Layer-Peeled Model: Toward Understanding Well-Trained Deep Neural Networks

Cong Fang* Hangfeng He† Qi Long‡ Weijie J. Su§

University of Pennsylvania

January 26, 2021

Abstract

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 in deep learning training [PHD20]. Moreover, 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 its confirmation by our computational experiments.

Contents

*fangcong@pku.edu.cn
†hangfeng@seas.upenn.edu
‡qlong@upenn.edu
§suw@wharton.upenn.edu

1 Introduction

In the past decade, deep learning has achieved remarkable performance across a range of scientific and engineering domains [KSH17, LBH15, SHM+16]. 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 also 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[1]

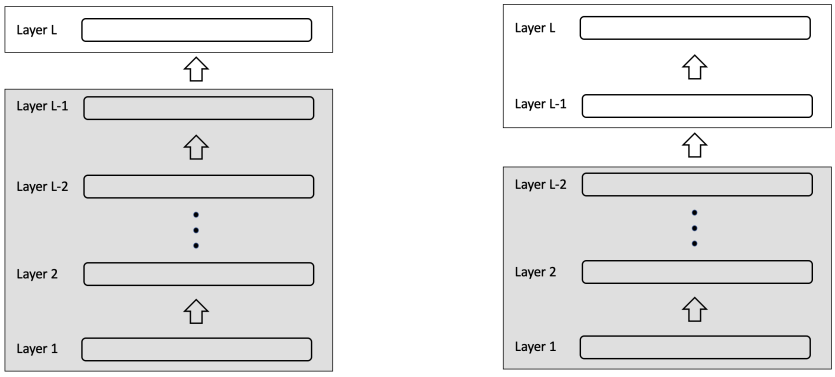
$$\boldsymbol{f}(\boldsymbol{x}; \boldsymbol{W}_{\text{full}}) = \boldsymbol{b}_L + \boldsymbol{W}_L \sigma\left(\boldsymbol{b}_{L-1} + \boldsymbol{W}_{L-1}\sigma(\cdots \sigma(\boldsymbol{b}_1 + \boldsymbol{W}_1 \boldsymbol{x}))\right).$$

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.[2] Owing to the complex and nonlinear interaction between the L layers, when applying stochastic gradient descent to the optimization problem

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

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

[2] The last-layer weights, $\boldsymbol{W}_L$, consist of K vectors that correspond to the K classes.



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

Figure 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.

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 $\boldsymbol{y}_k$,[3] $N = n_1 + \cdots + n_K$ 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, an 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 Program (1) with the goal of unveiling quantitative patterns of deep neural networks:

$$\begin{aligned} \min_{\boldsymbol{W}_L, \boldsymbol{H}} \quad & \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\mathcal{L}(\boldsymbol{W}_L\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \\ \text{subject to} \quad & \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 \le E_W, \\ & \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i=1}^{n_k}\|\boldsymbol{h}_{k,i}\|^2 \le E_H, \end{aligned} \tag{2}$$

where $\boldsymbol{W}_L = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K]^\top \in \mathbb{R}^{K\times p}$ is, as in Program (1), comprised of K linear classifiers in the last layer, $\boldsymbol{H} = [\boldsymbol{h}_{k,i} : 1 \le k \le K, 1 \le i \le n_k] \in \mathbb{R}^{p\times N}$ corresponds to the p-dimensional last-layer activations/features of all N training examples,[4] and E_H and E_W are two positive scalars. Although still nonconvex, this new optimization program is presumably much more amenable to analysis than the old one (1) as the interaction now is only between two layers.

In relating Program (2) to the optimization problem (1), a first simple observation is that $\boldsymbol{f}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}) = \boldsymbol{W}_L\sigma(\boldsymbol{W}_{L-1}\sigma(\cdots\sigma(\boldsymbol{W}_1\boldsymbol{x}_{k,i})))$ in (1) is replaced by $\boldsymbol{W}_L\boldsymbol{h}_{k,i}$ in (2), where we ignore

[3] We often encode $\boldsymbol{y}_k$ as a K-dimensional one-hot vector with 1 in the k-th entry.

[4] Strictly speaking, $\boldsymbol{H}$ is used to model the activations from the $L-1$ layer. Note that the dimension of the vector $\boldsymbol{w}_k$ is also p.

the bias terms for simplicity. Put differently, the black-box nature of the last-layer features, namely $\sigma(\boldsymbol{W}_{L-1}\sigma(\cdots\sigma(\boldsymbol{W}_1\boldsymbol{x}_{k,i})))$, 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 via *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 [PHD20], who discovered a mathematically elegant and pervasive phenomenon, termed neural collapse, through massive deep learning experiments on datasets with balanced classes. This top-down approach was also taken in [WL90, SHN$^+$18, OS20, YCY$^+$20, Sha20] to investigate various aspects of deep learning models.

1.1 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 even 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 Program (1).

Balanced Data. Roughly speaking, neural collapse refers to the emergence of certain geometric patterns of the last-layer features $\sigma(\boldsymbol{W}_{L-1}\sigma(\cdots\sigma(\boldsymbol{W}_1\boldsymbol{x}_{k,i})))$ and the last-layer classifiers $\boldsymbol{W}_L$, when the neural network is well-trained on a balanced dataset in the sense that it is toward not only zero misclassification error but also negligible[5] cross-entropy loss [PHD20]. Specifically, the authors experimentally observed the following properties: 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 1.2.

While it seems hopeless to rigorously prove neural collapse for multiple-layer neural networks (1) at the moment, alternatively, we seek to show that this phenomenon emerges in the surrogate model (2). More precisely, when the size of each class $n_k = n$ for all k, is it true that any global minimizer $\boldsymbol{W}_L^\star = [\boldsymbol{w}_1^\star, \ldots, \boldsymbol{w}_K^\star]^\top, \boldsymbol{H}^\star = [\boldsymbol{h}_{k,i}^\star : 1 \le k \le K, 1 \le i \le n]$ of Program (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., [CKNH20]). As an immediate implication, this result provides evidence of the Layer-Peeled Model's ability to characterize well-trained deep learning models.

[5]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 Program (1) is bounded away from 0.

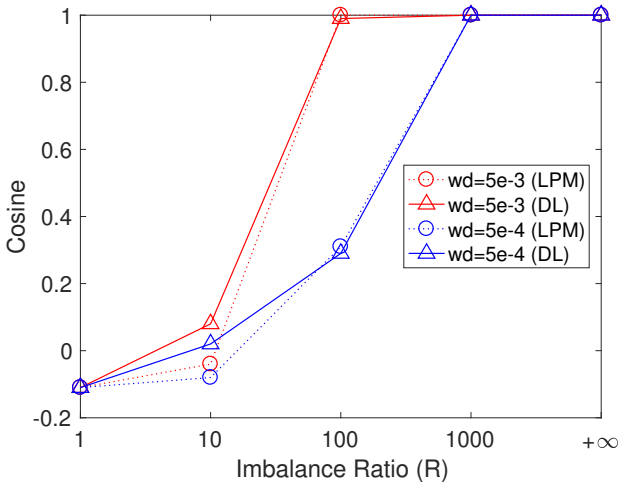


Figure 2: Minority Collapse predicted by the Layer-Peeled Model (LPM, in dotted lines) and empirically observed in deep learning (DL, in solid lines) on imbalanced datasets with $K_A = 7$ and $K_B = 3$. The y-axis denotes the average cosine of the angles between any pair of the minority classifier $\boldsymbol{w}_{K_A+1}^\star, \ldots, \boldsymbol{w}_K^\star$ for both LPM and DL. The datasets we use are subsets of the CIFAR10 datasets [Kri09] and the size of the majority classes is fixed to 5000. The experiments use VGG13 [SZ14] 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 4.3.

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 [JK19], 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 (2) and the asymmetry due to the imbalance in class sizes.

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 (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., [Yar17]). 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.

1.2 Related Work

There is a venerable line of work attempting to gain insights into deep learning from a theoretical point of view [JGH18, DLL$^+$19, AZLS19, ZCZG18, COB19, EMW19, BFT17, HS20, PBL20, MMN18, SS19, RVE18, FLYZ20, KWL$^+$19, SSJ20]. See also the reviews [FDZ21, HT20, FMZ19, Sun19] and references therein.

The work of neural collapse by [PHD20] in this body of work is particularly noticeable with its mathematically elegant and convincing insights. In brief, [PHD20] observed the following four properties of the last-layer features and classifiers in deep learning training on balanced datasets:[6]

> (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 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 [SH03, PHD20].

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

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

where $\boldsymbol{I}_K \in \mathbb{R}^{K\times K}$ is the identity matrix, $\mathbf{1}_K$ is the ones vector, and $\boldsymbol{P} \in \mathbb{R}^{p\times K}$ $(p \geq K)$[7] 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, [PHD20] 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 [MBB18, BHMM19, LR20, BLLT20, LSS20]) during the terminal phase of training—when the

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

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

trained model interpolates the in-sample training data. In passing, we remark that while preparing the manuscript, we became aware of [MPP20, EW20, LS20], which produced neural collapse using different models.

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 (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 rewriting (1) as

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

where the last-layer feature function $\boldsymbol{h}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{-L}) := \sigma(\boldsymbol{W}_{L-1}\sigma(\cdots \sigma(\boldsymbol{W}_1 \boldsymbol{x}_{k,i})))$ 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

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

for some positive numbers C_1 and C_2.[8] To clear up any confusion, note that due to its nonconvexity, (3) may admit multiple global minima and each in general corresponds to different values of C_1, C_2. Next, we can equivalently write (4) as

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

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

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

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

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

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 (5), we obtain the following optimization program, which we call the Layer-Peeled Model:

$$\begin{aligned} \min_{\boldsymbol{W},\boldsymbol{H}} \quad & \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i},\boldsymbol{y}_k) \\ \text{s.t.} \quad & \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 \le E_W, \\ & \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i=1}^{n_k}\|\boldsymbol{h}_{k,i}\|^2 \le E_H. \end{aligned} \tag{7}$$

For simplicity, above and henceforth we write $\boldsymbol{W} := \boldsymbol{W}_L \equiv [\boldsymbol{w}_1,\dots,\boldsymbol{w}_K]^\top$ 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 (7) uses the ℓ_q norm for any $q \neq 2$ and $q > 1$, in place of the ℓ_2 norm. This fact in turn justifies in part the ansatz (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—meaning $n_k = n$ for all classes $1 \le k \le K$—and our main finding is that the Layer-Peeled Model displays the neural collapse phenomenon, just as in deep learning training [PHD20]. The proofs are all deferred to Appendix A.1. Throughout this section, we assume $p \ge K-1$ unless otherwise specified. This assumption is satisfied in many popular architectures, where p is usually tens or hundreds of times of K.

3.1 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

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

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 (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 (7) with the cross-entropy loss admits an almost closed-form expression.

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

$$\boldsymbol{h}_{k,i}^\star = C\boldsymbol{w}_k^\star = C'\boldsymbol{m}_k^\star \tag{8}$$

for all $1 \le i \le n, 1 \le k \le K$*, 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 (note the rotation $\boldsymbol{P}$ 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, (8) says that all within-class (last-layer) features are the same: $\boldsymbol{h}_{k,i}^\star = \boldsymbol{h}_{k,i'}^\star$ for all $1 \le i, i' \le n$; 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

$$\cos \measuredangle(\boldsymbol{h}_k^\star, \boldsymbol{h}_{k'}^\star) = -\frac{1}{K-1} \tag{9}$$

for any $k \neq k'$ by Definition 1;[9] in addition, (8) 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 [MPP20, EW20, LS20] (see also [PL20]). For example, [EW20, LS20] considered models that impose a norm constraint for each individual class, rather than an overall constraint as employed in the Layer-Peeled Model.

3.2 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.

[9] 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 [PHD20], the largest angle implies a large-margin solution [SHN+18].

Contrastive Loss. Contrastive losses have been extensively used recently in both supervised and unsupervised deep learning [PSM14, AKK+19, CKNH20, BZMA20]. These losses pull similar training examples together in their embedding space while pushing apart dissimilar examples. Here we consider the supervised contrastive loss [KTW+20], which (in the balanced) case is defined through the last-layer features as

$$\mathcal{L}_c(\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) = \frac{1}{n}\sum_{j=1}^{n} -\log\left(\frac{\exp(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^{K}\sum_{\ell=1}^{n}\exp(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k',\ell}/\tau)}\right), \tag{10}$$

where $\tau > 0$ is a parameter. As the loss does not involve the last-layer classifiers explicitly, the Layer-Peeled Model in this case takes the form[10]

$$\begin{aligned} \min_{\boldsymbol{H}} \quad & \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n}\mathcal{L}_c(\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \\ \text{s.t.} \quad & \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{h}_{k,i}\|^2 \le E_H. \end{aligned} \tag{11}$$

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* (11) *satisfies*

$$\boldsymbol{h}_{k,i}^{\star} = \sqrt{E_H}\boldsymbol{m}_k^{\star} \tag{12}$$

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

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 (10), 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 function, both defined on $(0, 1)$. We consider a softmax-based loss function that takes the form

$$\mathcal{L}(\boldsymbol{z}, \boldsymbol{y}_k) = g_1\left(\boldsymbol{S}(\boldsymbol{z})(k)\right) + \sum_{k'=1,\ k'\neq k}^{K} g_2\left(\boldsymbol{S}(\boldsymbol{z})(k')\right). \tag{13}$$

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 [PGM+19].

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

[10] In (10), $\boldsymbol{h}_{k,i} \equiv \boldsymbol{h}(\boldsymbol{x}_{k,i}, \boldsymbol{W}_{-L})$ depends on the data, whereas in (11) $\boldsymbol{h}_{k,i}$'s form the decision variable $\boldsymbol{H}$.

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* (13), $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ *given by* (8) *is a global minimizer of Program* (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* (8).

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 (8). 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 (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 Appendix A.1 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 [WLW$^+$16, HLLT16, MR17]. For example, in the Places2 challenge dataset [ZKL$^+$16], the number of images in its majority scene categories is about eight times that in its minority classes. Another example is the Ontonotes dataset for part-of-speech tagging [HMP$^+$06], 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., [JK19]), 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 short, 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 Appendix A.2.

4.1 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 (7), relying on the well-known result for relaxing a quadratically constrained quadratic program as a semidefinite program (see, e.g., [SZ03]). 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:[11]

$$\begin{aligned}\min_{\boldsymbol{X}\in\mathbb{R}^{2K\times 2K}}\quad & \sum_{k=1}^{K}\frac{n_k}{N}\mathcal{L}(\boldsymbol{z}_k,\boldsymbol{y}_k)\\ \text{s.t.}\quad & \boldsymbol{z}_k=[\boldsymbol{X}(k,K+1),\boldsymbol{X}(k,K+2),\ldots,\boldsymbol{X}(k,2K)\,]^\top,\quad \text{for all } 1\leq k\leq K,\\ & \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{X}(k,k)\leq E_H,\quad \frac{1}{K}\sum_{k=K+1}^{2K}\boldsymbol{X}(k,k)\leq E_W,\\ & \boldsymbol{X}\succeq 0.\end{aligned}\tag{14}$$

Lemma 1 below relates the solutions of (14) to that of (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* (14)*. Define $(\boldsymbol{H}^\star,\boldsymbol{W}^\star)$ as*

$$\begin{aligned}&\left[\boldsymbol{h}_1^\star,\boldsymbol{h}_2^\star,\ldots,\boldsymbol{h}_K^\star,\ (\boldsymbol{W}^\star)^\top\right]=\boldsymbol{P}(\boldsymbol{X}^\star)^{1/2},\\ &\boldsymbol{h}_{k,i}^\star=\boldsymbol{h}_k^\star,\quad \textit{for all } 1\leq i\leq n, 1\leq k\leq K,\end{aligned}\tag{15}$$

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 (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* (7) *are in the form of* (15)*.*

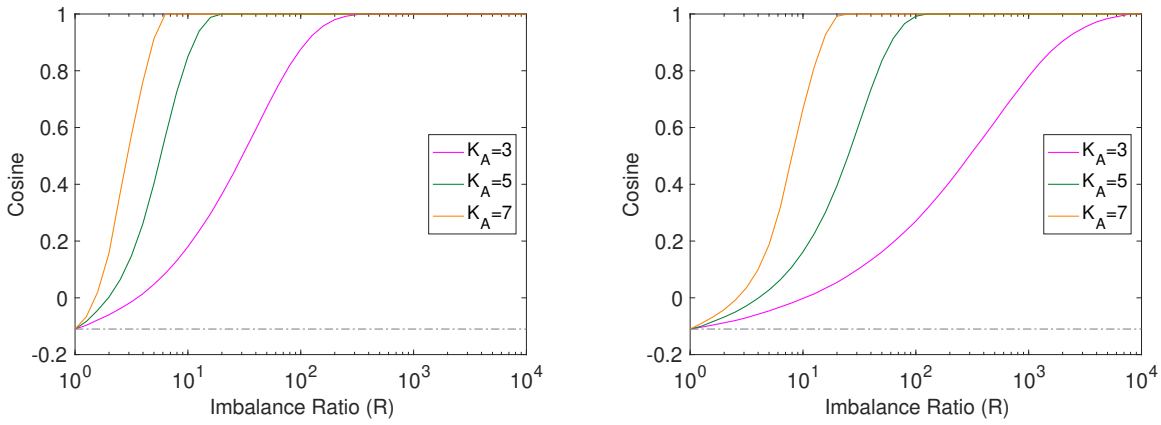
This lemma in effect says that the relaxation does *not* lead to any loss of information when we study the Layer-Peeled Model through a convex program, thereby offering a computationally efficient tool for gaining insights into the terminal phase of training deep neural networks on imbalanced datasets. An appealing feature is that the size of the program (14) 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 Section B).

The assumption of the convexity of $\mathcal{L}$ in the first argument is satisfied by a large class of loss functions, such as the cross-entropy loss. We also remark that (14) is not the unique convex relaxation. An alternative is to relax (7) via a nuclear norm-constrained convex program [BMP08, HV19] (see more details in Section B).

[11] Although Program (14) involves a semidefinite constraint, it is not a semidefinite program in the strict sense because a semidefinite program uses a linear objective function.

4.2 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 nontrivial patterns in this regime. 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) $E_W = 1$, $E_H = 10$

Figure 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 (9). The between-majority-class angles can still be large even when Minority Collapse emerges. For example, in the case $K_A = 5$ in Plot (a), the average between-majority-class cosine is -0.17, which corresponds to 100°, when Minority Collapse first occurs. Notably, our simulation suggests that the minority classifiers exhibit an equiangular frame and so do the majority classifiers.

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 (14). This figure reveals a two-phase behavior of the minority classifiers $\boldsymbol{w}^\star_{K_A+1}, \boldsymbol{w}^\star_{K_A+2}, \ldots, \boldsymbol{w}^\star_K$ 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.

We refer to the phenomenon that appears in the second phase as Minority Collapse. While it can be expected that the minority classifiers get 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, and 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 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 (7) with the cross-entropy loss. As $R \equiv n_A/n_B \to \infty$, we have*

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

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.

4.3 Experiments

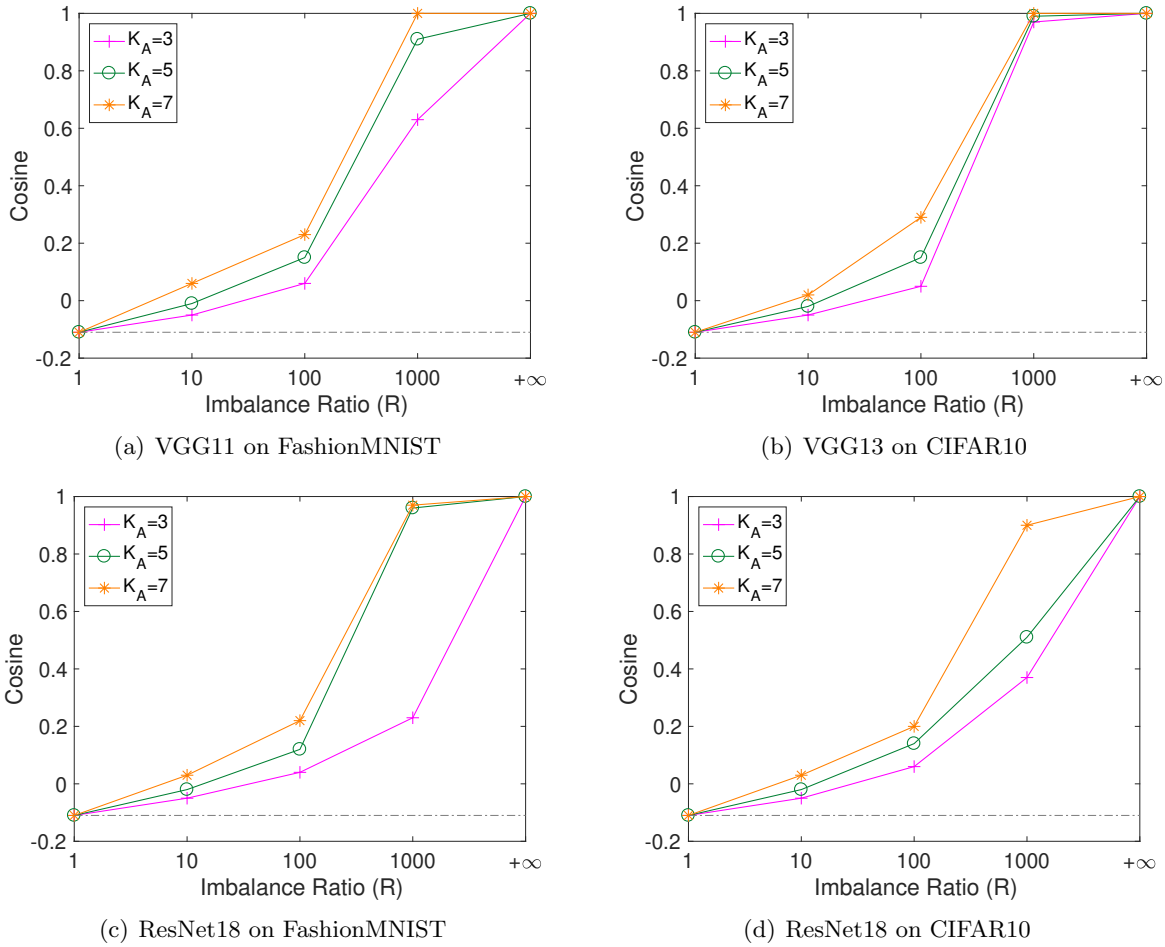
At the moment, Minority Collapse is merely a prediction of the Layer-Peeled Model. An immediate question thus is: does this phenomenon really occur in real-world neural networks? At first glance, it does not necessarily have to be the case since the Layer-Peeled Model is a dramatic simplification of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

To this end, we resort to computational experiments.[12] Explicitly, we consider training two network architectures, VGG and ResNet [HZRS16], on the FashionMNIST [XRV17] and CIFAR10 datasets, and in particular, replace the dropout layers in VGG with batch normalization [IS15]. 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 [PHD20]. 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 [PHD20] 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.

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,

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

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.



(a) VGG11 on FashionMNIST

(b) VGG13 on CIFAR10

(c) ResNet18 on FashionMNIST

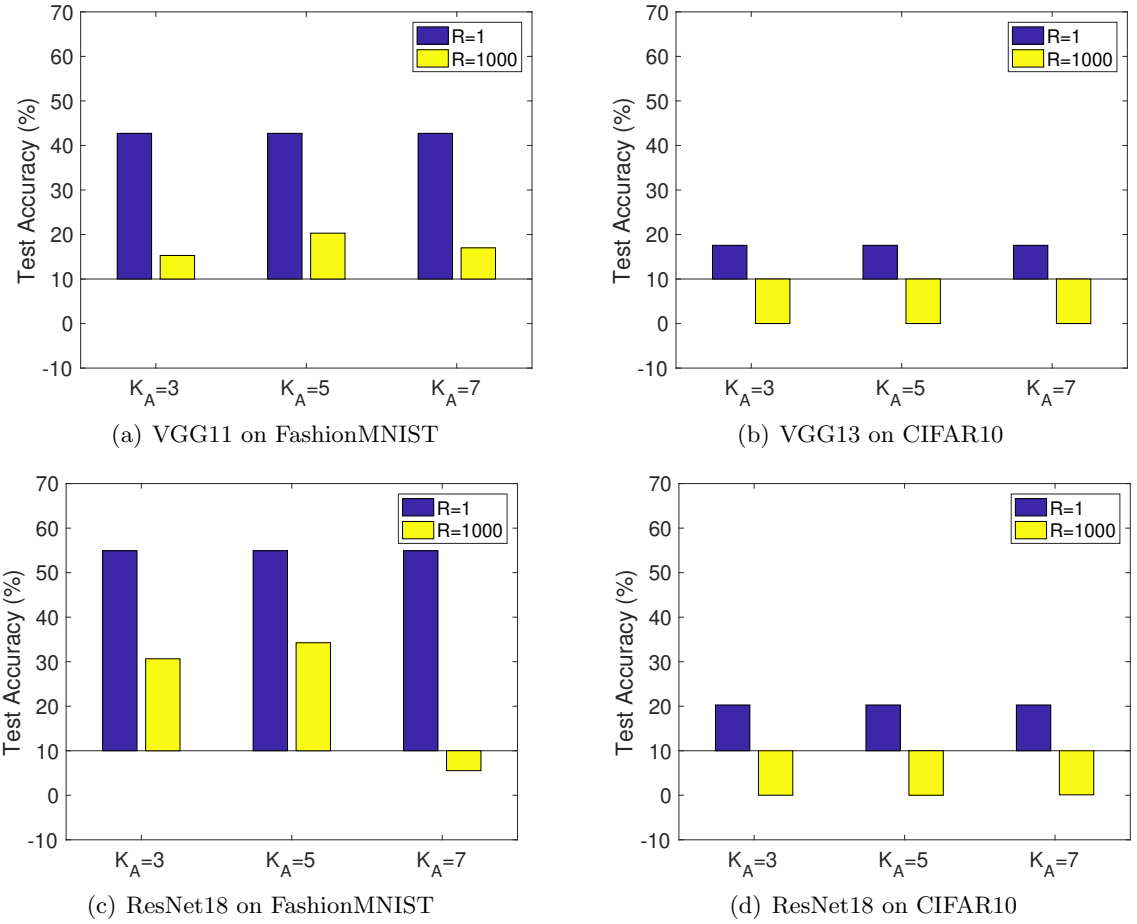
(d) ResNet18 on CIFAR10

Figure 4: The 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.

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 if 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.

Dataset	FashionMNIST					
Network architecture	VGG11			ResNet18		
No. of majority classes	$K_A = 3$	$K_A = 5$	$K_A = 7$	$K_A = 3$	$K_A = 5$	$K_A = 7$
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}
Dataset	CIFAR10					
Network architecture	VGG13			ResNet18		
No. of majority classes	$K_A = 3$	$K_A = 5$	$K_A = 7$	$K_A = 3$	$K_A = 5$	$K_A = 7$
Norm variation	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 $\text{Std}(\|\boldsymbol{w}_B^\star\|)/\text{Avg}(\|\boldsymbol{w}_B^\star\|)$, where $\text{Std}(\|\boldsymbol{w}_B^\star\|)$ 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

Figure 5: Comparison of the test accuracy on the minority classes between $R = 1$ and $R = 1000$. We fix $K_A + K_B = 10$. 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.

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 training process where Minority Collapse occurs is a departure from the terminal phase of training of neural collapse. While this fact seems to contradict conventional wisdom on the approximation power of deep learning, it is important to recognize 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., [PHD20]). 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 (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 focus is on the evaluation of some simple techniques used for imbalanced datasets.

Among many approaches to handling class imbalance in deep learning (see the review [JK19]), perhaps the most popular one is to oversample training examples from the minority classes [BMM18, SXY+19, CJL+19, CWG+19]. 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 Program (1) with

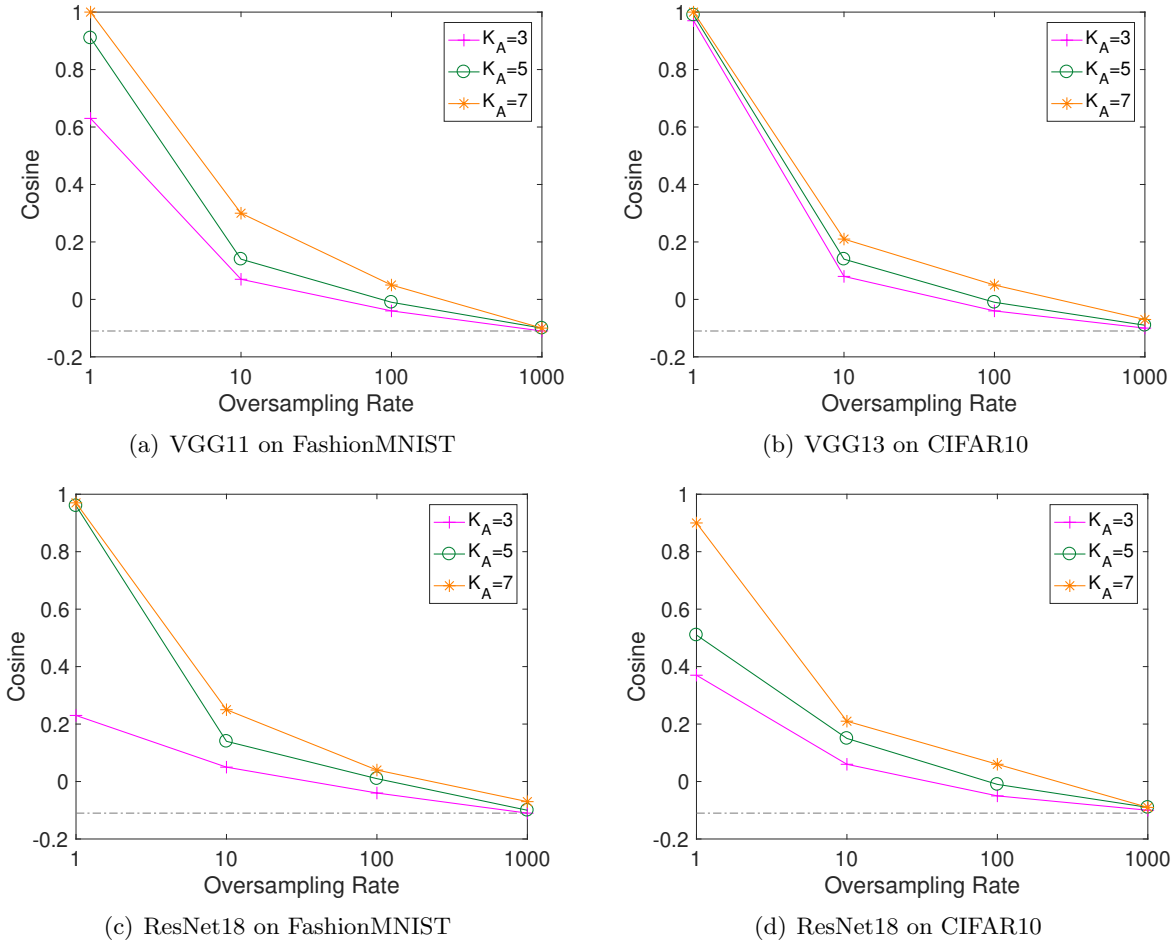
$$\frac{1}{n_A K_A + w_r n_B K_B} \left[\sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}), \boldsymbol{y}_k) + w_r \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B} \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}), \boldsymbol{y}_k) \right] \tag{16}$$

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 Section B).

A close look at (16) 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:

$$\begin{aligned}\min_{\boldsymbol{H},\boldsymbol{W}} \quad & \frac{1}{n_A K_A + w_r n_B K_B}\left[\sum_{k=1}^{K_A}\sum_{i=1}^{n_A}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i},\boldsymbol{y}_k) + w_r\sum_{k=K_A+1}^{K}\sum_{i=1}^{n_B}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i},\boldsymbol{y}_k)\right] \\ \text{s.t.} \quad & \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 \le E_W, \\ & \frac{1}{K}\sum_{k=1}^{K_A}\frac{1}{n_A}\sum_{i=1}^{n_A}\|\boldsymbol{h}_{k,i}\|^2 + \frac{1}{K}\sum_{k=K_A+1}^{K}\frac{1}{n_B}\sum_{i=1}^{n_B}\|\boldsymbol{h}_{k,i}\|^2 \le E_H.\end{aligned} \tag{17}$$



(a) VGG11 on FashionMNIST

(b) VGG13 on CIFAR10

(c) ResNet18 on FashionMNIST

(d) ResNet18 on CIFAR10

Figure 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 $p \geq 2K$ and the loss function $\mathcal{L}$ is convex in the first argument. Let $\boldsymbol{X}^\star$ be any minimizer of the convex program* (14) *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*

$$\begin{aligned} &\left[\boldsymbol{h}_1^\star, \boldsymbol{h}_2^\star, \ldots, \boldsymbol{h}_K^\star, (\boldsymbol{W}^\star)^\top\right] = \boldsymbol{P}(\boldsymbol{X}^\star)^{1/2}, \\ &\boldsymbol{h}_{k,i}^\star = \boldsymbol{h}_k^\star, \quad \textit{for all } 1 \leq i \leq n_A, 1 \leq k \leq K_A, \\ &\boldsymbol{h}_{k,i}^\star = \boldsymbol{h}_k^\star, \quad \textit{for all } 1 \leq i \leq n_B, K_A < k \leq K, \end{aligned} \tag{18}$$

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 (17)*. 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* (17) *are in the form of* (18)*.*

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 of 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 brought by oversampling for certain choices of the oversampling rates. The improvement is noticeable on both the minority classes and all classes.

A closer look at the results of Table 2, however, reveals that issues remain when addressing Minority Collapse by oversampling. Perhaps the most critical one is that although oversampling with a very large value of w_r can mitigate Minority Collapse on the training set, it is at the cost of degrading test accuracy. More specifically, 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 [PHD20]. 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 (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 Appendix A.1.

Proposition 2. *Assume $p \geq K$. For any $q \in (1,2) \cup (2,\infty)$, consider the optimization problem*

$$\begin{aligned} \min_{\boldsymbol{W},\boldsymbol{H}} \quad & \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i},\boldsymbol{y}_k) \\ \text{s.t.} \quad & \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 \leq E_W, \\ & \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{h}_{k,i}\|_q^q \leq E_H, \end{aligned}$$

where $\mathcal{L}$ is the cross-entropy loss. Then, any global minimizer of this program does not satisfy (8) *for any positive numbers C and C'. That is, neural collapse does not emerge in this model.*

While the paper has demonstrated its noticeable effectiveness, the Layer-Peeled Model 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? 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:

$$\begin{aligned} \min_{\boldsymbol{W},\boldsymbol{H}} \quad & \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\mathcal{L}(\boldsymbol{f}(\boldsymbol{h}_{k,i},\boldsymbol{W}_{(L-m+1):L}),\boldsymbol{y}_k) \\ \text{s.t.} \quad & \frac{1}{K}\|\boldsymbol{W}_{(L-m+1):L}\|^2 \le E_W, \\ & \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i=1}^{n_k}\|\boldsymbol{h}_{k,i}\|^2 \le E_H. \end{aligned}$$

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 (14) offers an analytical and computationally efficient technique to identify and mitigate bias induced by class imbalance when training deep learning models. First, an interesting question is to extend Minority Collapse from the case of two-valued class sizes to general imbalanced datasets. Second, as suggested by our findings in Section 5, how should we choose loss functions in order to mitigate Minority Collapse [CWG$^+$19]? Last, a possible use case of the Layer-Peeled Model is to design more efficient sampling schemes to take into account fairness considerations [BG18, ZS18, MMS$^+$19].

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.

Acknowledgments

We are grateful to X.Y. Han for helpful discussions about some results of [PHD20]. This work was supported in part by NIH through RF1AG063481, NSF through CAREER DMS-1847415 and CCF-1934876, an Alfred Sloan Research Fellowship, and the Wharton Dean's Research Fund.

References

[AKK$^+$19] Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019.

[AZLS19] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pages 2388–2464, 2019.

[BCN18] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *Siam Review*, 60(2):223–311, 2018.

[BFT17] Peter Bartlett, Dylan Foster, and Matus Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in Neural Information Processing Systems*, 30:6241–6250, 2017.

[BG18] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91, 2018.

[BHMM19] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

[BLLT20] Peter L Bartlett, Philip M Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 2020.

[BMM18] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

[BMP08] Francis Bach, Julien Mairal, and Jean Ponce. Convex sparse matrix factorizations. *arXiv preprint arXiv:0812.1869*, 2008.

[BZMA20] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *arXiv preprint arXiv:2006.11477*, 2020.

[CJL+19] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019.

[CKNH20] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[COB19] Lenaic Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. In *Advances in Neural Information Processing Systems*, 2019.

[CWG+19] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in Neural Information Processing Systems*, volume 32, pages 1567–1578, 2019.

[DLL+19] Simon S Du, Jason D Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, 2019.

[EMW19] Weinan E, Chao Ma, and Lei Wu. A comparative analysis of the optimization and generalization property of two-layer neural network and random feature models under gradient descent dynamics. *arXiv preprint arXiv:1904.04326*, 2019.

[EW20] Weinan E and Stephan Wojtowytsch. On the emergence of tetrahedral symmetry in the final and penultimate layers of neural network classifiers. *arXiv preprint arXiv:2012.05420*, 2020.

[FDZ21] Cong Fang, Han Dong, and Tong Zhang. Mathematical models of overparameterized neural networks. *Proceedings of the IEEE*, pages 1–21, 2021.

[FLLZ18] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *Advances in Neural Information Processing Systems*, pages 689–699, 2018.

[FLYZ20] Cong Fang, Jason D Lee, Pengkun Yang, and Tong Zhang. Modeling from features: a mean-field framework for over-parameterized deep neural networks. *arXiv preprint arXiv:2007.01452*, 2020.

[FLZ19] Cong Fang, Zhouchen Lin, and Tong Zhang. Sharp analysis for nonconvex SGD escaping from saddle points. In *Annual Conference on Learning Theory*, pages 1192–1234, 2019.

[FMZ19] Jianqing Fan, Cong Ma, and Yiqiao Zhong. A selective overview of deep learning. *arXiv preprint arXiv:1904.05526*, 2019.

[HLLT16] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5375–5384, 2016.

[HMP+06] Eduard Hovy, Mitch Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. Ontonotes: the 90% solution. In *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, 2006.

[HS20] Hangfeng He and Weijie J Su. The local elasticity of neural networks. In *International Conference on Learning Representations*, 2020.

[HT20] Fengxiang He and Dacheng Tao. Recent advances in deep learning theory. *arXiv preprint arXiv:2012.10931*, 2020.

[HV19] Benjamin D Haeffele and René Vidal. Structured low-rank matrix factorization: Global optimality, algorithms, and applications. *IEEE transactions on pattern analysis and machine intelligence*, 42(6):1468–1482, 2019.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[IS15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015.

[JGH18] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, 2018.

[JK19] Justin M Johnson and Taghi M Khoshgoftaar. Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1):27, 2019.

[Kri09] A Krizhevsky. Learning multiple layers of features from tiny images. *Master's thesis, University of Tront*, 2009.

[KSH17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[KTW+20] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020.

[KWL+19] Rohith Kuditipudi, Xiang Wang, Holden Lee, Yi Zhang, Zhiyuan Li, Wei Hu, Sanjeev Arora, and Rong Ge. Explaining landscape connectivity of low-cost solutions for multilayer nets. In *Advances in Neural Information Processing Systems*, pages 14601–14610, 2019.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[LR20] Tengyuan Liang and Alexander Rakhlin. Just interpolate: Kernel "ridgeless" regression can generalize. *Annals of Statistics*, 48(3):1329–1347, 2020.

[LS20] Jianfeng Lu and Stefan Steinerberger. Neural collapse with cross-entropy loss. *arXiv preprint arXiv:2012.08465*, 2020.

[LSS20] Zhu Li, Weijie Su, and Dino Sejdinovic. Benign overfitting and noisy features. *arXiv preprint arXiv:2008.02901*, 2020.

[MBB18] Siyuan Ma, Raef Bassily, and Mikhail Belkin. The power of interpolation: Understanding the effectiveness of sgd in modern over-parametrized learning. In *International Conference on Machine Learning*, pages 3325–3334. PMLR, 2018.

[MMN18] Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671, 2018.

[MMS+19] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *arXiv preprint arXiv:1908.09635*, 2019.

[MPP20] Dustin G Mixon, Hans Parshall, and Jianzong Pi. Neural collapse with unconstrained features. *arXiv preprint arXiv:2011.11619*, 2020.

[MR17] K Madasamy and M Ramaswami. Data imbalance and classifiers: impact and solutions from a big data perspective. *International Journal of Computational Intelligence Research*, 13(9):2267–2281, 2017.

[OS20] Samet Oymak and Mahdi Soltanolkotabi. Towards moderate overparameterization: global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 2020.

[PBL20] Tomaso Poggio, Andrzej Banburski, and Qianli Liao. Theoretical issues in deep networks. *Proceedings of the National Academy of Sciences*, 2020.

[PGM+19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019.

[PHD20] Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020.

[PL20] Tomaso Poggio and Qianli Liao. Explicit regularization and implicit bias in deep network classifiers trained with the square loss. *arXiv preprint arXiv:2101.00072*, 2020.

[PSM14] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[RVE18] Grant M Rotskoff and Eric 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.

[SH03] Thomas Strohmer and Robert W. Heath. Grassmannian frames with applications to coding and communication. *Applied and Computational Harmonic Analysis*, 14(3):257–275, 2003.

[Sha20] Ohad Shamir. Gradient methods never overfit on separable data. *arXiv preprint arXiv:2007.00028*, 2020.

[SHM+16] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[SHN+18] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878, 2018.

[SS19] Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks: A central limit theorem. *Stochastic Processes and their Applications*, 2019.

[SSJ20] Bin Shi, Weijie J Su, and Michael I Jordan. On learning rates and Schrödinger operators. *arXiv preprint arXiv:2004.06977*, 2020.

[Sun19] Ruoyu Sun. Optimization for deep learning: theory and algorithms. *arXiv preprint arXiv:1912.08957*, 2019.

[SXY+19] Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. *arXiv preprint arXiv:1902.07379*, 2019.

[SZ03] Jos F Sturm and Shuzhong Zhang. On cones of nonnegative quadratic functions. *Mathematics of Operations Research*, 28(2):246–267, 2003.

[SZ14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[WL90] Andrew R Webb and David Lowe. The optimised internal representation of multilayer classifier networks performs nonlinear discriminant analysis. *Neural Networks*, 3(4):367–375, 1990.

[WLW+16] Shoujin Wang, Wei Liu, Jia Wu, Longbing Cao, Qinxue Meng, and Paul J Kennedy. Training deep neural networks on imbalanced data sets. In *2016 international joint conference on neural networks (IJCNN)*, pages 4368–4374. IEEE, 2016.

[XRV17] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[Yar17] Dmitry Yarotsky. Error bounds for approximations with deep ReLU networks. *Neural Networks*, 94:103–114, 2017.

[YCY+20] Yaodong Yu, Kwan Ho Ryan Chan, Chong You, Chaobing Song, and Yi Ma. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. *Advances in Neural Information Processing Systems*, 33, 2020.

[ZCZG18] Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Stochastic gradient descent optimizes over-parameterized deep relu networks. In *Advances in Neural Information Processing Systems*, 2018.

[ZKL+16] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Antonio Torralba, and Aude Oliva. Places: An image database for deep scene understanding. *arXiv preprint arXiv:1610.02055*, 2016.

[ZS18] James Zou and Londa Schiebinger. AI can be sexist and racist—it's time to make it fair, 2018.

A Proofs

For simplicity, in this appendix 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.1 Balanced Case

A.1.1 Proofs of Theorem 1 and Proposition 2

Because there are multiplications of variables in the objective function, Program (7) is nonconvex. Thus the KKT condition is not sufficient for optimality. To prove Theorem 1, we directly determine the global minimum of (7). During this procedure, one key step is to show that program (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 the same constraints with suitable conditions. Finally, by checking all the conditions to reach the minimum, we obtain the minimizer of (7). 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

$$\begin{aligned}
&-\log\left(\frac{\boldsymbol{z}(k)}{\sum_{k'=1}^{K}\boldsymbol{z}(k')}\right)\\
=&-\log(\boldsymbol{z}(k)) + \log\left(\frac{C_a}{C_a+C_b}\left(\frac{(C_a+C_b)\ \boldsymbol{z}(k)}{C_a}\right) + C_c\sum_{k'=1,k'\neq k}^{K}\frac{\boldsymbol{z}(k')}{C_c}\right)\\
\overset{a}{\ge}&-\log(\boldsymbol{z}(k)) + \frac{C_a}{C_a+C_b}\log\left(\frac{(C_a+C_b)\ \boldsymbol{z}(k)}{C_a}\right) + C_c\sum_{k'=1,k'\neq k}^{K}\log\left(\frac{\boldsymbol{z}(k')}{C_c}\right)\\
\overset{b}{=}&-\frac{C_b}{C_a+C_b}\left[\log(\boldsymbol{z}(k)) - \frac{1}{K-1}\sum_{k'=1,k'\neq k}^{K}\log(\boldsymbol{z}(k'))\right] + C_d,
\end{aligned} \tag{19}$$

where $\overset{a}{\ge}$ applies the concavity of $\log(\cdot)$ and in $\overset{b}{=}$, we define $C_d := \frac{C_a}{C_a+C_b}\log(\frac{C_a+C_b}{C_a}) + \frac{C_b}{C_a+C_b}\log(1/C_c)$. Note that in (19), 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 (7). Applying (19) on the objective, we have

$$\begin{aligned}
&\frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \qquad (20)\\
\ge&\frac{C_b}{(C_a+C_b)N(K-1)}\sum_{i=1}^{n}\left[\left(\sum_{k=1}^{K}\boldsymbol{h}_{k,i}\right)^{\top}\left(\sum_{k=1}^{K}\boldsymbol{w}_k\right) - K\sum_{k=1}^{K}\boldsymbol{h}_{k,i}^{\top}\boldsymbol{w}_k\right] + C_d.
\end{aligned}$$

Defining $\bar{\boldsymbol{h}}_i := \frac{1}{K}\sum_{k=1}^K \boldsymbol{h}_{k,i}$ for $i \in [n]$, it follows by Young's inequality that

$$\begin{aligned}
&\sum_{i=1}^n \left[\left(\sum_{k=1}^K \boldsymbol{h}_{k,i} \right)^\top \left(\sum_{k=1}^K \boldsymbol{w}_k \right) - K \sum_{k=1}^K \boldsymbol{h}_{k,i}^\top \boldsymbol{w}_k \right] \\
=& K \sum_{i=1}^n \sum_{k=1}^K (\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i})^\top \boldsymbol{w}_k \\
\geq& -\frac{K}{2} \sum_{k=1}^K \sum_{i=1}^n \|\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i}\|^2 / C_e - \frac{C_e N}{2} \sum_{k=1}^K \|\boldsymbol{w}_k\|^2, \qquad (21)
\end{aligned}$$

where we pick $C_e := \sqrt{E_H / E_W}$. The two terms in the right hand side of (21) can be bounded via the constraints of (7). Especially, we have

$$\frac{C_e N}{2} \sum_{k=1}^K \|\boldsymbol{w}_k\|^2 \leq \frac{KN\sqrt{E_H E_W}}{2}, \qquad (22)$$

and

$$\begin{aligned}
\frac{K}{2} \sum_{k=1}^K \sum_{i=1}^n \|\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i}\|^2 / C_e &\overset{a}{=} \frac{K^2}{2C_e} \sum_{i=1}^n \left(\frac{1}{K} \sum_{k=1}^K \|\boldsymbol{h}_{k,i}\|^2 - \|\bar{\boldsymbol{h}}_i\|^2 \right) \\
&\leq \frac{K}{2C_e} \sum_{k=1}^K \sum_{i=1}^n \|\boldsymbol{h}_{k,i}\|^2 \leq \frac{KN\sqrt{E_H E_W}}{2}, \qquad (23)
\end{aligned}$$

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 (21), (22), and (23) into (20), we have

$$\frac{1}{N} \sum_{k=1}^K \sum_{i=1}^n \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq \frac{C_b}{C_a + C_b} \frac{K\sqrt{E_H E_W}}{K-1} + C_d := L_0. \qquad (24)$$

Now we check the conditions to make the equality in (24) hold.

By the strict concavity of $\log(\cdot)$, the equality in (20) holds if and only if

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

for all $(k,i,k') \in \{(k,i,k') : k \in [K], k' \in [K], k' \neq k, i \in [n]\}$. The equality in (21) holds if and only if

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

The equalities in (22) and (23) hold if and only if:

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

Applying Lemma 2 shown in the end of the section, we have $(\boldsymbol{H}, \boldsymbol{W})$ satisfies (8).

Reversely, it is easy to verify that the equality for (24) is reachable when $(\boldsymbol{H}, \boldsymbol{W})$ admits (8). So L_0 is the global minimum of (7) and $(\boldsymbol{H}, \boldsymbol{W})$ in (8) is the unique form for the minimizers. We complete the proof of Theorem 1. □

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

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

We can examine that $\mathcal{S}_R$ admits the constraints of (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 $\frac{1}{N}\sum_{k=1}^K \sum_{i=1}^n \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k)$ 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

$$\begin{aligned} &\frac{1}{N}\sum_{k=1}^K \sum_{i=1}^n \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \\ =& -\log\left(\frac{\exp(B_1 B_2 B_3 b^2 [a^2 + (K-1)])}{\exp(B_1 B_2 B_3 b^2 [a^2 + K - 1]) + (K-1)\exp(B_1 B_2 B_3 b^2 [K - 2 - 2a])} \right) \\ =& -\log\left(\frac{1}{1 + (K-1)\exp(-B_1 B_2 B_3 b^2 (a+1)^2)} \right). \end{aligned}$$

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

$$\begin{aligned} \left[B_1 B_2 B_3 b^2 (a+1)^2\right]^2 &\overset{a}{\leq} E_H E_W \left[B_3^2 b^2 (a+1)^2\right] \left[b^2 (a+1)^2\right] \\ &= E_H E_W \left[\frac{(a+1)^2}{a^2 + (K-1)}\right] \left[\frac{(a+1)^q}{a^q + K - 1}\right]^{2/q}. \end{aligned} \tag{25}$$

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 (25) cannot achieve the maximum at $a = K-1$, which is our desired result. Indeed, we have

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

So $a = K-1$ will not be the maximizer of (25) unless $q = 2$. We complete the proof. □

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

$$\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i} = -\sqrt{\frac{E_H}{E_W}} \boldsymbol{w}_k, \quad k \in [K], \quad i \in [n], \tag{26}$$

and

$$\frac{1}{K}\sum_{k=1}^{K}\frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{h}_{k,i}\|^2 = E_H, \quad \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 = E_W, \quad \bar{\boldsymbol{h}}_i = \boldsymbol{0}_p,\ i\in[n], \tag{27}$$

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*

$$\boldsymbol{h}_{k,i}\cdot\boldsymbol{w}_k = \boldsymbol{h}_{k,i}\cdot\boldsymbol{w}_{k'} + C. \tag{28}$$

Then $(\boldsymbol{H},\boldsymbol{W})$ *satisfies* (8).

Proof. Combining (26) with the last equality in (27), 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

$$\boldsymbol{W} = \sqrt{E_W}\ (\boldsymbol{M}^\star)^\top, \tag{29}$$

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 (28), we have, for all $(k,k')\in\{(k,k'): k\in[K], k'\in[K], k'\neq k\}$,

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

and

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

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

On the other hand, 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

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

which implies (29). We complete the proof. □

A.1.2 Proofs of Theorems 3 and 4

The proof of Theorems 3 and 4 follow a similar argument of Theorem 1.

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

$$E_{k,i,k'} := \frac{1}{n}\sum_{j=1}^{n}\exp(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k',j}/\tau).$$

For constants $C_a := \exp\left(\sqrt{E_H E_W}\right)$ and $C_b := \exp\left(-\sqrt{E_H E_W}/(K-1)\right)$, let $C_c := \frac{C_b}{(C_a+C_b)(K-1)}$. Using a similar argument as (19), we have for $j \in [n]$,

$$
\begin{aligned}
& -\log\left(\frac{\exp(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^{K} E_{k,i,k'}}\right) \qquad (30)\\
=& -\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j}/\tau + \log\left(\frac{C_a}{C_a+C_b}\left(\frac{(C_a+C_b)\ E_{k,i,k}}{C_a}\right) + C_c \sum_{k'=1,\ k'\neq k}^{K} \frac{E_{k,i,k'}}{C_c}\right)\\
\overset{a}{\geq}& -\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j}/\tau + \frac{C_a}{C_a+C_b}\log\left(\frac{(C_a+C_b)\ E_{k,i,k}}{C_a}\right) + C_c \sum_{k'=1,\ k'\neq k}^{K} \log\left(\frac{E_{k,i,k'}}{C_c}\right)\\
\overset{b}{=}& -\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j}/\tau + \frac{C_a}{C_a+C_b}\log\left(E_{k,i,k}\right) + C_c \sum_{k'=1,\ k'\neq k}^{K} \log\left(E_{k,i,k'}\right) + C_d\\
\overset{c}{\geq}& -\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j}/\tau + \frac{C_a}{(C_a+C_b)n}\sum_{\ell=1}^{n}\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,\ell}/\tau + \frac{C_c}{n}\sum_{k'=1,\ k'\neq k}^{K}\sum_{\ell=1}^{n}\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k',\ell}/\tau + C_d.
\end{aligned}
$$

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 (30) into the objective function, we have

$$
\begin{aligned}
& \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n}\frac{1}{n}\sum_{j=1}^{n} -\log\left(\frac{\exp(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^{K}\sum_{\ell=1}^{n}\exp(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k',\ell})}\right) \qquad (31)\\
=& \frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n}\frac{1}{n}\sum_{j=1}^{n} -\log\left(\frac{\exp(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^{K} E_{k,i,k'}}\right) + \log(n)\\
\overset{(30)}{\geq}& \frac{C_b K}{(C_a+C_b)N(K-1)\tau}\sum_{k=1}^{K}\sum_{i=1}^{n}\left(-\frac{1}{n}\sum_{j=1}^{n}\left(\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k,j} - \frac{1}{K}\sum_{k'=1}^{K}\boldsymbol{h}_{k,i}\cdot\boldsymbol{h}_{k',j}\right)\right) + C_d + \log(n).
\end{aligned}
$$

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 (21) and (23) gives that

$$\begin{aligned}
&\sum_{k=1}^K \sum_{i=1}^n \left(-\frac{1}{n}\sum_{j=1}^n \left(\boldsymbol{h}_{k,i}\cdot \boldsymbol{h}_{k,j} - \frac{1}{K}\sum_{k'=1}^K \boldsymbol{h}_{k,i}\cdot \boldsymbol{h}_{k',j} \right) \right) \\
=&\sum_{k=1}^K \sum_{i=1}^n \left(-\frac{1}{n}\sum_{j=1}^n \boldsymbol{h}_{k,i}\cdot (\boldsymbol{h}_{k,j} - \bar{\boldsymbol{h}}_j) \right) \\
\overset{a}{\geq} &-\frac{1}{2}\sum_{k=1}^K \sum_{i=1}^n \|\boldsymbol{h}_{k,i}\|^2 - \frac{1}{2}\sum_{k=1}^K \sum_{i=1}^n \left\|\boldsymbol{h}_{k,i} - \bar{\boldsymbol{h}}_i\right\|^2 \\
\overset{b}{\geq} &-\frac{1}{2}\sum_{k=1}^K \sum_{i=1}^n \|\boldsymbol{h}_{k,i}\|^2 - \frac{K}{2}\sum_{i=1}^n \left(\frac{1}{K}\sum_{k=1}^K \|\boldsymbol{h}_{k,i}\|^2 - \left\|\bar{\boldsymbol{h}}_i\right\|^2 \right) \\
\geq &-\sum_{k=1}^K \sum_{i=1}^n \|\boldsymbol{h}_{k,i}\|^2 \overset{c}{\geq} -NE_H,
\end{aligned} \tag{32}$$

where $\overset{a}{\geq}$ follows from Young's inequality, $\overset{b}{\geq}$ 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 (10). Therefore, plugging (32) into (31) yields that

$$\begin{aligned}
&\frac{1}{N}\sum_{k=1}^K \sum_{i=1}^n \frac{1}{n}\sum_{j=1}^n -\log\left(\frac{\exp(\boldsymbol{h}_{k,i}\cdot \boldsymbol{h}_{k,j}/\tau)}{\sum_{k'=1}^K \sum_{\ell=1}^n \exp(\boldsymbol{h}_{k,i}\cdot \boldsymbol{h}_{k',\ell}/\tau)} \right) \\
\geq &-\frac{C_b K E_H}{(C_a + C_b)(K-1)\tau} + C_d + \log(n).
\end{aligned} \tag{33}$$

Now we check the conditions to make the equality in (33) hold. By the strictly concavity of $\log(\cdot)$, the equality in (30) holds only if for all $(k,i,k') \in \{(k,i,k') : k \in [K], k' \in [K], k' \neq k, i \in [n]\}$,

$$\frac{E_{k,i,k}}{C_a} = \frac{E_{k,i,k'}}{C_b}. \tag{34}$$

The equality in (32) holds if and only if:

$$\boldsymbol{h}_{k,i} = \boldsymbol{h}_k,\ i \in [n],\ k \in [K], \quad \frac{1}{K}\sum_{k=1}^K \|\boldsymbol{h}_k\|^2 = E_H, \quad \sum_{k=1}^K \boldsymbol{h}_k = \boldsymbol{0}_p. \tag{35}$$

Plugging $\boldsymbol{h}_{k,i} = \boldsymbol{h}_k$ into (34), we have for $(k,k') \in \{k,k' : k \in [K], k' \in [K], k' \neq k\}$,

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

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 $k \in [K]$. On the other hand, 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 (12).

Reversely, it is easy to verify that the equality for (33) is reachable when $\boldsymbol{H}$ admits (12). We complete the proof of Theorem 3. □

Proof of Theorem 4. We first determine the minimum value of (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

$$\begin{aligned} \sum_{k'=1,\ k' \neq k}^{K} g_2 \left(\boldsymbol{S}(\boldsymbol{z}_{k,j})(k') \right) &\geq (K-1) g_2 \left(\frac{1}{K-1} \sum_{k'=1,\ k' \neq k}^{K} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k') \right) \\ &\overset{a}{=} (K-1) g_2 \left(1 - \frac{1}{K-1} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \right), \end{aligned} \tag{36}$$

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

$$\begin{aligned} &\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \qquad (37) \\ =&\frac{1}{N} \sum_{i=1}^{n} \sum_{k=1}^{K} \left[g_1 \left(\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \right) + \sum_{k'=1,\ k' \neq k}^{K} g_2 \left(\boldsymbol{S}(\boldsymbol{z}_{k',i})(k') \right) \right] \\ \overset{(36)}{\geq}& \frac{1}{N} \sum_{i=1}^{n} \sum_{k=1}^{K} \left[g_1 \left(\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \right) + (K-1) g_2 \left(1 - \frac{1}{K-1} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \right) \right] \\ \geq& g_1 \left(\frac{1}{N} \sum_{i=1}^{n} \sum_{k=1}^{K} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \right) + (K-1) g_2 \left(1 - \frac{1}{N(K-1)} \sum_{i=1}^{n} \sum_{k=1}^{K} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \right). \end{aligned}$$

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

$$\begin{aligned} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k) &= \frac{\exp(\boldsymbol{z}_{k,i}(k))}{\sum_{k'=1}^{K} \exp(\boldsymbol{z}_{k,i}(k'))} \\ &\leq \frac{\exp(\boldsymbol{z}_{k,i}(k))}{\exp(\boldsymbol{z}_{k,i}(k)) + (K-1) \exp\left(\frac{1}{K-1} \sum_{k'=1,\ k' \neq k}^{K} \boldsymbol{z}_{k,i}(k') \right)} \\ &= \frac{1}{1 + (K-1) \exp\left(\frac{1}{K-1} \sum_{k'=1,\ k' \neq k}^{K} \boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k) \right)}. \end{aligned} \tag{38}$$

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

$$g_0''(x) = -\frac{\exp(x)(1+C\exp(x))(1-C\exp(x))}{(1+C\exp(x))^4}. \tag{39}$$

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

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

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

(B) $i \in \mathcal{S}_2$ 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

$$L(t) := \begin{cases} N - \left(\frac{1}{2}t + \frac{K(n-t)}{1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)}\right), & t \in [0 : n-1], \\ N - \frac{n}{2}, & t = n. \end{cases} \tag{40}$$

We show in Lemma 3 (see the end of the proof) that

$$\frac{1}{N} \sum_{i=1}^{n} \sum_{k=1}^{K} \boldsymbol{S}(\boldsymbol{z}_{k,i})(i) \leq \frac{1}{N} L(0). \tag{41}$$

Plugging (41) into (37), the objective function can be lower bounded as:

$$\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq g_1\left(\frac{1}{N}L(0)\right) + (K-1)g_2\left(1 - \frac{1}{N(K-1)}L(0)\right) := L_0. \tag{42}$$

On the other hand, one can directly verify that the equality for (42) is reachable when $(\boldsymbol{H}, \boldsymbol{W})$ satisfies (8). So L_0 is the global minimum of (7) and (8) is a minimizer of (7).

Now we show all the solutions are in form (8) under the assumption that g_2 is strictly convex and g_1 (or g_2) are strictly monotone.

By the strict convexity of g_2, the equality in (36) holds if and only if for any $k \in [K]$ and $i \in [n]$ and $k' \in [K]$, $k'' \in [K]$ such that $k' \neq k$ and $k'' \neq k$, we have

$$\boldsymbol{S}(\boldsymbol{z}_{i,j})(k_1) = \boldsymbol{S}(\boldsymbol{z}_{i,j})(k_2),$$

which indicates that

$$\boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_{k'} = \boldsymbol{h}_{k,i} \cdot \boldsymbol{w}_{k''} \tag{43}$$

Again, by the strict convexity of g_2, (37) holds if and only if for all $k \in [K]$, $i \in [n]$, and a suitable number $C' \in (0, 1)$, we have

$$\mathcal{S}(\boldsymbol{z}_{k,i})(k) := C'. \tag{44}$$

Combining (43) with (44), 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 (42) 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 3 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

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

where $\bar{\boldsymbol{h}}_i := \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{h}_{k,i}$ with $i \in [n]$. In all, from Lemma 2, we have $(\boldsymbol{H}, \boldsymbol{W})$ satisfies (8), achieving the uniqueness argument. We complete the proof of Theorem 4. □

Lemma 3. *For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$, we have*

$$\sum_{i=1}^{n}\sum_{k=1}^{K}\boldsymbol{S}(\boldsymbol{W}\boldsymbol{h}_{k,i})(k) \leq L(0), \tag{45}$$

with L defined in (40)*. Moreover, recalling the definition of $\mathcal{S}_1$ and $\mathcal{S}_2$ in (A) and (B), the equality in* (45) *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

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

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

Proof of Lemma 3. For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$, we separately consider $\mathcal{S}_1$ and $\mathcal{S}_2$ defined in (A) and (B), 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 (38) that $S(\boldsymbol{z}_{k,i})(k) \leq 1/2$. Furthermore, for the 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
$$\sum_{i \in \mathcal{S}_1}\sum_{k=1}^{K}\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \leq t(1/2 + K - 1). \tag{46}$$

- For $i \in \mathcal{S}_2$, by the concavity of $g_0(x)$ when $x < \log\left(\frac{1}{K-1}\right)$ from (39), we have, for $\mathcal{S}_2 \neq \varnothing$,

$$\begin{aligned}
&\sum_{i\in\mathcal{S}_2}\sum_{k=1}^{K} \boldsymbol{S}(\boldsymbol{z}_{k,i})(k) \qquad (47)\\
\overset{(38)}{\leq}& \sum_{i\in\mathcal{S}_2}\sum_{k=1}^{K} \frac{1}{1+(K-1)\exp\left(\frac{1}{K-1}\sum_{k'=1,\ k'\neq k}^{K} \boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right)}\\
\leq& \frac{(n-t)K}{1+(K-1)\exp\left(\frac{1}{(n-t)K}\sum_{i\in\mathcal{S}_2}\sum_{k=1}^{K}\left(\frac{1}{K-1}\sum_{k'=1,\ k'\neq k}^{K} \boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right)\right)}.
\end{aligned}$$

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 in (21) and (23). Especially, recalling $\bar{\boldsymbol{h}}_i = \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{h}_{k,i}$ for $i \in [n]$, we have

$$\begin{aligned}
&\sum_{i\in\mathcal{S}_2}\sum_{k=1}^{K}\left(\frac{1}{K-1}\sum_{k'=1,\ k'\neq k}^{K} \boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right) \qquad (48)\\
=&\frac{1}{K-1}\sum_{i\in\mathcal{S}_2}\left[\left(\sum_{k=1}^{K}\boldsymbol{h}_{k,i}\right)^{\top}\left(\sum_{k=1}^{K}\boldsymbol{w}_k\right) - K\sum_{K=1}^{K}\boldsymbol{h}_{k,i}^{\top}\boldsymbol{w}_k\right]\\
\overset{(21)}{\geq}& -\frac{K}{2(K-1)}\sum_{k=1}^{K}\sum_{i\in\mathcal{S}_2}\|\bar{\boldsymbol{h}}_i - \boldsymbol{h}_{k,i}\|^2/C'' - \frac{C''K(n-t)}{2(K-1)}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2\\
\overset{(23)}{\geq}& -\frac{K}{2(K-1)}\sum_{k=1}^{K}\sum_{i\in\mathcal{S}_2}\|\boldsymbol{h}_{k,i}\|^2/C'' - \frac{C''K(n-t)}{2(K-1)}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2\\
\geq& -\frac{K}{2(K-1)}\sum_{k=1}^{K}\sum_{i=1}^{n}\|\boldsymbol{h}_{k,i}\|^2/C'' - \frac{C''K(n-t)}{2(K-1)}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2\\
\geq& -\frac{K^2}{(K-1)}\sqrt{E_H E_W(n-t)n},
\end{aligned}$$

where in the last inequality we follow from the constrains of (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 (48) into (47), using the monotonicity of $g_0(x)$, and adding (46), we have

$$\begin{aligned}
\sum_{k=1}^{n}\sum_{i=1}^{K}\boldsymbol{S}(\boldsymbol{z}_{k,i})(k) &\leq N - \left(\frac{1}{2}t + \frac{K}{1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)}(n-t)\right)\\
&= L(t). \qquad (49)
\end{aligned}$$

And when $t = n$, it directly follows from (47) 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 $L(t) \le L(0)$ 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

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

which implies that

$$\begin{aligned} L'(t) &= -\left[\frac{1}{2} - q(t) + q'(t)(n-t)\right] \\ &\le \frac{\frac{1}{2}\frac{K^2}{K-1}\sqrt{E_H E_W n}(n-t)^{-1/2} + K}{1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)} - \frac{1}{2} \\ &= \frac{K\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W}\right) + 2K - 1 - \exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)}{2\left[1+\exp\left(\frac{K}{K-1}\sqrt{n/(n-t)}\sqrt{E_H E_W} - \log(K-1)\right)\right]}. \end{aligned}$$

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

$$\sqrt{E_H E_W} > \frac{K-1}{K}\log\left(K^2\sqrt{E_H E_W} + (2K-1)(K-1)\right) \ge \frac{K-1}{K}\log\left(K(K-1)\right).$$

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

$$f(x) \le f\left(\frac{K}{K-1}\sqrt{E_H E_W}\right) = \frac{K^2}{K-1}\sqrt{E_H E_W} + 2K - 1 - \frac{1}{K-1}\exp\left(\frac{K}{K-1}\sqrt{E_H E_W}\right) \overset{a}{<} 0,$$

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 (45) from (49) with $t = 0$.

When $t = 0$, the equality in the first inequality of (48) holds if and only if:

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

The equality in the second and third inequalities of (48) holds if and only if:

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

We obtain Lemma 3. □

A.2 Imbalanced Case

A.2.1 Proofs of Lemma 1 and Proposition 1

Proof of Lemma 1. For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ for the original program (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 (14), we have

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

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 (14). Letting L_0 be the global minimum of (14), for any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$, we obtain

$$\begin{aligned}\frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) &= \sum_{k=1}^{K}\frac{n_k}{N}\left[\frac{1}{n_k}\sum_{k=1}^{n_k}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k)\right] \\ &\overset{a}{\geq} \sum_{k=1}^{K}\frac{n_k}{N}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_k, \boldsymbol{y}_k) = \sum_{k=1}^{K}\frac{n_k}{N}\mathcal{L}(\boldsymbol{z}_k, \boldsymbol{y}_k) \geq L_0, \end{aligned} \tag{50}$$

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 (15) with $\boldsymbol{X}^\star$ being a minimizer of (14), 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 (7) and have

$$\frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\mathcal{L}(\boldsymbol{W}^\star\boldsymbol{h}_{k,i}^\star, \boldsymbol{y}_k) = \sum_{k=1}^{K}\frac{n_k}{N}\mathcal{L}(\boldsymbol{z}_k^\star, \boldsymbol{y}_k) = L_0, \tag{51}$$

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 $k \in [K]$.

Combing (50) and (51), we conclude that L_0 is the global minimum of (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 (15). Let

$$\boldsymbol{h}_k' = \frac{1}{n_k}\sum_{i=1}^{n_k} \boldsymbol{h}_{k,i}',\ k \in [K], \quad \text{and} \quad \boldsymbol{X}' = \left[\boldsymbol{h}_1', \boldsymbol{h}_2', \ldots, \boldsymbol{h}_K', (\boldsymbol{W}')^\top\right]^\top \left[\boldsymbol{h}_1', \boldsymbol{h}_2', \ldots, \boldsymbol{h}_K', (\boldsymbol{W}')^\top\right].$$

(50) implies that $\boldsymbol{X}'$ is a minimizer of (14). As $(\boldsymbol{H}', \boldsymbol{W}')$ cannot be written as (15) 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

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

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 (7) are in form of (15). We complete the proof. □

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

A.2.2 Proof of Theorem 5

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

$$\begin{aligned}
\min_{\boldsymbol{H},\boldsymbol{W}} \quad & \frac{1}{K_A n_A}\sum_{k=1}^{K_A}\sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \\
\text{s.t.} \quad & \frac{1}{K}\sum_{k=1}^K \frac{1}{n_k}\sum_{i=1}^{n_k} \|\boldsymbol{h}_{k,i}\|^2 \leq E_H, \\
& \frac{1}{K}\sum_{k=1}^K \|\boldsymbol{w}_k\|^2 \leq E_W,
\end{aligned} \tag{52}$$

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 4 characterizes useful properties for the minimizer of (52).

Lemma 4. *Let $(\boldsymbol{H}, \boldsymbol{W})$ be a minimzer of (52). We have $\boldsymbol{h}_{k,i} = \mathbf{0}_p$ for all $k \in [K_A + 1 : K]$ and $i \in [n_B]$. Define L_0 as the global minimum of (52), i.e.,*

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

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 define $(\boldsymbol{H}^{\ell,\star}.\boldsymbol{W}^{\ell,\star})$ as a minimizer and separate the objective function into two parts. That is, we introduce

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

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:

$$\begin{aligned} \min_{\boldsymbol{H}^\ell, \boldsymbol{W}^\ell} \quad & \mathcal{L}_A^\ell\left(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell\right) \\ \text{s.t.} \quad & \frac{1}{K} \sum_{k=1}^{K} \left\|\boldsymbol{w}_k^\ell\right\|^2 \leq E_W, \\ & \frac{1}{K} \sum_{k=1}^{K_A} \frac{1}{n_A^\ell} \sum_{i=1}^{n_A^\ell} \|\boldsymbol{h}_{k,i}\|^2 + \frac{1}{K} \sum_{k=K_A+1}^{K} \frac{1}{n_B^\ell} \sum_{i=1}^{n_B^\ell} \|\boldsymbol{h}_{k,i}\|^2 \leq E_H, \end{aligned} \tag{53}$$

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

$$\begin{aligned} \min_{\boldsymbol{H}^\ell, \boldsymbol{W}^\ell} \quad & \mathcal{L}_B^\ell\left(\boldsymbol{H}^\ell, \boldsymbol{W}^\ell\right) \\ \text{s.t.} \quad & \frac{1}{K} \sum_{k=1}^{K} \left\|\boldsymbol{w}_k^\ell\right\|^2 \leq E_W, \\ & \frac{1}{K} \sum_{k=1}^{K_A} \frac{1}{n_A^\ell} \sum_{i=1}^{n_A^\ell} \|\boldsymbol{h}_{k,i}\|^2 + \frac{1}{K} \sum_{k=K_A+1}^{K} \frac{1}{n_B^\ell} \sum_{i=1}^{n_B^\ell} \|\boldsymbol{h}_{k,i}\|^2 \leq E_H. \end{aligned} \tag{54}$$

Note that Programs (53) and (54) and their minimizers have been studied in Lemma 4. 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 4 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 $\boldsymbol{h}_{k,i}^{\ell,A} = \mathbf{0}_p$ for all $k \in [K_A + 1 : K]$ and $i \in [n_B]$, we have

$$\mathcal{L}_B^\ell \left(\boldsymbol{H}^{\ell,A}, \boldsymbol{W}^{\ell,A} \right) = \log(K). \tag{55}$$

Now we prove Theorem 5 by contradiction. Suppose there exists a pair (k, k') such that $\lim_{\ell\to\infty} \boldsymbol{w}_k^{\ell,\star} - \boldsymbol{w}_{k'}^{\ell,\star} \neq \mathbf{0}_p$. 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

$$\begin{aligned}
\mathcal{L}^{a_\ell} \left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star} \right) \leq \mathcal{L}^{a_\ell} \left(\boldsymbol{H}^{a_\ell,A}, \boldsymbol{W}^{a_\ell,A} \right) &\overset{(55)}{=} \frac{K_A n_A^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}} L_A + \frac{K_B n_B^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}} \log(K) \\
&= L_A + \frac{1}{K_R R^{a_\ell} + 1} \left(\log(K) - L_A \right) \overset{\ell\to\infty}{\to} L_A,
\end{aligned} \tag{56}$$

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 4 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

$$\begin{aligned}
\mathcal{L}^{a_\ell} \left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star} \right) &= \mathcal{L}_A^{a_\ell} \left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star} \right) + \mathcal{L}_B^{a_\ell} \left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star} \right) \\
&\overset{a}{\geq} \mathcal{L}_A^{a_\ell} \left(\boldsymbol{H}^{a_\ell,\star}, \boldsymbol{W}^{a_\ell,\star} \right) + \mathcal{L}_B^{a_\ell} \left(\boldsymbol{H}^{a_\ell,B}, \boldsymbol{W}^{a_\ell,B} \right) \\
&= \frac{K_A n_A^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}} (L_A + \varepsilon_2) + \frac{K_B n_B^{a_\ell}}{K_A n_A^{a_\ell} + K_B n_B^{a_\ell}} L_B \\
&= L_A + \varepsilon_2 + \frac{1}{K_R R^{a_\ell} + 1} (L_B - L_A - \varepsilon_2) \overset{\ell\to\infty}{\to} L_A + \varepsilon_2,
\end{aligned} \tag{57}$$

where $\overset{a}{\geq}$ uses $\left(\boldsymbol{H}^{a_\ell,B}, \boldsymbol{W}^{a_\ell,B} \right)$ is the minimizer of (54). Thus we meet contradiction by comparing (56) with (57) and achieve Theorem 5. □

Proof of Lemma 4. 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 5 (see the end of the proof), for any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ of (52), the objective value can be bounded

from below by:

$$\begin{aligned}
&\frac{1}{K_A n_A}\sum_{k=1}^{K_A}\sum_{i=1}^{n_A}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i},\boldsymbol{y}_k) \qquad (58)\\
&\overset{a}{\geq} -\frac{C_g}{K_A}\sqrt{KE_H}\sqrt{\sum_{k=1}^{K_A}\|C_e\boldsymbol{w}_A+C_f\boldsymbol{w}_B-\boldsymbol{w}_k\|^2}+C_d\\
&\overset{b}{\geq} -\frac{C_g}{K_A}\sqrt{KE_H}\sqrt{KE_W-K_A\left(1/K_R-C_f^2-\frac{C_f^4}{C_e(2-C_e)}\right)\|\boldsymbol{w}_B\|^2-\sum_{k=K_A+1}^{K}\|\boldsymbol{w}_k-\boldsymbol{w}_B\|^2}+C_d,
\end{aligned}$$

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 (52). In the following, we separately consider three cases according to the values of K_A, K_B, and E_HE_W.

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

Then from $\overset{a}{\geq}$ in (58), we have

$$\begin{aligned}
&\frac{1}{K_A n_A}\sum_{k=1}^{K_A}\sum_{i=1}^{n_A}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i},\boldsymbol{y}_k)\\
&\overset{a}{\geq} -C_gC_f\sqrt{KE_H}\sqrt{\|\boldsymbol{w}_1-\boldsymbol{w}_B\|^2}+C_d\\
&= -C_gC_f\sqrt{KE_H}\sqrt{\|\boldsymbol{w}_1\|^2-2\boldsymbol{w}_1^\top\boldsymbol{w}_B+\|\boldsymbol{w}_B\|^2}+C_d\\
&\overset{b}{\geq} -C_gC_f\sqrt{KE_H}\sqrt{(1+1/K_B)(\|\boldsymbol{w}_1\|^2+K_B\|\boldsymbol{w}_B\|^2)}+C_d\\
&\overset{c}{\geq} -C_gC_f\sqrt{KE_H}\sqrt{(1+1/K_B)\left(KE_W-\sum_{k=2}^{K}\|\boldsymbol{w}_k-\boldsymbol{w}_B\|^2\right)}+C_d\\
&\geq -C_gC_f\sqrt{KE_H}\sqrt{(1+1/K_B)KE_W}+C_d := L_1, \qquad (59)
\end{aligned}$$

where $\overset{a}{\geq}$ uses $C_e+C_f=1$, $\overset{b}{\geq}$ follows from Young's inequality, 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

$$\begin{aligned}
\boldsymbol{w}_1=\sqrt{K_BE_W}\boldsymbol{u},\quad \boldsymbol{w}_k=-\sqrt{1/K_BE_W}\boldsymbol{u},\ k\in[2:K],\\
\boldsymbol{h}_{1,i}=\sqrt{(1+K_B)E_H}\boldsymbol{u},\ i\in[n_A],\qquad \boldsymbol{h}_{k,i}=\boldsymbol{0}_p,\ k\in[2:K],\ i\in[n_B],
\end{aligned}$$

where $\boldsymbol{u}$ is any unit vector, $(\boldsymbol{H}, \boldsymbol{W})$ can achieve the equality in (59). So L_1 is the global minimum of (52). Moreover, L_1 is achieved only if the equality in $\overset{a}{\geq}$ in (58) holds. From Lemma 52, we have 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

$$\sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2 \geq \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2 + \|\boldsymbol{w}_{k'} - \boldsymbol{w}_B\|^2 \geq \frac{\|\boldsymbol{w}_k - \boldsymbol{w}_{k'}\|^2}{2} = \varepsilon^2/2. \tag{60}$$

It follows from $\overset{c}{\geq}$ in (59) that

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

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 $\exp\big((1 + 1/K_R)\sqrt{E_H E_W}/(K_A - 1)\big) < \sqrt{1 + K_R} + 1$. Let us pick $C_a := \exp\big((1 + 1/K_R)\sqrt{E_H E_W}\big)$, $C_b := \exp\Big(-\frac{1}{K_A - 1}(1 + 1/K_R)\sqrt{E_H E_W}\Big)$, and $C_c := 1$.

Following from $\overset{b}{\geq}$ in (58), we know if $1/K_R - C_f^2 - \frac{C_f^4}{C_e(2 - C_f)} > 0$, then

$$\tfrac{1}{K_A n_A} \textstyle\sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq -C_g(1 + 1/K_R)\sqrt{E_H E_W} + C_d := L_2. \tag{61}$$

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

$$\begin{aligned}
& 1/K_R > C_f^2 - \frac{C_f^4}{C_e(2 - C_e)} && (\text{by } C_e + C_f = 1) \\
\iff \quad & C_e > \sqrt{\frac{1}{1 + K_R}} && \left(\text{by } C_e = \frac{K_B C_c}{K_A C_b + K_B C_c}\right) \\
\iff \quad & \frac{C_b}{C_c} > \frac{1}{\sqrt{1 + K_R} + 1} \\
\iff \quad & \exp\Big((1 + 1/K_R)\sqrt{E_H E_W}/(K_A - 1)\Big) < \sqrt{1 + K_R} + 1.
\end{aligned}$$

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

$$\begin{aligned}
[\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{K_A}] =& \sqrt{\frac{E_W}{E_H}} \left[\boldsymbol{h}_1, \ldots, \boldsymbol{h}_{K_A}\right]^\top = \sqrt{(1 + 1/K_R) E_W}\ (\boldsymbol{M}_A^\star)^\top, \\
\boldsymbol{h}_{k,i} =& \boldsymbol{h}_k, \quad k \in [K_A],\ i \in [n_A] \\
\boldsymbol{h}_{k,i} =& \boldsymbol{w}_k = \boldsymbol{0}_p, \quad k \in [K_A + 1 : K],\ i \in [n_B],
\end{aligned}$$

where $\boldsymbol{M}_A^\star$ is a K_A-simplex ETF, $(\boldsymbol{H}, \boldsymbol{W})$ can achieve the equality in (61). So L_2 is the global minimum of (52). Moreover, L_2 is achieved only if the equality in $\overset{a}{\geq}$ of (58) holds. From Lemma 5, we have 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 (60) into (58), we have

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq -\frac{C_g}{K_A} \sqrt{KE_H} \sqrt{KE_W - \varepsilon^2/2} + C_d := L_2 + \varepsilon_2, \tag{62}$$

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 (1) $g_b(0)/g_c(0) \geq \frac{1/C'_f - 1}{K_R}$ and (2) $g_b(1)/g_c(1) \leq \frac{1/C'_f - 1}{K_R}$.

(1) 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.

(2) 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 (58) that

$$\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \geq -C_g(1 + 1/K_R)\sqrt{E_H E_W} + C_d = L_2. \tag{63}$$

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

$$\begin{aligned}
\boldsymbol{w}_k &= g_N(x_0)\boldsymbol{P}_A \left[\frac{x_0}{\sqrt{(K_A - 1)K_A}}(K_A \boldsymbol{y}_k - \boldsymbol{1}_{K_A}) + \frac{1 - x_0}{\sqrt{K_A}} \boldsymbol{1}_{K_A} \right], \quad k \in [K_A], \\
\boldsymbol{w}_k &= -\frac{C_e(2 - C_e)}{C_f^2 K_A} \boldsymbol{P}_A \sum_{k=1}^{K_A} \boldsymbol{w}_k, \quad k \in [K_A + 1 : K], \\
\boldsymbol{h}_{k,i} &= \frac{\sqrt{(1 + 1/K_R)E_H}}{\| \boldsymbol{w}_i + \frac{C_e}{C_f K_A} \sum_{k=1}^{K_A} \boldsymbol{w}_k \|} \boldsymbol{P}_A \left[\boldsymbol{w}_i + \frac{C_e}{C_f K_A} \sum_{k=1}^{K_A} \boldsymbol{w}_k \right], \quad k \in [K_A],\ i \in [n_A], \\
\boldsymbol{h}_{k,i} &= \boldsymbol{0}_p, \quad k \in [K_A + 1 : K],\ i \in [n_B],
\end{aligned}$$

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 (63). Finally, following a same argument as Case (ii), we have that (1) L_2 is the global minimum of (52); (2) any minimizer satisfies that $\boldsymbol{h}_{k,i} = \boldsymbol{0}_p$ 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 (62) holds.

Combining the three cases, we obtain Lemma 4, completing the proof. □

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 $C_g := \frac{K_A C_b + K_B C_c}{C_a + (K_A - 1)C_b + K_B C_c} > 0$. For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ of (52), the objective value of (52) can be bounded from below by:*

$$\begin{aligned}
&\frac{1}{K_A n_A} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \qquad (64) \\
&\overset{a}{\geq} -\frac{C_g}{K_A}\sqrt{K E_H} \sqrt{\sum_{k=1}^{K_A} \|C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k\|^2} + C_d \\
&\overset{b}{\geq} -\frac{C_g}{K_A}\sqrt{K E_H} \sqrt{K E_W - K_A \left(1/K_R - C_f^2 - \frac{C_f^4}{C_e(2 - C_e)}\right) \|\boldsymbol{w}_B\|^2 - \sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2} + C_d,
\end{aligned}$$

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* $\boldsymbol{h}_{k,i} = \boldsymbol{0}_p$ *for all* $k \in [K_A + 1 : K]$.

Proof of Lemma 5. 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

$$
\begin{aligned}
&-\log\left(\frac{\exp(\boldsymbol{z}_{k,i}(i))}{\sum_{k'=1}^{K}\exp(\boldsymbol{z}_{k',i}(k))}\right)\\
=&-\boldsymbol{z}_{k,i}(k) + \log\left(C'_a\left(\frac{\exp(z_{k,i}(k))}{C'_a}\right) + \sum_{k'=1,\ k'\neq k}^{K_A} C'_b\left(\frac{\exp(z_{k,i}(k'))}{C'_b}\right) + \sum_{k'=K_A+1}^{K} C'_c\left(\frac{\exp(z_{k,i}(k'))}{C'_c}\right)\right)\\
\geq&-\boldsymbol{z}_{k,i}(k) + C'_a\boldsymbol{z}_{k,i}(k) + C'_b\sum_{k'=1,\ k'\neq k}^{K_A}\boldsymbol{z}_{k,i}(k') + C'_C\sum_{k'=K_A+1}^{K}\boldsymbol{z}_{i,j}(k) + C_d\\
=&C_gC_e\left(\frac{1}{K_A}\sum_{k'=1}^{K_A}\boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right) + C_gC_f\left(\frac{1}{K_B}\sum_{k'=K_A+1}^{K}\boldsymbol{z}_{k,i}(k') - \boldsymbol{z}_{k,i}(k)\right) + C_d.
\end{aligned}
$$

Therefore, integrating (65) 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

$$
\begin{aligned}
&\frac{1}{K_An_A}\sum_{k=1}^{K_A}\sum_{i=1}^{n_A}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) \qquad (65)\\
\geq&\frac{1}{K_An_A}\sum_{k=1}^{K_A}\sum_{i=1}^{n_A}C_g\left[C_e(\boldsymbol{h}_{k,i}\boldsymbol{w}_A - \boldsymbol{h}_{k,i}\boldsymbol{w}_k) + C_f(\boldsymbol{h}_{k,i}\boldsymbol{w}_B - \boldsymbol{h}_{k,i}\boldsymbol{w}_k)\right] + C_d\\
\overset{a}{=}&\frac{C_g}{K_A}\sum_{k=1}^{K_A}\boldsymbol{h}_k^\top(C_e\boldsymbol{w}_A + C_f\boldsymbol{w}_B - \boldsymbol{w}_k) + C_d,
\end{aligned}
$$

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

$$
\begin{aligned}
\sum_{k=1}^{K_A}\boldsymbol{h}_k^\top(C_e\boldsymbol{w}_A + C_f\boldsymbol{w}_B - \boldsymbol{w}_k) \geq&-\sqrt{\sum_{k=1}^{K_A}\|\boldsymbol{h}_k\|^2}\sqrt{\sum_{k=1}^{K_A}\|C_e\boldsymbol{w}_A + C_f\boldsymbol{w}_B - \boldsymbol{w}_k\|^2}\\
\overset{a}{\geq}&-\sqrt{\sum_{k=1}^{K_A}\frac{1}{n_k}\sum_{i=1}^{n_k}\|\boldsymbol{h}_{k,i}\|^2}\sqrt{\sum_{k=1}^{K_A}\|C_e\boldsymbol{w}_A + C_f\boldsymbol{w}_B - \boldsymbol{w}_k\|^2}\\
\overset{b}{\geq}&-\sqrt{KE_H}\sqrt{\sum_{k=1}^{K_A}\|C_e\boldsymbol{w}_A + C_f\boldsymbol{w}_B - \boldsymbol{w}_k\|^2}, \qquad (66)
\end{aligned}
$$

where $\overset{a}{\geq}$ follows from Jensens'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 $\frac{1}{K}\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i=1}^{n_k}\|\boldsymbol{h}_{k,i}\|^2 \leq E_H$. Moreover, we have $\sum_{k=1}^{K_A}\frac{1}{n_k}\sum_{i=1}^{n_k}\|\boldsymbol{h}_{k,i}\|^2 = E_H$ only if $\boldsymbol{h}_{k,i} = \boldsymbol{0}_p$ for all $k \in [K_A + 1 : K]$. Plugging (66) into (65), we obtain $\overset{a}{\geq}$ in (64).

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

$$\begin{aligned}
&\frac{1}{K_A}\sum_{k=1}^{K_A} \|C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B - \boldsymbol{w}_k\|^2 \\
=&\frac{1}{K_A}\sum_{k=1}^{K_A} \|\boldsymbol{w}_k\|^2 - 2\frac{1}{K_A}\sum_{k=1}^{K_A} \boldsymbol{w}_k \cdot (C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B) + \|C_e \boldsymbol{w}_A + C_f \boldsymbol{w}_B\|^2 \\
\overset{a}{=}&\frac{1}{K_A}\sum_{k=1}^{K_A} \|\boldsymbol{w}_k\|^2 - 2C_f^2 \boldsymbol{w}_A^\top \boldsymbol{w}_B - C_e(2-C_e)\|\boldsymbol{w}_A\|^2 + C_f^2\|\boldsymbol{w}_B\|^2. \qquad (67)
\end{aligned}$$

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

$$\begin{aligned}
&\frac{1}{K_A}\sum_{k=1}^{K_A} \|\boldsymbol{w}_k\|^2 - 2C_f^2 \boldsymbol{w}_A^\top \boldsymbol{w}_B - C_e(2-C_e)\|\boldsymbol{w}_A\|^2 + C_f^2\|\boldsymbol{w}_B\|^2 \qquad (68) \\
\le&\frac{K}{K_A}E_W^2 - \frac{1}{K_A}\sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k\|^2 - C_e(2-C_f)\left\|\boldsymbol{w}_A + \frac{C_f^2}{C_e(2-C_e)}\boldsymbol{w}_B\right\|^2 + \left(C_f^2 + \frac{C_f^4}{C_e(2-C_e)}\right)\|\boldsymbol{w}_B\|^2 \\
\overset{a}{=}&\frac{K}{K_A}E_W^2 - \left(1/K_R - C_f^2 - \frac{C_f^4}{C_e(2-C_e)}\right)\|\boldsymbol{w}_B\|^2 - \frac{1}{K_A}\sum_{k=K_A+1}^{K} \|\boldsymbol{w}_k - \boldsymbol{w}_B\|^2,
\end{aligned}$$

where $\overset{a}{\ge}$ 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 (67) and (68) into $\overset{a}{\ge}$ in (64), we obtain $\overset{b}{\ge}$ in (64), completing the proof. □

B Additional Results

Comparison of Oversampling and Weighted Adjusting. Oversampling and weight adjusting are two commonly-used tricks in deep learning [JK19]. Both of them actually consider the same objective as (16), 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, [BCN18, FLLZ18, FLZ19]). 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

$$\boldsymbol{W}_{\text{full}}^{t+1} = \boldsymbol{W}_{\text{full}}^{t} - \eta_w \boldsymbol{v}_w^t, \quad t = 0, 1, 2, \ldots, \qquad (69)$$

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

$$\begin{aligned} &\mathbb{E}\left[\boldsymbol{v}_w^t \mid \boldsymbol{W}_{\text{full}}^t\right] \\ =&\frac{1}{n_A K_A + n_B K_B} \left[\sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) + w_r \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B} \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right], \end{aligned} \tag{70}$$

and

$$\begin{aligned} \mathbb{E}\left[\|\boldsymbol{v}_w^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] =& \frac{1}{n_A K_A + n_B K_B} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \left\| \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right\|^2 \\ &+ \frac{w_r^2}{n_A K_A + n_B K_B} \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B} \left\| \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right\|^2. \end{aligned} \tag{71}$$

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

$$\boldsymbol{W}_{\text{full}}^{t+1} = \boldsymbol{W}_{\text{full}}^t - \eta_s \boldsymbol{v}_s^t, \quad t = 0, 1, 2, \ldots, \tag{72}$$

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

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

and

$$\begin{aligned} \mathbb{E}\left[\|\boldsymbol{v}_s^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] =& \frac{1}{n_A K_A + w_r n_B K_B} \sum_{k=1}^{K_A} \sum_{i=1}^{n_A} \left\| \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right\|^2 \\ &+ \frac{w_r}{n_A K_A + w_r n_B K_B} \sum_{k=K_A+1}^{K} \sum_{i=1}^{n_B} \left\| \nabla_{\boldsymbol{W}_{\text{full}}} \mathcal{L}(f(\boldsymbol{x}_{k,i}; \boldsymbol{W}_{\text{full}}^t), \boldsymbol{y}_k) \right\|^2. \end{aligned} \tag{73}$$

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 (71) with (73), 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 (73) implies that $\mathbb{E}\left[\|\boldsymbol{v}_s^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] \asymp \eta_s^2 a^2$. However, (71) reads that $\mathbb{E}\left[\|\boldsymbol{v}_w^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] \asymp \eta_s^2 \frac{n_A K_A + w_r^2 n_B K_B}{n_A K_A + w_r n_B K_B} a^2$. Furthermore, if we set $w_r \asymp n_A/n_B$, then $\mathbb{E}\left[\|\boldsymbol{v}_w^t\|^2 \mid \boldsymbol{W}_{\text{full}}^t\right] \asymp \eta_s^2 w_r a^2$. 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 (7) can also be relaxed as a nuclear norm-constrained convex optimization. The result heavily relies on the progress of matrix decomposition, e.g. [BMP08, HV19]. We will the use the equality (see e.g., [BMP08, Section 2]) that for any matrix $\boldsymbol{Z}$ and $a > 0$,

$$\|\boldsymbol{Z}\|_* = \inf_{r \in \mathbb{N}_+} \inf_{\boldsymbol{U}, \boldsymbol{V}: \boldsymbol{U}\boldsymbol{V}^\top = \boldsymbol{Z}} \frac{a}{2}\|\boldsymbol{U}\|^2 + \frac{1}{2a}\|\boldsymbol{V}\|^2, \tag{74}$$

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

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

$$\boldsymbol{h}_k = \frac{1}{n_k}\sum_{i=1}^{n_k} \boldsymbol{h}_{k,i},\ k \in [K], \quad \tilde{\boldsymbol{H}} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_K] \in \mathbb{R}^{p \times K}, \ \text{ and } \ \boldsymbol{Z} = \boldsymbol{W}\tilde{\boldsymbol{H}} \in \mathbb{R}^{K \times K}. \tag{75}$$

We consider the convex program:

$$\begin{aligned} \min_{\boldsymbol{Z} \in \mathbb{R}^{K \times K}} \quad & \sum_{k=1}^{K} \frac{n_k}{N} \mathcal{L}(\boldsymbol{Z}_k, \boldsymbol{y}_k) \\ \text{s.t.} \quad & \|\boldsymbol{Z}\|_* \le K\sqrt{E_H E_W}. \end{aligned} \tag{76}$$

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

Lemma 6. *Assume $p \ge K$ and the loss function $\mathcal{L}$ is convex on the first argument. Let $\boldsymbol{Z}^\star$ be a minimizer of the convex program (76). 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*

$$\begin{aligned} &\boldsymbol{W} = \boldsymbol{U}^\star \boldsymbol{\Sigma}_1^\star \boldsymbol{P}^\top, \quad [\boldsymbol{h}_1^\star, \boldsymbol{h}_2^\star, \ldots, \boldsymbol{h}_K^\star] = \boldsymbol{P}\boldsymbol{\Sigma}_2^\star \boldsymbol{V}^\star, \\ &\boldsymbol{h}_{k,i}^\star = \boldsymbol{h}_k^\star, \quad k \in [K],\ i \in [n_k], \end{aligned} \tag{77}$$

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 (7).

Proof of Lemma 6. For any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$ for the original program (7), define $\boldsymbol{h}_k$ for $k \in [K]$, $\tilde{\boldsymbol{H}}$, and $\boldsymbol{Z}$ by (75). We show $\boldsymbol{Z}$ is a feasible solution for the convex program (76). In fact, by (74) with $r = K$ and $a = \sqrt{E_H/E_W}$, we have

$$\begin{aligned}\|\boldsymbol{Z}\|_* &\le \frac{\sqrt{E_H/E_W}}{2}\|\boldsymbol{W}\|^2 + \frac{\sqrt{E_W/E_H}}{2}\left\|\tilde{\boldsymbol{H}}\right\|^2 \\ &\overset{a}{\le} \frac{\sqrt{E_H/E_W}}{2}\sum_{k=1}^{K}\|\boldsymbol{w}_k\|^2 + \frac{\sqrt{E_W/E_H}}{2}\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i=1}^{n_k}\|\boldsymbol{h}_{k,i}\|^2 \\ &\le K\sqrt{E_H E_W}, \end{aligned} \tag{78}$$

where $\overset{a}{\le}$ applies Jensen's inequality as:

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

Let L_0 be the global minimum of the convex problem (76). Since $\mathcal{L}$ is convex on the first argument, by the same argument as (50), we obtain, for any feasible solution $(\boldsymbol{H}, \boldsymbol{W})$,

$$\begin{aligned}\frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k) &= \sum_{k=1}^{K}\frac{n_k}{N}\left[\frac{1}{n_k}\sum_{k=1}^{n_k}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_{k,i}, \boldsymbol{y}_k)\right] \\ &\ge \sum_{k=1}^{K}\frac{n_k}{N}\mathcal{L}(\boldsymbol{W}\boldsymbol{h}_k, \boldsymbol{y}_k) = \sum_{k=1}^{K}\frac{n_k}{N}\mathcal{L}(\boldsymbol{Z}_k, \boldsymbol{y}_k) \ge L_0. \end{aligned} \tag{79}$$

On the other hand, for the solution $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ defined in (77) with $\boldsymbol{Z}^\star$, we can verify that $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a feasible solution for (7) and

$$\frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\mathcal{L}(\boldsymbol{W}^\star\boldsymbol{h}^\star_{k,i}, \boldsymbol{y}_k) = \sum_{k=1}^{K}\frac{n_k}{N}\mathcal{L}(\boldsymbol{Z}^\star_k, \boldsymbol{y}_k) = L_0. \tag{80}$$

Combining (79) and (80), we have that L_0 is the global minimum of (7) and $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a minimizer. □

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

(A) Any minimizer $\boldsymbol{Z}^\star$ of (76) satisfies that $\|\boldsymbol{Z}\|_* = \sqrt{E_H E_W}$.

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

$$\frac{1}{K}\sum_{k=1}^{K}\frac{1}{n}\sum_{i=1}^{n}\left\|\boldsymbol{h}^\star_{k,i}\right\|^2 = E_H, \quad \textit{and} \qquad \frac{1}{K}\sum_{k=1}^{K}\|\boldsymbol{w}^\star_k\|^2 = E_W.$$

(C) Any minimizer $\boldsymbol{X}^\star$ of (14) satisfies that

$$\frac{1}{K}\sum_{k=1}^{K}\boldsymbol{X}^\star(k,k) = E_H, \quad \textit{and} \qquad \frac{1}{K}\sum_{k=K+1}^{2K}\boldsymbol{X}^\star(k,k) = E_W.$$

Proof of Property 1. We first prove (A). Let $\boldsymbol{Z}^\star$ be any minimier of (76). 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 $\boldsymbol{Z}$ will reach the boundary of the constraint, achieving (A).

For (B), suppose there is a minimizer $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ of (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 $\frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{w}_k^\star\|^2 < E_W$. Letting $\boldsymbol{Z}^\star$ defined as (75), it follows from (79) that $\boldsymbol{Z}^\star$ is a minimizer of (76). However, by (78), we have $\|\boldsymbol{Z}^\star\|_* < \sqrt{E_H E_W}$, which is contradictory to (A). We obtain (B).

For (C), suppose there is a minimizer $\boldsymbol{X}^\star$ of (14) 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 in (15), $(\boldsymbol{H}^\star, \boldsymbol{W}^\star)$ is a minimizer of (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 $\frac{1}{K} \sum_{k=1}^{K} \|\boldsymbol{w}_k^\star\|^2 < E_W$, which is contradictory to (B). We complete the proof.

□