority classes each have $n_B = 6000/R$ ($n_B = 5000/R$) examples randomly sampled from the corresponding class. The rest experiment setup is basically the same as (1). In detail, we use the cross-entropy loss and stochastic gradient descent with momentum 0.9 and weight decay $\lambda = 5 \times 10^{-4}$. The networks are trained for 350 epochs with a batch size of 128. The initial learning is annealed by a factor of 10 at $1/3$ and $2/3$ of the 350 epochs. The only difference from (1) is that we simply set the learning rate to 0.1 instead of sweeping over 25 learning rates between 0.0001 and 0.25. This is because the test performance of our trained models is already comparable with their best reported test accuracy. Detailed training and test performance is displayed in Tables S1 and S2 in SI Appendix.

The results of the experiments above are displayed in Figure 4. This figure clearly indicates that the angles between the minority classifiers collapse to zero as soon as R is large enough. Moreover, the numerical examination in Table 1 shows that the norm of the classifier is constant across the minority classes. Taken together, these two pieces clearly give evidence for the emergence of Minority Collapse in these neural networks, thereby further demonstrating the effectiveness of our Layer-Peeled Model. Besides, Figure 4 also shows that the issue of Minority Collapse is compounded when there are more majority classes, which is consistent with Figure 3.

Next, in order to get a handle on how Minority Collapse impacts the test accuracy, we plot the results of another numerical study in Figure 5. The setting is the same as Figure 4, except that now we randomly sample 6 or 5 examples per class for the minority classes depending on whether the dataset is FashionMNIST or CIFAR10. The results show that the performance of the trained model deteriorates in the test data when the imbalance ratio $R = 1000$, when Minority Collapse has occurred or is about to occur. This is by no means intuitive a priori as the test performance is only restricted to the minority classes and a large value of R only leads to more training data in the majority classes without affecting the minority classes at all.

It is worthwhile to mention that the emergence of Minority Collapse would prevent the model from achieving zero training error. This is because its prediction is uniform over the minority classes and, therefore, the "argmax" rule does not give the correct label for a training example from a minority class. As such, the occurrence of Minority Collapse is a departure from the terminal phase of deep learning training. While this fact seems to contradict conventional wisdom on the approximation power of deep learning, it is important to note that the constraints in the Layer-Peeled Model or, equivalently, weight decay in neural networks limits the expressive power of deep learning models. Besides, it is equally important to recognize

---

[‡‡] Our code is publicly available at https://github.com/HornHehhf/LPM.

Dataset	FashionMNIST						CIFAR10					
Network architecture	VGG11			ResNet18			VGG13			ResNet18		
No. of majority classes	$K_A=3$	$K_A=5$	$K_A=7$	$K_A=3$	$K_A=5$	$K_A=7$	$K_A=3$	$K_A=5$	$K_A=7$	$K_A=3$	$K_A=5$	$K_A=7$
Norm variation	2.7×10^{-5}	4.4×10^{-8}	6.0×10^{-8}	1.4×10^{-5}	5.0^{-8}	6.3×10^{-8}	1.4×10^{-4}	9.0×10^{-7}	5.2×10^{-8}	5.4×10^{-5}	3.5×10^{-7}	5.4×10^{-8}

Table 1. Variability of the lengths of the minority classifiers when $R = \infty$. Each number in the row of "norm variation" is $\mathrm{Std}(\|\boldsymbol{w}_B^\star\|)/\mathrm{Avg}(\|\boldsymbol{w}_B^\star\|)$, where $\mathrm{Std}(\|\boldsymbol{w}_B^\star\|)$ denotes the standard deviation of the lengths of the K_B classifiers and the denominator denotes the average. The results indicate that the classifiers of the minority classes have almost the same length.
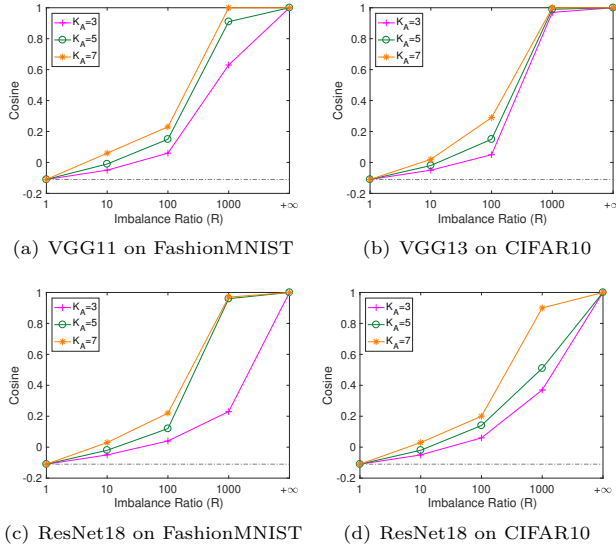

(a) VGG11 on FashionMNIST (b) VGG13 on CIFAR10

(c) ResNet18 on FashionMNIST (d) ResNet18 on CIFAR10

Fig. 4. Occurrence of Minority Collapse in deep neural networks. Each curve denotes the average between-minority-class cosine. We fix $K_A + K_B = 10$. In particular, Figure 4(b) shares the same setting with Figure 2 in Section 1, where the LPM-based predictions are given by (E_W, E_H) such that the two constraints in the Layer-Peeled Model become active for the weights of the trained networks. For ResNet 18, Minority Collapse also occurs as long as R is sufficiently large. Specifically, the average cosine would hit 1 for $K_A = 7$ when $R = 5000$ on CIFAR10, and when $R = 3000$ on FashionMNIST.
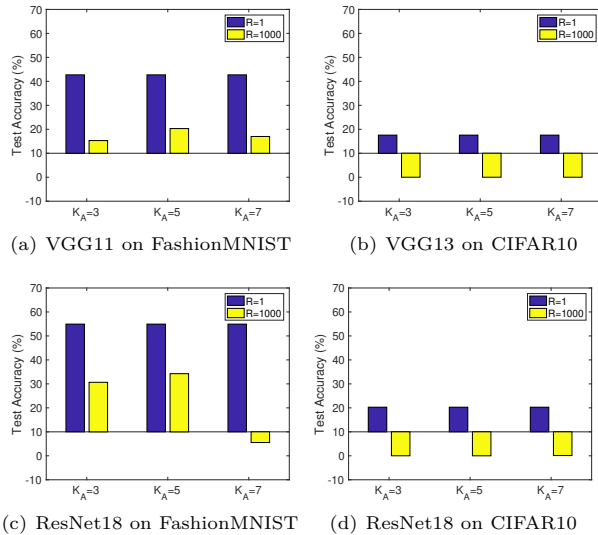

(a) VGG11 on FashionMNIST (b) VGG13 on CIFAR10

(c) ResNet18 on FashionMNIST (d) ResNet18 on CIFAR10

Fig. 5. Comparison of the test accuracy on the minority classes between $R = 1$ and $R = 1000$. We fix $K_A + K_B = 10$ and use $n_B = 6$ ($n_B = 5$) training examples from each minority class and $n_A = 6R$ ($n_A = 5R$) training examples from each majority class in FashionMNIST (CIFAR10). Note that when $R = 1000$, the test accuracy on the minority classes can be lower than 10% because the trained neural networks misclassify many examples in the minority classes as some majority classes.

that the training error, which mostly occurs in the minority classes, is actually very small when Minority Collapse emerges since the minority examples only account for a small portion of the entire training set. In this spirit, the aforementioned departure is not as significant as it appears at first glance since the training error is generally, if not always, not exactly zero (see, e.g., (1)). From an optimization point of view, a careful examination indicates that Minority Collapse can be attributed to the two constraints in the Layer-Peeled Model or the ℓ_2 regularization in Eq. (1). For example, Figure 2 shows that Minority Collapse occurs earlier with a larger value of λ. However, this issue does not disappear by simply setting a small penalty coefficient λ as the imbalance ratio can be arbitrarily large.

5. How to Mitigate Minority Collapse?

In this section, we further exploit the use of the Layer-Peeled Model in an attempt to lessen the detrimental effect of Minority Collapse. Instead of aiming to develop a full set of methodologies to overcome this issue, which is beyond the scope of the paper, our aim is to evaluate some simple techniques used for imbalanced datasets.

Among many approaches to handling class imbalance in deep learning (see the review (11)), perhaps the most popular one is to oversample training examples from the minority classes (61–64). In its simplest form, this sampling scheme retains all majority training examples while duplicating each training example from the minority classes for w_r times, where the oversampling rate w_r is a positive integer. Oversampling in effect transforms the original problem to the minimization of a new optimization problem by replacing the risk term in Eq. (1) with

$$\frac{1}{n_A K_A + w_r n_B K_B} \left[\sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}), \boldsymbol{y}_k) + w_r \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B} \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}), \boldsymbol{y}_k) \right] \quad [17]$$

while keeping the penalty term $\frac{\lambda}{2}\|\boldsymbol{W}_{\text{full}}\|^2$. Note that oversampling is closely related to weight adjusting (see more discussion in SI Appendix).

A close look at Eq. (17) suggests that the neural network obtained by minimizing this new program might behave as if it were trained on a (larger) dataset with n_A and $w_r n_B$ examples in each majority class and minority class, respectively. To formalize this intuition, as earlier, we start by considering the

Layer-Peeled Model in the case of oversampling:

$$\min_{\boldsymbol{H},\boldsymbol{W}} \frac{1}{N'}\left[\sum_{k=1}^{K_A}\sum_{i=1}^{n_A}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i},\boldsymbol{y}_k) + w_r\sum_{k=K_A+1}^{K}\sum_{i=1}^{n_B}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i},\boldsymbol{y}_k)\right]$$

$$\text{s.t. } \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 \le E_W, \qquad [18]$$

$$\frac{1}{K}\sum_{k=1}^{K_A}\frac{1}{n_A}\sum_{i=1}^{n_A}\|\boldsymbol{h}_{k,i}\|^2 + \frac{1}{K}\sum_{k=K_A+1}^{K}\frac{1}{n_B}\sum_{i=1}^{n_B}\|\boldsymbol{h}_{k,i}\|^2 \le E_H,$$

where $N' := n_A K_A + w_r n_B K_B$.

(a) VGG11 on FashionMNIST (b) VGG13 on CIFAR10

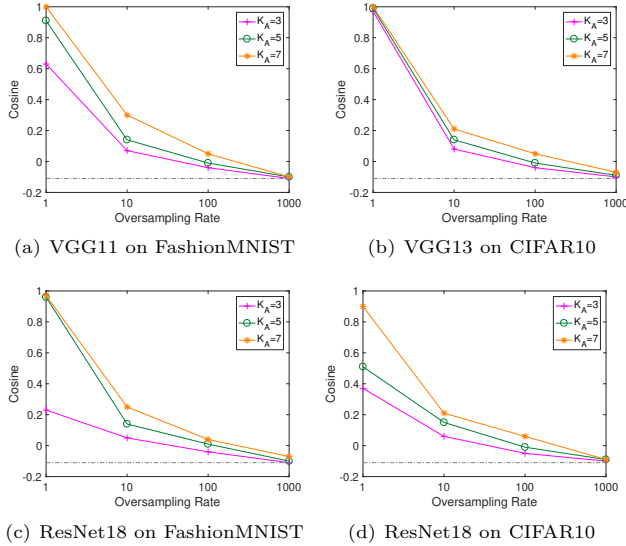(c) ResNet18 on FashionMNIST (d) ResNet18 on CIFAR10

Fig. 6. Effect of oversampling when the imbalance ratio is $R = 1000$. Each plot shows the average cosine of the between-minority-class angles. The results indicate that increasing the oversampling rate would enlarge the between-minority-class angles.

The following result confirms our intuition that oversampling indeed boosts the size of the minority classes for the Layer-Peeled Model.

Proposition 1. *Assume $p \ge 2K$ and the loss function $\mathcal{L}$ is convex in the first argument. Let $\boldsymbol{X}^\star$ be any minimizer of the convex program (15) with $n_1 = n_2 = \cdots = n_{K_A} = n_A$ and $n_{K_A+1} = n_{K_A+2} = \cdots = n_K = w_r n_B$. Define $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ as*

$$\left[\boldsymbol{h}_1^\star, \boldsymbol{h}_2^\star, \ldots, \boldsymbol{h}_K^\star, (\boldsymbol{W}^\star)^\top\right] = \boldsymbol{P}(\boldsymbol{X}^\star)^{1/2},$$
$$\boldsymbol{h}_{k,i}^\star = \boldsymbol{h}_k^\star, \quad \text{for all } 1 \le i \le n_A, 1 \le k \le K_A, \qquad [19]$$
$$\boldsymbol{h}_{k,i}^\star = \boldsymbol{h}_k^\star, \quad \text{for all } 1 \le i \le n_B, K_A < k \le K,$$

where $\boldsymbol{P} \in \mathbb{R}^{p \times 2K}$ is any partial orthogonal matrix such that $\boldsymbol{P}^\top \boldsymbol{P} = \boldsymbol{I}_{2K}$. Then $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a global minimizer of the oversampling-adjusted Layer-Peeled Model Eq. (18). Moreover, if all $\boldsymbol{X}^\star$'s satisfy $\frac{1}{K}\sum_{k=1}^{K}\boldsymbol{X}^\star(k,k) = E_H$, then all the solutions of Eq. (18) are in the form of Eq. (19).

Together with Lemma 1, Proposition 1 shows that the number of training examples in each minority class is now in effect $w_r n_B$ instead of n_B in the Layer-Peeled Model. In the special case $w_r = n_A/n_B \equiv R$, the results show that all the angles are equal between any given pair of the last-layer

classifiers, no matter if they fall in the majority or minority classes.

We turn to Figure 6 for an illustration of the effects of oversampling on real-world deep learning models, using the same experimental setup as in Figure 5. From Figure 6, we see that the angles between pairs of the minority classifiers become larger as the oversampling rate w_r increases. Consequently, the issue of Minority Collapse becomes less detrimental in terms of training accuracy as w_r increases. This again corroborates the predictive ability of the Layer-Peeled Model.

Network architecture	VGG11			ResNet18		
No. of majority classes	$K_A = 3$	$K_A = 5$	$K_A = 7$	$K_A = 3$	$K_A = 5$	$K_A = 7$
Original (minority)	15.29	20.30	17.00	30.66	34.26	5.53
Oversampling (minority)	**41.13**	**57.22**	**30.50**	**37.86**	**53.46**	**8.13**
Improvement (minority)	25.84	36.92	13.50	7.20	19.20	2.60
Original (overall)	40.10	57.61	69.09	50.88	64.89	66.13
Oversampling (overall)	58.25	76.17	73.37	55.91	74.56	67.10
Improvement (overall)	18.15	18.56	4.28	5.03	9.67	0.97

Table 2. Test accuracy (%) on FashionMNIST when $R = 1000$. **For example, "Original (minority)" means that the test accuracy is evaluated only on the minority classes and oversampling is not used. When oversampling is used, we report the best test accuracy among four oversampling rates: 1, 10, 100, and 1000. The best test accuracy is never achieved at $w_r = 1000$, indicating that oversampling with a large w_r would impair the test performance.**

Next, we refer to Table 2 for effect on the test performance. The results clearly demonstrate the improvement in test accuracy using oversampling, with certain choices of the oversampling rate. The improvement is noticeable on both the minority classes and all classes.

Behind the results of Table 2, however, it reveals an issue when addressing Minority Collapse by oversampling. Specifically, this technique might lead to degradation of test performance using a very large oversampling rate w_r, which though can mitigate Minority Collapse. How can we efficiently select an oversampling rate for optimal test performance? More broadly, Minority Collapse does not seem likely to be fully resolved by sampling-based approaches alone, and the doors are widely open for future investigation.

6. Discussion

In this paper, we have developed the Layer-Peeled Model as a simple yet effective modeling strategy toward understanding well-trained deep neural networks. The derivation of this model follows a top-down strategy by isolating the last layer from the remaining layers. Owing to the analytical and numerical tractability of the Layer-Peeled Model, we provide some explanation of a recently observed phenomenon called neural collapse in deep neural networks trained on balanced datasets (1). Moving to imbalanced datasets, an analysis of this model suggests that the last-layer classifiers corresponding to the minority classes would collapse to a single vector once the imbalance level is above a certain threshold. This new phenomenon, which we refer to as Minority Collapse, occurs consistently in our computational experiments.

The efficacy of the Layer-Peeled Model in analyzing well-trained deep learning models implies that the ansatz Eq. (6)—a crucial step in the derivation of this model—is at least a useful approximation. Moreover, this ansatz can be further justified by the following result in an indirect manner, which, together with Theorem 1, shows that the ℓ_2 norm suggested by the

ansatz happens to be the only choice among all the ℓ_q norms that is consistent with empirical observations. Its proof is given in SI Appendix.

Proposition 2. *Assume $K \geq 3$ and $p \geq K$.*[§§] *For any $q \in (0,2) \cup (2,\infty)$, consider the optimization problem*

$$\min_{\boldsymbol{W},\boldsymbol{H}} \quad \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k)$$

$$\text{s.t.} \quad \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 \leq E_W, \qquad [20]$$

$$\frac{1}{K} \sum_{k=1}^{K} \frac{1}{n} \sum_{i=1}^{n} \|\boldsymbol{h}_{k,i}\|_q^q \leq E_H,$$

where $\mathcal{L}$ is the cross-entropy loss. Then, any global minimizer of this program does not satisfy Eq. (9) *for any positive numbers C and C'. That is, neural collapse does not emerge in this model.*

While the Layer-Peeled Model has demonstrated its noticeable effectiveness, it requires future investigation for consolidation and extension. First, an analysis of the gap between the Layer-Peeled Model and well-trained deep learning models would be a welcome advance. For example, how does the gap depend on the neural network architectures? How to take into account the sparsity of the last-layer features when using the ReLU activation function? From a different angle, a possible extension is to retain multiple layers following the top-down viewpoint. Explicitly, letting $1 \leq m < L$ be the number of the top layers we wish to retain in the model, we can represent the prediction of the neural network as $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{W}_{\text{full}}) = \boldsymbol{f}(\boldsymbol{h}(\boldsymbol{x}; \boldsymbol{W}_{1:(L-m)}), \boldsymbol{W}_{(L-m+1):L})$ by letting $\boldsymbol{W}_{1:(L-m)}$ and $\boldsymbol{W}_{(L-m+1):L}$ be the first $L - m$ layers and the last m layers, respectively. Consider the m-Layer-Peeled Model:

$$\min_{\boldsymbol{W},\boldsymbol{H}} \quad \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}(\boldsymbol{f}(\boldsymbol{h}_{k,i}, \boldsymbol{W}_{(L-m+1):L}), \boldsymbol{y}_k)$$

$$\text{s.t.} \quad \frac{1}{K} \|\boldsymbol{W}_{(L-m+1):L}\|^2 \leq E_W,$$

$$\frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \|\boldsymbol{h}_{k,i}\|^2 \leq E_H.$$

The two constraints might be modified to take into account the network architectures. An immediate question is whether this model with $m = 2$ is capable of capturing new patterns of deep learning training.

From a practical standpoint, the Layer-Peeled Model together with its convex relaxation Eq. (15) offers an analytical and computationally efficient technique to identify and mitigate bias induced by class imbalance. An interesting question is to extend Minority Collapse from the case of two-valued class sizes to general imbalanced datasets. Next, as suggested by our findings in Section 5, how should we choose loss functions in order to mitigate Minority Collapse (64)? Last, a possible use case of the Layer-Peeled Model is to design more efficient sampling schemes to take into account fairness considerations (65–67).

Broadly speaking, insights can be gained not only from the Layer-Peeled Model but also from its modeling strategy. The details of empirical deep learning models, though formidable, can often be simplified by rendering a certain part of the network *modular*. When the interest is about the top few layers, for example, this paper clearly demonstrates the benefits of taking a top-down strategy for modeling neural networks especially in consolidating our understanding of previous results and in discovering new patterns. Owing to its mathematical convenience, the Layer-Peeled Model shall open the door for future research extending these benefits.

1. V Papyan, X Han, DL Donoho, Prevalence of neural collapse during the terminal phase of deep learning training. *Proc. Natl. Acad. Sci.* **117**, 24652–24663 (2020).
2. A Krizhevsky, I Sutskever, GE Hinton, Imagenet classification with deep convolutional neural networks. *Commun. ACM* **60**, 84–90 (2017).
3. Y LeCun, Y Bengio, G Hinton, Deep learning. *Nature* **521**, 436–444 (2015).
4. D Silver, et al., Mastering the game of go with deep neural networks and tree search. *Nature* **529**, 484–489 (2016).
5. AR Webb, D Lowe, The optimised internal representation of multilayer classifier networks performs nonlinear discriminant analysis. *Neural Networks* **3**, 367–375 (1990).
6. D Soudry, E Hoffer, MS Nacson, S Gunasekar, N Srebro, The implicit bias of gradient descent on separable data. *The J. Mach. Learn. Res.* **19**, 2822–2878 (2018).
7. S Oymak, M Soltanolkotabi, Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE J. on Sel. Areas Inf. Theory* **1**, 84–105 (2020).
8. Y Yu, KHR Chan, C You, C Song, Y Ma, Learning diverse and discriminative representations via the principle of maximal coding rate reduction. *Adv. Neural Inf. Process. Syst.* **33** (2020).
9. O Shamir, Gradient methods never overfit on separable data. arXiv:2007.00028 (10 Sep 2020).
10. T Chen, S Kornblith, M Norouzi, G Hinton, A simple framework for contrastive learning of visual representations in *Proceedings of the 37th International Conference on Machine Learning*, Proceedings of Machine Learning Research, eds. HD III, A Singh. (PMLR), Vol. 119, pp. 1597–1607 (2020).
11. JM Johnson, TM Khoshgoftaar, Survey on deep learning with class imbalance. *J. Big Data* **6**, 27 (2019).
12. A Krizhevsky, Master's thesis (University of Toronto) (2009).
13. K Simonyan, A Zisserman, Very deep convolutional networks for large-scale image recognition in *International Conference on Learning Representations*. (2015).
14. D Yarotsky, Error bounds for approximations with deep ReLU networks. *Neural Networks* **94**, 103–114 (2017).
15. A Jacot, F Gabriel, C Hongler, Neural tangent kernel: Convergence and generalization in neural networks in *Advances in Neural Information Processing Systems*. (2018).
16. SS Du, JD Lee, H Li, L Wang, X Zhai, Gradient descent finds global minima of deep neural networks in *International Conference on Machine Learning*. (2019).
17. Z Allen-Zhu, Y Li, Z Song, A convergence theory for deep learning via over-parameterization in *International Conference on Machine Learning*. pp. 2388–2464 (2019).
18. D Zou, Y Cao, D Zhou, Q Gu, Stochastic gradient descent optimizes over-parameterized deep relu networks in *Advances in Neural Information Processing Systems*. (2018).
19. L Chizat, E Oyallon, F Bach, On lazy training in differentiable programming in *Advances in Neural Information Processing Systems*. (2019).
20. W E, C Ma, L Wu, A comparative analysis of the optimization and generalization property of two-layer neural network and random feature models under gradient descent dynamics. arXiv:1904.04326 (21 Feb 2020).
21. P Bartlett, D Foster, M Telgarsky, Spectrally-normalized margin bounds for neural networks. *Adv. Neural Inf. Process. Syst.* **30**, 6241–6250 (2017).
22. H He, WJ Su, The local elasticity of neural networks in *International Conference on Learning Representations*. (2020).
23. T Poggio, A Banburski, Q Liao, Theoretical issues in deep networks. *Proc. Natl. Acad. Sci.* **117**, 30039–30045 (2020).
24. S Mei, A Montanari, PM Nguyen, A mean field view of the landscape of two-layer neural networks. *Proc. Natl. Acad. Sci.* **115**, E7665–E7671 (2018).
25. J Sirignano, K Spiliopoulos, Mean field analysis of neural networks: A central limit theorem. *Stoch. Process. their Appl.* **130**, 1820–1852 (2020).
26. GM Rotskoff, E Vanden-Eijnden, Neural networks as interacting particle systems: Asymptotic convexity of the loss landscape and universal scaling of the approximation error in *Advances in Neural Information Processing Systems*. (2018).
27. C Fang, JD Lee, P Yang, T Zhang, Modeling from features: a mean-field framework for over-parameterized deep neural networks. arXiv:2007.01452 (3 July 2020).

[§§]See discussion in the case $K = 2$ in SI Appendix.

28. R Kuditipudi, et al., Explaining landscape connectivity of low-cost solutions for multilayer nets in *Advances in Neural Information Processing Systems*. pp. 14601–14610 (2019).

29. B Shi, WJ Su, MI Jordan, On learning rates and Schrödinger operators. arXiv:2004.06977 (15 Apr 2020).

30. C Fang, H Dong, T Zhang, Mathematical models of overparameterized neural networks. arXiv:2012.13982 (27 Dec 2020).

31. F He, D Tao, Recent advances in deep learning theory. arXiv:2012.10931 (20 Dec 2020).

32. J Fan, C Ma, Y Zhong, A selective overview of deep learning. arXiv:1904.05526 (15 Apr 2019).

33. R Sun, Optimization for deep learning: theory and algorithms. arXiv:1912.08957 (19 Dec 2019).

34. T Strohmer, RW Heath, Grassmannian frames with applications to coding and communication. *Appl. Comput. Harmon. Analysis* **14**, 257–275 (2003).

35. S Ma, R Bassily, M Belkin, The power of interpolation: Understanding the effectiveness of sgd in modern over-parametrized learning in *International Conference on Machine Learning*. (PMLR), pp. 3325–3334 (2018).

36. M Belkin, D Hsu, S Ma, S Mandal, Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proc. Natl. Acad. Sci.* **116**, 15849–15854 (2019).

37. T Liang, A Rakhlin, Just interpolate: Kernel "ridgeless" regression can generalize. *Annals Stat.* **48**, 1329–1347 (2020).

38. PL Bartlett, PM Long, G Lugosi, A Tsigler, Benign overfitting in linear regression. *Proc. Natl. Acad. Sci.* **117**, 30063–30070 (2020).

39. Z Li, W Su, D Sejdinovic, Benign overfitting and noisy features. arXiv:2008.02901 (6 Aug 2020).

40. DG Mixon, H Parshall, J Pi, Neural collapse with unconstrained features. arXiv:2011.11619 (23 Nov 2020).

41. W E, S Wojtowytsch, On the emergence of tetrahedral symmetry in the final and penultimate layers of neural network classifiers. arXiv:2012.05420 (19 Dec 2020).

42. J Lu, S Steinerberger, Neural collapse with cross-entropy loss. arXiv:2012.08465 (18 Jan 2021).

43. T Ergen, M Pilanci, Convex duality of deep neural networks. arXiv preprint arXiv:2002.09773 (22 Feb 2020).

44. T Poggio, Q Liao, Explicit regularization and implicit bias in deep network classifiers trained with the square loss. arXiv:2101.00072 (31 Dec 2020).

45. J Pennington, R Socher, CD Manning, Glove: Global vectors for word representation in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. pp. 1532–1543 (2014).

46. N Saunshi, O Plevrakis, S Arora, M Khodak, H Khandeparkar, A theoretical analysis of contrastive unsupervised representation learning in *Proceedings of the 36th International Conference on Machine Learning*, Proceedings of Machine Learning Research, eds. K Chaudhuri, R Salakhutdinov. (PMLR), Vol. 97, pp. 5628–5637 (2019).

47. A Baevski, H Zhou, A Mohamed, M Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations. arXiv:2006.11477 (22 Oct 2020).

48. P Khosla, et al., Supervised contrastive learning. arXiv:2004.11362 (10 Dec 2020).

49. A Paszke, et al., Pytorch: An imperative style, high-performance deep learning library in *Advances in neural information processing systems*. pp. 8026–8037 (2019).

50. S Wang, et al., Training deep neural networks on imbalanced data sets in *2016 international joint conference on neural networks (IJCNN)*. (IEEE), pp. 4368–4374 (2016).

51. C Huang, Y Li, CC Loy, X Tang, Learning deep representation for imbalanced classification in *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 5375–5384 (2016).

52. K Madasamy, M Ramaswami, Data imbalance and classifiers: impact and solutions from a big data perspective. *Int. J. Comput. Intell. Res.* **13**, 2267–2281 (2017).

53. B Zhou, A Khosla, A Lapedriza, A Torralba, A Oliva, Places: An image database for deep scene understanding. arXiv:1610.02055 (6 Oct 2016).

54. E Hovy, M Marcus, M Palmer, L Ramshaw, R Weischedel, Ontonotes: the 90% solution in *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*. pp. 57–60 (2006).

55. JF Sturm, S Zhang, On cones of nonnegative quadratic functions. *Math. Oper. Res.* **28**, 246–267 (2003).

56. F Bach, J Mairal, J Ponce, Convex sparse matrix factorizations. arXiv:0812.1869 (10 Dec 2008).

57. BD Haeffele, R Vidal, Structured low-rank matrix factorization: Global optimality, algorithms, and applications. *IEEE transactions on pattern analysis machine intelligence* **42**, 1468–1482 (2019).

58. K He, X Zhang, S Ren, J Sun, Deep residual learning for image recognition in *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 770–778 (2016).

59. H Xiao, K Rasul, R Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv:1708.07747 (15 Sep 2017).

60. S Ioffe, C Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift in *International Conference on Machine Learning*. pp. 448–456 (2015).

61. M Buda, A Maki, MA Mazurowski, A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks* **106**, 249–259 (2018).

62. J Shu, et al., Meta-weight-net: Learning an explicit mapping for sample weighting in *Advances in Neural Information Processing Systems*, eds. H Wallach, et al. (Curran Associates, Inc.), Vol. 32, pp. 1919–1930 (2019).

63. Y Cui, M Jia, TY Lin, Y Song, S Belongie, Class-balanced loss based on effective number of samples in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 9268–9277 (2019).

64. K Cao, C Wei, A Gaidon, N Arechiga, T Ma, Learning imbalanced datasets with label-distribution-aware margin loss in *Advances in Neural Information Processing Systems*. Vol. 32, pp. 1567–1578 (2019).

65. J Buolamwini, T Gebru, Gender shades: Intersectional accuracy disparities in commercial gender classification in *Conference on fairness, accountability and transparency*. pp. 77–91 (2018).

66. J Zou, L Schiebinger, AI can be sexist and racist—it's time to make it fair (2018).

67. N Mehrabi, F Morstatter, N Saxena, K Lerman, A Galstyan, A survey on bias and fairness in machine learning. arXiv:1908.09635 (17 Sep 2019).

68. L Bottou, FE Curtis, J Nocedal, Optimization methods for large-scale machine learning. *Siam Rev.* **60**, 223–311 (2018).

69. C Fang, CJ Li, Z Lin, T Zhang, Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator in *Advances in Neural Information Processing Systems*. pp. 689–699 (2018).

70. C Fang, Z Lin, T Zhang, Sharp analysis for nonconvex SGD escaping from saddle points in *Annual Conference on Learning Theory*. pp. 1192–1234 (2019).

For simplicity, we define $[m_1 : m_2] := \{m_1, m_1 + 1, \ldots, m_2\}$ for $m_1, m_2 \in \mathbb{N}$ with $m_1 \le m_2$ and $[m_2] := [1 : m_2]$ for $m_2 \ge 1$.

A. Balanced Case.

A.1. Proofs of Theorem 1 and Proposition 2 . Because there are multiplications of variables in the objective function, Eq. (7) is nonconvex. Thus the KKT condition is not sufficient for optimality. To prove Theorem 1, we directly determine the global minimum of Eq. (7). During this procedure, one key step is to show that minimizing Eq. (7) is equivalent to minimize a symmetric quadratic function:

$$\sum_{i=1}^{n} \left[\left(\sum_{k=1}^{K} \boldsymbol{h}_{k,i} \right)^{\top} \left(\sum_{k=1}^{K} \boldsymbol{w}_k \right) - K \sum_{k=1}^{K} \boldsymbol{h}_{k,i}^{\top} \boldsymbol{w}_k \right]$$

under suitable conditions. The detail is shown below.

Proof of Theorem 1. By the concavity of $\log(\cdot)$, for any $\boldsymbol{z} \in \mathbb{R}^K$, $k \in [K]$, constants $C_a, C_b > 0$, letting $C_c = \frac{C_b}{(C_a + C_b)(K - 1)}$, we have

$$-\log\left(\frac{\boldsymbol{z}(k)}{\sum_{k'=1}^{K} \boldsymbol{z}(k')} \right) = -\log(\boldsymbol{z}(k)) + \log\left(\sum_{k'=1}^{K} \boldsymbol{z}(k') \right)$$

$$= -\log(\boldsymbol{z}(k)) + \log\left(\frac{C_a}{C_a + C_b} \left(\frac{(C_a + C_b)\, \boldsymbol{z}(k)}{C_a} \right) + C_c \sum_{k'=1, k' \ne k}^{K} \frac{\boldsymbol{z}(k')}{C_c} \right). \tag{21}$$

Recognizing the equality

$$\frac{C_a}{C_a + C_b} + \underbrace{C_c + \cdots + C_c}_{K-1} = \frac{C_a}{C_a + C_b} + (K - 1)\frac{C_b}{(C_a + C_b)(K - 1)} = 1$$

and the concavity of $\log(\cdot)$, we see that the Jensen inequality gives

$$\log\left(\frac{C_a}{C_a + C_b} \left(\frac{(C_a + C_b)\, \boldsymbol{z}(k)}{C_a} \right) + C_c \sum_{k'=1, k' \ne k}^{K} \frac{\boldsymbol{z}(k')}{C_c} \right) \ge \frac{C_a}{C_a + C_b} \log\left(\frac{(C_a + C_b)\, \boldsymbol{z}(k)}{C_a} \right) + C_c \sum_{k'=1, k' \ne k}^{K} \log\left(\frac{\boldsymbol{z}(k')}{C_c} \right). \tag{22}$$

Plugging this inequality into Eq. (21), we get

$$-\log\left(\frac{\boldsymbol{z}(k)}{\sum_{k'=1}^{K} \boldsymbol{z}(k')} \right) \ge -\log(\boldsymbol{z}(k)) + \frac{C_a}{C_a + C_b} \log\left(\frac{(C_a + C_b)\, \boldsymbol{z}(k)}{C_a} \right) + C_c \sum_{k'=1, k' \ne k}^{K} \log\left(\frac{\boldsymbol{z}(k')}{C_c} \right)$$

$$= -\frac{C_b}{C_a + C_b} \left[\log(\boldsymbol{z}(k)) - \frac{1}{K - 1} \sum_{k'=1, k' \ne k}^{K} \log(\boldsymbol{z}(k')) \right] + C_d,$$

where the constant $C_d := \frac{C_a}{C_a + C_b} \log(\frac{C_a + C_b}{C_a}) + \frac{C_b}{C_a + C_b} \log(1/C_c)$. Note that in Eq. (21), C_a and C_b can be any positive numbers. To prove Theorem 1, we set $C_a := \exp\left(\sqrt{E_H E_W} \right)$ and $C_b := \exp\left(-\sqrt{E_H E_W}/(K - 1) \right)$, which shall lead to the tightest lower bound for the objective of Eq. (7). Applying Eq. (21) to the objective, we have

$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \tag{23}$$

$$\ge \frac{C_b}{(C_a + C_b)N(K - 1)} \sum_{i=1}^{n} \left[\left(\sum_{k=1}^{K} \boldsymbol{h}_{k,i} \right)^{\top} \left(\sum_{k=1}^{K} \boldsymbol{w}_k \right) - K \sum_{k=1}^{K} \boldsymbol{h}_{k,i}^{\top} \boldsymbol{w}_k \right] + C_d.$$

Defining $\bar{\boldsymbol{h}}_i := \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{h}_{k,i}$ for $i \in [n]$, it follows from the simple inequality $2ab \le a^2 + b^2$ that

$$\sum_{i=1}^{n} \left[\left(\sum_{k=1}^{K} \boldsymbol{h}_{k,i} \right)^{\top} \left(\sum_{k=1}^{K} \boldsymbol{w}_k \right) - K \sum_{k=1}^{K} \boldsymbol{h}_{k,i}^{\top} \boldsymbol{w}_k \right]$$

$$= K \sum_{i=1}^{n} \sum_{k=1}^{K} (\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i})^{\top} \boldsymbol{w}_k$$

$$\ge -\frac{K}{2} \sum_{k=1}^{K} \sum_{i=1}^{n} \|\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i}\|^2 / C_e - \frac{C_e N}{2} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2, \tag{24}$$

where we pick $C_e := \sqrt{E_H/E_W}$. The two terms in the right hand side of Eq. (24) can be bounded via the constraints of Eq. (7). Specifically, we have

$$\frac{C_e N}{2} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 \le \frac{K N \sqrt{E_H E_W}}{2}, \tag{25}$$

and

$$\frac{K}{2} \sum_{k=1}^{K} \sum_{i=1}^{n} \|\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i}\|^2 / C_e \overset{a}{=} \frac{K^2}{2C_e} \sum_{i=1}^{n} \left(\frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{h}_{k,i}\|^2 - \|\bar{\boldsymbol{h}}_i\|^2 \right)$$

$$\leq \frac{K}{2C_e} \sum_{k=1}^{K} \sum_{i=1}^{n} \|\boldsymbol{h}_{k,i}\|^2 \leq \frac{KN\sqrt{E_H E_W}}{2}, \qquad [26]$$

where $\overset{a}{=}$ uses the fact that $\mathbb{E}\|\boldsymbol{a} - \mathbb{E}[\boldsymbol{a}]\|^2 = \mathbb{E}\|\boldsymbol{a}\|^2 - \|\mathbb{E}[\boldsymbol{a}]\|^2$. Thus plugging Eq. (24), Eq. (25), and Eq. (26) into Eq. (23), we have

$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq -\frac{C_b}{C_a + C_b} \frac{K\sqrt{E_H E_W}}{K-1} + C_d := L_0. \qquad [27]$$

Now we check the conditions that reduce Eq. (27) to an equality.

By the strict concavity of $\log(\cdot)$, Eq. (22) reduces to an equality only if

$$\frac{(C_a + C_b)\, \boldsymbol{z}(k)}{C_a} = \frac{\boldsymbol{z}(k')}{C_c}$$

for $k' \neq k$. Therefore, Eq. (23) reduces to an equality only if

$$\frac{(C_a + C_b)\boldsymbol{h}_{k,i}^{\top}\boldsymbol{w}_k}{C_a} = \frac{\boldsymbol{h}_{k,i}^{\top}\boldsymbol{w}_{k'}}{C_c}.$$

Recognizing $C_c = \frac{C_b}{(C_a + C_b)(K-1)}$ and taking the logarithm of both sides of the above equation, we obtain

$$\boldsymbol{h}_{k,i}\boldsymbol{w}_k = \boldsymbol{h}_{k,i}\boldsymbol{w}_{k'} + \log\left(\frac{C_a(K-1)}{C_b} \right),$$

for all $(k, i, k') \in \{(k, i, k') : k \in [K], k' \in [K], k' \neq k, i \in [n]\}$. Eq. (24) becomes equality if and only if

$$\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i} = -C_e\boldsymbol{w}_k, \quad k \in [K],\ i \in [n].$$

Eq. (25) and Eq. (26) become equalities if and only if:

$$\frac{1}{K} \sum_{k=1}^{K} \frac{1}{n} \sum_{i=1}^{n} \left\| \boldsymbol{h}_{k,i} \right\|^2 = E_H, \quad \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 = E_W, \quad \bar{\boldsymbol{h}}_i = \boldsymbol{0}_p,\ i \in [n].$$

Applying Lemma 2 shown in the end of the section, we have $(\boldsymbol{H}, \boldsymbol{W})$ satisfies Eq. (9).

Reversely, it is easy to verify that Eq. (27) reduces to equality when $(\boldsymbol{H}, \boldsymbol{W})$ admits Eq. (9). So L_0 is the global minimum of Eq. (7) and Eq. (9) is the unique form for the minimizers. We complete the proof of Theorem 1. $\qquad \square$

Lemma 2. *Suppose* $(\boldsymbol{H}, \boldsymbol{W})$ *satisfies*

$$\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i} = -\sqrt{\frac{E_H}{E_W}}\boldsymbol{w}_k, \quad k \in [K], \quad i \in [n], \qquad [28]$$

and

$$\frac{1}{K} \sum_{k=1}^{K} \frac{1}{n} \sum_{i=1}^{n} \left\| \boldsymbol{h}_{k,i} \right\|^2 = E_H, \quad \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 = E_W, \quad \bar{\boldsymbol{h}}_i = \boldsymbol{0}_p,\ i \in [n], \qquad [29]$$

where $\bar{\boldsymbol{h}}_i := \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{h}_{k,i}$ *with* $i \in [n]$. *Moreover, there exists a constant* C *such that for all* $(k, i, k') \in \{(k, i, k') : k \in [K], k' \in [K], k' \neq k, i \in [n]\}$, *we have*

$$\boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_k = \boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_{k'} + C. \qquad [30]$$

Then $(\boldsymbol{H}, \boldsymbol{W})$ *satisfies Eq. (9).*

Proof. Combining Eq. (28) with the last equality in Eq. (29), we have

$$\boldsymbol{W} = \sqrt{\frac{E_W}{E_H}} \left[\boldsymbol{h}_1, \ldots, \boldsymbol{h}_K \right]^{\top}, \qquad \boldsymbol{h}_{k,i} = \boldsymbol{h}_k,\ k \in [K],\ i \in [n].$$

Thus it remains to show

$$\boldsymbol{W} = \sqrt{E_W}\ (\boldsymbol{M}^{\star})^{\top}, \qquad [31]$$

where $\boldsymbol{M}^{\star}$ is a K-simplex ETF.

Plugging $\boldsymbol{h}_k = \boldsymbol{h}_{k,i} = \sqrt{\frac{E_W}{E_H}}\boldsymbol{w}_k$ into Eq. (30), we have, for all $(k, k') \in \{(k, k') : k \in [K], k' \in [K], k' \neq k\}$,

$$\sqrt{\frac{E_H}{E_W}}\|\boldsymbol{w}_k\|^2 = \boldsymbol{h}_k \cdot \boldsymbol{w}_k = \boldsymbol{h}_k \cdot \boldsymbol{w}_{k'} + C = \sqrt{\frac{E_H}{E_W}}\boldsymbol{w}_k \cdot \boldsymbol{w}_{k'} + C,$$

and

$$\sqrt{\frac{E_H}{E_W}}\|\boldsymbol{w}_{k'}\|^2 = \boldsymbol{h}_{k'} \cdot \boldsymbol{w}_{k'} = \boldsymbol{h}_{k'} \cdot \boldsymbol{w}_k + C = \sqrt{\frac{E_H}{E_W}}\boldsymbol{w}_{k'} \cdot \boldsymbol{w}_k + C.$$

Therefore, from $\frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 = E_W$, we have $\|\boldsymbol{w}_k\| = \sqrt{E_W}$ and $\boldsymbol{h}_k\boldsymbol{w}_{k'} = C' := \sqrt{E_H E_W} - C$.

Furthermore, recalling that $\bar{\boldsymbol{h}}_i = \boldsymbol{0}_p$ for $i \in [n]$, we have $\sum_{k=1}^{K} \boldsymbol{h}_k = \boldsymbol{0}_p$, which further yields $\sum_{k=1}^{K} \boldsymbol{h}_k \cdot \boldsymbol{w}_{k'} = 0$ for $k' \in [K]$. Then it follows from $\boldsymbol{h}_k\boldsymbol{w}_{k'} = C'$ and $\boldsymbol{h}_k\boldsymbol{w}_k = \sqrt{E_H E_W}$ that $\boldsymbol{h}_k\boldsymbol{w}_{k'} = -\sqrt{E_H E_W}/(K-1)$. Thus we obtain

$$\boldsymbol{WW}^{\top} = \sqrt{\frac{E_W}{E_H}}\boldsymbol{W}[\boldsymbol{h}_1,\dots,\boldsymbol{h}_K] = E_W \left[\frac{K}{K-1}\left(\boldsymbol{I}_K - \frac{1}{K}\boldsymbol{1}_K\boldsymbol{1}_K^{\top}\right)\right],$$

which implies Eq. (31). We complete the proof. $\qquad\square$

Proof of Proposition 2. We introduce the set $\mathcal{S}_R$ as

$$\mathcal{S}_R := \left\{ (\boldsymbol{H},\boldsymbol{W}) : \begin{array}{c} [\boldsymbol{h}_1,\dots,\boldsymbol{h}_K] = B_1 b \boldsymbol{P}\left[(a+1)\boldsymbol{I}_K - \boldsymbol{1}_K\boldsymbol{1}_K^{\top}\right], \\ \boldsymbol{W} = B_2 B_3 b \left[(a+1)\boldsymbol{I}_K - \boldsymbol{1}_K\boldsymbol{1}_K^{\top}\right]^{\top}\boldsymbol{P}^{\top}, \\ \boldsymbol{h}_{k,i} = \boldsymbol{h}_k, \quad k \in [K],\ i \in [n], \\ b \geq 0,\ a \geq 0,\ b^q[a^q + (K-1)] = 1, \\ |B_1| \leq \sqrt{E_H},\ |B_2| \leq \sqrt{E_W},\ B_3 \geq 0,\ B_3^2 b^2[a^2 + (K-1)] = 1, \\ \boldsymbol{P} \in \mathbb{R}^{p \times K},\ \boldsymbol{P}^{\top}\boldsymbol{P} = \boldsymbol{I}_K. \end{array} \right\}$$

We can examine that $\mathcal{S}_R$ admits the constraints of Eq. (7). So any $(\boldsymbol{H},\boldsymbol{W}) \in \mathcal{S}_R$ is a feasible solution. Moreover, one can observe that this feasible solution has a special symmetry structure: for each $k \in [K]$, the features in class k collapse to their mean $\boldsymbol{h}_k$, i.e., (NC1), and $\boldsymbol{w}_k$ is parallel to $\boldsymbol{h}_k$, i.e., (NC3). However, weights do not form the vertices of ETF unless $a = K-1$. Therefore, it suffices to show that the minimizer of $\frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n}\mathcal{L}(\boldsymbol{Wh}_{k,i},\boldsymbol{y}_k)$ in the set $\mathcal{S}_R$ do not satisfy $a = K-1$.

In fact, for any $(\boldsymbol{H},\boldsymbol{W}) \in \mathcal{S}_R$, the objective function value can be written as a function of B_1, B_2, B_3, a, and b. We have

$$\frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n}\mathcal{L}(\boldsymbol{Wh}_{k,i},\boldsymbol{y}_k)$$

$$= -\log\left(\frac{\exp(B_1 B_2 B_3 b^2[a^2 + (K-1)])}{\exp(B_1 B_2 B_3 b^2[a^2 + K - 1]) + (K-1)\exp(B_1 B_2 B_3 b^2[K - 2 - 2a])}\right)$$

$$= -\log\left(\frac{1}{1 + (K-1)\exp(-B_1 B_2 B_3 b^2(a+1)^2)}\right).$$

Then it follows to maximize $B_1 B_2 B_3 b^2(a+1)^2$ or equivalently $\left[B_1 B_2 B_3 b^2(a+1)^2\right]^2$. By $B_3^2 b^2[a^2 + (K-1)] = 1$ and $b^q[a^q + (K-1)] = 1$, we have

$$\left[B_1 B_2 B_3 b^2(a+1)^2\right]^2 \overset{a}{\leq} E_H E_W \left[B_3^2 b^2(a+1)^2\right]\left[b^2(a+1)^2\right]$$

$$= E_H E_W \left[\frac{(a+1)^2}{a^2 + (K-1)}\right]\left[\frac{(a+1)^q}{a^q + K - 1}\right]^{2/q}, \qquad [32]$$

where $\overset{a}{\leq}$ picks $B_1 = \sqrt{E_H}$ and $B_2 = \sqrt{E_W}$. Let us consider function $g : [0, +\infty) \to \mathbb{R} : g(x) = \left[\frac{(x+1)^2}{x^2 + (K-1)}\right]\left[\frac{(x+1)^q}{x^q + K - 1}\right]^{2/q}$. Note that by the first-order optimality, once if $g'(K-1) \neq 0$, then Eq. (32) cannot achieve the maximum at $a = K-1$, which is our desired result. Indeed, we have

$$g'(K-1) = \frac{2K^4}{[(K-1)^2 + (K-1)]\left[(K-1)^q + K - 1\right]^{2/q+1}}\left[(K-1) - (K-1)^{q-1}\right].$$

Therefore, $a = K-1 \geq 2$ is not the maximizer of Eq. (32), unless $q = 2$. We complete the proof. $\qquad\square$

Following the proof of Proposition 2, for completeness we discuss the structure of the global minimizers of Program (20) in the case $K = 2$. In short, we show that when $q \in (1,2) \cup (2,\infty)$, although the global minimizers of Eq. (20) remain in the form of Eq. (9), they are no longer rotationally invariant due to certain constraints on the solutions. This is in contrast to a K-simplex ETF, which is rotationally invariant (see Definition 1).

For simplicity of notation, we assume that there is one training example in each class (the case of multiple training examples can be directly extended). Program (20) in the case $K = 2$ takes the following form:

$$\begin{aligned} \min_{\boldsymbol{W},\boldsymbol{H}} \quad & -\log\left(\frac{\exp(\boldsymbol{w}_1^{\top}\boldsymbol{h}_1)}{\exp(\boldsymbol{w}_1^{\top}\boldsymbol{h}_1) + \exp(\boldsymbol{w}_2^{\top}\boldsymbol{h}_1)}\right) - \log\left(\frac{\exp(\boldsymbol{w}_2^{\top}\boldsymbol{h}_2)}{\exp(\boldsymbol{w}_1^{\top}\boldsymbol{h}_2) + \exp(\boldsymbol{w}_2^{\top}\boldsymbol{h}_2)}\right) \\ \text{s.t.} \quad & \|\boldsymbol{w}_1\|^2 + \|\boldsymbol{w}_2\|^2 \leq 2E_W, \\ & \|\boldsymbol{h}_1\|_q^q + \|\boldsymbol{h}_2\|_q^q \leq 2E_H. \end{aligned} \qquad [33]$$

We show that the optimal solution to (33) satisfies some specific ETF structures. In brief, when $q > 2$, both the features and weights are parallel to a certain vector, and when $1 < q < 2$, the solution is sparse in the sense that only one entry is nonzero for both the features and the weights.

Lemma 3. *For $q > 2$, any global minimizer of Eq. (33) satisfies*

$$\boldsymbol{h}_1^{\star} = -\boldsymbol{h}_2^{\star} = C_1\boldsymbol{w}_1^{\star} = -C_1\boldsymbol{w}_2^{\star} = C_2(\pm\boldsymbol{1}_p), \qquad [34]$$

where the constants $C_1 = \left(\frac{E_H}{p}\right)^{1/q}\left(\frac{E_W}{p}\right)^{-1/2}, C_2 = \left(\frac{E_H}{p}\right)^{1/q}$, and $\pm\boldsymbol{1}_p$ denotes a p-dimensional vector such that each entry is either 1 or -1 (there are in total 2^p such vectors). For $1 < q < 2$, any global minimizer of Eq. (33) satisfies

$$\boldsymbol{h}_1^{\star} = -\boldsymbol{h}_2^{\star} = C_3\boldsymbol{w}_1^{\star} = -C_3\boldsymbol{w}_2^{\star}, \quad \|\boldsymbol{h}_1^{\star}\|_0 = 1, \quad \|\boldsymbol{h}_1^{\star}\| = C_4, \qquad [35]$$

where the constants $C_3 = E_H^{1/q}E_W^{-1/2}$ and $C_4 = E_H^{1/q}$.

Proof. For any constants $C_a, C_b > 0$, letting $C_c = \frac{C_b}{C_a + C_b}$, using the same arguments as Eq. (21) and Eq. (23), we have

$$-\log\left(\frac{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_1)}{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_1) + \exp(\boldsymbol{w}_2^\top \boldsymbol{h}_1)}\right) - \log\left(\frac{\exp(\boldsymbol{w}_2^\top \boldsymbol{h}_2)}{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_2) + \exp(\boldsymbol{w}_2^\top \boldsymbol{h}_2)}\right)$$

$$\geq \frac{C_b}{C_a + C_b}\left[(\boldsymbol{h}_1 + \boldsymbol{h}_2)^\top (\boldsymbol{w}_1 + \boldsymbol{w}_2) - 2(\boldsymbol{h}_1^\top \boldsymbol{w}_1 + \boldsymbol{h}_2^\top \boldsymbol{w}_2)\right] + C_d. \tag{36}$$

Then it follows that

$$(\boldsymbol{h}_1 + \boldsymbol{h}_2)^\top (\boldsymbol{w}_1 + \boldsymbol{w}_2) - 2(\boldsymbol{h}_1^\top \boldsymbol{w}_1 + \boldsymbol{h}_2^\top \boldsymbol{w}_2) = -(\boldsymbol{h}_1 - \boldsymbol{h}_2)^\top (\boldsymbol{w}_1 - \boldsymbol{w}_2) \geq -\|\boldsymbol{h}_1 - \boldsymbol{h}_2\| \|\boldsymbol{w}_1 - \boldsymbol{w}_2\|. \tag{37}$$

We have

$$\|\boldsymbol{w}_1 - \boldsymbol{w}_2\|^2 = \|\boldsymbol{w}_1\|^2 + \|\boldsymbol{w}_2\|^2 - 2\boldsymbol{w}_1^\top \boldsymbol{w}_2 \leq 2\|\boldsymbol{w}_1\|^2 + 2\|\boldsymbol{w}_2\|^2 \leq 4E_W \tag{38}$$

and

$$\|\boldsymbol{h}_1 - \boldsymbol{h}_2\|^2 = \sum_{i=1}^p |\boldsymbol{h}_1(i) - \boldsymbol{h}_2(i)|^2 \leq \sum_{i=1}^p (|\boldsymbol{h}_1(i)| + |\boldsymbol{h}_2(i)|)^2 \overset{a}{\leq} 2^{2-2/q}\left[\sum_{i=1}^p (|\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q)^{\frac{2}{q}}\right], \tag{39}$$

where $\boldsymbol{h}_1(i)$ and $\boldsymbol{h}_2(i)$ denotes the i-th entry of $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$, respectively. In $\overset{a}{\leq}$, we use Jensen's inequality that $\left(\frac{|\boldsymbol{h}_1(i)| + |\boldsymbol{h}_2(i)|}{2}\right)^q \leq \frac{|\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q}{2}$ since $|x|^q$ is strictly convex.

- When $1 < q < 2$, we pick $C_a = \exp\left(E_H^{1/q} E_W^{1/2}\right)$ and $C_b = 1/C_a$. We have

$$\sum_{i=1}^p (|\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q)^{\frac{2}{q}} \leq \left(\sum_{i=1}^p |\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q\right)^{2/q} \leq 2^{2/q} E_H^{2/q}, \tag{40}$$

where the first inequality uses that $\sum_{i=1}^p |x_i|^o \leq \left(\sum_{i=1}^p |x_i|\right)^o$ for $o > 1$ and the equality holds if and only if the non-zero elements of $\{x_i\}_{i=1}^p$ is at most 1. Then by plugging Eq. (38), Eq. (39), and Eq. (40) into Eq. (36), using Eq. (37), we have

$$-\log\left(\frac{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_1)}{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_1) + \exp(\boldsymbol{w}_2^\top \boldsymbol{h}_1)}\right) - \log\left(\frac{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_2)}{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_2) + \exp(\boldsymbol{w}_2^\top \boldsymbol{h}_2)}\right) \geq -\frac{C_b}{C_a + C_b}\sqrt{2^{4-4/q} E_W E_H^{2/q}} + C_d. \tag{41}$$

Now we check the conditions that reduce Eq. (41) to an equality. Eq. (37) reduces to an equality if and only if there exists a constant $C_5 > 0$ such that $\boldsymbol{h}_1 - \boldsymbol{h}_2 = C_5(\boldsymbol{w}_1 - \boldsymbol{w}_2)$. Eq. (38) reduces to an equality if and only if $\boldsymbol{w}_1 = -\boldsymbol{w}_2$ and $\|\boldsymbol{w}_1\|^2 = E_W$. Eq. (39) reduces to an equality if and only if $\boldsymbol{h}_1 = -\boldsymbol{h}_2$. Finally, Eq. (40) reduces to an equality if and only if there is only one non-zero entry i such that we exactly have $|\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q = 2E_H$. We can obtain Eq. (34).

When $q > 2$, we pick $C_a = \exp\left(p\left(\frac{E_H}{p}\right)^{1/q}\left(\frac{E_W}{p}\right)^{1/2}\right)$ and $C_b = 1/C_a$. We have

$$\sum_{i=1}^p (|\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q)^{\frac{2}{q}} \leq p^{1-2/q}\left(\sum_{i=1}^p |\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q\right)^{2/q} \leq p^{1-2/q} 2^{2/q} E_H^{2/q}, \tag{42}$$

where the first inequality uses Jensen's inequality that $\left(\frac{1}{p}\sum_{i=1}^p |x_i|\right)^a \leq \frac{1}{p}\sum_{i=1}^p |x_i|^a$ for $a > 1$ since $|x|^a$ is strictly convex with respect to x, and let $a = q/2$, and $x_i = (|\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q)^{2/q} \geq 0$. Then by plugging Eq. (38), Eq. (39), and Eq. (42) into Eq. (36), using Eq. (37), we have

$$-\log\left(\frac{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_1)}{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_1) + \exp(\boldsymbol{w}_2^\top \boldsymbol{h}_1)}\right) - \log\left(\frac{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_2)}{\exp(\boldsymbol{w}_1^\top \boldsymbol{h}_2) + \exp(\boldsymbol{w}_2^\top \boldsymbol{h}_2)}\right) \geq -\frac{C_b}{C_a + C_b}\sqrt{2^{4-4/q} p^{1-2/q} E_W E_H^{2/q}} + C_d. \tag{43}$$

Now we check the conditions that reduce Eq. (41) to an equality. In fact, by the strict convexity, Eq. (42) reduces to an equality if and only if $|\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q = |\boldsymbol{h}_1(j)|^q + |\boldsymbol{h}_2(j)|^q$ for all $i \neq j$ and $\sum_{i=1}^p |\boldsymbol{h}_1(i)|^q + |\boldsymbol{h}_2(i)|^q = 2E_H$. Then by combining the conditions to reduce Eq. (37), Eq. (38), and Eq. (39) to equalities, we can obtain Eq. (34). $\qquad\square$

Figure 7 displays simulation results concerning the last-layer weights for binary classification using deep neural networks. The results show that last-layer weights exhibit neither the all-ones nor the sparse pattern as in Lemma 3, thereby implying that the ℓ_2 norm is the best choice among all ℓ_q norms for modeling deep neural networks using the Layer-Peeled Model.

In the case where $q \leq 1$, we conjecture that the ℓ_q norm regularizer would also render the solution to Eq. (33) sparse. We leave this for future work.

A.2. Proofs of Theorems 3 and 4. The proofs of Theorems 3 and 4 follow from the similar argument of Theorem 1.

Proof of Theorem 3. For $k \in [K]$, $i \in [n]$, and $k' \in [K]$, define

$$E_{k,i,k'} := \frac{1}{n}\sum_{j=1}^n \exp(\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k',j}/\tau).$$
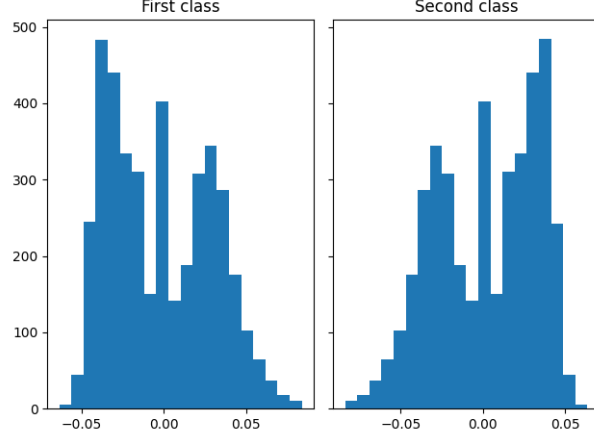
Fig. 7. Histograms of last-layer weights of VGG11 trained on the first two classes in FashionMNIST. Each histogram shows the empirical distribution of all the entries of $\boldsymbol{w}_1$ or $\boldsymbol{w}_2$. If the prediction of Lemma 3 applied to real neural networks for binary classification, then we would observe a mixture of one or two point masses in the histograms, which however is not the case. There are 6000 examples in each class, and we use the same experimental settings as in Section C. The training and test accuracies are 100% and 99.75%, respectively.

For constants $C_a := \exp\left(\sqrt{E_H E_W}\right)$ and $C_b := (K-1)\exp\left(-\sqrt{E_H E_W}\right)$, let $C_c := \frac{C_b}{(C_a + C_b)(K-1)}$. Using a similar argument as Eq. (21), we have for $j \in [n]$,

$$
-\log\left(\frac{\exp(\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^{K} E_{k,i,k'}}\right) \tag{44}
$$

$$
= -\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j}/\tau + \log\left(\frac{C_a}{C_a + C_b}\left(\frac{(C_a + C_b)\, E_{k,i,k}}{C_a}\right) + C_c \sum_{k'=1,\ k'\neq k}^{K} \frac{E_{k,i,k'}}{C_c}\right)
$$

$$
\overset{a}{\geq} -\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j}/\tau + \frac{C_a}{C_a + C_b} \log\left(\frac{(C_a + C_b)\, E_{k,i,k}}{C_a}\right) + C_c \sum_{k'=1,\ k'\neq k}^{K} \log\left(\frac{E_{k,i,k'}}{C_c}\right)
$$

$$
\overset{b}{=} -\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j}/\tau + \frac{C_a}{C_a + C_b} \log\left(E_{k,i,k}\right) + C_c \sum_{k'=1,\ k'\neq k}^{K} \log\left(E_{k,i,k'}\right) + C_d
$$

$$
\overset{c}{\geq} -\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j}/\tau + \frac{C_a}{(C_a + C_b)n} \sum_{\ell=1}^{n} \boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,\ell}/\tau + \frac{C_c}{n} \sum_{k'=1,\ k'\neq k}^{K} \sum_{\ell=1}^{n} \boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k',\ell}/\tau + C_d.
$$

where $\overset{a}{\geq}$ and $\overset{c}{\geq}$ apply the concavity of $\log(\cdot)$ and in $\overset{b}{=}$ we define $C_d := \frac{C_a}{C_a + C_b} \log(\frac{C_a + C_b}{C_a}) + \frac{C_b}{C_a + C_b} \log(1/C_c)$. Then plugging Eq. (44) into the objective function, we have

$$
\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n} \frac{1}{n} \sum_{j=1}^{n} -\log\left(\frac{\exp(\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^{K} \sum_{\ell=1}^{n} \exp(\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k',\ell})}\right) \tag{45}
$$

$$
= \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n} \frac{1}{n} \sum_{j=1}^{n} -\log\left(\frac{\exp(\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^{K} E_{k,i,k'}}\right) + \log(n)
$$

$$
\overset{\text{Eq. (44)}}{\geq} \frac{C_b K}{(C_a + C_b)N(K-1)\tau} \sum_{k=1}^{K} \sum_{i=1}^{n} \left(-\frac{1}{n} \sum_{j=1}^{n}\left(\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j} - \frac{1}{K} \sum_{k'=1}^{K} \boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k',j}\right)\right) + C_d + \log(n).
$$

Now defining $\bar{\boldsymbol{h}}_i := \frac{1}{K}\sum_{k=1}^K \boldsymbol{h}_{k,i}$ for $i \in [n]$, a similar argument as Eq. (24) and Eq. (26) gives that

$$\sum_{k=1}^K \sum_{i=1}^n \left(-\frac{1}{n}\sum_{j=1}^n \left(\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j} - \frac{1}{K}\sum_{k'=1}^K \boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k',j} \right) \right)$$

$$= \sum_{k=1}^K \sum_{i=1}^n \left(-\frac{1}{n}\sum_{j=1}^n \boldsymbol{h}_{k,i} \cdot (\boldsymbol{h}_{k,j} - \bar{\boldsymbol{h}}_j) \right)$$

$$\overset{a}{\geq} -\frac{1}{2}\sum_{k=1}^K \sum_{i=1}^n \left\| \boldsymbol{h}_{k,i} \right\|^2 - \frac{1}{2}\sum_{k=1}^K \sum_{i=1}^n \left\| \boldsymbol{h}_{k,i} - \bar{\boldsymbol{h}}_i \right\|^2$$

$$\overset{b}{\geq} -\frac{1}{2}\sum_{k=1}^K \sum_{i=1}^n \left\| \boldsymbol{h}_{k,i} \right\|^2 - \frac{K}{2}\sum_{i=1}^n \left(\frac{1}{K}\sum_{k=1}^K \left\| \boldsymbol{h}_{k,i} \right\|^2 - \left\| \bar{\boldsymbol{h}}_i \right\|^2 \right)$$

$$\geq -\sum_{k=1}^K \sum_{i=1}^n \left\| \boldsymbol{h}_{k,i} \right\|^2 \overset{c}{\geq} -NE_H, \tag{46}$$

where $\overset{a}{\geq}$ follows from $2ab \leq a^2 + b^2$, $\overset{b}{\geq}$ follows from $\mathbb{E}\|\boldsymbol{a} - \mathbb{E}[\boldsymbol{a}]\|^2 = \mathbb{E}\|\boldsymbol{a}\|^2 - \|\mathbb{E}[\boldsymbol{a}]\|^2$, and $\overset{c}{\geq}$ uses the constraint of Eq. (11). Therefore, plugging Eq. (46) into Eq. (45) yields that

$$\frac{1}{N}\sum_{k=1}^K \sum_{i=1}^n \frac{1}{n}\sum_{j=1}^n -\log\left(\frac{\exp(\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^K \sum_{\ell=1}^n \exp(\boldsymbol{h}_{k,i} \cdot \boldsymbol{h}_{k',\ell}/\tau)} \right)$$

$$\geq -\frac{C_b K E_H}{(C_a + C_b)(K-1)\tau} + C_d + \log(n). \tag{47}$$

Now we check the conditions that can make Eq. (47) reduce equality. By the strictly concavity of $\log(\cdot)$, Eq. (44) reduce to equalities only if for all $(k, i, k') \in \{(k, i, k') : k \in [K], k' \in [K], k' \neq k, i \in [n]\}$,

$$\frac{E_{k,i,k}}{C_a(K-1)} = \frac{E_{k,i,k'}}{C_b}. \tag{48}$$

Eq. (46) reduce to equalities if and only if:

$$\boldsymbol{h}_{k,i} = \boldsymbol{h}_k, \ i \in [n], \ k \in [K], \quad \frac{1}{K}\sum_{k=1}^K \|\boldsymbol{h}_k\|^2 = E_H, \quad \sum_{k=1}^K \boldsymbol{h}_k = \boldsymbol{0}_p. \tag{49}$$

Plugging $\boldsymbol{h}_{k,i} = \boldsymbol{h}_k$ into Eq. (48), we have for $(k, k') \in \{k, k' : k \in [K], k' \in [K], k' \neq k\}$,

$$\frac{\exp(\|\boldsymbol{h}_k\|^2)}{C_a(K-1)} = \frac{\exp(\boldsymbol{h}_k \cdot \boldsymbol{h}_{k'})}{C_b} = \frac{\exp(\|\boldsymbol{h}_{k'}\|^2)}{C_a(K-1)}.$$

Then it follows from $\frac{1}{K}\sum_{k=1}^K \|\boldsymbol{h}_k\|^2 = E_H$ that $\|\boldsymbol{h}_k\|^2 = E_H$ for $k \in [K]$. Moreover, since $\sum_{k=1}^K \boldsymbol{h}_k = \boldsymbol{0}_p$, we obtain

$$\boldsymbol{h}_k \cdot \boldsymbol{h}_{k'} = -\frac{E_H}{K-1}$$

for $(k, k') \in \{k, k' : k \in [K], k' \in [K], k' \neq k\}$. Therefore,

$$[\boldsymbol{h}_1, \ldots, \boldsymbol{h}_K]^\top [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_K] = E_H\left[\frac{K}{K-1}\left(\boldsymbol{I}_K - \boldsymbol{1}_K \boldsymbol{1}_K^\top \right) \right],$$

which implies Eq. (13).

Reversely, it is easy to verify that the equality for Eq. (47) is reachable when $\boldsymbol{H}$ admits Eq. (13). We complete the proof of Theorem 3. □

Proof of Theorem 4. We first determine the minimum of Eq. (7). For the simplicity of our expressions, we introduce $\boldsymbol{z}_{k,i} := \boldsymbol{W}\boldsymbol{h}_{k,i}$ for $k \in [K]$ and $i \in [n]$. By the convexity of g_2, for any $k \in [K]$ and $i \in [n]$, we have

$$\sum_{k'=1, \ k'\neq k}^K g_2\left(\boldsymbol{S}(\boldsymbol{z}_{k,i})(k') \right) \geq (K-1)g_2\left(\frac{1}{K-1}\sum_{k'=1, \ k'\neq k}^K \boldsymbol{S}(\boldsymbol{z}_{k,i})(k') \right)$$

$$\overset{a}{=} (K-1)g_2\left(1 - \frac{1}{K-1}\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \right), \tag{50}$$

where $\overset{a}{=}$ uses $\sum_{k=1}^{K} \boldsymbol{S}(\boldsymbol{a})(k) = 1$ for any $\boldsymbol{a} \in \mathbb{R}^K$. Then it follows by the convexity of g_1 and g_2 that

$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \tag{51}$$

$$= \frac{1}{N} \sum_{i=1}^{n} \sum_{k=1}^{K} \left[g_1\left(\boldsymbol{S}(\boldsymbol{z}_{k,i})(k)\right) + \sum_{k'=1,\ k'\neq k}^{K} g_2\left(\boldsymbol{S}(\boldsymbol{z}_{k',i})(k')\right) \right]$$

$$\overset{\text{Eq. (50)}}{\geq} \frac{1}{N} \sum_{i=1}^{n} \sum_{k=1}^{K} \left[g_1\left(\boldsymbol{S}(\boldsymbol{z}_{k,i})(k)\right) + (K-1)g_2\left(1 - \frac{1}{K-1}\boldsymbol{S}(\boldsymbol{z}_{k,i})(k)\right) \right]$$

$$\geq g_1\left(\frac{1}{N} \sum_{i=1}^{n} \sum_{k=1}^{K} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \right) + (K-1)g_2\left(1 - \frac{1}{N(K-1)} \sum_{i=1}^{n} \sum_{k=1}^{K} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \right).$$

Because $g_1(x) + (K-1)g_2(1 - \frac{x}{K-1})$ is monotonously deceasing, it suffices to maximize

$$\frac{1}{N} \sum_{i=1}^{n} \sum_{k=1}^{K} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k).$$

To begin with, for any $\boldsymbol{z}_{k,i}$ with $k \in [K]$ and $i \in [n]$, by convexity of exponential function and the monotonicity of $q(x) = \frac{a}{a+x}$ for $x > 0$ if $a > 0$, we have

$$\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) = \frac{\exp(\boldsymbol{z}_{k,i}(k))}{\sum_{k'=1}^{K} \exp(\boldsymbol{z}_{k,i}(k'))}$$

$$\leq \frac{\exp(\boldsymbol{z}_{k,i}(k))}{\exp(\boldsymbol{z}_{k,i}(k)) + (K-1)\exp\left(\frac{1}{K-1} \sum_{k'=1,\ k'\neq k}^{K} \boldsymbol{z}_{k,i}(k') \right)}$$

$$= \frac{1}{1 + (K-1)\exp\left(\frac{1}{K-1} \sum_{k'=1,\ k'\neq k}^{K} \boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k) \right)}. \tag{52}$$

Consider function $g_0 : \mathbb{R} \to \mathbb{R}$ as $g_0(x) = \frac{1}{1+C\exp(x)}$ with $C := (K-1) \geq 1$. We have

$$g_0''(x) = -\frac{\exp(x)(1 + C\exp(x))(1 - C\exp(x))}{(1 + C\exp(x))^4}. \tag{53}$$

For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ of Eq. (7), we divide the index set $[n]$ into two subsets $\mathcal{S}_1$ and $\mathcal{S}_2$ defined below:

(i) $i \in \mathcal{S}_1$ if there exists at least one $k \in [K]$ such that

$$\frac{1}{K-1} \sum_{k'=1,\ k'\neq k}^{K} \boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k) \geq \log\left(\frac{1}{K-1} \right).$$

(ii) $i \in \mathcal{S}_2$ if for all $k \in [K]$, $\frac{1}{K-1} \sum_{k'=1,\ k'\neq k}^{K} \boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k) < \log\left(\frac{1}{K-1} \right)$.

Clearly, $\mathcal{S}_1 \cap \mathcal{S}_2 = \varnothing$. Let $|\mathcal{S}_1| = t$, then $|\mathcal{S}_2| = n - t$. Define function $L : [n] \to \mathbb{R}$ as

$$L(t) := \begin{cases} N - \left(\frac{1}{2}t + \frac{K(n-t)}{1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1) \right)} \right), & t \in [0 : n-1], \\ N - \frac{n}{2}, & t = n. \end{cases} \tag{54}$$

We show in Lemma 4 (see the end of the proof) that

$$\frac{1}{N} \sum_{i=1}^{n} \sum_{k=1}^{K} \boldsymbol{S}(\boldsymbol{z}_{k,i})(i) \leq \frac{1}{N}L(0). \tag{55}$$

Plugging Eq. (55) into Eq. (51), the objective function can be lower bounded as:

$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq g_1\left(\frac{1}{N}L(0) \right) + (K-1)g_2\left(1 - \frac{1}{N(K-1)}L(0) \right) := L_0. \tag{56}$$

On the other hand, one can directly verify that the equality for Eq. (56) is reachable when $(\boldsymbol{H}, \boldsymbol{W})$ satisfies Eq. (9). So L_0 is the global minimum of Eq. (7) and Eq. (9) is a minimizer of Eq. (7).

Now we show all the solutions are in the form of Eq. (9) under the assumption that g_2 is strictly convex and g_1 (or g_2) is strictly monotone.

By the strict convexity of g_2, the equality in Eq. (50) holds if and only if for any $k \in [K]$ and $i \in [n]$ and $k' \in [K]$, $k'' \in [K]$ such that for all $k' \neq k$ and $k'' \neq k$, we have

$$\boldsymbol{S}(\boldsymbol{z}_{i,j})(k') = \boldsymbol{S}(\boldsymbol{z}_{i,j})(k''),$$

which indicates that
$$\boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_{k'} = \boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_{k''}. \tag{57}$$

Again, by the strict convexity of g_2, Eq. (51) holds if and only if for all $k \in [K]$, $i \in [n]$, and a suitable number $C' \in (0,1)$, we have
$$\mathcal{S}(\boldsymbol{z}_{k,i})(k) := C'. \tag{58}$$

Combining Eq. (57) with Eq. (58), we have for all $(k,i,k') \in \{(k,i,k') : k \in [K], k' \in [K], k' \neq k, i \in [n]\}$,
$$\frac{\exp(\boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_k)}{\exp(\boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_{k'})} = \frac{C'(K-1)}{1-C'},$$

which implies that
$$\boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_k = \boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_{k'} + \log\left(\frac{C'(K-1)}{1-C'}\right).$$

On the other hand, by the strict monotonicity of $g_1(x) + (K-1)g_2(1 - \frac{x}{K-1})$, the equality in Eq. (56) holds if and only if $\frac{1}{N}\sum_{i=1}^{n}\sum_{k=1}^{K}\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) = L(0)$. Thus Lemma 4 reads
$$\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i} = -\sqrt{\frac{E_H}{E_W}}\boldsymbol{w}_k, \quad k \in [K], \quad i \in [n],$$

and
$$\frac{1}{K}\sum_{k=1}^{K}\frac{1}{n}\sum_{i=1}^{n}\left\|\boldsymbol{h}_{k,i}\right\|^2 = E_H, \quad \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 = E_W, \quad \bar{\boldsymbol{h}}_i = \boldsymbol{0}_p, \ i \in [n],$$

where $\bar{\boldsymbol{h}}_i := \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{h}_{k,i}$ with $i \in [n]$. Putting the pieces together, from Lemma 2, we have $(\boldsymbol{H}, \boldsymbol{W})$ satisfies Eq. (9), achieving the uniqueness argument. We complete the proof of Theorem 4. □

Lemma 4. *For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$, we have*
$$\sum_{i=1}^{n}\sum_{k=1}^{K}\boldsymbol{S}(\boldsymbol{W}\boldsymbol{h}_{k,i})(k) \leq L(0), \tag{59}$$

with L defined in Eq. (54). Moreover, recalling the definition of $\mathcal{S}_1$ and $\mathcal{S}_2$ in (i) and (ii), respectively, the equality in Eq. (59) holds if and only if $|\mathcal{S}_1| = 0$,
$$\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i} = -\sqrt{\frac{E_H}{E_W}}\boldsymbol{w}_k, \quad k \in [K], \quad i \in [n],$$

and
$$\frac{1}{K}\sum_{k=1}^{K}\frac{1}{n}\sum_{i=1}^{n}\left\|\boldsymbol{h}_{k,i}\right\|^2 = E_H, \quad \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 = E_W, \quad \bar{\boldsymbol{h}}_i = \boldsymbol{0}_p, \ i \in [n],$$

where $\bar{\boldsymbol{h}}_i := \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{h}_{k,i}$ with $i \in [n]$.

Proof of Lemma 4. For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$, we separately consider $\mathcal{S}_1$ and $\mathcal{S}_2$ defined in (i) and (ii), respectively. Let $t := |\mathcal{S}_1|$.

- For $i \in \mathcal{S}_1$, let $k \in [K]$ be any index such that $\frac{1}{K-1}\sum_{k' \neq k}\boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k) \geq \log\left(\frac{1}{K-1}\right)$, where $\boldsymbol{z}_{k,i} := \boldsymbol{W}\boldsymbol{h}_{k,i}$. By the monotonicity of $g_0(x)$, it follows from Eq. (52) that $S(\boldsymbol{z}_{k,i})(k) \leq 1/2$. Furthermore, for any other index $k' \in [K]$ such that $k' \neq k$, using that $\frac{\exp(\boldsymbol{z}_{k',i}(k'))}{\sum_{k''=1}^{K}\exp(\boldsymbol{z}_{k',i}(k''))} \leq 1$, we have
$$\sum_{i \in \mathcal{S}_1}\sum_{k=1}^{K}\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \leq t(1/2 + K - 1). \tag{60}$$

- For $i \in \mathcal{S}_2$, by the concavity of $g_0(x)$ when $x < \log\left(\frac{1}{K-1}\right)$ from Eq. (53), we have, for $\mathcal{S}_2 \neq \varnothing$,
$$\sum_{i \in \mathcal{S}_2}\sum_{k=1}^{K}\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \tag{61}$$
$$\overset{\text{Eq. (52)}}{\leq} \sum_{i \in \mathcal{S}_2}\sum_{k=1}^{K}\frac{1}{1 + (K-1)\exp\left(\frac{1}{K-1}\sum_{k'=1, \ k' \neq k}^{K}\boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right)}$$
$$\leq \frac{(n-t)K}{1 + (K-1)\exp\left(\frac{1}{(n-t)K}\sum_{i \in \mathcal{S}_2}\sum_{k=1}^{K}\left(\frac{1}{K-1}\sum_{k'=1, \ k' \neq k}^{K}\boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right)\right)}.$$

We can bound $\sum_{i\in\mathcal{S}_2}\sum_{k=1}^{K}\left(\frac{1}{K-1}\sum_{k'=1,\ k'\neq k}^{K}\boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right)$ using the similar arguments as Eq. (24) and Eq. (26). Specifically, recalling $\bar{\boldsymbol{h}}_i = \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{h}_{k,i}$ for $i\in[n]$, we have

$$\sum_{i\in\mathcal{S}_2}\sum_{k=1}^{K}\left(\frac{1}{K-1}\sum_{k'=1,\ k'\neq k}^{K}\boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right) \tag{62}$$

$$=\frac{1}{K-1}\sum_{i\in\mathcal{S}_2}\left[\left(\sum_{k=1}^{K}\boldsymbol{h}_{k,i}\right)^{\top}\left(\sum_{k=1}^{K}\boldsymbol{w}_k\right) - K\sum_{K=1}^{K}\boldsymbol{h}_{k,i}^{\top}\boldsymbol{w}_k\right]$$

$$\overset{\text{Eq. (24)}}{\geq} -\frac{K}{2(K-1)}\sum_{k=1}^{K}\sum_{i\in\mathcal{S}_2}\|\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i}\|^2/C'' - \frac{C''K(n-t)}{2(K-1)}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2$$

$$\overset{\text{Eq. (26)}}{\geq} -\frac{K}{2(K-1)}\sum_{k=1}^{K}\sum_{i\in\mathcal{S}_2}\|\boldsymbol{h}_{k,i}\|^2/C'' - \frac{C''K(n-t)}{2(K-1)}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2$$

$$\geq -\frac{K}{2(K-1)}\sum_{k=1}^{K}\sum_{i=1}^{n}\|\boldsymbol{h}_{k,i}\|^2/C'' - \frac{C''K(n-t)}{2(K-1)}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2$$

$$\geq -\frac{K^2}{(K-1)}\sqrt{E_H E_W(n-t)n},$$

where in the last inequality we follow from the constraints of Eq. (7) and set $C'' := \sqrt{\frac{nE_H}{(n-t)E_W}}$.

We combine the above two cases. When $t\in[0, n-1]$, by plugging Eq. (62) into Eq. (61), using the monotonicity of $g_0(x)$, and adding Eq. (60), we have

$$\sum_{k=1}^{n}\sum_{i=1}^{K}\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \leq N - \left(\frac{1}{2}t + \frac{K}{1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)}(n-t)\right)$$

$$= L(t). \tag{63}$$

And when $t = n$, it directly follows from Eq. (61) that

$$\sum_{k=1}^{n}\sum_{i=1}^{K}\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \leq N - \frac{n}{2} = L(n).$$

Therefore, it suffices to show $L(t) \leq L(0)$ for all $t\in[0:n]$. We first consider the case when $t\in[0:N-1]$. We show that $L(t)$ is monotonously decreasing. Indeed, define

$$q(t) := \frac{K}{1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)}.$$

We have

$$q'(t) = \frac{-\frac{1}{2}K\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)\frac{K}{K-1}\sqrt{E_H E_W n}(n-t)^{-3/2}}{\left[1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)\right]^2}$$

$$\geq \frac{-\frac{1}{2}\frac{K^2}{K-1}\sqrt{E_H E_W n}(n-t)^{-3/2}}{1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)},$$

which implies that

$$L'(t) = -\left[\frac{1}{2} - q(t) + q'(t)(n-t)\right]$$

$$\leq \frac{\frac{1}{2}\frac{K^2}{K-1}\sqrt{E_H E_W n}(n-t)^{-1/2} + K}{1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)} - \frac{1}{2}$$

$$= \frac{K\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W}\right) + 2K - 1 - \exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)}{2\left[1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)\right]}.$$

Consider function $f(x) : \left[\frac{K}{K-1}\sqrt{E_H E_W}, \frac{K}{K-1}\sqrt{E_H E_W n}\right] \to R$ as:

$$f(x) = Kx + 2K - 1 - \exp(x - \log(K-1)).$$

We have
$$f'(x) = K - \exp(x)/(K-1) < 0$$
when $x \in \left[\frac{K}{K-1} \sqrt{E_H E_W}, \frac{K}{K-1} \sqrt{E_H E_W n} \right]$, where we use the assumption that
$$\sqrt{E_H E_W} > \frac{K-1}{K} \log \left(K^2 \sqrt{E_H E_W} + (2K-1)(K-1) \right) \geq \frac{K-1}{K} \log \left(K(K-1) \right).$$
Therefore, for all $x \in \left[\frac{K}{K-1} \sqrt{E_H E_W}, \frac{K}{K-1} \sqrt{E_H E_W n} \right]$, we have
$$f(x) \leq f \left(\frac{K}{K-1} \sqrt{E_H E_W} \right) = \frac{K^2}{K-1} \sqrt{E_H E_W} + 2K - 1 - \frac{1}{K-1} \exp \left(\frac{K}{K-1} \sqrt{E_H E_W} \right) \overset{a}{<} 0,$$
where $\overset{a}{<}$ use our assumption again. We obtain $L'(t) < 0$ for all $t \in [0:N-1]$. So $L(t)$ reaches the maximum if and only if $t = 0$ when $t \in [0:N-1]$. Moreover, under our assumption, one can verify that $L(N) < L(0)$. We obtain Eq. (59) from Eq. (63) with $t = 0$.

When $t = 0$, the first inequality of Eq. (62) reduces to equality if and only if:
$$\bar{h}_i - h_{k,i} = -\sqrt{\frac{E_H}{E_W}} w_k, \quad k \in [K], \quad i \in [n].$$
The second and third inequalities of Eq. (62) reduce to equalities if and only if:
$$\frac{1}{K} \sum_{k=1}^{K} \frac{1}{n} \sum_{i=1}^{n} \left\| h_{k,i} \right\|^2 = E_H, \quad \frac{1}{K} \sum_{k=1}^{K} \| w_k \|^2 = E_W, \quad \bar{h}_i = \mathbf{0}_p, \ i \in [n].$$
We obtain Lemma 4. $\qquad\square$

B. Imbalanced Case.

B.1. Proofs of Lemma 1 and Proposition 1.

Proof of Lemma 1. For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ for the original program Eq. (7), we define
$$\boldsymbol{h}_k := \frac{1}{n_k} \sum_{i=1}^{n_k} \boldsymbol{h}_{k,i}, \ k \in [K], \quad \text{and} \quad \boldsymbol{X} := \left[\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_K, \boldsymbol{W}^\top \right]^\top \left[\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_K, \boldsymbol{W}^\top \right].$$
Clearly, $\boldsymbol{X} \succeq 0$. For the other two constraints of Eq. (15), we have
$$\frac{1}{K} \sum_{k=1}^{K} \boldsymbol{X}(k,k) = \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{h}_k\|^2 \overset{a}{\leq} \frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \left\| \boldsymbol{h}_{k,i} \right\|^2 \overset{b}{\leq} E_H,$$
and
$$\frac{1}{K} \sum_{k=K+1}^{2K} \boldsymbol{X}(k,k) = \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 \overset{c}{\leq} E_W,$$
where $\overset{a}{\leq}$ applies Jensen's inequality and $\overset{b}{\leq}$ and $\overset{c}{\leq}$ use that $(\boldsymbol{H}, \boldsymbol{W})$ is a feasible solution. So $\boldsymbol{X}$ is a feasible solution for the convex program Eq. (15). Letting L_0 be the global minimum of Eq. (15), for any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$, we obtain
$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) = \sum_{k=1}^{K} \frac{n_k}{N} \left[\frac{1}{n_k} \sum_{k=1}^{n_k} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \right]$$
$$\overset{a}{\geq} \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_k, \boldsymbol{y}_k) = \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}(\boldsymbol{z}_k, \boldsymbol{y}_k) \geq L_0, \qquad [64]$$
where in $\overset{a}{\geq}$, we use $\mathcal{L}$ is convex on the first argument, and so $\mathcal{L}(\boldsymbol{W}\boldsymbol{h}, \boldsymbol{y}_k)$ is convex on $\boldsymbol{h}$ given $\boldsymbol{W}$ and $k \in [K]$.

On the other hand, considering the solution $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ defined in Eq. (16) with $\boldsymbol{X}^\star$ being a minimizer of Eq. (15), we have $\left[\boldsymbol{h}_1^\star, \boldsymbol{h}_2^\star, \ldots, \boldsymbol{h}_K^\star, (\boldsymbol{W}^\star)^\top \right]^\top \left[\boldsymbol{h}_1^\star, \boldsymbol{h}_2^\star, \ldots, \boldsymbol{h}_K^\star, (\boldsymbol{W}^\star)^\top \right] = \boldsymbol{X}^\star$ ($p \geq 2K$ guarantees the existence of $\left[\boldsymbol{h}_1^\star, \boldsymbol{h}_2^\star, \ldots, \boldsymbol{h}_K^\star, (\boldsymbol{W}^\star)^\top \right]$). We can verify that $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a feasible solution for Eq. (7) and have
$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}(\boldsymbol{W}^\star \boldsymbol{h}_{k,i}^\star, \boldsymbol{y}_k) = \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}(\boldsymbol{z}_k^\star, \boldsymbol{y}_k) = L_0, \qquad [65]$$
where $\boldsymbol{z}_k^\star = [\boldsymbol{X}^\star(k, 1+K), \boldsymbol{X}^\star(k, 2+K), \ldots, \boldsymbol{X}^\star(k, 2K)]^\top$ for $k \in [K]$.

Combining Eq. (64) and Eq. (65), we conclude that L_0 is the global minimum of Eq. (7) and $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a minimizer.

Suppose there is a minimizer $(\boldsymbol{H}', \boldsymbol{W}')$ that cannot be written as Eq. (16). Let
$$\boldsymbol{h}_k' = \frac{1}{n_k} \sum_{i=1}^{n_k} \boldsymbol{h}_{k,i}', \ k \in [K], \quad \text{and} \quad \boldsymbol{X}' = \left[\boldsymbol{h}_1', \boldsymbol{h}_2', \ldots, \boldsymbol{h}_K', (\boldsymbol{W}')^\top \right]^\top \left[\boldsymbol{h}_1', \boldsymbol{h}_2', \ldots, \boldsymbol{h}_K', (\boldsymbol{W}')^\top \right].$$

Eq. (64) implies that $\boldsymbol{X}'$ is a minimizer of Eq. (15). As $(\boldsymbol{H}', \boldsymbol{W}')$ cannot be written as Eq. (16) with $\boldsymbol{X}^\star = \boldsymbol{X}'$, then there is a $k' \in [K]$, $i, j \in [n_{k'}]$ with $i \neq j$ such that $\boldsymbol{h}_{k',i} \neq \boldsymbol{h}_{k',j}$. We have

$$
\begin{aligned}
\frac{1}{K} \sum_{k=1}^{K} \boldsymbol{X}'(k,k) &= \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{h}_k'\|^2 \\
&= \frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \left\|\boldsymbol{h}_{k,i}'\right\|^2 - \frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{k=1}^{K} \|\boldsymbol{h}_{k,i}' - \boldsymbol{h}_k'\|^2 \\
&\leq \frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \left\|\boldsymbol{h}_{k,i}'\right\|^2 - \frac{1}{K} \frac{1}{n_{k'}} (\|\boldsymbol{h}_{k',i}' - \boldsymbol{h}_{k'}'\|^2 + \|\boldsymbol{h}_{k',j}' - \boldsymbol{h}_{k'}'\|^2) \\
&\leq \frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \left\|\boldsymbol{h}_{k,i}'\right\|^2 - \frac{1}{K} \frac{1}{2n_{k'}} \|\boldsymbol{h}_{k',i}' - \boldsymbol{h}_{k',j}'\|^2 \\
&< E_H.
\end{aligned}
$$

By contraposition, if all $\boldsymbol{X}^\star$ satisfy that $\frac{1}{K} \sum_{k=1}^{K} \boldsymbol{X}^\star(k,k) = E_H$, then all the solutions of Eq. (7) are in the form of Eq. (16). We complete the proof. □

Proposition 1 can be obtained by the same argument. We omit the proof here.

B.2. Proof of Theorem 5. To prove Theorem 5, we first study a limit case where we only learn the classification for a partial classes. We solve the optimization program:

$$
\begin{aligned}
\min_{\boldsymbol{H}, \boldsymbol{W}} \quad & \frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \\
\text{s.t.} \quad & \frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \left\|\boldsymbol{h}_{k,i}\right\|^2 \leq E_H, \\
& \frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 \leq E_W,
\end{aligned}
\qquad [66]
$$

where $n_1 = n_2 = \cdots = n_{K_A} = n_A$ and $n_{K_A+1} = n_{K_A+2} = \cdots = n_K = n_B$. Lemma 5 characterizes useful properties for the minimizer of Eq. (66).

Lemma 5. *Let* $(\boldsymbol{H}, \boldsymbol{W})$ *be a minimzer of* Eq. (66). *We have* $\boldsymbol{h}_{k,i} = \boldsymbol{0}_p$ *for all* $k \in [K_A + 1 : K]$ *and* $i \in [n_B]$. *Let* L_0 *be the global minimum of* Eq. (66). *We have*

$$
L_0 = \frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k).
$$

Then L_0 *only depends on* K_A, K_B, E_H, *and* E_W. *Moreover, for any feasible solution* $(\boldsymbol{H}', \boldsymbol{W}')$, *if there exist* $k, k' \in [K_A + 1 : K]$ *such that* $\|\boldsymbol{w}_k - \boldsymbol{w}_{k'}\| = \varepsilon > 0$, *we have*

$$
\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}\left(\boldsymbol{W}'\boldsymbol{h}_{k,i}', \boldsymbol{y}_k\right) \geq L_0 + \varepsilon',
$$

where $\varepsilon' > 0$ *depends on* ε, K_A, K_B, E_H, *and* E_W.

Now we are ready to prove Theorem 5. The proof is based on the contradiction.

Proof of Theorem 5. Consider sequences n_A^ℓ and n_B^ℓ with $R^\ell := n_A^\ell / n_B^\ell$ for $\ell = 1, 2, \ldots$. We have $R^\ell \to \infty$. For each optimization program indexed by $\ell \in \mathbb{N}_+$, we introduce $(\boldsymbol{H}^{\ell,\star}, \boldsymbol{W}^{\ell,\star})$ as a minimizer and separate the objective function into two parts. We consider

$$
\mathcal{L}^\ell\left(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell\right) = \frac{K_A n_A^\ell}{K_A n_A^\ell + K_B n_B^\ell} \mathcal{L}_A^\ell\left(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell\right) + \frac{K_B n_B^\ell}{K_A n_A^\ell + K_B n_B^\ell} \mathcal{L}_B^\ell\left(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell\right),
$$

with

$$
\mathcal{L}_A^\ell\left(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell\right) := \frac{1}{K_A n_A^\ell} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A^\ell} \mathcal{L}\left(\boldsymbol{W}^\ell \boldsymbol{h}_{k,i}^\ell, \boldsymbol{y}_k\right)
$$

and

$$
\mathcal{L}_B^\ell\left(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell\right) := \frac{1}{K_B n_B^\ell} \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B^\ell} \mathcal{L}\left(\boldsymbol{W}^\ell \boldsymbol{h}_{k,i}^\ell, \boldsymbol{y}_k\right).
$$

We define $\left(\boldsymbol{H}^{\ell,A}, \boldsymbol{W}^{\ell,A}\right)$ as a minimizer of the optimization program:

$$\min_{\boldsymbol{H}^\ell, \boldsymbol{W}^\ell} \quad \mathcal{L}_A^\ell\left(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell\right)$$

$$\text{s.t.} \quad \frac{1}{K}\sum_{k=1}^{K}\left\|\boldsymbol{w}_k^\ell\right\|^2 \leq E_W,$$

$$\frac{1}{K}\sum_{k=1}^{K_A}\frac{1}{n_A^\ell}\sum_{i=1}^{n_A^\ell}\left\|\boldsymbol{h}_{k,i}^\ell\right\|^2 + \frac{1}{K}\sum_{k=K_A+1}^{K}\frac{1}{n_B^\ell}\sum_{i=1}^{n_B^\ell}\left\|\boldsymbol{h}_{k,i}^\ell\right\|^2 \leq E_H, \tag{67}$$

and $\left(\boldsymbol{H}^{\ell,B}, \boldsymbol{W}^{\ell,B}\right)$ as a minimizer of the optimization program:

$$\min_{\boldsymbol{H}^\ell, \boldsymbol{W}^\ell} \quad \mathcal{L}_B^\ell\left(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell\right)$$

$$\text{s.t.} \quad \frac{1}{K}\sum_{k=1}^{K}\left\|\boldsymbol{w}_k^\ell\right\|^2 \leq E_W,$$

$$\frac{1}{K}\sum_{k=1}^{K_A}\frac{1}{n_A^\ell}\sum_{i=1}^{n_A^\ell}\left\|\boldsymbol{h}_{k,i}^\ell\right\|^2 + \frac{1}{K}\sum_{k=K_A+1}^{K}\frac{1}{n_B^\ell}\sum_{i=1}^{n_B^\ell}\left\|\boldsymbol{h}_{k,i}^\ell\right\|^2 \leq E_H. \tag{68}$$

Note that Programs Eq. (67) and Eq. (68) and their minimizers have been studied in Lemma 5. We define:

$$L_A := \mathcal{L}_A^\ell\left(\boldsymbol{H}^{\ell,A}, \boldsymbol{W}^{\ell,A}\right) \quad \text{and} \quad L_B := \mathcal{L}_B^\ell\left(\boldsymbol{H}^{\ell,B}, \boldsymbol{W}^{\ell,B}\right).$$

Then Lemma 5 implies that L_A and L_B only depend on K_A, K_B, E_H, and E_W, and are independent of ℓ. Moreover, since $\boldsymbol{h}_{k,i}^{\ell,A} = \boldsymbol{0}_p$ for all $k \in [K_A+1:K]$ and $i \in [n_B]$, we have

$$\mathcal{L}_B^\ell\left(\boldsymbol{H}^{\ell,A}, \boldsymbol{W}^{\ell,A}\right) = \log(K). \tag{69}$$

Now we prove Theorem 5 by contradiction. Suppose there exists a pair (k,k') such that $\lim_{\ell\to\infty} \boldsymbol{w}_k^{\ell,\star} - \boldsymbol{w}_{k'}^{\ell,\star} \neq \boldsymbol{0}_p$. Then there exists $\varepsilon > 0$ such that for a subsequence $\{(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star})\}_{\ell=1}^\infty$ and an index ℓ_0 when $\ell \geq \ell_0$, we have $\left\|\boldsymbol{w}_k^{a_\ell,\star} - \boldsymbol{w}_{k'}^{a_\ell,\star}\right\| \geq \varepsilon$. Now we figure out a contradiction by estimating the objective function value on $(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star})$. In fact, because $(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star})$ is a minimizer of $\mathcal{L}^\ell(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell)$, we have

$$\mathcal{L}^{a_\ell}\left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star}\right) \leq \mathcal{L}^{a_\ell}\left(\boldsymbol{H}^{a_\ell,A}, \boldsymbol{W}^{a_\ell,A}\right) \stackrel{\text{Eq. (69)}}{=} \frac{K_A n_A^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}} L_A + \frac{K_B n_B^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}} \log(K)$$

$$= L_A + \frac{1}{K_R R^{a_\ell} + 1}\left(\log(K) - L_A\right) \stackrel{\ell\to\infty}{\to} L_A, \tag{70}$$

where we define $K_R := K_A/K_B$ and use $R^\ell = n_A^\ell/n_B^\ell$.

However, when $\ell > \ell_0$, because $\left\|\boldsymbol{w}_k^{a_\ell,\star} - \boldsymbol{w}_{k'}^{a_\ell,\star}\right\| \geq \varepsilon > 0$, Lemma 5 implies that

$$\mathcal{L}_A^{a_\ell}\left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star}\right) \geq L_A + \varepsilon_2,$$

where $\varepsilon_2 > 0$ only depends on ε, K_A, K_B, E_H, and E_W, and is independent of ℓ. We obtain

$$\mathcal{L}^{a_\ell}\left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star}\right) = \frac{K_A n_A^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}}\mathcal{L}_A^{a_\ell}\left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star}\right) + \frac{K_B n_B^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}}\mathcal{L}_B^{a_\ell}\left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star}\right)$$

$$\stackrel{a}{\geq} \frac{K_A n_A^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}}\mathcal{L}_A^{a_\ell}\left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star}\right) + \frac{K_B n_B^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}}\mathcal{L}_B^{a_\ell}\left(\boldsymbol{H}^{a_\ell,B}, \boldsymbol{W}^{a_\ell,B}\right)$$

$$= \frac{K_A n_A^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}}(L_A + \varepsilon_2) + \frac{K_B n_B^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}} L_B$$

$$= L_A + \varepsilon_2 + \frac{1}{K_R R^{a_\ell} + 1}(L_B - L_A - \varepsilon_2) \stackrel{\ell\to\infty}{\to} L_A + \varepsilon_2, \tag{71}$$

where $\stackrel{a}{\geq}$ uses $\left(\boldsymbol{H}^{a_\ell,B}, \boldsymbol{W}^{a_\ell,B}\right)$ is the minimizer of Eq. (68). Thus we meet contradiction by comparing Eq. (70) with Eq. (71) and achieve Theorem 5. $\qquad\square$

Proof of Lemma 5. For any constants $C_a > 0$, $C_b > 0$, and $C_c > 0$, define $C_a' := \frac{C_a}{C_a + (K_A-1)C_b + K_B C_c} \in (0,1)$, $C_b' := \frac{C_b}{C_a + (K_A-1)C_b + K_B C_c} \in (0,1)$, and $C_c' := \frac{C_c}{C_a + (K_A-1)C_b + K_B C_c} \in (0,1)$, $C_d := -C_a' \log(C_a') - C_b'(K_A - 1)\log(C_b') - K_B C_c' \log(C_c')$, $C_e := \frac{K_A C_b}{K_A C_b + K_B C_c} \in (0,1)$, $C_f := \frac{K_B C_c}{K_A C_b + K_B C_c} \in (0,1)$, and $C_g := \frac{K_A C_b + K_B C_c}{C_a + (K_A-1)C_b + K_B C_c} > 0$. Using a similar argument as Theorem 1, we show in Lemma 6 (see the end of the proof), for any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ of Eq. (66), the objective value can be bounded from

below by:

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \tag{72}$$

$$\overset{a}{\geq} - \frac{C_g}{K_A} \sqrt{KE_H} \sqrt{\sum_{k=1}^{K_A} \left\| C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k \right\|^2} + C_d$$

$$\overset{b}{\geq} - \frac{C_g}{K_A} \sqrt{KE_H} \sqrt{KE_W - K_A \left(1/K_R - C_f^2 - \frac{C_f^4}{C_e(2 - C_e)} \right) \|\boldsymbol{w}_B\|^2 - \sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2} + C_d,$$

where $\boldsymbol{w}_A := \frac{1}{K_A} \sum_{k=1}^{K_A} \boldsymbol{w}_k$, $\boldsymbol{w}_B := \frac{1}{K_B} \sum_{k=K_A+1}^{K} \boldsymbol{w}_k$, and $K_R := \frac{K_A}{K_B}$. Moreover, the equality in $\overset{a}{\geq}$ holds only if $\boldsymbol{h}_{k,i} = \boldsymbol{0}_p$ for all $k \in [K_A + 1 : K]$ and $i \in [n_B]$.

Though C_a, C_b, and C_c can be any positive numbers, we need to carefully pick them to exactly reach the global minimum of Eq. (66). In the following, we separately consider three cases according to the values of K_A, K_B, and $E_H E_W$.

(i) Consider the case when $K_A = 1$. We pick $C_a := \exp\left(\sqrt{K_B(1 + K_B)E_H E_W}\right)$, $C_b := 1$, and $C_c := \exp\left(-\sqrt{(1 + K_B)E_H E_W / K_B}\right)$.

Then from $\overset{a}{\geq}$ in Eq. (72), we have

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k)$$

$$\overset{a}{\geq} - C_g C_f \sqrt{KE_H} \sqrt{\|\boldsymbol{w}_1 - \boldsymbol{w}_B\|^2} + C_d$$

$$= - C_g C_f \sqrt{KE_H} \sqrt{\|\boldsymbol{w}_1\|^2 - 2\boldsymbol{w}_1^\top \boldsymbol{w}_B + \|\boldsymbol{w}_B\|^2} + C_d$$

$$\overset{b}{\geq} - C_g C_f \sqrt{KE_H} \sqrt{(1 + 1/K_B)(\|\boldsymbol{w}_1\|^2 + K_B\|\boldsymbol{w}_B\|^2)} + C_d$$

$$\overset{c}{\geq} - C_g C_f \sqrt{KE_H} \sqrt{(1 + 1/K_B)\left(KE_W - \sum_{k=2}^{K} \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2 \right)} + C_d$$

$$\geq - C_g C_f \sqrt{KE_H} \sqrt{(1 + 1/K_B)KE_W} + C_d := L_1, \tag{73}$$

where $\overset{a}{\geq}$ uses $C_e + C_f = 1$, $\overset{b}{\geq}$ follows from $2ab \leq a^2 + b^2$, i.e., $-2\boldsymbol{w}_1^\top \boldsymbol{w}_B \leq (1/K_B)\|\boldsymbol{w}_1\|^2 + K_B\|\boldsymbol{w}_B\|^2$, and $\overset{c}{\geq}$ follows from $\sum_{k=2}^{K} \|\boldsymbol{w}_k\|^2 = K_B\|\boldsymbol{w}_B\|^2 + \sum_{k=2}^{K} \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2$ and the constraint that $\sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 \leq KE_W$.

On the other hand, when $(\boldsymbol{H}, \boldsymbol{W})$ satisfies that

$$\boldsymbol{w}_1 = \sqrt{K_B E_W}\boldsymbol{u}, \quad \boldsymbol{w}_k = -\sqrt{1/K_B E_W}\boldsymbol{u}, \; k \in [2 : K],$$

$$\boldsymbol{h}_{1,i} = \sqrt{(1 + K_B)E_H}\boldsymbol{u}, \; i \in [n_A], \qquad \boldsymbol{h}_{k,i} = \boldsymbol{0}_p, \; k \in [2 : K], \; i \in [n_B],$$

where $\boldsymbol{u}$ is any unit vector, the inequalities in Eq. (73) reduce to equalities. So L_1 is the global minimum of Eq. (66). Moreover, L_1 is achieved only if $\overset{a}{\geq}$ in Eq. (72) reduces to inequality. From Lemma 66, we have that any minimizer satisfies that $\boldsymbol{h}_{k,i} = \boldsymbol{0}_p$ for all $k \in [K_A + 1 : K]$ and $i \in [n_B]$.

Finally, for any feasible solution $(\boldsymbol{H}', \boldsymbol{W}')$, if there exist $k, k' \in [K_A + 1 : K]$ such that $\|\boldsymbol{w}_k - \boldsymbol{w}_{k'}\| = \varepsilon > 0$, we have

$$\sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2 \geq \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2 + \|\boldsymbol{w}_{k'} - \boldsymbol{w}_B\|^2 \geq \frac{\|\boldsymbol{w}_k - \boldsymbol{w}_{k'}\|^2}{2} = \varepsilon^2/2. \tag{74}$$

It follows from $\overset{c}{\geq}$ in Eq. (73) that

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq -C_g C_f \sqrt{KE_H} \sqrt{(1 + 1/K_B)(KE_W - \varepsilon^2/2)} + C_d := L_1 + \varepsilon_1$$

with $\varepsilon_1 > 0$ depending on ε, K_A, K_B, E_H, and E_W.

- (ii) Consider the case when $K_A > 1$ and $\exp\left((1 + 1/K_R)\sqrt{E_H E_W}/(K_A - 1)\right) < \sqrt{1 + K_R} + 1$. Let us pick $C_a := \exp\left((1 + 1/K_R)\sqrt{E_H E_W}\right)$, $C_b := \exp\left(-\frac{1}{K_A - 1}(1 + 1/K_R)\sqrt{E_H E_W}\right)$, and $C_c := 1$.

Following from $\overset{b}{\geq}$ in Eq. (72), we know if $1/K_R - C_f^2 - \frac{C_f^4}{C_e(2 - C_f)} > 0$, then

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq -C_g(1 + 1/K_R)\sqrt{E_H E_W} + C_d := L_2. \tag{75}$$

In fact, we do have $1/K_R - C_f^2 - \frac{C_f^4}{C_e(2-C_f)} > 0$ because

$$1/K_R > C_f^2 - \frac{C_f^4}{C_e(2-C_e)} \qquad \left(\text{by } C_e + C_f = 1\right)$$

$$\iff \quad C_e > \sqrt{\frac{1}{1+K_R}} \qquad \left(\text{by } C_e = \frac{K_B C_c}{K_A C_b + K_B C_c}\right)$$

$$\iff \quad \frac{C_b}{C_c} > \frac{1}{\sqrt{1+K_R}+1}$$

$$\iff \quad \exp\left((1+1/K_R)\sqrt{E_H E_W}/(K_A-1)\right) < \sqrt{1+K_R}+1.$$

On the other hand, when $(\boldsymbol{H}, \boldsymbol{W})$ satisfies that

$$\left[\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{K_A}\right] = \sqrt{\frac{E_W}{E_H}} \left[\boldsymbol{h}_1, \ldots, \boldsymbol{h}_{K_A}\right]^\top = \sqrt{(1+1/K_R)E_W} \, (\boldsymbol{M}_A^\star)^\top,$$

$$\boldsymbol{h}_{k,i} = \boldsymbol{h}_k, \quad k \in [K_A], \ i \in [n_A]$$

$$\boldsymbol{h}_{k,i} = \boldsymbol{w}_k = \boldsymbol{0}_p, \quad k \in [K_A+1:K], \ i \in [n_B],$$

where $\boldsymbol{M}_A^\star$ is a K_A-simplex ETF, Eq. (75) reduces to equality. So L_2 is the global minimum of Eq. (66). Moreover, L_2 is achieved only if $\overset{a}{\geq}$ of Eq. (72) reduces to equality. From Lemma 6, we have that any minimizer satisfies that $\boldsymbol{h}_{k,i} = \boldsymbol{0}_p$ for all $k \in [K_A+1:K]$ and $i \in [n_B]$.

Finally, for any feasible solution $(\boldsymbol{H}', \boldsymbol{W}')$, if there exist $k, k' \in [K_A+1:K]$ such that $\|\boldsymbol{w}_k - \boldsymbol{w}_{k'}\| = \varepsilon > 0$, plugging Eq. (74) into $\overset{b}{\geq}$ in Eq. (72), we have

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq -\frac{C_g}{K_A}\sqrt{KE_H}\sqrt{KE_W - \varepsilon^2/2} + C_d := L_2 + \varepsilon_2, \qquad [76]$$

with $\varepsilon_2 > 0$ depending on ε, K_A, K_B, E_H, and E_W.

- (iii) Consider the case when $K_A > 1$ and $\exp((1+1/K_R)\sqrt{E_H E_W}/(K_A-1)) \geq \sqrt{1+K_R}+1$. Let $C_f' := \frac{1}{\sqrt{K_R+1}}$ and $C_e' := 1 - C_f'$. For $x \in [0,1]$, we define:

$$g_N(x) := \sqrt{\frac{(1+K_R)E_W}{K_R x^2 + (K_R + K_R^2)(1-x)^2}},$$

$$g_a(x) := \exp\left(\frac{g_N(x)\sqrt{(1+K_R)E_H/K_R}}{\sqrt{x^2 + \left(1+\frac{C_e'}{C_f'}\right)^2(1-x)^2}}\left[x^2 + \left(1+\frac{C_e'}{C_f'}\right)(1-x)^2\right]\right),$$

$$g_b(x) := \exp\left(\frac{g_N(x)\sqrt{(1+K_R)E_H/K_R}}{\sqrt{x^2 + \left(1+\frac{C_e'}{C_f'}\right)^2(1-x)^2}}\left[-\frac{1}{K_A-1}x^2 + \left(1+\frac{C_e'}{C_f'}\right)(1-x)^2\right]\right),$$

$$g_c(x) := \exp\left(\frac{g_N(x)\sqrt{(1+K_R)E_H/K_R}}{\sqrt{x^2 + \left(1+\frac{C_e'}{C_f'}\right)^2(1-x)^2}}\left[-\left(1+\frac{C_e'}{C_f'}\right)K_R(1-x)^2\right]\right).$$

Let $x_0 \in [0,1]$ be a root of the equation

$$g_b(x)/g_c(x) = \frac{1/C_f' - 1}{K_R}.$$

We first show that the solution x_0 exists. First of all, one can directly verify when $x \in [0,1]$, $g_b(x)/g_c(x)$ is continuous. It suffices to prove that (A) $g_b(0)/g_c(0) \geq \frac{1/C_f'-1}{K_R}$ and (B) $g_b(1)/g_c(1) \leq \frac{1/C_f'-1}{K_R}$.

(A) When $x = 0$, we have $g_b(x)/g_c(x) \geq \exp(0) = 1$. At the same time, $\frac{1/C_f'-1}{K_R} = \frac{\sqrt{K_R+1}-1}{K_R} = \frac{1}{\sqrt{K_R+1}+1} \leq 1$. Thus (i) is achieved.

(B) When $x = 1$, we have $g_N(1) = \sqrt{(1+1/K_R)E_W}$, so

$$g_b(1)/g_c(1) = \exp\left(-(1+1/K_R)\sqrt{E_H E_W}/(K_A-1)\right) \overset{a}{\leq} \frac{1}{\sqrt{K_R+1}+1} = \frac{1/C_f'-1}{K_R}.$$

where $\overset{a}{\leq}$ is obtained by the condition that

$$\exp\left((1+1/K_R)\sqrt{E_H E_W}/(K_A-1)\right) \geq \sqrt{1+K_R}+1.$$

Now we pick $C_a := g_a(x_0)$, $C_b := g_b(x_0)$, and $C_c := g_c(x_0)$, because $\frac{C_b}{C_c} = \frac{1/C_f' - 1}{K_R}$, we have $C_e = C_e'$ and $C_f = C_f'$ and $1/K_R = C_f^2 + \frac{C_f^4}{C_e(2-C_e)}$. Then it follows from $\overset{b}{\geq}$ in Eq. (72) that

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq -C_g(1 + 1/K_R)\sqrt{E_H E_W} + C_d = L_2. \tag{77}$$

On the other hand, consider the solution $(\boldsymbol{H}, \boldsymbol{W})$ that satisfies

$$\boldsymbol{w}_k = g_N(x_0)\boldsymbol{P}_A \left[\frac{x_0}{\sqrt{(K_A-1)K_A}}(K_A\boldsymbol{y}_k - \mathbf{1}_{K_A}) + \frac{1-x_0}{\sqrt{K_A}}\mathbf{1}_{K_A} \right], \quad k \in [K_A],$$

$$\boldsymbol{w}_k = -\frac{C_e(2-C_e)}{C_f^2 K_A}\boldsymbol{P}_A \sum_{k=1}^{K_A} \boldsymbol{w}_k, \quad k \in [K_A + 1 : K],$$

$$\boldsymbol{h}_{k,i} = \frac{\sqrt{(1+1/K_R)E_H}}{\|\boldsymbol{w}_i + \frac{C_e}{C_f K_A}\sum_{k=1}^{K_A}\boldsymbol{w}_k\|}\boldsymbol{P}_A \left[\boldsymbol{w}_i + \frac{C_e}{C_f K_A}\sum_{k=1}^{K_A} \boldsymbol{w}_k \right], \quad k \in [K_A], \ i \in [n_A],$$

$$\boldsymbol{h}_{k,i} = \mathbf{0}_p, \quad k \in [K_A + 1 : K], \ i \in [n_B],$$

where $\boldsymbol{y}_k \in \mathbb{R}^K$ is the vector containing one in the k-th entry and zero elsewhere and $\boldsymbol{P}_A \in \mathbb{R}^{p \times K_A}$ is a partial orthogonal matrix such that $\boldsymbol{P}_A^\top \boldsymbol{P}_A = \boldsymbol{I}_{K_A}$. We have $\exp\left(\boldsymbol{h}_{k,i}^\top \boldsymbol{w}_k\right) = g_a(x_0)$ for $i \in [n_A]$ and $k \in [K_A]$, $\exp\left(\boldsymbol{h}_{k,i}^\top \boldsymbol{w}_{k'}\right) = g_b(x_0)$ for $i \in [n_A]$ and $k, k' \in [K_A]$ such that $k \neq k'$, and $\exp\left(\boldsymbol{h}_{k,i}^\top \boldsymbol{w}_{k'}\right) = g_c(x_0)$ for $i \in [n_A]$, $k \in [K_A]$, and $k' \in [K_B]$. Moreover, $(\boldsymbol{H}, \boldsymbol{W})$ can achieve the equality in Eq. (77). Finally, following the same argument as Case (ii), we have that (1) L_2 is the global minimum of Eq. (66); (2) any minimizer satisfies that $\boldsymbol{h}_{k,i} = \mathbf{0}_p$ for all $k \in [K_A + 1 : K]$ and $i \in [n_B]$; (3) for any feasible solution $(\boldsymbol{H}', \boldsymbol{W}')$, if there exist $k, k' \in [K_A + 1 : K]$ such that $\|\boldsymbol{w}_k - \boldsymbol{w}_{k'}\| = \varepsilon > 0$, then Eq. (76) holds.

Combining the three cases, we obtain Lemma 5, completing the proof. $\qquad \square$

Lemma 6. *For any constants $C_a > 0$, $C_b > 0$, and $C_c > 0$, define $C_a' := \frac{C_a}{C_a + (K_A-1)C_b + K_B C_c} \in (0,1)$, $C_b' := \frac{C_b}{C_a + (K_A-1)C_b + K_B C_c} \in (0,1)$, and $C_c' := \frac{C_c}{C_a + (K_A-1)C_b + K_B C_c} \in (0,1)$, $C_d := -C_a' \log(C_a') - C_b'(K_A - 1)\log(C_b') - K_B C_c' \log(C_c')$, $C_e := \frac{K_A C_b}{K_A C_b + K_B C_c} \in (0,1)$, $C_f := \frac{K_B C_c}{K_A C_b + K_B C_c} \in (0,1)$, and $C_g := \frac{K_A C_b + K_B C_c}{C_a + (K_A-1)C_b + K_B C_c} > 0$. For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ of Eq. (66), the objective value of Eq. (66) can be bounded from below by:*

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \tag{78}$$

$$\overset{a}{\geq} -\frac{C_g}{K_A}\sqrt{KE_H}\sqrt{\sum_{k=1}^{K_A}\left\|C_e\boldsymbol{w}_A + C_f\boldsymbol{w}_B - \boldsymbol{w}_k\right\|^2} + C_d$$

$$\overset{b}{\geq} -\frac{C_g}{K_A}\sqrt{KE_H}\sqrt{KE_W - K_A\left(1/K_R - C_f^2 - \frac{C_f^4}{C_e(2-C_e)}\right)\|\boldsymbol{w}_B\|^2 - \sum_{k=K_A+1}^{K}\|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2} + C_d,$$

where $\boldsymbol{w}_A := \frac{1}{K_A}\sum_{k=1}^{K_A}\boldsymbol{w}_k$, $\boldsymbol{w}_B := \frac{1}{K_B}\sum_{k=K_A+1}^{K}\boldsymbol{w}_k$, and $K_R := \frac{K_A}{K_B}$. Moreover, the equality in $\overset{a}{\geq}$ hold only if $\boldsymbol{h}_{k,i} = \mathbf{0}_p$ for all $k \in [K_A + 1 : K]$.

Remark 6. Note that the case $\boldsymbol{h}_{k,i} = \mathbf{0}_p$ does not imply that the network activations all die for the classes $k \in [K_A + 1 : K]$. This is because our analysis does not include the bias term for simplicity.

Proof of Lemma 6. For $k \in [K_A]$ and $i \in [n_k]$, we introduce $\boldsymbol{z}_{k,i} = \boldsymbol{W}\boldsymbol{h}_{k,i}$. Because that $C_a' + (K_A - 1)C_b' + K_B C_c' = 1$, $C_a' > 0$, $C_b' > 0$, and $C_c' > 0$, by the concavity of $\log(\cdot)$, we have

$$-\log\left(\frac{\exp(\boldsymbol{z}_{k,i}(i))}{\sum_{k'=1}^{K}\exp(\boldsymbol{z}_{k',i}(k))}\right) \tag{79}$$

$$= -\boldsymbol{z}_{k,i}(k) + \log\left(C_a'\left(\frac{\exp(\boldsymbol{z}_{k,i}(k))}{C_a'}\right) + \sum_{k'=1, \ k'\neq k}^{K_A} C_b'\left(\frac{\exp(\boldsymbol{z}_{k,i}(k'))}{C_b'}\right) + \sum_{k'=K_A+1}^{K} C_c'\left(\frac{\exp(\boldsymbol{z}_{k,i}(k'))}{C_c'}\right)\right)$$

$$\geq -\boldsymbol{z}_{k,i}(k) + C_a'\boldsymbol{z}_{k,i}(k) + C_b'\sum_{k'=1, \ k'\neq k}^{K_A}\boldsymbol{z}_{k,i}(k') + C_C'\sum_{k'=K_A+1}^{K}\boldsymbol{z}_{i,j}(k) + C_d$$

$$= C_g C_e\left(\frac{1}{K_A}\sum_{k'=1}^{K_A}\boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right) + C_g C_f\left(\frac{1}{K_B}\sum_{k'=K_A+1}^{K}\boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right) + C_d.$$

Therefore, integrating Eq. (79) with $k \in [K_A]$ and $i \in [n_A]$, recalling that $\boldsymbol{w}_A = \frac{1}{K_A} \sum_{k=1}^{K_A} \boldsymbol{w}_k$ and $\boldsymbol{w}_B = \frac{1}{K_B} \sum_{k=K_A+1}^{K} \boldsymbol{w}_k$, we have

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \tag{80}$$

$$\geq \frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} C_g \left[C_e(\boldsymbol{h}_{k,i}\boldsymbol{w}_A - \boldsymbol{h}_{k,i}\boldsymbol{w}_k) + C_f(\boldsymbol{h}_{k,i}\boldsymbol{w}_B - \boldsymbol{h}_{k,i}\boldsymbol{w}_k) \right] + C_d$$

$$\overset{a}{=} \frac{C_g}{K_A} \sum_{k=1}^{K_A} \boldsymbol{h}_k^\top (C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k) + C_d,$$

where in $\overset{a}{=}$, we introduce $\boldsymbol{h}_k := \frac{1}{n_k} \sum_{i=1}^{n_k} \boldsymbol{h}_{k,i}$ for $k \in [K]$, and use $C_e + C_f = 1$. Then it is sufficient to bound $\sum_{k=1}^{K_A} \boldsymbol{h}_k^\top (C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k)$. By the Cauchy–Schwarz inequality, we have

$$\sum_{k=1}^{K_A} \boldsymbol{h}_k^\top (C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k) \geq - \sqrt{\sum_{k=1}^{K_A} \|\boldsymbol{h}_k\|^2} \sqrt{\sum_{k=1}^{K_A} \left\| C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k \right\|^2}$$

$$\overset{a}{\geq} - \sqrt{\sum_{k=1}^{K_A} \frac{1}{n_k} \sum_{i=1}^{n_k} \|\boldsymbol{h}_{k,i}\|^2} \sqrt{\sum_{k=1}^{K_A} \left\| C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k \right\|^2}$$

$$\overset{b}{\geq} - \sqrt{K E_H} \sqrt{\sum_{k=1}^{K_A} \left\| C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k \right\|^2}, \tag{81}$$

where $\overset{a}{\geq}$ follows from Jensen's inequality $\frac{1}{n_k} \sum_{i=1}^{n_k} \|\boldsymbol{h}_{k,i}\|^2 \geq \boldsymbol{h}_k$ for $k \in [K_A]$ and $\overset{b}{\geq}$ uses the constraint that $\frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \left\| \boldsymbol{h}_{k,i} \right\|^2 \leq E_H$. Moreover, we have $\sum_{k=1}^{K_A} \frac{1}{n_k} \sum_{i=1}^{n_k} \left\| \boldsymbol{h}_{k,i} \right\|^2 = E_H$ only if $\boldsymbol{h}_{k,i} = \boldsymbol{0}_p$ for all $k \in [K_A + 1 : K]$. Plugging Eq. (81) into Eq. (80), we obtain $\overset{a}{\geq}$ in Eq. (78).

We then bound $\sum_{k=1}^{K_A} \left\| C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k \right\|^2$. First, we have

$$\frac{1}{K_A} \sum_{k=1}^{K_A} \left\| C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k \right\|^2$$

$$= \frac{1}{K_A} \sum_{k=1}^{K_A} \|\boldsymbol{w}_k\|^2 - 2 \frac{1}{K_A} \sum_{k=1}^{K_A} \boldsymbol{w}_k \cdot (C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B) + \|C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B\|^2$$

$$\overset{a}{=} \frac{1}{K_A} \sum_{k=1}^{K_A} \|\boldsymbol{w}_k\|^2 - 2 C_f^2 \boldsymbol{w}_A^\top \boldsymbol{w}_B - C_e(2 - C_e)\|\boldsymbol{w}_A\|^2 + C_f^2 \|\boldsymbol{w}_B\|^2. \tag{82}$$

where $\overset{a}{=}$ uses $\sum_{k=1}^{K_A} \boldsymbol{w}_k = K_A \boldsymbol{w}_A$. Then using the constraint that $\sum_{k=1}^{K} \|\boldsymbol{w}_k\| \leq K E_W$ yields that

$$\frac{1}{K_A} \sum_{k=1}^{K_A} \|\boldsymbol{w}_k\|^2 - 2 C_f^2 \boldsymbol{w}_A^\top \boldsymbol{w}_B - C_e(2 - C_e)\|\boldsymbol{w}_A\|^2 + C_f^2 \|\boldsymbol{w}_B\|^2 \tag{83}$$

$$\leq \frac{K}{K_A} E_W^2 - \frac{1}{K_A} \sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k\|^2 - C_e(2 - C_f) \left\| \boldsymbol{w}_A + \frac{C_f^2}{C_e(2 - C_e)} \boldsymbol{w}_B \right\|^2 + \left(C_f^2 + \frac{C_f^4}{C_e(2 - C_e)} \right) \|\boldsymbol{w}_B\|^2$$

$$\overset{a}{=} \frac{K}{K_A} E_W^2 - \left(1/K_R - C_f^2 - \frac{C_f^4}{C_e(2 - C_e)} \right) \|\boldsymbol{w}_B\|^2 - \frac{1}{K_A} \sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2,$$

where $\overset{a}{\geq}$ applies $\sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k\|^2 = K_B \|\boldsymbol{w}_B\|^2 + \sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2$. Plugging Eq. (82) and Eq. (83) into $\overset{a}{\geq}$ in Eq. (78), we obtain $\overset{b}{\geq}$ in Eq. (78), completing the proof. $\square$

C. Additional Results.

Comparison of Oversampling and Weighted Adjusting. Oversampling and weight adjusting are two commonly-used tricks in deep learning (11). Both of them actually consider the same objective as Eq. (17), but applies different optimization algorithms to minimize the objective. It was observed that oversampling is more stable than weight adjusting in optimization. As a by product of this work, we compare the two algorithms below and shows that the variance of updates for oversampling will be potentially much smaller than that of weight adjusting. It was well-known in stochastic optimization field that the variance of the updates decides the convergence of an optimization algorithm

(see e.g, ([68](#)–[70](#))). Thus we offer a reasonable justification for the stability of the oversampling technique. We simply consider sampling the training data without replacement. It slightly differs from the deep learning training methods in practice. Besides, we only consider sampling a single data in each update. The analysis can be directly extended to the mini-batch setting.

We first introduce the two methods. The weight adjusting algorithm in each update randomly samples a training data, and updates the parameters $\boldsymbol{W}_{\text{full}}$ by the Stochastic Gradient Descent algorithm as

$$\boldsymbol{W}_{\text{full}}^{t+1} = \boldsymbol{W}_{\text{full}}^{t} - \eta_w \boldsymbol{v}_w^t, \quad t = 0, 1, 2, \ldots, \tag{84}$$

where $\boldsymbol{W}_{\text{full}}^t$ denotes the parameters at iteration step t, η_w is a positive step size, and the stochastic gradient $\boldsymbol{v}_w^t$ satisfies that

$$\boldsymbol{v}_w^t = \begin{cases} \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k), & k \in [K_A], i \in [n_A], \text{ with probability } \frac{1}{K_A n_A + K_B n_B}, \\ w_r \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k), & k \in [K_A + 1 : K_B], i \in [n_B], \text{ with probability } \frac{1}{K_A n_A + K_B n_B}. \end{cases}$$

We have

$$\mathbb{E}\left[\boldsymbol{v}_w^t \mid \boldsymbol{W}_{\text{full}}^t\right] \tag{85}$$

$$= \frac{1}{n_A K_A + n_B K_B} \left[\sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) + w_r \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B} \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right],$$

and

$$\mathbb{E}\left[\|\boldsymbol{v}_w^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] = \frac{1}{n_A K_A + n_B K_B} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \left\| \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right\|^2$$

$$+ \frac{w_r^2}{n_A K_A + n_B K_B} \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B} \left\| \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right\|^2. \tag{86}$$

For the oversampling method, the algorithm in effect duplicates the data by w_r times and runs Stochastic Gradient Descent on the "whole" data. Therefore, the update goes as

$$\boldsymbol{W}_{\text{full}}^{t+1} = \boldsymbol{W}_{\text{full}}^{t} - \eta_s \boldsymbol{v}_s^t, \quad t = 0, 1, 2, \ldots, \tag{87}$$

where $\boldsymbol{v}_s^t$ satisfies that

$$\boldsymbol{v}_s^t = \begin{cases} \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k), & k \in [K_A], i \in [n_A], \text{ with probability } \frac{1}{K_A n_A + K_B w_r n_B}, \\ \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k), & k \in [K_A + 1 : K_B], i \in [n_B], \text{ with probability } \frac{w_r}{K_A n_A + K_B w_r n_B}. \end{cases}$$

We obtain

$$\mathbb{E}\left[\boldsymbol{v}_s^t \mid \boldsymbol{W}_{\text{full}}^t\right] = \frac{1}{n_A K_A + w_r n_B K_B} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k)$$

$$+ \frac{w_r}{n_A K_A + w_r n_B K_B} \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B} \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k),$$

and

$$\mathbb{E}\left[\|\boldsymbol{v}_s^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] = \frac{1}{n_A K_A + w_r n_B K_B} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \left\| \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right\|^2$$

$$+ \frac{w_r}{n_A K_A + w_r n_B K_B} \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B} \left\| \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right\|^2. \tag{88}$$

We suppose the two updates in expectation are in a same scale. That means we assume $\eta_w = \frac{n_A K_A + w_r n_B K_B}{n_A K_A + n_B K_B} \eta_s$. Then $\eta_w \mathbb{E}\left[\boldsymbol{v}_w^t \mid \boldsymbol{W}_{\text{full}}^t\right] = \eta_s \mathbb{E}\left[\boldsymbol{v}_s^t \mid \boldsymbol{W}_{\text{full}}^t\right]$. In fact, if $K_A \asymp 1$, $K_B \asymp 1$, $n_A \gg n_B$, and $1 \ll w_r \lesssim (n_A/n_B)$, we have $\frac{n_A K_A + w_r n_B K_B}{n_A K_A + n_B K_B} \asymp 1$ and so $\eta_w \asymp \eta_s$. Now by comparing Eq. ([86](#)) with Eq. ([88](#)), we obtain that the second moment of $\eta_w \boldsymbol{v}_w^t$ is much smaller than that of $\eta_s \boldsymbol{v}_s^t$ since the order of w_r for the latter is larger by 1. For example, let us assume that all the norms of the gradients are in a same order, i.e., $\left\| \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right\| \asymp a$ for all k and i, where $a > 0$. Then Eq. ([88](#)) implies that $\mathbb{E}\left[\|\eta_s \boldsymbol{v}_s^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] \asymp \eta_s^2 a^2$. However, Eq. ([86](#)) reads that $\mathbb{E}\left[\|\eta_w \boldsymbol{v}_w^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] \asymp \eta_s^2 \frac{n_A K_A + w_r^2 n_B K_B}{n_A K_A + w_r n_B K_B} a^2$. Furthermore, if we set $w_r \asymp n_A/n_B$, then $\mathbb{E}\left[\|\eta_w \boldsymbol{v}_w^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] \asymp \eta_s^2 w_r a^2$. Thus the second moment for $\eta_w \boldsymbol{v}_w^t$ is around w_r times of that for $\eta_s \boldsymbol{v}_s^t$. And this fact also holds for the variance because $\left\| \eta_s \mathbb{E}\left[\boldsymbol{v}_s^t \mid \boldsymbol{W}_{\text{full}}^t\right] \right\| \asymp \eta_s a$ and the property that $\mathbb{E}\|\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}]\|^2 = \mathbb{E}\|\boldsymbol{x}\|^2 - \|\mathbb{E}[\boldsymbol{x}]\|^2$ for any random variable $\boldsymbol{x}$. Therefore, we can conclude that the variance of updates for oversampling is potentially much smaller than that of weight adjusting.

More Discussions on Convex Relaxation and Cross-Entropy Loss.   We show Program Eq. ([7](#)) can also be relaxed as a nuclear norm-constrained convex optimization. The result heavily relies on the progress of matrix decomposition, e.g. ([56](#), [57](#)). We will use the equality (see e.g., ([56](#), Section 2)) that for any matrix $\boldsymbol{Z}$ and $a > 0$,

$$\|\boldsymbol{Z}\|_* = \inf_{r \in \mathbb{N}_+} \inf_{\boldsymbol{U}, \boldsymbol{V} : \boldsymbol{U}\boldsymbol{V}^\top = \boldsymbol{Z}} \frac{a}{2} \|\boldsymbol{U}\|^2 + \frac{1}{2a} \|\boldsymbol{V}\|^2, \tag{89}$$

where r is the number of columns for U and $\|\cdot\|_*$ denotes the nuclear norm.

For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ for the original program Eq. (7), we define

$$\boldsymbol{h}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \boldsymbol{h}_{k,i}, \; k \in [K], \quad \tilde{\boldsymbol{H}} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_K] \in \mathbb{R}^{p \times K}, \;\; \text{and} \;\; \boldsymbol{Z} = \boldsymbol{W}\tilde{\boldsymbol{H}} \in \mathbb{R}^{K \times K}. \tag{90}$$

We consider the convex program:

$$\min_{\boldsymbol{Z} \in \mathbb{R}^{K \times K}} \quad \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}(\boldsymbol{Z}_k, \boldsymbol{y}_k) \tag{91}$$

$$\text{s.t.} \quad \|\boldsymbol{Z}\|_* \leq K \sqrt{E_H E_W}.$$

where $\boldsymbol{Z}_k$ denotes the k-th column of $\boldsymbol{Z}$ for $k \in [K]$.

Lemma 7. *Assume $p \geq K$ and the loss function $\mathcal{L}$ is convex on the first argument. Let $\boldsymbol{Z}^\star$ be a minimizer of the convex program Eq. (91). Let r be the rank of $\boldsymbol{Z}^\star$ and consider thin Singular Value Decomposition (SVD) of $\boldsymbol{Z}^\star$ as $\boldsymbol{Z}^\star = \boldsymbol{U}^\star \boldsymbol{\Sigma}^\star \boldsymbol{V}^\star$. Introduce two diagonal matrices $\boldsymbol{\Sigma}_1^\star$ and $\boldsymbol{\Sigma}_2^\star$ with the entries defined as $\boldsymbol{\Sigma}_1^\star(i,i) = \sqrt{\frac{E_W}{E_H}} \sqrt{|\boldsymbol{\Sigma}^\star(i,i)|}$ and $\boldsymbol{\Sigma}_2^\star(i,i) = \sqrt{\frac{E_H}{E_W}} \boldsymbol{\Sigma}^\star(i,i)/\sqrt{|\boldsymbol{\Sigma}^\star(i,i)|}$ for $i \in [r]$, respectively. Let $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ be*

$$\boldsymbol{W} = \boldsymbol{U}^\star \boldsymbol{\Sigma}_1^\star \boldsymbol{P}^\top, \quad [\boldsymbol{h}_1^\star, \boldsymbol{h}_2^\star, \ldots, \boldsymbol{h}_K^\star] = \boldsymbol{P}\boldsymbol{\Sigma}_2^\star \boldsymbol{V}^\star,$$
$$\boldsymbol{h}_{k,i}^\star = \boldsymbol{h}_k^\star, \quad k \in [K], \; i \in [n_k], \tag{92}$$

where $\boldsymbol{P} \in \mathbb{R}^{p \times r}$ is any partial orthogonal matrix such that $\boldsymbol{P}^\top \boldsymbol{P} = \boldsymbol{I}_r$. Then $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a minimizer of Eq. (7).

Proof of Lemma 7. For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ for the original program Eq. (7), define $\boldsymbol{h}_k$ for $k \in [K]$, $\tilde{\boldsymbol{H}}$, and $\boldsymbol{Z}$ by Eq. (90). We show $\boldsymbol{Z}$ is a feasible solution for the convex program Eq. (91). In fact, by Eq. (89) with $r = K$ and $a = \sqrt{E_H/E_W}$, we have

$$\begin{aligned}
\|\boldsymbol{Z}\|_* &\leq \frac{\sqrt{E_H/E_W}}{2} \|\boldsymbol{W}\|^2 + \frac{\sqrt{E_W/E_H}}{2} \left\|\tilde{\boldsymbol{H}}\right\|^2 \\
&\stackrel{a}{\leq} \frac{\sqrt{E_H/E_W}}{2} \sum_{k=1}^{K} \|\boldsymbol{w}_k\|^2 + \frac{\sqrt{E_W/E_H}}{2} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \left\|\boldsymbol{h}_{k,i}\right\|^2 \\
&\leq K \sqrt{E_H E_W},
\end{aligned} \tag{93}$$

where $\stackrel{a}{\leq}$ applies Jensen's inequality as:

$$\left\|\tilde{\boldsymbol{H}}\right\|^2 = \sum_{k=1}^{K} \|\boldsymbol{h}_k\|^2 \leq \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} \left\|\boldsymbol{h}_{k,i}\right\|^2.$$

Let L_0 be the global minimum of the convex problem Eq. (91). Since $\mathcal{L}$ is convex on the first argument, by the same argument as Eq. (64), we obtain, for any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$,

$$\begin{aligned}
\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) &= \sum_{k=1}^{K} \frac{n_k}{N} \left[\frac{1}{n_k} \sum_{k=1}^{n_k} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \right] \\
&\geq \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_k, \boldsymbol{y}_k) = \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}(\boldsymbol{Z}_k, \boldsymbol{y}_k) \geq L_0.
\end{aligned} \tag{94}$$

On the other hand, for the solution $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ defined in Eq. (92) with $\boldsymbol{Z}^\star$, we can verify that $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a feasible solution for Eq. (7) and

$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}(\boldsymbol{W}^\star \boldsymbol{h}_{k,i}^\star, \boldsymbol{y}_k) = \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}(\boldsymbol{Z}_k^\star, \boldsymbol{y}_k) = L_0. \tag{95}$$

Combining Eq. (94) and Eq. (95), we have that L_0 is the global minimum of Eq. (7) and $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a minimizer. $\square$

Property 1. *For the cross-entropy loss, we have the following properties.*

(i) Any minimizer $\boldsymbol{Z}^\star$ of Eq. (91) satisfies that $\|\boldsymbol{Z}\|_* = \sqrt{E_H E_W}$.

(ii) Any minimizer $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ of Eq. (7) satisfies

$$\frac{1}{K} \sum_{k=1}^{K} \frac{1}{n} \sum_{i=1}^{n} \left\|\boldsymbol{h}_{k,i}^\star\right\|^2 = E_H, \quad and \quad \frac{1}{K} \sum_{k=1}^{K} \left\|\boldsymbol{w}_k^\star\right\|^2 = E_W.$$

(iii) Any minimizer $\boldsymbol{X}^\star$ of Eq. (15) satisfies that

$$\frac{1}{K} \sum_{k=1}^{K} \boldsymbol{X}^\star(k,k) = E_H, \quad and \quad \frac{1}{K} \sum_{k=K+1}^{2K} \boldsymbol{X}^\star(k,k) = E_W.$$
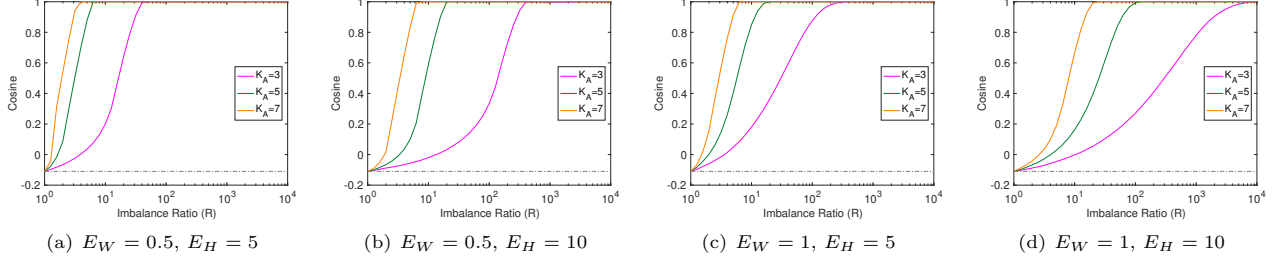
Fig. 8. The average cosine of the angles between any pair of the minority classifier solved from the Layer-Peeled Model. The average cosine reaches 1 once R is above some threshold. The total number of classes $K_A + K_B$ is fixed to 10. The gray dash-dotted line indicates the value of $-\frac{1}{K-1}$, which is given by Eq. (10).

Proof of Property 1. We first prove (i). Let $\boldsymbol{Z}^\star$ be any minimizer of Eq. (91). Then by the Karush–Kuhn–Tucker conditions, there is a pair $(\lambda, \boldsymbol{\xi})$ with $\lambda \geq 0$ and $\boldsymbol{\xi} \in \partial \|\boldsymbol{Z}^\star\|_*$ such that

$$\nabla_{\boldsymbol{Z}}\left[\sum_{k=1}^K \frac{n_k}{N}\mathcal{L}(\boldsymbol{Z}_k^\star, \boldsymbol{y}_k)\right] + \lambda\boldsymbol{\xi} = \boldsymbol{0}^{K\times K},$$

where $\partial\|\boldsymbol{Z}\|_*$ denotes the set of sub-gradient of $\|\boldsymbol{Z}\|_*$. For the cross-entropy loss, one can verify that $\nabla_{\boldsymbol{Z}}\left[\sum_{k=1}^K \frac{n_k}{N}\mathcal{L}(\boldsymbol{Z}_k, \boldsymbol{y}_k)\right] \neq \boldsymbol{0}^{K\times K}$ for all $\boldsymbol{Z}$. So $\lambda \neq 0$. By the complementary slackness condition, we have that $\boldsymbol{Z}$ will reach the boundary of the constraint, achieving (i).

For (ii), suppose there is a minimizer $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ of Eq. (7) such that $\frac{1}{K}\sum_{k=1}^K \frac{1}{n}\sum_{i=1}^n \left\|\boldsymbol{h}_{k,i}^\star\right\|^2 < E_H$ or $\frac{1}{K}\sum_{k=1}^K \left\|\boldsymbol{w}_k^\star\right\|^2 < E_W$. Letting $\boldsymbol{Z}^\star$ defined by Eq. (90), it follows from Eq. (94) that $\boldsymbol{Z}^\star$ is a minimizer of Eq. (91). However, by Eq. (93), we have $\|\boldsymbol{Z}^\star\|_* < \sqrt{E_H E_W}$, which is contradictory to (i). We obtain (ii).

For (iii), suppose there is a minimizer $\boldsymbol{X}^\star$ of Eq. (15) such that $\frac{1}{K}\sum_{k=1}^K \boldsymbol{X}^\star(k,k) < E_H$ or $\frac{1}{K}\sum_{k=K+1}^{2K}\boldsymbol{X}^\star(k,k) < E_W$. Then letting $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ defined by Eq. (16), $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a minimizer of Eq. (7) from Theorem 1. However, we have $\frac{1}{K}\sum_{k=1}^K \frac{1}{n}\sum_{i=1}^n \left\|\boldsymbol{h}_{k,i}^\star\right\|^2 < E_H$ or $\frac{1}{K}\sum_{k=1}^K \left\|\boldsymbol{w}_k^\star\right\|^2 < E_W$, which contradicts to (ii). We complete the proof. $\square$

D. Additional Experimental Results. In this part, we provide some additional experimental results for Minority Collapse. As for the experiments for Minority Collapse in Figure 4, the corresponding training and test accuracy are shown in Tables 3-4. Furthermore, we find that the pre-trained neural networks on ImageNet (an imbalanced dataset with $K = 1000$ classes) that are officially released by Pytorch[¶¶] also do not converge to a Simplex ETF, indicating that neural collapse does not emerge during the terminal phase of imbalanced training. Specifically, the minimal (maximal) between-class angle of pre-trained classifiers for VGG19 and ResNet152 are $43°$ ($103°$) and $37°$ ($102°$), respectively. The corresponding standard deviation of between-class angles of pre-trained classifiers for VGG19 and ResNet152 are $4.1°$ and $3.6°$, respectively. More details can be found in Figure 9. The phase transition point of the imbalance ratio is in Figure 8 with multiple choices of E_W and E_H.

Dataset	FashionMNIST						CIFAR10					
Network architecture	VGG11			ResNet18			VGG13			ResNet18		
No. of majority classes	$K_A = 3$	$K_A = 5$	$K_A = 7$	$K_A = 3$	$K_A = 5$	$K_A = 7$	$K_A = 3$	$K_A = 5$	$K_A = 7$	$K_A = 3$	$K_A = 5$	$K_A = 7$
$R = 1$	100	100	100	100	100	100	100	100	100	100	100	100
$R = 10$	100	100	100	100	100	100	100	100	100	100	100	100
$R = 100$	100	100	100	100	100	100	100	100	100	100	100	100
$R = 1000$	99.87	99.94	99.97	99.97	99.93	99.97	99.80	99.90	99.96	99.90	100	99.97
$R = inf$	100	100	100	100	100	100	100	100	100	100	100	100

Table 3. Training accuracy (%) for different settings.

Dataset	FashionMNIST						CIFAR10					
Network architecture	VGG11			ResNet18			VGG13			ResNet18		
No. of majority classes	$K_A = 3$	$K_A = 5$	$K_A = 7$	$K_A = 3$	$K_A = 5$	$K_A = 7$	$K_A = 3$	$K_A = 5$	$K_A = 7$	$K_A = 3$	$K_A = 5$	$K_A = 7$
$R = 1$	93.02	93.02	93.02	93.80	93.80	93.80	88.62	88.62	88.62	88.72	88.72	88.72
$R = 10$	87.12	89.79	92.00	86.07	88.77	92.78	65.55	71.80	80.41	58.66	66.44	78.79
$R = 100$	73.48	85.00	88.03	70.82	84.62	86.24	30.87	48.36	64.52	28.90	45.97	62.91
$R = 1000$	40.10	57.61	69.09	45.51	57.95	66.13	28.48	45.44	61.82	28.57	45.10	60.89
$R = inf$	29.39	47.61	63.86	29.44	47.72	64.59	28.31	44.87	61.40	28.44	45.27	61.16

Table 4. Test accuracy (%) for different settings.

¶¶ https://pytorch.org/vision/stable/models.html.
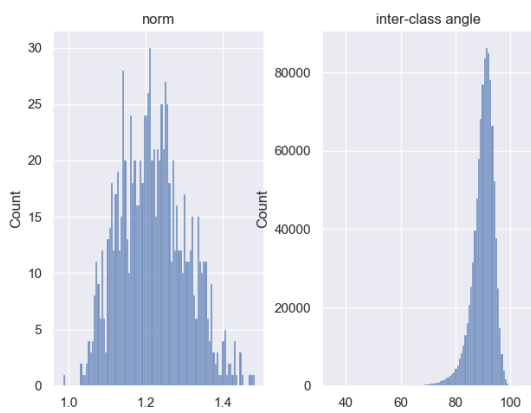
(a) Pre-trained VGG19 on ImageNet   (b) Pre-trained ResNet152 on ImageNet
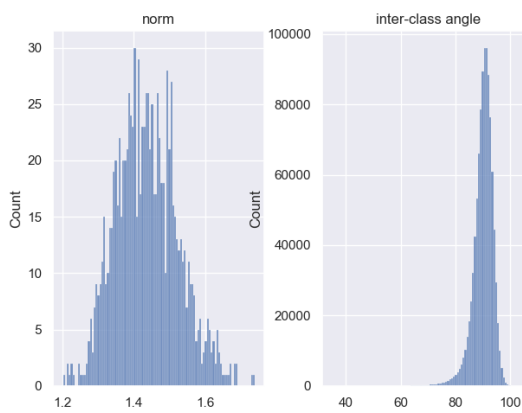
Fig. 9. The neural networks that are pre-trained on ImageNet by PyTorch do not converge to a Simplex ETF.