Generalized Neural Collapse for a Large Number of Classes

Jiachen Jiang [*1] **Jinxin Zhou** [*1] **Peng Wang** [2] **Qing Qu** [2] **Dustin G. Mixon** [3] **Chong You** [4] **Zhihui Zhu** [1]

Abstract

Neural collapse provides an elegant mathematical characterization of learned last-layer representations, also known as features, and classifier weights within deep classification models. The result not only provides insights into deep models but also catalyzes the development of new techniques for improving them. However, most of the existing empirical and theoretical studies into neural collapse center around scenarios where the number of classes is small relative to the dimensionality of the feature space. This paper introduces a generalization of neural collapse to encompass scenarios where the number of classes surpasses the dimension of feature space, which broadly occurs for language models, information retrieval systems, and face recognition applications. A key technical contribution is the introduction of the concept of *softmax code*, defined as a collection of points that maximizes the minimum one-vs-rest margin, to describe the arrangement of class-mean features. We provide empirical study to verify the prevalence of generalized neural collapse in practical deep neural networks. Moreover, we provide theoretical study to show that the generalized neural collapse provably occurs under an unconstrained feature model with spherical constraint, subject to specific technical conditions on feature dimension and the number of classes.

1. Introduction

Over the past decade, deep neural networks (DNNs) have achieved remarkable success across numerous machine learning tasks and have significantly enhanced the state-of-the-art in many practical applications including computer vision, natural language processing, and information retrieval systems. Despite their tremendous success, a comprehensive understanding of how DNNs work is still lacking. Towards demystifying DNNs, the recent work Papyan et al. (2020); Papyan (2020) examined the last-layer features and classifier of DNNs and empirically uncovered an intriguing phenomenon called *Neural Collapse* ($\mathcal{NC}$), which can be briefly summarized as the following characteristics:

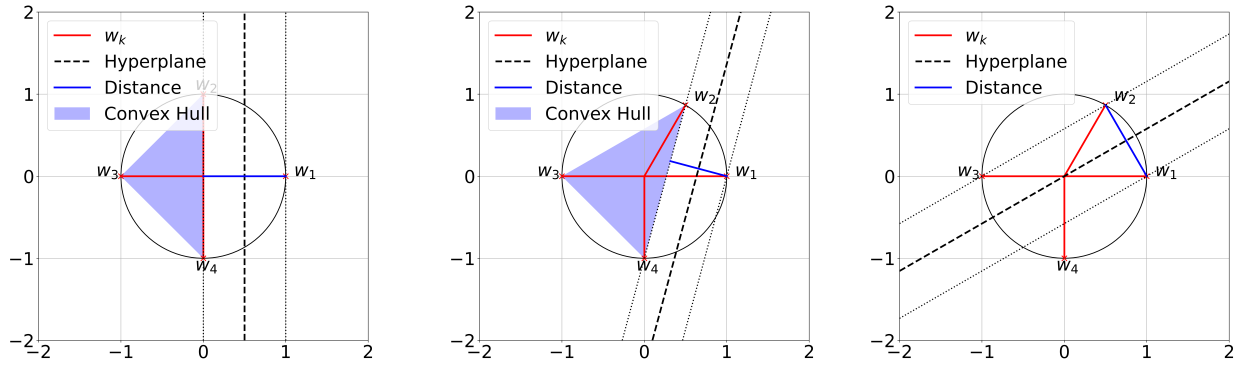
- ***Variability Collapse*** **($\mathcal{NC}_1$):** Within-class variability of features collapses to zero.
- ***Convergence to Simplex ETF*** **($\mathcal{NC}_2$):** Class-mean features converge to a simplex Equiangular Tight Frame (ETF), achieving equal lengths, equal pair-wise angles, and maximal distance in the feature space.
- ***Self-Duality*** **($\mathcal{NC}_3$):** Linear classifiers converge to class-mean features, up to a global rescaling.

Neural collapse provides a mathematically elegant characterization of learned representations or features in deep learning based classification models, independent of network architectures, dataset properties, and optimization algorithms. Building on the so-called *unconstrained feature model* (Mixon et al., 2020) or the *layer-peeled model* (Fang et al., 2021), subsequent research (Zhu et al., 2021; Lu & Steinerberger, 2020; Ji et al., 2021; Yaras et al.; Wojtowytsch et al., 2020; Ji et al.; Zhou et al.; Han et al.; Tirer & Bruna, 2022; Zhou et al., 2022a; Poggio & Liao, 2020; Thrampoulidis et al., 2022; Tirer et al., 2023; Nguyen et al., 2022; Li et al., 2024) has provided theoretical evidence for the existence of the $\mathcal{NC}$ phenomenon when using a family of loss functions including cross-entropy (CE) loss, mean-square-error (MSE) loss and variants of CE loss. Theoretical results regarding $\mathcal{NC}$ not only contribute to a new understanding of the working of DNNs but also provide inspiration for developing new techniques to enhance their practical performance in various settings, such as imbalanced learning (Xie et al., 2023; Liu et al., 2023b), transfer learning (Galanti et al., 2022a; Li et al., 2022; Xie et al., 2022; Galanti et al., 2022b), continual learning (Yu et al., 2022; Yang et al., 2023), loss and architecture designs (Chan et al., 2022; Yu et al., 2020; Zhu et al., 2021; Wang et al., 2024), etc.

However, most of the existing empirical and theoretical stud-

[*]Equal contribution [1]Department of Computer Science, The Ohio State University, Columbus, OH, USA [2]Department of Electrical Engineering & Computer Science, University of Michigan, Ann Arbor, MI, USA [3]Department of Mathematics, The Ohio State University, Columbus, OH, USA [4]Google Research, New York City, NY, USA. Correspondence to: Chong You and Zhihui Zhu <cyou@google.com and zhu.3440@osu.edu>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.



(a) One-vs-rest distance: Case 1 (b) One-vs-rest distance: Case 2 (c) One-vs-one distance

Figure 1. In Generalized Neural Collapse ($\mathcal{GNC}$), the optimal classifier weight $\{\boldsymbol{w}_k\}$ is a *Softmax Code* defined from maximizing the *one-vs-rest distance* (see Definition 2.1). *(a, b)* Illustration of the one-vs-rest distance using the example of $\boldsymbol{w}_1$-vs-$\{\boldsymbol{w}_2, \boldsymbol{w}_3, \boldsymbol{w}_4\}$ distance, under two configurations of $\{\boldsymbol{w}_k\}_{k=1}^4$ in a two-dimensional space. The distance in Case 1 is larger than that in Case 2. *(c)* Illustration of the *one-vs-one distance* used to define the Tammes problem (see Eq. (8)). We prove $\mathcal{GNC}$ under technical conditions on Softmax Code and Tammes problem (see Section 3).

ies in $\mathcal{NC}$ focus on the case that the number of classes is small relative to the dimension of the feature space. Nevertheless, there are many cases in practice where the number of classes can be very large, such as

- Person identification (Deng et al., 2019), where each identity is regarded as one class.
- Language models (Devlin et al., 2018), where the number of classes equals the vocabulary size[1].
- Retrieval systems (Mitra et al., 2018), where each document in the dataset represents one class.
- Contrastive learning (Chen et al., 2020a), where each training data can be regarded as one class.

In such cases, it is usually infeasible to have a feature dimension commensurate with the number of classes due to computational and memory constraints. Therefore, it is crucial to develop a comprehensive understanding of the characteristics of learned features in such cases, particularly with the increasing use of web-scale datasets that have a vast number of classes.

Contributions. This paper studies the geometric properties of the learned last-layer features and the classifiers for the cases where the number of classes can be arbitrarily large compared to the feature dimension. Motivated by the use of spherical constraints in learning with a large number of classes, such as person identification and contrastive learning, we consider networks trained with *spherical constraints* on the features and classifiers. Our contributions can be summarized as follows.

- **The Arrangement Problem: Generalizing $\mathcal{NC}$ to a Large Number of Classes.** In Section 2 we introduce the generalized $\mathcal{NC}$ ($\mathcal{GNC}$) for describing the last-layer features and classifier. In particular, $\mathcal{GNC}_1$ and $\mathcal{GNC}_3$ state the same as $\mathcal{NC}_1$ and $\mathcal{NC}_3$, respectively. $\mathcal{GNC}_2$ states that the classifier weight is a *Softmax Code*, which generalizes the notion of a simplex ETF and is defined as the collection of points on the unit hyper-sphere that maximizes the minimum one-vs-all distance (see Figure 1 (a,b) for an illustration). Empirically, we verify that the $\mathcal{GNC}$ approximately holds in practical DNNs trained with a small temperature in CE loss. Furthermore, we conduct theoretical study in Section 3 to show that under the unconstrained features model (UFM) (Mixon et al., 2020; Fang et al., 2021; Zhu et al., 2021) and with a vanishing temperature, the global solutions satisfy $\mathcal{GNC}$ under technical conditions on Softmax Code and solutions to the Tammes problem (Tammes, 1930), the latter defined as a collection of points on the unit hyper-sphere that maximizes the minimum one-vs-one distance (see Figure 1(c) for an illustration).

- **The Assignment Problem: Implicit Regularization of Class Semantic Similarity.** Unlike the simplex ETF (which is used to describe $\mathcal{NC}_2$) where the distance between any pair of vectors is the same, not all pairs in a Softmax Code are of equal distant when the number of classes is greater than the feature dimension. This leads to the "assignment" problem, i.e., the correspondence between the classes and the weights in a Softmax Code. In Section 4, we show empirically an implicit regularization effect by the semantic similarity of the classes, i.e., conceptually similar classes (e.g., Cat and Dog) are often assigned to closer classifier weights in Softmax Code,

[1]Language models are usually trained to classify a token (or a collection of them) that is either masked in the input (as in BERT (Devlin et al., 2018)), or the next one following the context (as in language modeling), or a span of masked tokens in the input (as in T5 (Raffel et al., 2020)). In such cases, the number of classes equals the number of all possible tokens, i.e., the vocabulary size.

compared to those that are conceptually dissimilar (e.g., Cat and Truck). Moreover, such an implicit regularization is beneficial, i.e., enforcing other assignments produces inferior model quality.

- **Cost Reduction for Practical Network Training/Fine-tuning.** The universality of alignment between classifier weights and class means (i.e., $\mathcal{GNC}_3$) implies that training the classifier is unnecessary and the weight can be simply replaced by the class-mean features. Our experiments in Section 5 show that such a strategy achieves comparable performance to classical training methods, and even better out-of-distribution performance than classical fine-tuning methods with significantly reduced parameters.

Related work. The recent work Liu et al. (2023a) also introduces a notion of generalized $\mathcal{NC}$ for the case of large number of classes, which predicts equal-spaced features. However, their work focuses on networks trained with weight decay, for which empirical results in Appendix B.2 and Yaras et al. (2023) show to not produce equal-length and equal-spaced features for a relatively large number of classes. Due to limited space, we refer to Appendix B.2 for the detailed comparison between different geometric properties of the learned features and classifiers for the weight decay and spherical constraints formulations. Moreover, the work Liu et al. (2023a) relies on a specific choice of kernel function to describe the uniformity. Instead, we concretely define $\mathcal{GNC}_2$ through the softmax code. When preparing this submission, we notice a concurrent work Gao et al. (2023) that provides analysis for generalized $\mathcal{NC}$, but again for networks trained with weight decay. In addition, Gao et al. (2023) analyzes gradient flow for the corresponding UFM with a particular choice of weight decay, while our work studies the global optimality of the training problem. The work Zhou et al. (2022a) empirically shows that MSE loss is inferior to the CE loss when $K > d+1$, but no formal analysis is provided for CE loss. Finally, the global optimality of the UFM with spherical constraints has been studied in Lu & Steinerberger (2022); Yaras et al. (2023) but only for the cases $K \leq d+1$ or $K \to \infty$.

2. Generalized Neural Collapse for A Large Number of Classes

In this section, we begin by providing a brief overview of DNNs and introducing notations used in this study in Section 2.1. We will also introduce the concept of the UFM which is used in theoretical study of the subsequent section. Next, we introduce the notion of *Softmax Code* for describing the distribution of a collection of points on the unit sphere, which prepares us to present a formal definition of *Generalized Neural Collapse* and empirical verification of its validity in Section 2.2.

2.1. Basics Concepts of DNNs

A DNN classifier aims to learn a feature mapping $\phi_{\boldsymbol{\theta}}(\cdot) : \mathbb{R}^D \to \mathbb{R}^d$ with learnable parameters $\boldsymbol{\theta}$ that maps from input $\boldsymbol{x} \in \mathbb{R}^D$ to a deep representation called the feature $\phi_{\boldsymbol{\theta}}(\boldsymbol{x}) \in \mathbb{R}^d$, and a linear classifier $\boldsymbol{W} = [\boldsymbol{w}_1 \quad \boldsymbol{w}_2 \quad \cdots \quad \boldsymbol{w}_K] \in \mathbb{R}^{d\times K}$ such that the output (also known as the logits) $\Psi_{\boldsymbol{\Theta}}(\boldsymbol{x}) = \boldsymbol{W}^\top \phi_{\boldsymbol{\theta}}(\boldsymbol{x}) \in \mathbb{R}^K$ can make a correct prediction. Here, $\boldsymbol{\Theta} = \{\boldsymbol{\theta}, \boldsymbol{W}\}$ represents *all* the learnable parameters of the DNN.[2]

Given a balanced training set $\{(\boldsymbol{x}_{k,i}, \boldsymbol{y}_k)\}_{i\in[n],k\in[K]} \subseteq \mathbb{R}^D \times \mathbb{R}^K$, where $\boldsymbol{x}_{k,i}$ is the i-th sample in the k-th class and $\boldsymbol{y}_k$ is the corresponding one-hot label with all zero entries except for unity in the k-th entry, the network parameters $\boldsymbol{\Theta}$ are typically optimized by minimizing the CE loss

$$\min_{\boldsymbol{\Theta}} \frac{1}{nK} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}_{\text{CE}} \left(\Psi_{\boldsymbol{\Theta}} \left(\boldsymbol{x}_{k,i} \right), \boldsymbol{y}_k, \tau \right),$$

$$\mathcal{L}_{\text{CE}} \left(\boldsymbol{z}, \boldsymbol{y}_k, \tau \right) = -\log \Big(\frac{\exp(z_k/\tau)}{\sum_{j=1}^{K} \exp(z_j/\tau)} \Big). \tag{1}$$

In above, we assume that a spherical constraint is imposed on the feature and classifier weights and that the logit z_k is divided by the temperature parameter τ. This is a common practice when dealing with a large number of classes (Wang et al., 2018b; Chang et al., 2019; Chen et al., 2020a). Specifically, we enforce $\{\boldsymbol{w}_k, \phi_{\boldsymbol{\Theta}}(\boldsymbol{x}_{k,i})\} \subseteq \mathbb{S}^{d-1} := \{\boldsymbol{a} \in \mathbb{R}^d : \|\boldsymbol{a}\|_2 = 1\}$ for all $i \in [n]$ and $k \in [K]$. An alternative regularization is weight decay on the model parameters $\boldsymbol{\Theta}$, the effect of which we study in Appendix B.

To simplify the notation, we denote the *oblique manifold* embedded in Euclidean space by $\mathcal{OB}(d, K) := \left\{ \boldsymbol{W} \in \mathbb{R}^{d\times K} \mid \boldsymbol{w}_k \in \mathbb{S}^{d-1},\ \forall k \in [K] \right\}$. In addition, we denote the last-layer features centered at their global-mean features by $\boldsymbol{h}_{k,i} := \phi_{\boldsymbol{\theta}}(\boldsymbol{x}_{k,i}) - \sum_{k=1}^{K}\sum_{i=1}^{n} \phi_{\boldsymbol{\theta}}(\boldsymbol{x}_{k,i})$. We rewrite all the features in a matrix form as

$$\boldsymbol{H} := [\boldsymbol{H}_1 \quad \boldsymbol{H}_2 \quad \cdots \quad \boldsymbol{H}_K] \in \mathbb{R}^{d\times nK},$$
$$\text{with } \boldsymbol{H}_k := \begin{bmatrix} \boldsymbol{h}_{k,1} & \cdots & \boldsymbol{h}_{k,n} \end{bmatrix} \in \mathbb{R}^{d\times n}.$$

Also we denote by $\overline{\boldsymbol{h}}_k := \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{h}_{k,i}$ the class-mean feature for each class.

Unconstrained Features Model (UFM). The UFM (Mixon et al., 2020) or layer-peeled model (Fang et al., 2021), wherein the last-layer features are treated as free optimization variables, are widely used for theoretically understanding the $\mathcal{NC}$ phenomena. In this paper, we will consider

[2] We ignore the bias term in the linear classifier since (i) the bias term is used to compensate the global mean of the features and vanishes when the global mean is zero (Papyan et al., 2020; Zhu et al., 2021), (ii) it is the default setting across a wide range of applications such as person identification (Wang et al., 2018b; Deng et al., 2019), contrastive learning (Chen et al., 2020a; He et al., 2020), etc.

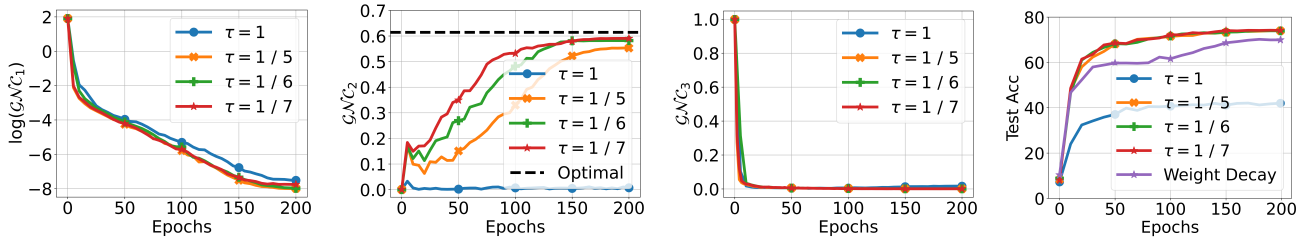


Figure 2. **Illustration of $\mathcal{GNC}$ and test accuracy across different temperatures** τ in training a ResNet18 on CIFAR100 with $d = 10$ and $K = 100$. "Optimal" in the second left figure refers to $\max_{\boldsymbol{W} \in \mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W})$. Please refer to Appendix B for more details of the optimization of the one-vs-rest distance.

the following UFM with a spherical constraint on classifier weights $\boldsymbol{W}$ and unconstrained features $\boldsymbol{H}$:

$$\min_{\boldsymbol{W},\boldsymbol{H}} \frac{1}{nK} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}_{\text{CE}}\left(\boldsymbol{W}^\top \boldsymbol{h}_{k,i}, \boldsymbol{y}_k, \tau\right) \quad (2)$$
$$\text{s.t.} \quad \boldsymbol{W} \in \mathcal{OB}(d,K), \ \boldsymbol{H} \in \mathcal{OB}(d,nK).$$

2.2. Generalized Neural Collapse

We start by introducing the notion of *softmax code* which will be used for describing $\mathcal{GNC}$.

Definition 2.1 (Softmax Code)**.** *Given positive integers d and K, a softmax code is an arrangement of K points on a unit sphere of $\mathbb{R}^d$ that maximizes the minimal distance between one point and the convex hull of the others:*

$$\max_{\boldsymbol{W} \in \mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}),$$
$$\rho_{\text{one-vs-rest}}(\boldsymbol{W}) \doteq \min_k \operatorname{dist}\Big(\boldsymbol{w}_k, \{\boldsymbol{w}_j\}_{j\in[K]\setminus k}\Big). \quad (3)$$

In above, the distance between a point $\boldsymbol{v}$ and a set $\mathcal{W}$ is defined as $\operatorname{dist}(\boldsymbol{v}, \mathcal{W}) = \inf_{\boldsymbol{w}\in\operatorname{conv}(\mathcal{W})}\{\|\boldsymbol{v} - \boldsymbol{w}\|\}$*, where* $\operatorname{conv}(\cdot)$ *denotes the convex hull of a set.*

We now extend $\mathcal{NC}$ to the *Generalized Neural Collapse* ($\mathcal{GNC}$) that captures the properties of the features and classifiers at the terminal phase of training. With a vanishing temperature (i.e., $\tau \to 0$), the last-layer features and classifier exhibit the following $\mathcal{GNC}$ phenomenon:

- ***Variability Collapse*** **($\mathcal{GNC}_1$).** All features of the same class collapse to the corresponding class mean. Formally, as used in (Papyan et al., 2020), the quantity $\mathcal{GNC}_1 \doteq \frac{1}{K} \operatorname{tr}\left(\boldsymbol{\Sigma}_W \boldsymbol{\Sigma}_B^\dagger\right) \to 0$, where $\boldsymbol{\Sigma}_B := \frac{1}{K}\sum_{k=1}^{K} \overline{\boldsymbol{h}}_k \overline{\boldsymbol{h}}_k^\top$ and $\boldsymbol{\Sigma}_W := \frac{1}{nK}\sum_{k=1}^{k}\sum_{i=1}^{n}\left(\boldsymbol{h}_{k,i} - \overline{\boldsymbol{h}}_k\right)\left(\boldsymbol{h}_{k,i} - \overline{\boldsymbol{h}}_k\right)^\top$ denote the between-class and within-class covariance matrices, respectively.
- ***Softmax Codes*** **($\mathcal{GNC}_2$).** Classifier weights converge to the softmax code in Definition 2.1. This property may be measured by $\mathcal{GNC}_2 \doteq \rho_{\text{one-vs-rest}}(\boldsymbol{W}) \to \max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W})$.
- ***Self-Duality*** **($\mathcal{GNC}_3$).** Linear classifiers converge to the class-mean features. Formally, this alignment can be measured by $\mathcal{GNC}_3 \doteq \frac{1}{K}\sum_{k=1}^{K}\left(1 - \boldsymbol{w}_k^\top \overline{\boldsymbol{h}}_k\right) \to 0$.

The main difference between $\mathcal{GNC}$ and $\mathcal{NC}$ lies in $\mathcal{GNC}_2$ / $\mathcal{NC}_2$, which describe the configuration of the classifier weight $\boldsymbol{W}$. In $\mathcal{NC}_2$, the classifier weights corresponding to different classes are described as a simplex ETF, which is a configuration of vectors that have equal pair-wise distance and that distance is maximized. Such a configuration does not exist in general when the number of classes is large, i.e., $K > d+1$. $\mathcal{GNC}_2$ introduces a new configuration described by the notion of softmax code. By Definition 2.1, a softmax code is a configuration where each vector is maximally separated from all the other points, measured by its distance to their convex hull. Such a definition is motivated from theoretical analysis (see Section 3). In particular, it reduces to simplex ETF when $K \leq d+1$ (see Theorem 3.3).

Interpretation of Softmax Code. Softmax Code abides a max-distance interpretation. Specifically, consider the features $\{\boldsymbol{h}_{k,i}\}_{k\in[K],i\in[n]}$ from K classes. In multi-class classification, one commonly used distance (or margin) measurement is the one-vs-rest (also called one-vs-all or one-vs-other) distance (Murphy, 2022), i.e., the distance of class k vis-a-vis other classes. Noting that the distance between two classes is equivalent to the distance between the convex hulls of the data from each class (Murphy, 2022), the distance of class k vis-a-vis other classes is given by $\operatorname{dist}(\{\boldsymbol{h}_{k,i}\}_{i\in[n]}, \{\boldsymbol{h}_{k',i}\}_{k'\in[K]\setminus k, i\in[n]})$. From $\mathcal{GNC}_1$ and $\mathcal{GNC}_3$ we can rewrite the distance as

$$\operatorname{dist}\left(\{\boldsymbol{h}_{k,i}\}_{i\in[n]}, \{\boldsymbol{h}_{k',i}\}_{k'\in[K]\setminus k, i\in[n]}\right) =$$
$$\operatorname{dist}\left(\overline{\boldsymbol{h}}_k, \{\overline{\boldsymbol{h}}_{k'}\}_{k'\in[K]\setminus k}\right) = \operatorname{dist}\left(\boldsymbol{w}_k, \{\boldsymbol{w}_{k'}\}_{k'\in[K]\setminus k}\right). \quad (4)$$

By noticing that the rightmost term is minimized in a Softmax Code, it follows from $\mathcal{GNC}_2$ that the learned features satisfy that their one-vs-rest distance minimized over all classes $k \in [K]$ is maximized. In other words, measured by one-vs-rest distance, the learned features are are maximally separated. Finally, we mention that the separation of classes may be characterized by other measures of distance as well,

such as the one-vs-one distance (also known as the sample margin in (Cao et al., 2019; Zhou et al., 2022b)) which leads to the well-known Tammes problem, or the distances captured in the Thomson problems (Thomson, 1904; Hars).We will discuss this in Section 3.2.

Experimental Verification of $\mathcal{GNC}$. We verify the occurence of $\mathcal{GNC}$ by training a ResNet18 (He et al., 2016) for image classification on the CIFAR100 dataset (Krizhevsky, 2009), and report the results in Figure 2. To simulate the case of $K > d+1$, we use a modified ResNet18 where the feature dimension is 10. From Figure 2, we can observe that both $\mathcal{GNC}_1$ and $\mathcal{GNC}_3$ converge to 0, and $\mathcal{GNC}_2$ converges towards the spherical code with relatively small temperature τ. Additionally, selecting a small τ is not only necessary for achieving $\mathcal{GNC}$, but also for attaining high testing performance. Due to limited space, we present experimental details and other experiments with different architectures and datasets in Appendix B. In the next section, we provide a theoretical justification for $\mathcal{GNC}$ under UFM in (2).

3. Theoretical Analysis of GNC

In this section, we provide a theoretical analysis of $\mathcal{GNC}$ under the UFM in (2). We first show in Section 3.1 that under appropriate temperature parameters, the solution to (2) can be approximated by the solution to a "HardMax" problem, which is of a simpler form amenable for subsequent analysis. We then provide a theoretical analysis of $\mathcal{GNC}$ in Section 3.2, by first proving the optimal classifier forms a Softmax Code ($\mathcal{GNC}_2$), and then establishing $\mathcal{GNC}_1$ and $\mathcal{GNC}_3$ under technical conditions on Softmax Code and solutions to the Tammes problem. In addition, we provide insights for the design of feature dimension d given a number of classes K by analyzing the upper and lower bound for the one-vs-rest distance of a Softmax Code. All proofs can be found in Appendix C.

3.1. Preparation: the Asymptotic CE Loss

Due to the nature of the softmax function which blends the output vector, analyzing the CE loss can be difficult even for the unconstrained features model. The previous work (Yaras et al., 2023) analyzing the case $K \leq d+1$ relies on the simple structure of the global solutions, where the classifiers form a simplex ETF. However, this approach cannot be directly applied to the case $K > d+1$ due to the absence of an informative characterization of the global solution. Motivated by the fact that the temperature τ is often selected as a small value ($\tau < 1$, e.g., $\tau = 1/30$ in (Wang et al., 2018b)) in practical applications (Wang et al., 2018b; Chen et al., 2020a), we consider the case of $\tau \to 0$ where the CE loss (2) converges to the "HardMax" [3] problem:

$$\min_{\substack{\boldsymbol{W}\in\mathcal{OB}(d,K)\\ \boldsymbol{H}\in\mathcal{OB}(d,nK)}} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W},\boldsymbol{H}), \text{ where}$$
$$\mathcal{L}_{\text{HardMax}}(\boldsymbol{W},\boldsymbol{H}) \doteq \max_{k\in[K]}\max_{i\in[n]}\max_{k'\neq k}\langle \boldsymbol{w}_{k'}-\boldsymbol{w}_k, \boldsymbol{h}_{k,i}\rangle, \tag{5}$$

where $\langle\cdot,\cdot\rangle$ denotes the inner-product operator. As the CE loss (2) may not have unique solutions, to circumvent the technical issue of defining the limit for uncountable sequence of sets (i.e., sets of solutions), by $\tau \to 0$ we will consider a countable sequence, i.e., letting $\tau = 1/p$ with $p \in \mathbb{Z}$ and $p \to \infty$. With this, we have the following result.

Lemma 3.1 (Convergence to the HardMax problem)**.** *For any positive integers K and n, we have*

$$\limsup_{\tau\to 0}\left(\operatorname*{arg\,min}_{\substack{\boldsymbol{W}\in\mathcal{OB}(d,K)\\ \boldsymbol{H}\in\mathcal{OB}(d,nK)}} \frac{1}{nK}\sum_{k=1}^{K}\sum_{i=1}^{n}\mathcal{L}_{CE}\left(\boldsymbol{W}^\top \boldsymbol{h}_{k,i}, \boldsymbol{y}_k, \tau\right)\right)$$
$$\subseteq \operatorname*{arg\,min}_{\substack{\boldsymbol{W}\in\mathcal{OB}(d,K)\\ \boldsymbol{H}\in\mathcal{OB}(d,nK)}} \mathcal{L}_{HardMax}(\boldsymbol{W},\boldsymbol{H}).$$

Our goal is not to replace CE with the HardMax function in practice. Instead, we will analyze the HardMax problem in (5) to gain insight into the global solutions and the $\mathcal{GNC}$ phenomenon.

3.2. Main Result: Theoretical Analysis of $\mathcal{GNC}$

$\mathcal{GNC}_2$ and Softmax Code. Our main result for $\mathcal{GNC}_2$ is the following.

Theorem 3.2 ($\mathcal{GNC}_2$)**.** *Let $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ be an optimal solution to (5). Then, it holds that $\boldsymbol{W}^\star$ is a Softmax Code,*

$$\boldsymbol{W}^\star = \operatorname*{arg\,max}_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}). \tag{6}$$

$\mathcal{GNC}_2$ is described by the Softmax Code, which is defined from an optimization problem (see Definition 2.1). This optimization problem may not have a closed form solution in general. Nonetheless, the one-vs-rest distance that is used to define Softmax Code has a clear geometric meaning, making an intuitive interpretation of Softmax Code tractable. Specifically, maximizing the one-vs-rest distance results in the classifier weight vectors $\{\boldsymbol{w}_k^\star\}$ to be maximally distant.

[3] While our main results focus on this asymptotic CE loss with a small temperature parameter, which may be a potential limitation, this asymptotic analysis offers several advantages. Firstly, a small temperature parameter is necessary for achieving a large margin, as indicated in Figure 2, aligning with common practices across various applications. Secondly, it provides a profound geometric interpretation, as discussed in the following parts.

As shown in Figures 1a and 1b for a simple setting of four classes in a 2D plane, the weight vectors $\{\boldsymbol{w}_k\}$ that are uniformly distributed (and hence maximally distant) have a larger margin than the non-uniform case.

For certain choices of (d, K) the Softmax Code bears a simple form.

Theorem 3.3. *For any positive integers K and d, let $\boldsymbol{W}^\star \in \mathcal{OB}(d, K)$ be a Softmax Code. Then,*

- *$d = 2$: $\{\boldsymbol{w}_k^\star\}$ is uniformly distributed on the unit circle, i.e., $\{\boldsymbol{w}_k^\star\} = \{(\cos(\frac{2\pi k}{K} + \alpha), \sin(\frac{2\pi k}{K} + \alpha))\}$ for some α;*
- *$K \leq d + 1$: $\{\boldsymbol{w}_k^\star\}$ forms a simplex ETF, i.e., $\boldsymbol{W}^\star = \sqrt{\frac{K}{K-1}}\boldsymbol{P}(\boldsymbol{I}_K - \frac{1}{K}\boldsymbol{1}_K\boldsymbol{1}_K^\top)$ for some orthonomal $\boldsymbol{P} \in \mathbb{R}^{d\times K}$;*
- *$d + 1 < K \leq 2d$: $\rho_{\text{one-vs-rest}}(\boldsymbol{W}^\star) = 1$ which can be achieved when $\{\boldsymbol{w}_k^\star\}$ are a subset of vertices of a cross-polytope[4];*

For the cases of $K \leq d + 1$, the optimal $\boldsymbol{W}^\star$ from Theorem 3.3 is the same as that of (Lu & Steinerberger, 2022). However, Theorem 3.3 is an analysis of the HardMax loss while (Lu & Steinerberger, 2022) analyzed the CE loss.

$\mathcal{GNC}_1$ and Within-class Variability Collapse. To establish the within-class variability collapse property, we require a technical condition associated with the Softmax Code. Recall that Softmax Codes are those that maximize the *minimum* one-vs-rest distance over all classes. We introduce *rattlers*, which are classes that do not attain such a *minimum*.

Definition 3.4 (Rattler of Softmax Code)**.** *Given positive integers d and K, a rattler associated with a Softmax Code $\boldsymbol{W}^{SC} \in \mathcal{OB}(d, K)$ is an index $k_{rattler} \in [K]$ for which*

$$\min_{k\in[K]} \operatorname{dist}(\boldsymbol{w}_k^{SC}, \{\boldsymbol{w}_j^{SC}\}_{j\in[K]\setminus k}) \neq \operatorname{dist}(\boldsymbol{w}_{k_{rattler}}^{SC}, \{\boldsymbol{w}_j^{SC}\}_{j\in[K]\setminus k_{rattler}}).$$

In other words, rattlers are points in a Softmax Code with no neighbors at the minimum one-to-rest distance. This notion is borrowed from the literature of the *Tammes Problem* (Cohn, 2022; Wang, 2009), which we will soon discuss in more detail[5].

We are now ready to present the main results for $\mathcal{GNC}_1$.

Theorem 3.5 ($\mathcal{GNC}_1$)**.** *Let $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ be an optimal solution to (5). For all k that is not a rattler of $\boldsymbol{W}^\star$, it holds that*

$$\begin{aligned}\overline{\boldsymbol{h}}_k^\star \doteq \boldsymbol{h}_{k,1}^\star = \cdots = \boldsymbol{h}_{k,n}^\star \\ = \mathcal{P}_{\mathbb{S}^{d-1}}\left(\boldsymbol{w}_k^\star - \mathcal{P}_{\{\boldsymbol{w}_j^\star\}_{j\in[K]\setminus k}}(\boldsymbol{w}_k^\star)\right),\end{aligned}$$

where $\mathcal{P}_{\mathcal{W}}(\boldsymbol{v}) \doteq \arg\min_{\boldsymbol{w}\in\operatorname{conv}(\mathcal{W})}\{\|\boldsymbol{v} - \boldsymbol{w}\|_2\}$ denotes the projection of $\boldsymbol{v}$ on the hypersphere of $\operatorname{conv}(\mathcal{W})$.

The following result shows that the requirement in the Theorem 3.5 that k is not a rattler is satisfied in certain cases.

Theorem 3.6. *If $d = 2$, or $K \leq d + 1$, Softmax Code has no rattler for all classes.*

$\mathcal{GNC}_3$ and Self-Duality. To motivate our technical conditions for establishing self-duality, assume that any optimal solution $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ to (5) satisfies self-duality as well as $\mathcal{GNC}_1$. This implies that

$$\begin{aligned}&\arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K), \boldsymbol{H}\in\mathcal{OB}(d,nK)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) \\ =\ &\arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,nK)} \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{w}_k\rangle. \quad (7)\end{aligned}$$

After simplification we may rewrite the optimization problem on the right hand side equivalently as:

$$\begin{aligned}&\max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-one}}(\boldsymbol{W}), \\ &\rho_{\text{one-vs-one}}(\boldsymbol{W}) \doteq \min_{k\in[K]} \min_{k'\neq k} \operatorname{dist}(\boldsymbol{w}_k, \boldsymbol{w}_{k'}). \quad (8)\end{aligned}$$

Eq. (8) is the well-known *Tammes problem*. Geometrically, the problem asks for a distribution of K points on the unit sphere of $\mathbb{R}^d$ so that the minimum distance between any pair of points is maximized. The Tammes problem is unsolved in general, except for certain pairs of (K, d).

Both the Tammes problem and the Softmax Code are problems of arranging points to be maximally separated on the unit sphere, with their difference being the specific measures of separation. Comparing (8) and (3), the Tammes problem maximizes for all $k \in [K]$ the *one-vs-one distance*, i.e., $\min_{k'\neq k} \operatorname{dist}(\boldsymbol{w}_k, \boldsymbol{w}_{k'})$, whereas the Softmax Code maximizes the minimum *one-vs-rest distance*, i.e., $\operatorname{dist}(\boldsymbol{w}_k, \{\boldsymbol{w}_j\}_{j\in[K]\setminus k})$. Both one-vs-one distance and one-vs-rest distances characterize the separation of the weight vector $\boldsymbol{w}_k$ from $\{\boldsymbol{w}_j\}_{j\in[K]\setminus k}$. As illustrated in Figure 1, taking $k = 1$, the former is the distance between $\boldsymbol{w}_1$ and its closest point in the set $\{\boldsymbol{w}_2, \boldsymbol{w}_3, \boldsymbol{w}_4\}$, in this case $\boldsymbol{w}_2$ (see Figure 1c), whereas the later captures the minimal distance from $\boldsymbol{w}_1$ to the convex hull of the rest vectors $\{\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3\}$ (see Figure 1b).

Since the Tammes problem can be derived from the self-duality constraint on the HardMax problem, it may not be surprising that the Tammes problem can be used to describe a condition for establishing self-duality. Specifically, we have the following result.

[4]Indeed, any sphere code $\boldsymbol{W}$ that achieves equality in Rankin's orthoplex bound (Fickus et al., 2017) $\max_{k\neq j}\langle \boldsymbol{w}_k, \boldsymbol{w}_j\rangle \geq 0$ is a softmax code.

[5]The occurrence of rattlers is rare: Among the 182 pairs of (d, K) for which the solution to Tammes problem is known, only 31 have rattlers (Cohn, 2022). This has excluded the cases of $d = 2$ or $K \leq 2d$ where there is no rattler. The occurrence of ratter in Softmax Code may be rare as well.

Theorem 3.7 ($\mathcal{GNC}_3$)**.** *For any K, d such that both Tammes problem and Softmax Code have no rattler, the following two statements are equivalent:*

- *Any optimal solution $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ to (5) satisfies $\boldsymbol{h}^\star_{k,i} = \boldsymbol{w}^*_k, \forall i \in [n], \forall k \in [K]$;*
- *The Tammes problem and the Softmax codes are equivalent, i.e.,* $\arg\max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}) = \arg\max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-one}}(\boldsymbol{W})$.

In words, Theorem 3.7 states that $\mathcal{GNC}_3$ holds if and only if the Tammes problem in (8) and the Softmax codes are equivalent. As both the Tammes problem and Softmax Code maximize separation between one vector and the others, though their notions of separation are different, we conjecture that they are equivalent and share the same optimal solutions. We prove this conjecture for some special cases and leave the study for the general case as future work[6].

Theorem 3.8. *If $d = 2$, or $K \leq d+1$, the Tammes problem and the Softmax codes are equivalent.*

3.3. Insights for Choosing Feature Dimension d Given Class Number K

Given a class number K, how does the choice of feature dimension d affect the model performance? Intuitively, smaller d reduces the separability between classes in a Softmax Code. We define this rigorously by providing bounds for the one-vs-rest distance of a Softmax Code based on d and K.

Theorem 3.9. *Assuming $K \geq \sqrt{2\pi\sqrt{e}d}$ and letting $\Gamma(\cdot)$ denote the Gamma function, we have*

$$\frac{1}{2}\left[\frac{\sqrt{\pi}}{K}\frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}+1\right)}\right]^{\frac{2}{d-1}} \leq \max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}) \leq 2\left[\frac{2\sqrt{\pi}}{K}\frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}\right]^{\frac{1}{d-1}}. \quad (9)$$

The bounds characterize the separability for K classes in d-dimensional space. Given the number of classes K and desired margin ρ, the minimal feature dimension is roughly an order of $\log(K^2/\rho)$, showing classes separate easily in higher dimensions. This also provides a justification for applications like face classification and self-supervised learning, where the number of classes (e.g., millions of classes) could be significantly larger than the dimensionality of the features (e.g., $d = 512$).

By conducting experiments on ResNet-50 with varying feature dimensions for ImageNet classification, we further corroborate the relationship between feature dimension and network performance in Figure 3. First, we observe that

[6]We numerically verify the equivalence for all the cases with $d \leq 100$ in Table 1 of (Cohn & Kumar, 2007).

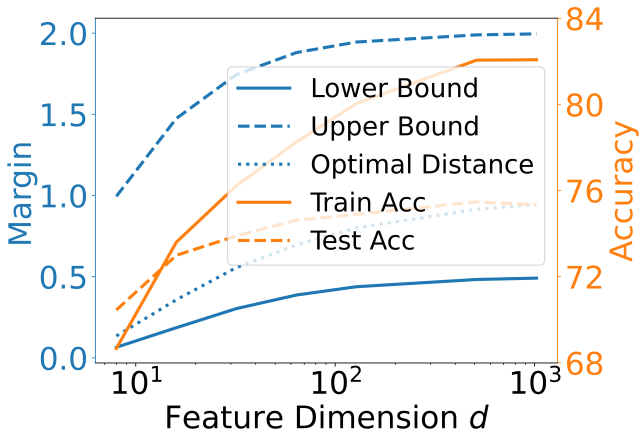


Figure 3. Effect of feature dimension d on (Left y-axis): $\rho_{\text{one-vs-rest}}(\boldsymbol{W}^\star)$ and its upper/lower bounds (in Theorem 3.9), and (Right y-axis): training and test accuracies on ImageNet.

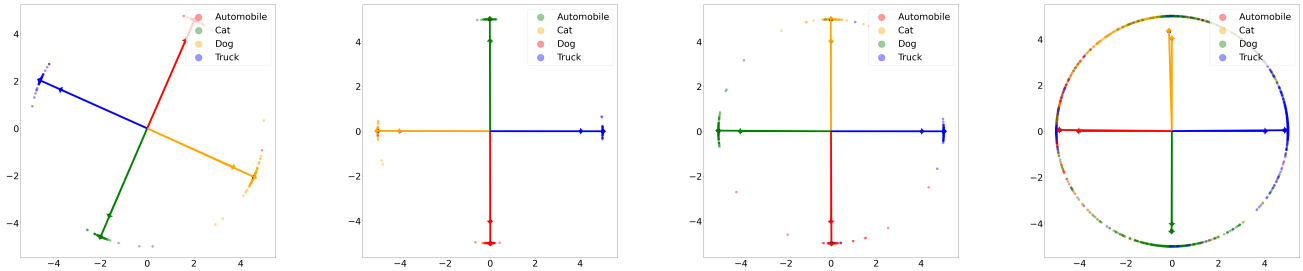
the curve of the optimal distance is closely aligned with the curve of testing performance, indicating a strong correlation between distance and testing accuracy. Moreover, both the distance and performance curves exhibit a slow (exponential) decrease as the feature dimension d decreases, which is consistent with the bounds in Theorem 3.9.

4. The Assignment Problem: An Empirical Study

Unlike the case $d \geq K-1$ where the optimal classifier (simplex ETF) has equal angles between any pair of the classifier weights, when $d < K-1$, not all pairs of classifier weights are equally distant with the optimal $\boldsymbol{W}$ (Softmax Code) predicted in Theorem 3.2. Consequently, this leads to a "class assignment" problem. To illustrate this, we train a ResNet18 network with $d = 2$ on four classes {Automobile, Cat, Dog, Truck} from CIFAR10 dataset that are selected due to their clear semantic similarity and discrepancy. In this case, according to Theorem 3.3, the optimal classifiers are given by $[1, 0], [-1, 0], [0, 1], [0, -1]$, up to a rotation. Consequently, there are three distinct class assignments, as illustrated in Figures 4b to 4d.

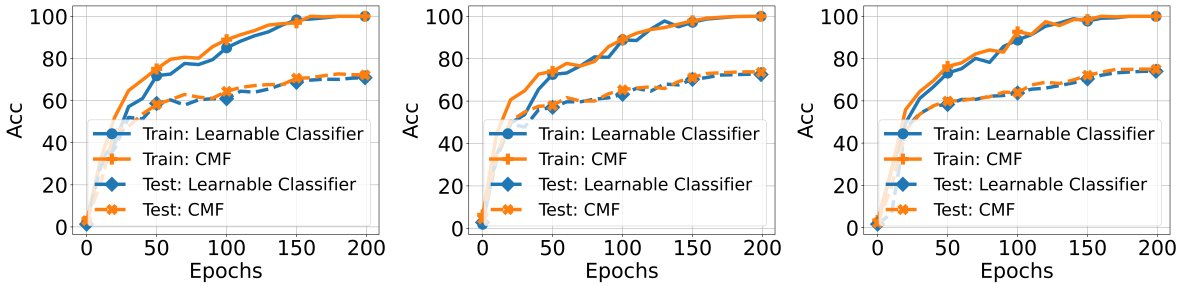
When doing standard training, the classifier consistently converges to the case where Cat and Dog are closer together across 5 different trials; Figure 4a shows the learned features (dots) and classifier weights (arrows) in one of such trials. This demonstrates the implicit algorithmic regularization in training DNNs, which naturally attracts (semantically) similar classes and separates dissimilar ones.

We also conduct experiments with the classifier fixed to be one of the three arrangements, and present the results in Figures 4b to 4d. Among them, we observe that the case



(a) Trainable (91.45%) (b) Fixed A1 (91.90%) (c) Fixed A2 (92.13%) (d) Fixed A3 (89.95%)

Figure 4. **Assignment of classes to classifier weights** for a ResNet18 with 2-dimensional feature space trained on the 4 classes {Automobile, Cat, Dog, Truck} from CIFAR10. *(a)* Learned classifier. *(b-d)* Classifiers fixed to be three different assignments. Test accuracy is reported in the bracket.



(a) ResNet (b) DenseNet (c) ResNeXt

Figure 5. **Comparison of the learning curves (training and testing accuracies) with learned classifiers vs. CMF classifiers** trained with various networks on CIFAR100 dataset and $d = 10$.

where Cat and Dog are far apart achieves a testing accuracy of 89.95%, lower than the other two cases with accuracies of 91.90% and 92.13%. This demonstrates the important role of class assignment to the generalization of DNNs, and that the implicit bias of the learned classifier is benign, i.e., leads to a more generalizable solutions. A comprehensive study of this phenomenon is deferred to future work.

5. Implications for Practical Network Training/Fine-tuning

Since the classifier always converges to a simplex ETF when $K \leq d+1$, prior work proposes to fix the classifier as a simplex ETF for reducing training cost (Zhu et al., 2021) and handling imbalance dataset (Yang et al., 2022). When $K > d+1$, the optimal classifier is also known to be a Softmax Code according to $\mathcal{GNC}_2$. However, the same method as in prior work may become sub-optimal due to the class assignment problem (see Section 4). To address this, we introduce the method of class-mean features (CMF) classifiers, where the classifier weights are set to be the exponential moving average of the mini-batch class-mean features during the training process. This approach is motivated from $\mathcal{GNC}_3$ where the optimal classifier converges to the class-mean features. We explain the details in Appendix B. As in prior work, CMF can reduce trainable parameters as well. For instance, it can reduce 30.91% of total parameters in a ResNet18 for BUPT-CBFace-50 dataset (Zhang & Deng, 2020). Here, we compare CMF with the standard training where the classifier is learned together with the feature mapping, in both training from scratch and fine-tuning.

Training from Scratch. We train a ResNet18 on CIFAR100 by using a learnable classifier or the CMF classifier. The learning curves in Figure 5 indicate that the approach with CMF classifier achieves comparable performance to the classical training protocols.

Fine-tuning. To verify the effectiveness of the CMF classifiers on fine-tuning, we follow the setting in (Kumar et al., 2022) to measure the performance of the fine-tuned model on both in-distribution (ID) task (i.e., CIFAR10 (Krizhevsky, 2009)) and OOD task (STL10 (Coates et al., 2011)). We compare the standard approach that fine-tunes both the classifier (randomly initialized) and the pre-trained feature mapping with our approach (using the CMF classifier). Our experiments show that the approach with CMF classifier achieves slightly better ID accuracy (98.00% VS 97.00%) and a better OOD performance (90.67% VS 87.42%). The improvement of OOD performance stems from the ability to align the classifier with the class-means through the entire

process, which better preserves the OOD property of the pre-trained model. Our approach also simplifies the two-stage approach of linearly probing and subsequent full fine-tuning in (Kumar et al., 2022).

6. Conclusion

In this work, we have introduced generalized neural collapse ($\mathcal{GNC}$) for characterizing learned last-layer features and classifiers in DNNs under an arbitrary number of classes and feature dimensions. We empirically validate the $\mathcal{GNC}$ phenomenon on practical DNNs that are trained with a small temperature in the CE loss and subject to spherical constraints on the features and classifiers. Building upon the unconstrained features model we have proven that $\mathcal{GNC}$ holds under certain technical conditions. $\mathcal{GNC}$ could offer valuable insights for the design, training, and generalization of DNNs. For example, the minimal one-vs-rest distance provides implications for designing feature dimensions when dealing with a large number of classes. Additionally, we have leveraged $\mathcal{GNC}$ to enhance training efficiency and fine-tuning performance by fixing the classifier as class-mean features. Further exploration of $\mathcal{GNC}$ in other scenarios, such as imbalanced learning, is left for future work. It is also of interest to further study the problem of optimally assigning classifiers from Softmax Code for each class, which could shed light on developing techniques for better classification performance.

Acknowledgment

JJ, JZ, and ZZ acknowledge support from NSF grants CCF-2240708 and IIS-2312840. Q.Q. and P.W.. acknowledge support from ONR N00014-22-1-2529 and NSF CAREER CCF-214390. Q.Q. also acknowledge supports from NSF CCF-2212066, NSF CCF-2212326, and NSF IIS 2312842, an AWS AI Award, a gift grant from KLA, and MICDE Catalyst Grant. We thank CloudBank (supported by NSF under Award #1925001) for providing the computational resources.

Impact Statement

This paper advances the understanding of deep representation learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Boyd, S. P. and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Braides, A. A handbook of γ-convergence. In *Handbook of Differential Equations: stationary partial differential equations*, volume 3, pp. 101–213. Elsevier, 2006.

Cao, K., Wei, C., Gaidon, A., Arechiga, N., and Ma, T. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019.

Carathéodory, C. Über den variabilitätsbereich der fourier'schen konstanten von positiven harmonischen funktionen. *Rendiconti Del Circolo Matematico di Palermo (1884-1940)*, 32(1):193–217, 1911.

Chan, K. H. R., Yu, Y., You, C., Qi, H., Wright, J., and Ma, Y. Redunet: A white-box deep network from the principle of maximizing rate reduction. *The Journal of Machine Learning Research*, 23(1):4907–5009, 2022.

Chang, W.-C., Felix, X. Y., Chang, Y.-W., Yang, Y., and Kumar, S. Pre-training tasks for embedding-based large-scale retrieval. In *International Conference on Learning Representations*, 2019.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Chen, X. and He, K. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15750–15758, 2021.

Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning, 2020b.

Coates, A., Ng, A., and Lee, H. An analysis of single-layer networks in unsupervised feature learning. In Gordon, G., Dunson, D., and Dudík, M. (eds.), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 215–223, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR. URL https://proceedings.mlr.press/v15/coates11a.html.

Cohn, H. Small spherical and projective codes. 2022.

Cohn, H. and Kumar, A. Universally optimal distribution of points on spheres. *Journal of the American Mathematical Society*, 20(1):99–148, 2007.

Deng, J., Guo, J., Xue, N., and Zafeiriou, S. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699, 2019.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

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

Fickus, M., Jasper, J., Mixon, D. G., and Watson, C. E. A brief introduction to equi-chordal and equi-isoclinic tight fusion frames. In *Wavelets and Sparsity XVII*, volume 10394, pp. 186–194. SPIE, 2017.

Galanti, T., György, A., and Hutter, M. Generalization bounds for transfer learning with pretrained classifiers. *arXiv preprint arXiv:2212.12532*, 2022a.

Galanti, T., György, A., and Hutter, M. On the role of neural collapse in transfer learning. In *International Conference on Learning Representations*, 2022b.

Gao, P., Xu, Q., Wen, P., Shao, H., Yang, Z., and Huang, Q. A study of neural collapse phenomenon: Grassmannian frame, symmetry, generalization, 2023.

Han, X., Papyan, V., and Donoho, D. L. Neural collapse under mse loss: Proximity to and dynamics on the central path. In *International Conference on Learning Representations*.

Hars, L. Numerical solutions of the thomson-p problems.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition, 2015.

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

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Ji, W., Lu, Y., Zhang, Y., Deng, Z., and Su, W. J. An unconstrained layer-peeled perspective on neural collapse. In *International Conference on Learning Representations*.

Ji, W., Lu, Y., Zhang, Y., Deng, Z., and Su, W. J. An unconstrained layer-peeled perspective on neural collapse. *arXiv preprint arXiv:2110.02796*, 2021.

Krizhevsky, A. Learning multiple layers of features from tiny images. pp. 32–33, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Kumar, A., Raghunathan, A., Jones, R., Ma, T., and Liang, P. Fine-tuning can distort pretrained features and underperform out-of-distribution, 2022.

Li, P., Li, X., Wang, Y., and Qu, Q. Neural collapse in multi-label learning with pick-all-label loss. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=y8NevOhrnW.

Li, X., Liu, S., Zhou, J., Lu, X., Fernandez-Granda, C., Zhu, Z., and Qu, Q. Principled and efficient transfer learning of deep models via neural collapse. *arXiv preprint arXiv:2212.12206*, 2022.

Lindgren, E., Reddi, S., Guo, R., and Kumar, S. Efficient training of retrieval models using negative cache. *Advances in Neural Information Processing Systems*, 34: 4134–4146, 2021.

Liu, W., Yu, L., Weller, A., and Schölkopf, B. Generalizing and decoupling neural collapse via hyperspherical uniformity gap. *arXiv preprint arXiv:2303.06484*, 2023a.

Liu, X., Zhang, J., Hu, T., Cao, H., Yao, Y., and Pan, L. Inducing neural collapse in deep long-tailed learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 11534–11544. PMLR, 2023b.

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

Lu, J. and Steinerberger, S. Neural collapse under cross-entropy loss. *Applied and Computational Harmonic Analysis*, 59:224–241, 2022.

Mitra, B., Craswell, N., et al. An introduction to neural information retrieval. *Foundations and Trends® in Information Retrieval*, 13(1):1–126, 2018.

Mixon, D. G., Parshall, H., and Pi, J. Neural collapse with unconstrained features, 2020.

Moore, M. H. Vector packing in finite dimensional vector spaces. *Linear Algebra and its Applications*, 8(3):213–224, 1974. ISSN 0024-3795. doi: https://doi.org/10.1016/0024-3795(74)90067-6. URL https://www.sciencedirect.com/science/article/pii/0024379574900676.

Murphy, K. P. *Probabilistic machine learning: an introduction*. MIT press, 2022.

Nguyen, D. A., Levie, R., Lienen, J., Kutyniok, G., and Hüllermeier, E. Memorization-dilation: Modeling neural collapse under noise. *arXiv preprint arXiv:2206.05530*, 2022.

Papyan, V. Traces of class/cross-class structure pervade deep learning spectra. *Journal of Machine Learning Research*, 21(252):1–64, 2020.

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

Poggio, T. and Liao, Q. Explicit regularization and implicit bias in deep network classifiers trained with the square loss. *arXiv preprint arXiv:2101.00072*, 2020.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Rangamani, A., Lindegaard, M., Galanti, T., and Poggio, T. A. Feature learning in deep classifiers through intermediate neural collapse. In *International Conference on Machine Learning*, pp. 28729–28745. PMLR, 2023.

Rockafellar, R. T. and Wets, R. J.-B. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

Tammes, P. M. L. On the origin of number and arrangement of the places of exit on the surface of pollen-grains. *Recueil des travaux botaniques néerlandais*, 27(1):1–84, 1930.

Thomson, J. J. Xxiv. on the structure of the atom: an investigation of the stability and periods of oscillation of a number of corpuscles arranged at equal intervals around the circumference of a circle; with application of the results to the theory of atomic structure. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 7(39):237–265, 1904.

Thrampoulidis, C., Kini, G. R., Vakilian, V., and Behnia, T. Imbalance trouble: Revisiting neural-collapse geometry. *Advances in Neural Information Processing Systems*, 35: 27225–27238, 2022.

Tirer, T. and Bruna, J. Extended unconstrained features model for exploring deep neural collapse. In *International Conference on Machine Learning*, pp. 21478–21505. PMLR, 2022.

Tirer, T., Huang, H., and Niles-Weed, J. Perturbation analysis of neural collapse. In *International Conference on Machine Learning*, pp. 34301–34329. PMLR, 2023.

Wang, F., Xiang, X., Cheng, J., and Yuille, A. L. Normface: L2 hypersphere embedding for face verification. In *Proceedings of the 25th ACM international conference on Multimedia*, pp. 1041–1049, 2017.

Wang, F., Cheng, J., Liu, W., and Liu, H. Additive margin softmax for face verification. *IEEE Signal Processing Letters*, 25(7):926–930, 2018a.

Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., and Liu, W. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5265–5274, 2018b.

Wang, J. Finding and investigating exact spherical codes. *Experimental Mathematics*, 18(2):249–256, 2009.

Wang, P., Liu, H., Pai, D., Yu, Y., Zhu, Z., Qu, Q., and Ma, Y. A global geometric analysis of maximal coding rate reduction. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=u9qmjV2khT.

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

Xie, L., Yang, Y., Cai, D., and He, X. Neural collapse inspired attraction-repulsion-balanced loss for imbalanced learning. *Neurocomputing*, 2023.

Xie, S., Qiu, J., Pasad, A., Du, L., Qu, Q., and Mei, H. Hidden state variability of pretrained language models can guide computation reduction for transfer learning. *arXiv preprint arXiv:2210.10041*, 2022.

Yang, Y., Xie, L., Chen, S., Li, X., Lin, Z., and Tao, D. Do we really need a learnable classifier at the end of deep neural network? *arXiv preprint arXiv:2203.09081*, 2022.

Yang, Y., Yuan, H., Li, X., Lin, Z., Torr, P., and Tao, D. Neural collapse inspired feature-classifier alignment for few-shot class incremental learning. *arXiv preprint arXiv:2302.03004*, 2023.

Yaras, C., Wang, P., Zhu, Z., Balzano, L., and Qu, Q. Neural collapse with normalized features: A geometric analysis over the riemannian manifold. In *Advances in Neural Information Processing Systems*.

Yaras, C., Wang, P., Zhu, Z., Balzano, L., and Qu, Q. Neural collapse with normalized features: A geometric analysis over the riemannian manifold, 2023.

Yi, X., Yang, J., Hong, L., Cheng, D. Z., Heldt, L., Kumthekar, A., Zhao, Z., Wei, L., and Chi, E. Sampling-bias-corrected neural modeling for large corpus item recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 269–277, 2019.

You, C. *Sparse methods for learning multiple subspaces from large-scale, corrupted and imbalanced data*. PhD thesis, Johns Hopkins University, 2018.

Yu, L., Hu, T., Hong, L., Liu, Z., Weller, A., and Liu, W. Continual learning by modeling intra-class variation. *arXiv preprint arXiv:2210.05398*, 2022.

Yu, Y., Chan, K. H. R., You, C., Song, C., and Ma, Y. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. *Advances in Neural Information Processing Systems*, 33:9422–9434, 2020.

Zhang, Y. and Deng, W. Class-balanced training for deep face recognition. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition workshops*, pp. 824–825, 2020.

Zhang, Y., Jiang, H., Miura, Y., Manning, C. D., and Langlotz, C. P. Contrastive learning of medical visual representations from paired images and text. In *Machine Learning for Healthcare Conference*, pp. 2–25. PMLR, 2022.

Zhou, J., You, C., Li, X., Liu, K., Liu, S., Qu, Q., and Zhu, Z. Are all losses created equal: A neural collapse perspective. In *Advances in Neural Information Processing Systems*.

Zhou, J., Li, X., Ding, T., You, C., Qu, Q., and Zhu, Z. On the optimization landscape of neural collapse under mse loss: Global optimality with unconstrained features. In *International Conference on Machine Learning*, pp. 27179–27202. PMLR, 2022a.

Zhou, X., Liu, X., Zhai, D., Jiang, J., Gao, X., and Ji, X. Learning towards the largest margins. *arXiv preprint arXiv:2206.11589*, 2022b.

Zhu, Z., Ding, T., Zhou, J., Li, X., You, C., Sulam, J., and Qu, Q. A geometric analysis of neural collapse with unconstrained features. *Advances in Neural Information Processing Systems*, 34:29820–29834, 2021.

Appendix

The organization of the appendix is as follows. Firstly, we introduce some useful results that are utilized throughout the appendices, and discuss related works on using CE loss with spherical constraints. In Appendix B, we offer comprehensive information regarding the datasets and computational resources for each figure, along with presenting additional experimental results on practical datasets. Lastly, in Appendix C, we provide the theoretical proofs for all the theorems mentioned in Section 3.

A. Basic and Common Practice of Using Spherical Constraints.

A.1. Useful Results

Lemma A.1 (LogSumExp). *Let denote* $\boldsymbol{x} = \left[x_1, x_2, \cdots, x_n\right] \in \mathcal{R}^n$ *and the LogSumExp function* $LSE(\boldsymbol{x}, \tau) = \log\left(\sum_{i=1}^n \exp(x_i/\tau)\right)$. *With* $\tau \to 0$, *we have*

$$\tau LSE(\boldsymbol{x}, \tau) \to \max_i x_i$$

In other words, the LogSumExp function is a smooth approximation to the maximum function.

Proof of Lemma A.1. According to the definition of the LogSumExp function, we have

$$\max_i x_i \leq \tau \text{LSE}(\boldsymbol{x}, \tau) \leq \max_i x_i + \tau \log n$$

where the first inequality is strict unless $n = 1$ and the second inequality is strict unless all arguments are equal $x_1 = x_2 = \cdots = x_n$. Therefore, with $\tau \to 0$, $\tau \text{LSE}(\boldsymbol{x}, \tau) \to \max_i x_i$. □

Theorem A.2 (Carathéodory's theorem (Carathéodory, 1911)). *Given* $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K \in \mathbb{R}^d$, *if a point* $\boldsymbol{v}$ *lies in the convex hull* $\text{conv}(\{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K\})$ *of* $\{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K\}$, *then* $\boldsymbol{v}$ *resides in the convex hull of at most* $d+1$ *of the points in* $\{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K\}$.

A.2. Common Practice of Using Spherical Constraints

In cases where the number of classes is larger than the feature dimensions, it is common practice to use a modified version of CE loss (1) with normalization and a small temperature τ. This approach is prevalent in face recognition, information retrieval and self-supervised contrastive learning areas. Below we present some representative works.

- **Face recognition.** AdditiveFace (Wang et al., 2018a), ArcFace (Deng et al., 2019), NormFace (Wang et al., 2017) and CosFace (Wang et al., 2018b) use the cross entropy loss with the feature and weight normalization and small temperature parameter. They observe that the feature and weight normalization are necessary since this practice strengthens the cosine constraint and improves the model's ability to distinguish between different face classes. For the temperature values, AdditiveFace (Wang et al., 2018a) uses the temperature $\tau = 1/30$ (in the first sentence of page 3); ArcFace (Deng et al., 2019) sets $\tau = 1/64$ (in the second paragraph of experiment setting in section 4.2); In the NormFace (Wang et al., 2017), the authors conduct comprehensive experiments for the effect of temperature across different datasets. For example, the superior performance is chosen at temperature $\tau = 1/40$ on the LFW dataset in figure 8, which is also better than CE loss without any normalization. The CosFace (Wang et al., 2018b) sets the temperature parameter as $\tau = 1/64$ ("training part" of section 4.1) for CASIA-WebFace and $\tau = 1/30$ also performs well in the practical implementation [link].
- **Information retrieval.** (Yi et al., 2019) also uses the modified cross entropy loss when training dual encoders (the last paragraph in section 3) to learn the representations of query and candidates. They observe that setting a small temperature value from $\tau = 0.05$ to 0.07 results in the best performance in table 1. Similarly, (Lindgren et al., 2021) also uses the modified cross entropy loss when training dual encoders in Equation 1 & 2 of section 2.2. It uses a temperature value of $\tau = 0.05$ for document retrieval (see the "set up the cache loss" in [link]).
- **Contrastive learning.** SimCLR (Chen et al., 2020a) treats each data sample as a class and computes the cosine similarity between pairs of positive and negative examples using the normalized inner product of features. It chooses the inverse temperature between 10 and 20 (which is the inverse of our τ) in table 5 to obtain the best performance. Similarly, MoCo (He et al., 2020) sets temperature as $\tau = 0.07$ according to the "technical details" part of section 3.3. In multi-domains constrative learning, ConVIRT (Zhang et al., 2022) uses normalization and temperature for the CE

loss to learn contrastive representations between medical image and text (see equation 2 & 3) and chooses $\tau = 0.01$ (in table 4). CLIP (Radford et al., 2021) model applies normalization to both image and text embeddings to ensure consistent scales(see figure 3), and it initializes the temperature $\tau = 0.07$ (in section 2.5 of page 5).

B. Experiments

In this section, we begin by providing additional details regarding the datasets and the computational resources utilized in the paper. Specifically, CIFAR10, CIFAR100, and BUPT-CBFace datasets are publicly available for academic purposes under the MIT license. Additionally, all experiments were conducted on 4xV100 GPU with 32G memory. Furthermore, we present supplementary experiments and implementation specifics for each figure.

B.1. Implementation details

We present implementation details for results in the paper.

Results in Figure 2. To illustrate the occurrence of the $\mathcal{GNC}$ phenomenon in practical multi-class classification problems, we trained a ResNet18 network (He et al., 2016) on the CIFAR100 dataset (Krizhevsky, 2009) using CE loss with varying temperature parameters. In this experiment, we set the dimensions of the last-layer features to 10. This is achieved by setting the number of channels in the second convolutional layer of the last residual block before the classifier layer to be 10. Prior to training, we applied standard preprocessing techniques, which involved normalizing the images (channel-wise) using their mean and standard deviation. We also employed standard data augmentation methods. For optimization, we utilized SGD with a momentum of 0.9 and an initial learning rate of 0.1, which decayed according to the CosineAnnealing over a span of 200 epochs.

The optimal margin (represented by the dotted line) in the second left subfigure is obtained from numerical optimization of the Softmax Code problem. According to Theorem 3.2, solving the Softmax Codes is equivalent to solving the "HardMax" problem in (5), which is an optimization over both $\boldsymbol{W}$ and $\boldsymbol{H}$. Thus, we optimize (5) numerically using projected gradient descent. Likewise, when given $\boldsymbol{W}$, we find the one-vs-rest distance by optimizing the "HardMax" problem over $\boldsymbol{H}$. The equivalence is proved in Lemma C.4 in Appendix C. Consequently, we can use the same projected gradient descent to find the one-vs-rest distance. In this optimization process, we used an initial learning rate of 0.1 and decreased it by a factor of 10 every 1000 iterations, for a total of 5000 iterations.

Results in Figure 4. To demonstrate the implicit algorithmic regularization in training DNNs, we train a ResNet18 network on four classes Automobile, Cat, Dog, Truck from CIFAR10 dataset. We set the dimensions of the last-layer features to 2 so that the learned features can be visualized, and we used a temperature parameter of 0.05. We optimized the networks for a total of 800 epochs using SGD with a momentum of 0.9 and an initial learning rate of 0.1, which is decreased by a factor of 10 every 200 epochs.

Results in Figure 5. To assess the effectiveness of the proposed CMF (Class Mean Feature) method, we utilized the ResNet18 architecture (He et al., 2015) as the feature encoder on the CIFAR100 dataset (Krizhevsky, 2009), where we set the feature dimension to $d = 20$ and the temperature parameter to $\tau = 0.1$. Since the model can only access a mini-batch of the dataset for each iteration, it becomes computationally prohibitive to calculate the class-mean features of the entire dataset. As a solution, we updated the classifier by employing the exponential moving average of the feature class mean, represented as $\boldsymbol{W}^{(t+1)} \leftarrow \beta \boldsymbol{W}^{(t)} + (1-\beta)\overline{\boldsymbol{H}}^{(t)}$. Here, $\overline{\boldsymbol{H}}^{(t)} \in \mathcal{R}^{K\times d}$ denotes the class mean feature of the mini-batch in iteration t, $\boldsymbol{W} \in \mathcal{R}^{K\times d}$ represents the classifier weights, and $\beta \in [0, 1)$ denotes the momentum coefficient (in our experiment, $\beta = 0.9$). We applied the same data preprocessing and augmentation techniques mentioned previously and employed an initial learning rate of 0.1 with the CosineAnnealing scheduler.

Results in the fine-tuning experiment of Section 5. We fine-tune a pretrained ResNet50 on MoCo v2 (Chen et al., 2020b) on CIFAR10 (Krizhevsky, 2009). The CIFAR10 test dataset was chosen as the in-distribution (ID) task, while the STL10 dataset (Coates et al., 2011) served as the out-of-distribution (OOD) task. To preprocess the dataset, we resized the images to 224×224 using BICUBIC interpolation and normalized them by their mean and standard deviation. Since there is no "monkey" class in CIFAR10 dataset, we remove the "monkey" class in the STL10 dataset. Additionally, we reassign the labels of CIFAR10 datasets to the STL10 dataset in order to match the classifier output. For optimization, we employed the Adam optimizer with a learning rate of $1e-5$ and utilized the CosineAnnealing scheduler. The models are fine-tuned for 5

epochs with the batch size of 100. We reported the best ID testing accuracy achieved during the fine-tuning process and also provided the corresponding OOD accuracy.

Results in Figure 6. To visualize the different structures learned under weight decay or spherical constraint, we conducted experiments in both practical and unconstrained feature model settings. In the practical setting, we trained a ResNet18 network (He et al., 2016) on the first 30 classes of the CIFAR100 dataset (Krizhevsky, 2009) using either weight decay or spherical constraint with the cross-entropy (CE) loss. For visualization purposes, we set the dimensions of the last-layer features to 2, and for the spherical constraint, we used a temperature parameter of 70. Prior to training, we applied standard preprocessing techniques, which included normalizing the images (channel-wise) using their mean and standard deviation. Additionally, we employed standard data augmentation methods. We optimized the networks for a total of 1600 epochs using SGD with a momentum of 0.9 and an initial learning rate of 0.1, which decreased by a factor of 10 every 700 epochs. For the unconstrained feature model setting, we considered only one sample per class and treated the feature (class-mean features) and weights as free optimization variables. In this case, we initialized the learning rate at 0.1 and decreased it by a factor of 10 every 1000 iterations, for totaling 5000 iterations. The temperature parameter was set to 0.02 for the spherical constraint.

Results in Figure 11. To demonstrate the implicit algorithmic regularization in training DNNs, we train a ResNet18 network on four classes Automobile, Cat, Deer, Dog, Horse, Truck from CIFAR10 dataset. We set the dimensions of the last-layer features to 2 so that the learned features can be visualized, and we used a temperature parameter of 0.2. We optimized the networks for a total of 600 epochs using SGD with a momentum of 0.9 and an initial learning rate of 0.1, which is decreased by a factor of 10 every 200 epochs.

B.2. Effect of regularization: Spherical constraint vs. weight decay

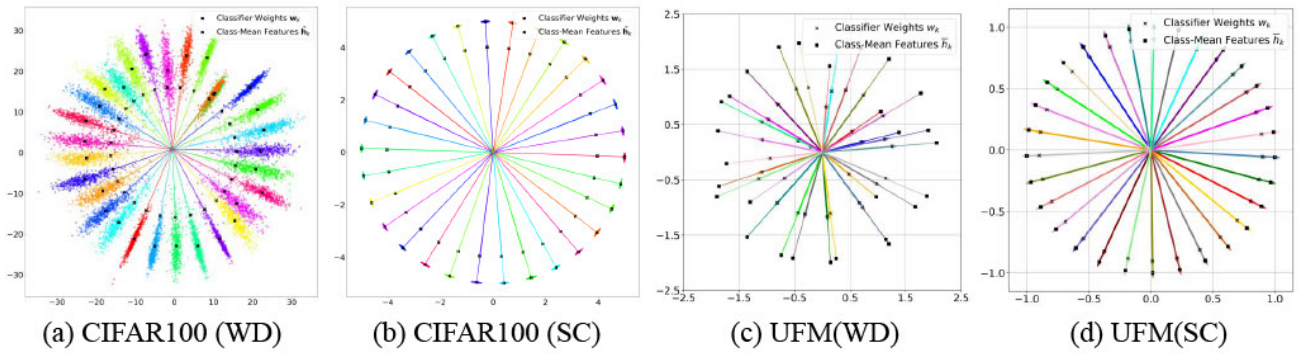


Figure 6. **Comparison of classifier weights and last-layer features with weight decay (WD) vs spherical constraint (SC) on features and classifiers.** (a) vs (b): Results with a ResNet18 trained on CFIAR100. (c) vs (d): Results with the unconstrained feature model in (10) and (2), respectively. In all settings, we set $K = 30$ and $d = 2$ for visualization; the first $K = 30$ classes from CIFAR100 dataset are used to train ResNet18.

We study the features and classifiers of networks trained with weight decay for the case $K > d + 1$, and compare with those obtained with spherical constraint. For the purpose of visualization, we train a ResNet18 with $d = 2$ on the first $K = 30$ classes of CIFAR100 and display the learned features and classifiers in Figure 6(a). Below we summarize observations from Figure 6(a).

- **Non-equal length and non-uniform distribution.** The vectors of class-mean features $\{\overline{\boldsymbol{h}}_k\}$ *do not* have equal length, and also appear to be *not* equally spaced when normalized to the unit sphere.
- **Self-duality only in direction.** The classifiers point towards the same direction as their corresponding class-mean features, but they have different lengths, and there exists no global scaling to exactly align them. As shown in Figure 7, the ratios between the lengths of classifier weights and class-mean features vary across different classes. Therefore, there exists no global scaling to align them exactly.

In contrast, using spherical constraint produces equal-length and uniformly distributed features, as well as aligned classifier and classifier weights not only in direction but also in length, see Figure 6 (b). Such a result is aligned with $\mathcal{GNC}$. This

observation may explain the common practice of using feature normalization in applications with an extremely large number of classes (Chen & He, 2021; Wang et al., 2018b), and justify our study of networks trained with sphere constraints in (2) rather than weight decay as in (Liu et al., 2023a).

We note that the discrepancy between the approaches using weight decay and spherical constraints is not due to the insufficient expressiveness of the networks. In fact, we also observe different performances in the unconstrained feature model with spherical constraints as in (2) and with the following regularized form (Zhu et al., 2021; Mixon et al., 2020; Tirer & Bruna, 2022; Zhou et al., 2022a):

$$\min_{\boldsymbol{W},\boldsymbol{H}} \frac{1}{nK}\sum_{k=1}^{K}\sum_{i=1}^{n}\mathcal{L}_{\mathrm{CE}}\left(\boldsymbol{W}^{\top}\boldsymbol{h}_{k,i},\boldsymbol{y}_k,\tau\right)+\frac{\lambda}{2}(\|\boldsymbol{W}\|_F^2+\|\boldsymbol{H}\|_F^2), \tag{10}$$

where λ represents the weight decay parameters. We observe similar phenomena in Figure 6(c, d) as in Figure 6(a, b) that the weight decay formulation results in features with non-equal lengths, non-uniform distribution, and different lengths than the classifiers. In the next section, we provide a theoretical justification for $\mathcal{GNC}$ under the UFM for Problem (2).

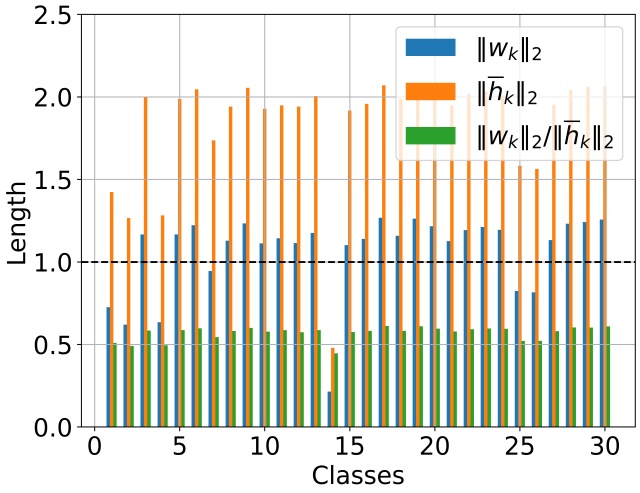


Figure 7. Illustration of the length of the classifier weights and class-mean features in unconstrained feature model (UFM) with weight decay (WD) form in Figure 6. The ratios between the lengths of classifier weights and class-mean features vary across different classes.

B.3. Additional results on prevalence of $\mathcal{GNC}$

We provide additional evidence on the occurrence of the $\mathcal{GNC}$ phenomenon in practical multi-class classification problems. Towards that, we train ResNet18, DenseNet121, and ResNeXt50 network on the CIFAR100, Tiny-ImageNet and BUPT-CBFace-50 datasets using CE loss. To illustrate the case where the feature dimension is smaller than the number of classes, we insert another linear layer before the last-layer classifier and set the dimensions of the features as $d = 10$ for CIFAR100 and Tiny-ImageNet, and $d = 512$ for BUPT-CBFace-50.

The results are reported in Figure 8. It can be seen that in all the cases the $\mathcal{GNC}_1$, $\mathcal{GNC}_2$ and $\mathcal{GNC}_3$ measures converge mostly monotonically as a function of the training epochs towards the values predicted by $\mathcal{GNC}$(i.e., 0 for $\mathcal{GNC}_1$ and $\mathcal{GNC}_3$ and the objective of Softmax Code for $\mathcal{GNC}_2$, see Section 2.2).

Effect of temperature τ. The results on BUPT-CBFace-50 reported in Figure 8 uses a temperature $\tau = 0.02$. To examine the effect of τ, we conduct experiments with varying τ and report the $\mathcal{GNC}_2$ in Figure 9. It can be seen that the $\mathcal{GNC}_2$ measure at convergence monotonically increases as τ decreases.

Implementation detail. We choose the temperature $\tau = 0.1$ for CIFAR100 and Tiny-ImageNet and $\tau = 0.02$ for the BUPT-CBFace-50 dataset. The CIFAR100 dataset consists of $60,000$ 32×32 color images in 100 classes and Tiny-ImageNet contains $100,000$ 64×64 color images in 200 classes. The BUPT-CBFace-50 dataset consists of $500,000$ images in $10,000$

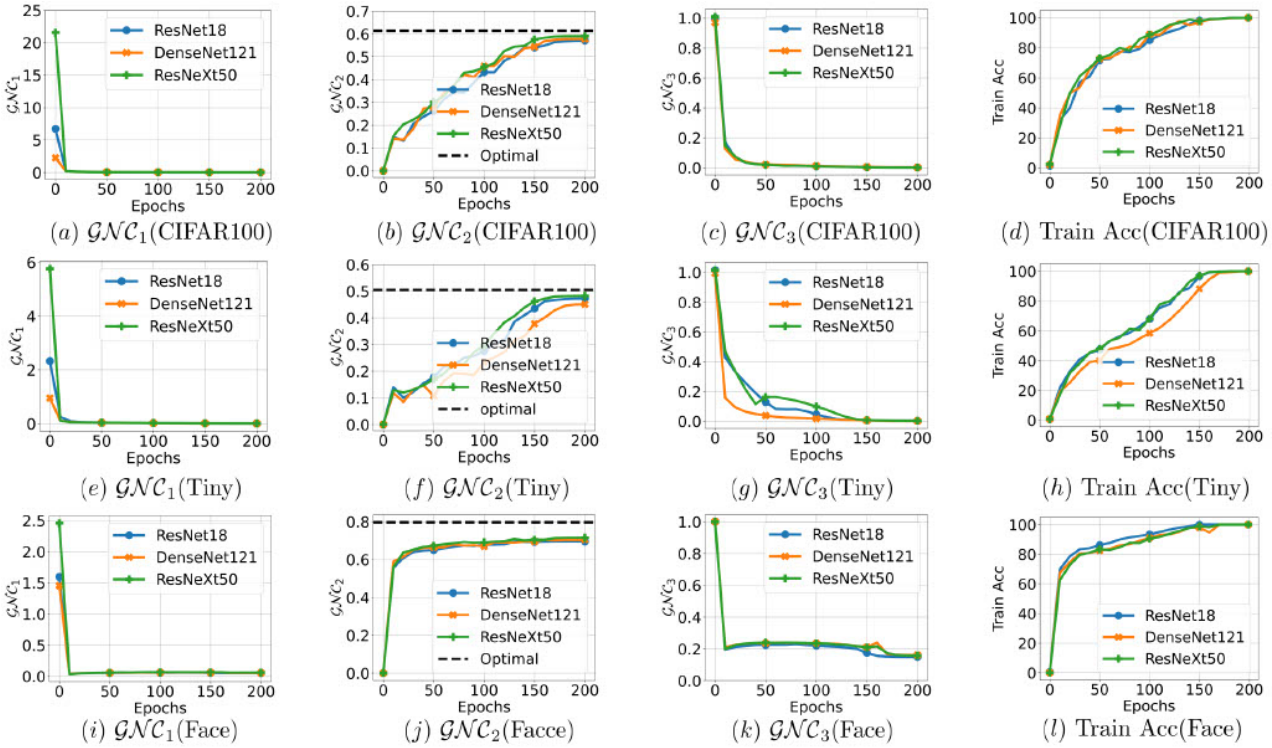


Figure 8. **Illustration of $\mathcal{GNC}$ and train accuracy across different network architectures on CIFAR100(top), Tiny-ImageNet(middle) and BUPT-CBFace-50(bottom) datasets.** We train the networks on CIFAR100 with $d = 10,\ K = 100$, Tiny-ImageNet with $d = 10,\ K = 200$ and BUPT-CBFace-50 with $d = 512,\ K = 10000$.

classes and all images are resized to the size of 50×50. To adapt to the smaller size images, we modify these architectures by changing the first convolutional layer to have a kernel size of 3, a stride of 1, and a padding size of 1. For our data augmentation strategy, we employ the random crop with a size of 32 and padding of 4, random horizontal flip with a probability of 0.5, and random rotation with degrees of 15 to increase the diversity of our training data. Then we normalize the images (channel-wise) using their mean and standard deviation. For optimization, we utilized SGD with a momentum of 0.9 and an initial learning rate of 0.1, which decayed according to the CosineAnnealing over a span of 200 epochs. The optimal margin (represented by the dash line) in the second from left column is obtained through numerical optimization of the Softmax Codes problem.

B.4. The Nearest Centroid Classifier

As the learned features exhibit within-class variability collapse and are maximally distant between classes, the classifier also converges to the nearest centroid classifier (a.k.a nearest class-center classifier (NCC), where each sample is classified with the nearest class-mean features), which is termed as $\mathcal{NC}_4$ in (Papyan et al., 2020) and exploited in (Galanti et al., 2022b;a; Rangamani et al., 2023) for studying $\mathcal{NC}$. To evaluate the convergence in terms of NCC accuracy, we use the same setup as in Figure 2, i.e., train a ResNet18 network on the CIFAR100 dataset with CE loss using different temperatures. We then classify the features by the NCC. The result is presented in Figure 10. We can observe that with a relatively small temperature τ, the NCC accuracy converges to 100% and hence the classifier also converges to a NCC.

B.5. Additional results on the effect of assignment problem

We provide additional evidence on the effect of "class assignment" problem in practical multi-class classification problems. To illustrate this, we train a ResNet18 network with $d = 2$ on six classes {Automobile, Cat, Deer, Dog, Horse, Truck} from CIFAR10 dataset that are selected due to their clear semantic similarity and discrepancy. Consequently, according to Theorem 3.3, there are fifteen distinct class assignments up to permutation.

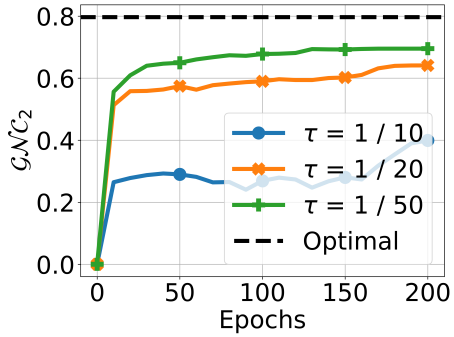


Figure 9. **Illustration of $\mathcal{GNC}_2$ on BUPT-CBFace-50 dataset across varying and temperatures.** We train the networks on BUPT-CBFace-50 with $d = 512,\ K = 10,000$.

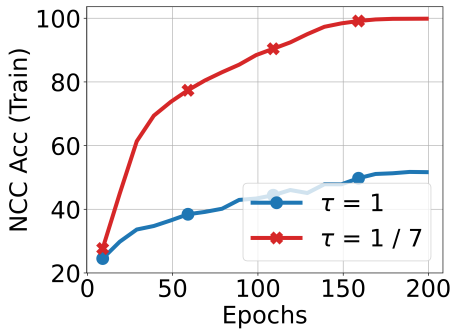


Figure 10. **Illustration of NCC training accuracy with different temperatures.**

When doing standard training, the classifier consistently converges to the case where Cat-Dog, Automobile-Truck and Deer-Horse pairs are closer together across 5 different trials; Figure 11(a) shows the learned features (dots) and classifier weights (arrows) in one of such trials. This demonstrates the implicit algorithmic regularization in training DNNs, which naturally attracts (semantically) similar classes and separates dissimilar ones.

We also conduct experiments with the classifier fixed to be three of the fifteen arrangements, and present the results in Figure 11 (b)-(d). Among them, we observe that the case where Cat-Dog, Automobile-Truck and Deer-Horse pairs are far apart achieves a testing accuracy of 86.37%, which is lower than the other two cases with testing accuracies of 91.60% and 91.95%. This demonstrates the important role of class assignment to the generalization of DNNs, and that the implicit bias of the learned classifier is benign, i.e., leads to a more generalizable solutions. Moreover, compared with the case of four classes in Figure 4, the discrepancy of test accuracy between different assignments increases from 2.18% to 5.58%. This suggests that as the number of classes grows, the importance of appropriate class assignments becomes increasingly paramount.

C. Theoretical Proofs

C.1. Proof of Lemma 3.1

We rewrite Lemma 3.1 below for convenience.

Lemma C.1 (Convergence to the "HardMax" problem)**.** *For any positive integers K and n, we have*

$$\limsup_{\tau\to 0}\left(\arg\min\frac{1}{nK}\sum_{k=1}^{K}\sum_{i=1}^{n}\mathcal{L}_{CE}\left(\boldsymbol{W}^\top\boldsymbol{h}_{k,i},\boldsymbol{y}_k,\tau\right)\right)\subseteq\arg\min\mathcal{L}_{HardMax}(\boldsymbol{W},\boldsymbol{H}). \tag{11}$$

In above, all $\arg\min$ *are taken over* $\boldsymbol{W}\in\mathcal{OB}(d,K),\boldsymbol{H}\in\mathcal{OB}(d,nK)$.

Proof. Our proof uses the fundamental theorem of Γ-convergence (Braides, 2006) (a.k.a. epi-convergence (Rockafellar &

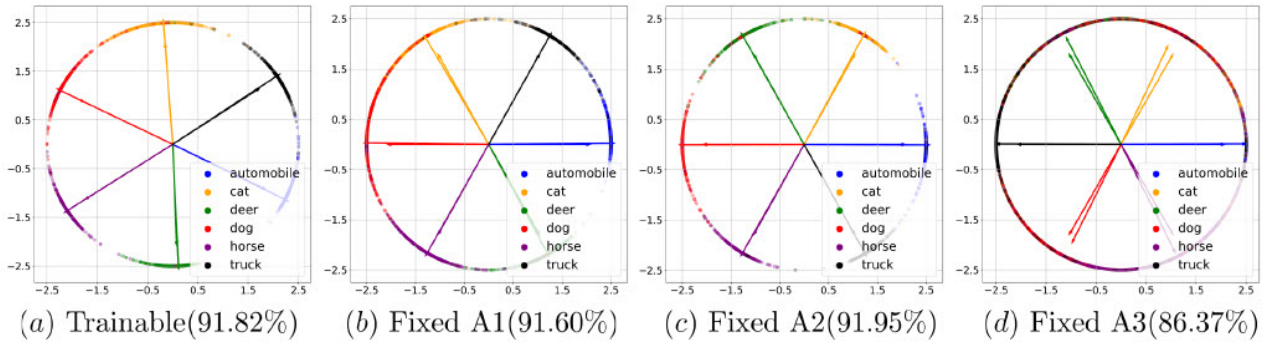


Figure 11. **Assignment of classes to classifier weights** for a ResNet18 with 2-dimensional feature space trained on the 6 classes {Automobile, Cat, Deer, Dog, Horse, Truck} from CIFAR10. *(a)* Learned classifier. *(b-d)* Classifiers fixed to be three different assignments. Test accuracy is reported in the bracket.

Wets, 2009)). Denote

$$\mathcal{L}_0(\boldsymbol{W},\boldsymbol{H},\tau) \doteq \tau \cdot \log \sum_{i=1}^{n}\sum_{k=1}^{K} \mathcal{L}_{\mathrm{CE}}(\boldsymbol{W}^\top \boldsymbol{h}_{k,i}, \boldsymbol{y}_k, \tau), \tag{12}$$

and let

$$\mathcal{WH}^- \doteq \{(\boldsymbol{W},\boldsymbol{H}) \in \mathcal{OB}(d,K)\times\mathcal{OB}(d,K) : (\boldsymbol{w}_{k'}-\boldsymbol{w}_k)^\top \boldsymbol{h}_{k,i} \le 0, \forall i\in[n], k\in[K], k'\in[K]\setminus k\}. \tag{13}$$

By Lemma C.2, the function $\mathcal{L}_0(\boldsymbol{W},\boldsymbol{H},\tau)$ converges uniformly to $\mathcal{L}_{\text{HardMax}}(\boldsymbol{W},\boldsymbol{H})$ on $\mathcal{WH}^-$ as $\epsilon\to 0$. Combining it with [Proposition 7.15](Rockafellar & Wets, 2009), we have $\mathcal{L}_0(\boldsymbol{W},\boldsymbol{H},\tau)$ Γ-converges to $\mathcal{L}_{\text{HardMax}}(\boldsymbol{W},\boldsymbol{H})$ on $\mathcal{WH}^-$ as well. By applying [Theorem 2.10](Braides, 2006), we have

$$\limsup_{\tau\to 0} \operatorname*{arg\,min}_{(\boldsymbol{W},\boldsymbol{H})\in\mathcal{WH}^-} \mathcal{L}_0(\boldsymbol{W},\boldsymbol{H},\tau) \subseteq \operatorname*{arg\,min}_{(\boldsymbol{W},\boldsymbol{H})\in\mathcal{WH}^-} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W},\boldsymbol{H}). \tag{14}$$

Note that in above, $\arg\min$ are taken over $\mathcal{WH}^-$ which is a strict subset of $\boldsymbol{W}\in\mathcal{OB}(d,K), \boldsymbol{H}\in\mathcal{OB}(d,nK)$. However, by Lemma C.3, we know that

$$\operatorname*{arg\,min}_{(\boldsymbol{W},\boldsymbol{H})\in\mathcal{WH}^-} \mathcal{L}_0(\boldsymbol{W},\boldsymbol{H},\tau) = \operatorname*{arg\,min}_{(\boldsymbol{W},\boldsymbol{H})\in\mathcal{OB}(d,K)\times\mathcal{OB}(d,K)} \mathcal{L}_0(\boldsymbol{W},\boldsymbol{H},\tau), \tag{15}$$

which holds for all τ sufficiently small. Hence, we have

$$\limsup_{\tau\to 0} \operatorname*{arg\,min}_{(\boldsymbol{W},\boldsymbol{H})\in\mathcal{OB}(d,K)\times\mathcal{OB}(d,K)} \mathcal{L}_0(\boldsymbol{W},\boldsymbol{H},\tau) \subseteq \operatorname*{arg\,min}_{(\boldsymbol{W},\boldsymbol{H})\in\mathcal{OB}(d,K)\times\mathcal{OB}(d,K)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W},\boldsymbol{H}), \tag{16}$$

which concludes the proof. □

Lemma C.2. *$\mathcal{L}_0(\boldsymbol{W},\boldsymbol{H},\tau)$ converges uniformly to $\mathcal{L}_{\text{HardMax}}(\boldsymbol{W},\boldsymbol{H})$ in the domain $(\boldsymbol{W},\boldsymbol{H})\in\mathcal{WH}^-$ as $\tau\to 0$.*

Proof. Recall from the definition of the CE loss in (1) that

$$\begin{aligned}\mathcal{L}_0(\boldsymbol{W},\boldsymbol{H},\tau) &\doteq \tau\cdot\log\sum_{i=1}^{n}\sum_{k=1}^{K}\mathcal{L}_{\mathrm{CE}}(\boldsymbol{W}^\top\boldsymbol{h}_{k,i},\boldsymbol{y}_k,\tau)\\ &= \tau\log\sum_{i=1}^{n}\sum_{k=1}^{K}\log\left(1+\sum_{k'\in[K]\setminus k}\exp\left((\boldsymbol{w}_{k'}-\boldsymbol{w}_k)^\top\boldsymbol{h}_{k,i}/\tau\right)\right)\end{aligned} \tag{17}$$

Denote $\alpha_{i,k,k'} = (\boldsymbol{w}_{k'}-\boldsymbol{w}_k)^\top\boldsymbol{h}_{k,i}, \forall i\in[n], k\in[K], k'\in[K]\setminus k$ for convenience. Fix any $i\in[n]$ and $k\in[K]$, by the property that $\frac{x}{1+x}\le\log(1+x)\le x$ for all $x>-1$, we have

$$\frac{\sum_{k'\in[K]\setminus k}\exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right)}{1+\sum_{k'\in[K]\setminus k}\exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right)} \le \log\left(1+\sum_{k'\in[K]\setminus k}\exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right)\right) \le \sum_{k'\in[K]\setminus k}\exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right). \tag{18}$$

Note that in the domain $(\boldsymbol{W}, \boldsymbol{H}) \in \mathcal{WH}^-$ we have $\alpha_{i,k,k'} \leq 0$ for all $i, k, k' \neq k$. It follows trivially from monotonicity of exponential function that $\sum_{k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right) < K - 1$ holds for all i, k. Hence, continuing from the inequality above we have

$$\frac{\sum_{k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right)}{K} \leq \log\left(1 + \sum_{k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right)\right) \leq \sum_{k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right). \tag{19}$$

Summing over i, k we obtain

$$\sum_{i,k} \frac{\sum_{k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right)}{K} \leq \sum_{i,k} \log\left(1 + \sum_{k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right)\right) \leq \sum_{i,k} \sum_{k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right), \tag{20}$$

Using the property of max function we further obtain

$$\begin{aligned}\frac{\max_{i,k,k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right)}{K} &\leq \sum_{i,k} \log\left(1 + \sum_{k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right)\right) \\ &\leq n \cdot K \cdot (K-1) \cdot \max_{i,k,k' \in [K] \setminus k} \exp\left(\frac{\alpha_{i,k,k'}}{\tau}\right). \end{aligned} \tag{21}$$

Taking logarithmic on both sides and multiplying all terms by τ we get

$$\max_{i,k,k' \in [K] \setminus k} \alpha_{i,k,k'} - \tau \log K \leq \mathcal{L}_0(\boldsymbol{W}, \boldsymbol{H}, \tau) \leq \tau \log(n \cdot K \cdot (K-1)) + \max_{i,k,k' \in [K] \setminus k} \alpha_{i,k,k'}. \tag{22}$$

Noting that $\max_{i,k,k' \in [K] \setminus k} \alpha_{i,k,k'} = \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H})$, we have

$$\mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) - \tau \log K \leq \mathcal{L}_0(\boldsymbol{W}, \boldsymbol{H}, \tau) \leq \tau \log(n \cdot K \cdot (K-1)) + \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}). \tag{23}$$

Hence, for any $\epsilon > 0$, by taking $\tau_0 = \frac{\epsilon}{\max\{\log K, \log(n \cdot K \cdot (K-1))\}}$, we have that for any $(\boldsymbol{W}, \boldsymbol{H}) \in \mathcal{WH}^-$, we have

$$|\mathcal{L}_0(\boldsymbol{W}, \boldsymbol{H}, \tau) - \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H})| \leq \tau \max\{\log K, \log(n \cdot K \cdot (K-1))\} < \epsilon, \tag{24}$$

for any $\tau < \tau_0$. That is, $\mathcal{L}_0(\boldsymbol{W}, \boldsymbol{H}, \tau)$ converges uniformly to $\mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H})$.

□

Lemma C.3. *Given any n, K there exists a constant τ_0 such that for any $\tau < \tau_0$ we have*

$$\operatorname*{arg\,min}_{(\boldsymbol{W}, \boldsymbol{H}) \in \mathcal{OB}(d,K) \times \mathcal{OB}(d,K)} \mathcal{L}_0(\boldsymbol{W}, \boldsymbol{H}, \tau) \in \mathcal{WH}^-. \tag{25}$$

Proof. Note that for any $(\boldsymbol{W}, \boldsymbol{H}) \notin \mathcal{OB}(d, K) \times \mathcal{OB}(d, K)$, there exists $\bar{\imath} \in [n], \bar{k} \in [K]$, and $\bar{k}' \in [K] \setminus \bar{k}$ such that $(\boldsymbol{w}_{\bar{k}'} - \boldsymbol{w}_{\bar{k}})^\top \boldsymbol{h}_{\bar{k},\bar{\imath}} \geq 0$. Hence, we have

$$\begin{aligned}\mathcal{L}_0(\boldsymbol{W}, \boldsymbol{H}, \tau) = \tau \log \sum_{i=1}^{n} \sum_{k=1}^{K} \log\left(1 + \sum_{k' \in [K] \setminus k} \exp\left((\boldsymbol{w}_{k'} - \boldsymbol{w}_k)^\top \boldsymbol{h}_{k,i}/\tau\right)\right) \\ \geq \tau \log\log(1 + \exp(\boldsymbol{w}_{\bar{k}'} - \boldsymbol{w}_{\bar{k}})^\top \boldsymbol{h}_{\bar{k},\bar{\imath}}/\tau) \geq \tau \log\log(2), \quad \forall \tau > 0. \end{aligned} \tag{26}$$

On the other hand, we show that one may construct a $(\boldsymbol{W}^*, \boldsymbol{H}^*) \in \mathcal{WH}^-$ such that $\mathcal{L}_0(\boldsymbol{W}^*, \boldsymbol{H}^*, \tau) < \tau \log\log(2)$ for any small enough τ. Towards that, we take $\boldsymbol{W}^*$ to be any matrix in $\mathcal{OB}(d, K)$ with distinct columns. Denote $M = \max_{k \neq k'} \langle \boldsymbol{w}_k^*, \boldsymbol{w}_{k'}^* \rangle$ the inner product of the closest pair of columns from $\boldsymbol{W}^*$, which by construction has value

$M < 1$. Take $\boldsymbol{h}_{k,i} = \boldsymbol{w}_k$ for all $k \in [K], i \in [n]$. We have $(\boldsymbol{w}^*_{k'} - \boldsymbol{w}^*_k)^\top \boldsymbol{h}^*_{k,i} \leq -(1-M)$, for all $i \in [n], k \in [K]$, and $k' \in [K] \setminus k$. Plugging this into the definition of $\mathcal{L}_0(\boldsymbol{W}, \boldsymbol{H}, \tau)$ we have

$$\mathcal{L}_0(\boldsymbol{W}^*, \boldsymbol{H}^*, \tau) = \tau \log \sum_{i=1}^{n} \sum_{k=1}^{K} \log \left(1 + \sum_{k' \in [K]\setminus k} \exp\left((\boldsymbol{w}^*_{k'} - \boldsymbol{w}^*_k)^\top \boldsymbol{h}^*_{k,i} / \tau \right) \right)$$
$$\leq \tau \log \left(nK \log(1 + (K-1)\exp(-\frac{1-M}{\tau})) \right) < \tau \log\log 2, \quad \forall \tau < \tau_0 \quad (27)$$

for some $\tau_0 > 0$ that depends only on M. In above, the last inequality holds because one can always find a $\tau_0 > 0$ such that $nK \log(1 + (K-1)\exp(-\frac{1-M}{\tau})) < \log 2$ for $\tau < \tau_0$. This implies that any $(\boldsymbol{W}, \boldsymbol{H}) \notin \mathcal{OB}(d, K) \times \mathcal{OB}(d, K)$ is not a minimizer of $\mathcal{L}_0(\boldsymbol{W}, \boldsymbol{H}, \tau)$ for a sufficiently small τ, which finishes the proof. □

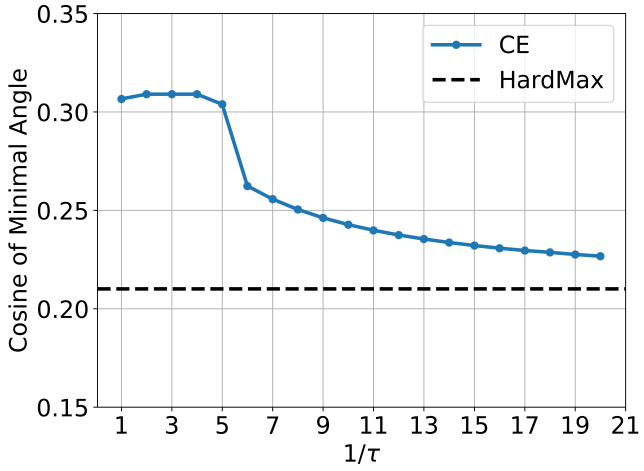


Figure 12. Verifying Lemma 3.1 under $d = 3$ and $K = 7$.

As depicted in Figure 12, we plot the cosine value of the minimal angle obtained from optimizing the CE loss (blue line) and the "HardMax" approach (black line) for different temperature parameters. The figure demonstrates that as the temperature parameter $\tau \to 0$, the blue line converges to the black line, thus validating our proof.

C.2. An important lemma

We present an important lemma which will be used to prove many of the subsequent results.

Lemma C.4 (Optimal Features for Fixed Classifier)**.** *For any $k \in [K]$, suppose $\boldsymbol{w}_k \notin \operatorname{conv}(\{\boldsymbol{w}_j\}_{j \in [K]\setminus k})$, then*

$$\min_{\boldsymbol{h} \in \mathbb{S}^{d-1}} \max_{k' \neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h} \rangle = -\operatorname{dist}(\boldsymbol{w}_k, \{\boldsymbol{w}_j\}_{j \in [K]\setminus k}). \quad (28)$$

In addition, the optimal $\boldsymbol{h}$ is given by

$$\boldsymbol{h} = \mathcal{P}_{\mathbb{S}^{d-1}} \left(\boldsymbol{w}_k - \mathcal{P}_{\{\boldsymbol{w}_j\}_{j \in [K]\setminus k}}(\boldsymbol{w}_k) \right) \quad (29)$$

where $\mathcal{P}_{\mathcal{W}}(\boldsymbol{v}) \doteq \arg\min_{\boldsymbol{w} \in \operatorname{conv}(\mathcal{W})} \{\|\boldsymbol{v} - \boldsymbol{w}\|_2\}$ denotes the projection of $\boldsymbol{v}$ on $\operatorname{conv}(\mathcal{W})$.

Proof. The proof follows from combining Lemma C.5 and Lemma C.6. □

Lemma C.5. *Suppose $\boldsymbol{w}_k \notin \operatorname{conv}(\{\boldsymbol{w}_j\}_{j \in [K]\setminus k})$. Then*

$$\min_{\boldsymbol{h} \in \mathbb{S}^{d-1}} \max_{k' \neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h} \rangle \equiv \min_{\|\boldsymbol{h}\|_2 \leq 1} \max_{k' \neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h} \rangle \quad (30)$$

where $\equiv$ means the two problems are equivalent, i.e., have the same optimal solutions.

Proof of Lemma C.5. By the separating hyperplane theorem (e.g. [Example 2.20](Boyd & Vandenberghe, 2004)), there exist a nonzero vector $\bar{\boldsymbol{h}}$ and $b \in \mathbb{R}$ such that $\max_{k' \neq k} \langle \boldsymbol{w}_{k'}, \bar{\boldsymbol{h}} \rangle < b$ and $\langle \boldsymbol{w}_k, \bar{\boldsymbol{h}} \rangle > b$, i.e., $\max_{k' \neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \bar{\boldsymbol{h}} \rangle < 0$. Let $\boldsymbol{h}^*$ be any optimal solution to the RHS of (30). Then, it holds that

$$\max_{k' \neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}^* \rangle \leq \max_{k' \neq k} \left\langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \frac{\bar{\boldsymbol{h}}}{\|\bar{\boldsymbol{h}}\|_2} \right\rangle < 0.$$

Hence, it must be the case that $\|\boldsymbol{h}^*\|_2 = 1$; otherwise, taking $\boldsymbol{h} = \boldsymbol{h}^*/\|\boldsymbol{h}^*\|_2$ gives lower objective for the RHS of (30), contradicting the optimality of $\boldsymbol{h}^*$. □

Lemma C.6. *Suppose $\boldsymbol{w}_k \notin \operatorname{conv}(\{\boldsymbol{w}_j\}_{j \in [K] \setminus k})$. Consider the (primal) problem*

$$\min_{\|\boldsymbol{h}\|_2 \leq 1} \max_{k' \neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h} \rangle . \tag{31}$$

Its dual problem is given by

$$\max_{\boldsymbol{v} \in \mathbb{R}^d} -\|\boldsymbol{w}_k - \sum_{k' \neq k} v_{k'} \boldsymbol{w}_{k'}\|_2 \;\; \textit{s.t.} \sum_{k' \neq k} v_{k'} = 1, \; \textit{and} \; v_{k'} \geq 0, \forall k' \neq k. \tag{32}$$

with zero duality gap. Moreover, for any primal optimal solution $\boldsymbol{h}^*$ there is a dual optimal solution $\boldsymbol{v}^*$ and they satisfy

$$\boldsymbol{h}^* = \frac{\boldsymbol{w}_k - \sum_{k'} v^*_{k'} \boldsymbol{w}_{k'}}{\|\boldsymbol{w}_k - \sum_{k'} v^*_{k'} \boldsymbol{w}_{k'}\|_2}. \tag{33}$$

Proof of Lemma C.6. We rewrite the primal problem as

$$\min_{\|\boldsymbol{h}\|_2 \leq 1, \boldsymbol{p} \in \mathbb{R}^d} \max_{k' \neq k} p_{k'} \;\; \text{s.t. } p_{k'} = \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h} \rangle . \tag{34}$$

Introducing the dual variable $\boldsymbol{v} \in \mathbb{R}^d$, the Lagragian function is

$$\mathcal{L}(\boldsymbol{h}, \boldsymbol{p}, \boldsymbol{v}) = \max_{k' \neq k} p_{k'} - \sum_{k' \neq k} v_{k'} (p_{k'} - \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h} \rangle), \;\; \|\boldsymbol{h}\|_2 \leq 1. \tag{35}$$

We now derive the dual problem, defined as

$$\begin{aligned} \max_{\boldsymbol{v}} \min_{\|\boldsymbol{h}\|_2 \leq 1, \boldsymbol{p} \in \mathbb{R}^d} \max_{k' \neq k} \mathcal{L}(\boldsymbol{h}, \boldsymbol{p}, \boldsymbol{v}) &= \max_{\boldsymbol{v}} \left(\min_{\boldsymbol{p}} \left(\max_{k' \neq k} p_{k'} - \sum_{k' \neq k} v_{k'} p_{k'} \right) + \min_{\|\boldsymbol{h}\|_2 \leq 1} \left\langle \sum_{k' \neq k} v_{k'} \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h} \right\rangle \right) \\ &= \max_{\boldsymbol{v}} \min_{\|\boldsymbol{h}\|_2 \leq 1} \left\langle \sum_{k' \neq k} v_{k'} \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h} \right\rangle \;\; \text{s.t.} \sum_{k' \neq k} v_{k'} = 1, \; v_{k'} \geq 0 \; \forall k' \neq k \\ &= \max_{\boldsymbol{v}} -\|\boldsymbol{w}_k - \sum_{k' \neq k} v_{k'} \boldsymbol{w}_{k'}\|_2 \;\; \text{s.t.} \sum_{k' \neq k} v_{k'} = 1, \; v_{k'} \geq 0 \; \forall k' \neq k. \end{aligned} \tag{36}$$

In above, the second equality follows from the fact that the conjugate (see e.g. (Boyd & Vandenberghe, 2004)) of the max function is the indicator function of the probability simplex. The third equality uses the assumption that $\boldsymbol{w}_k \notin \operatorname{conv}(\{\boldsymbol{w}_j\}_{j \in [K] \setminus k})$, which implies that $\sum_{k' \neq k} v_{k'} \boldsymbol{w}_{k'} - \boldsymbol{w}_k \neq 0$ under the simplex constraint of $\boldsymbol{v}$, hence the optimal $\boldsymbol{h}$ to the optimization in the second line can be easily obtained as $\boldsymbol{h} = \frac{\boldsymbol{w}_k - \sum_{k'} v_{k'} \boldsymbol{w}_{k'}}{\|\boldsymbol{w}_k - \sum_{k'} v_{k'} \boldsymbol{w}_{k'}\|_2}$.

Finally, the rest of the claims hold as the primal problem is convex with the Slater's condition satisfied. □

C.3. Proof of Theorem 3.2

Here we prove the following result which is a stronger version of Theorem 3.2.

Theorem C.7. *Let $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ be an optimal solution to (5). Then, it holds that $\boldsymbol{W}^\star$ is a Softmax Code, i.e.,*

$$\boldsymbol{W}^\star \in \operatorname*{arg\,max}_{\boldsymbol{W} \in \mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}). \tag{37}$$

Conversely, let $\boldsymbol{W}^{SC}$ be any Softmax Code. Then, there exists a $\boldsymbol{H}^{SC}$ such that $(\boldsymbol{W}^{SC}, \boldsymbol{H}^{SC})$ is an optimal solution to (5).

Proof. The proof is divided into two parts.

Any optimal solution to (5) is a Softmax Code. Our proof is based on providing a lower bound on the objective $\mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H})$. We distinguish two cases in deriving the lower bound.

- $\boldsymbol{W}$ has distinct columns. In this case we use the following bound:

$$\begin{aligned}\mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) &= \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_{k,i}\rangle \\ &\geq \max_{k\in[K]} \max_{i\in[n]} \min_{\bar{\boldsymbol{h}}_{k,i}\in\mathbb{S}^{d-1}} \max_{k'\neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \bar{\boldsymbol{h}}_{k,i}\rangle = -\min_{k\in[K]} \operatorname{dist}(\boldsymbol{w}_k, \{\boldsymbol{w}_j\}_{j\in[K]\setminus k}). \end{aligned} \tag{38}$$

  In above, the first equality follows directly from definition of the HardMax function. The inequality follows trivially from the property of the $\min$ operator. The last equality follows from Lemma C.4, which requires that $\boldsymbol{W}$ has distinct columns. Continuing on the rightmost term in (38), we have

$$-\min_{k\in[K]} \operatorname{dist}(\boldsymbol{w}_k, \{\boldsymbol{w}_j\}_{j\in[K]\setminus k}) = -\rho_{\text{one-vs-rest}}(\boldsymbol{W}) \geq -\rho_{\text{one-vs-rest}}(\boldsymbol{W}^{\text{SC}}), \tag{39}$$

  where $\boldsymbol{W}^{\text{SC}}$ is any Softmax Code. In above, the equality follows from the definition of the operator $\rho_{\text{one-vs-rest}}()$, and the inequality follows from the definition of the Softmax Code. In particular, by defining $\widehat{\boldsymbol{W}} = \boldsymbol{W}^{\text{SC}}$ and $\widehat{\boldsymbol{H}}$ as

$$\widehat{\boldsymbol{h}}_{k,i} = \frac{\widehat{\boldsymbol{w}}_k - \operatorname{proj}(\widehat{\boldsymbol{w}}_k, \{\widehat{\boldsymbol{w}}_j\}_{j\in[K]\setminus k})}{\|\widehat{\boldsymbol{w}}_k - \operatorname{proj}(\widehat{\boldsymbol{w}}_k, \{\widehat{\boldsymbol{w}}_j\}_{j\in[K]\setminus k})\|_2}, \tag{40}$$

  all inequalities in (38) and (39) holds with equality by taking $\boldsymbol{W} = \widehat{\boldsymbol{W}}$ and $\boldsymbol{H} = \widehat{\boldsymbol{H}}$, at which we have that $\mathcal{L}_{\text{HardMax}}(\widehat{\boldsymbol{W}}, \widehat{\boldsymbol{H}}) = -\rho_{\text{one-vs-rest}}(\boldsymbol{W}^{\text{SC}})$.

- $\boldsymbol{W}$ does not have distinct columns. Hence, there exists k_1, k_2 such that $k_1 \neq k_2$ but $\boldsymbol{w}_{k_1} = \boldsymbol{w}_{k_2}$. We have

$$\mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) = \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_{k,i}\rangle \geq \max_{i\in[n]} \langle \boldsymbol{w}_{k_2} - \boldsymbol{w}_{k_1}, \boldsymbol{h}_{k_1,i}\rangle = 0. \tag{41}$$

Combining the above two cases, and by noting that $-\rho_{\text{one-vs-rest}}(\boldsymbol{W}^{\text{SC}}) < 0$, we have that $(\widehat{\boldsymbol{W}}, \widehat{\boldsymbol{H}})$ is an optimal solution to the HardMax problem in (5). Moreover, since $(\boldsymbol{W}^*, \boldsymbol{H}^*)$ is an optimal solution to the HardMax problem, it must attain the lower bound, i.e.,

$$\mathcal{L}_{\text{HardMax}}(\boldsymbol{W}^*, \boldsymbol{H}^*) = -\rho_{\text{one-vs-rest}}(\boldsymbol{W}^{\text{SC}}). \tag{42}$$

Hence, $\boldsymbol{W}^*$ has to attain the equality in (39). By definition, this means that $\boldsymbol{W}^*$ is a Softmax Code, which concludes the proof of this part.

From any Softmax Code we can construct an optimal solution to (5). Let

$$\boldsymbol{W}^{\text{SC}} \in \operatorname*{arg\,max}_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}) \tag{43}$$

be any Softmax Code. Moreover, define $\boldsymbol{H}^{\text{SC}}$ to be such that

$$\boldsymbol{h}_{k,i}^{\text{SC}} = \operatorname*{arg\,min}_{\boldsymbol{h}_k\in\mathbb{S}^{d-1}} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^{\text{SC}} - \boldsymbol{w}_k^{\text{SC}}, \boldsymbol{h}_k\rangle, \forall k \in [K], \forall i \in [n]. \tag{44}$$

Note that the following result holds which will be used in the subsequent proof:

$$\begin{aligned}\boldsymbol{W}^{\text{SC}} &\in \operatorname*{arg\,max}_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \min_k \operatorname{dist}\Big(\boldsymbol{w}_k, \{\boldsymbol{w}_j\}_{j\in[K]\setminus k}\Big) && \text{(Definition of Softmax Code)} \\ &\in \operatorname*{arg\,min}_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \max_k \min_{\boldsymbol{h}_k\in\mathbb{S}^{d-1}} \max_{k'\neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_k\rangle. && \text{(Lemma C.4)}\end{aligned} \tag{45}$$

For any $(\widehat{\boldsymbol{W}}, \widehat{\boldsymbol{H}})$, we have

$$\begin{aligned}
\mathcal{L}_{\text{HardMax}}(\widehat{\boldsymbol{W}}, \widehat{\boldsymbol{H}}) &= \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \widehat{\boldsymbol{w}}_{k'} - \widehat{\boldsymbol{w}}_k, \widehat{\boldsymbol{h}}_{k,i} \rangle && \text{(Definition of } \mathcal{L}_{\text{HardMax}}) \\
&\geq \max_{k\in[K]} \min_{\boldsymbol{h}_k \in \mathbb{S}^{d-1}} \max_{k'\neq k} \langle \widehat{\boldsymbol{w}}_{k'} - \widehat{\boldsymbol{w}}_k, \boldsymbol{h}_k \rangle \\
&\geq \max_{k\in[K]} \min_{\boldsymbol{h}_k \in \mathbb{S}^{d-1}} \max_{k'\neq k} \langle \boldsymbol{w}^{\text{SC}}_{k'} - \boldsymbol{w}^{\text{SC}}_k, \boldsymbol{h}_k \rangle && \text{(Eq. (45))} \\
&= \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}^{\text{SC}}_{k'} - \boldsymbol{w}^{\text{SC}}_k, \boldsymbol{h}^{\text{SC}}_{k,i} \rangle && \text{(Eq. (44))} \\
&= \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}^{\text{SC}}, \boldsymbol{H}^{\text{SC}}). && \text{(Definition of } \mathcal{L}_{\text{HardMax}})
\end{aligned} \tag{46}$$

This implies that $(\boldsymbol{W}^{\text{SC}}, \boldsymbol{H}^{\text{SC}})$ is an optimal solution to the HardMax problem in (5), which concludes the proof of this part.

□

C.4. Proof of Theorem 3.3

Theorem C.8. *For any positive integers K and d, let $\boldsymbol{W}^\star \in \mathcal{OB}(d, K)$ be a Softmax Code. Then,*

- *$d = 2$: $\{\boldsymbol{w}^\star_k\}$ is uniformly distributed on the unit circle, i.e., $\{\boldsymbol{w}^\star_k\} = \{\big(\cos(\frac{2\pi k}{K} + \alpha), \sin(\frac{2\pi k}{K} + \alpha)\big)\}$ for some α;*
- *$K \leq d + 1$: $\{\boldsymbol{w}^\star_k\}$ forms a simplex ETF, i.e., $\boldsymbol{W}^\star = \sqrt{\frac{K}{K-1}}\boldsymbol{P}(\boldsymbol{I}_K - \frac{1}{K}\boldsymbol{1}_K\boldsymbol{1}_K^\top)$ for some orthonomal $\boldsymbol{P} \in \mathbb{R}^{d\times K}$;*
- *$d + 1 < K \leq 2d$: $\min_k \operatorname{dist}(\boldsymbol{w}^\star_k, \{\boldsymbol{w}^\star_j\}_{j\in[K]\setminus k}) = 1$ which can be achieved when $\{\boldsymbol{w}^\star_k\}$ are a subset of vertices of a cross-polytope;*
- *$K \to \infty$: $\{\boldsymbol{w}^\star_k\}$ are uniformly distributed on the unite sphere $\mathbb{S}^{d-1}$;*

Proof. We prove the results case by case as follows.

- **$d = 2$: $\{\boldsymbol{w}^\star_k\}$ are uniformly distributed on the unite sphere of $\mathbb{S}^1$.**

  Denote $\boldsymbol{w}^\star_k = [\cos\alpha^\star_k, \sin\alpha^\star_k], \forall k \in [K]$. Without loss of generality we may assume that $0 < \alpha^\star_1 < \alpha^\star_2 < \ldots < \alpha^\star_K \leq 2\pi$. Define

  $$\theta^\star_k = \begin{cases} \alpha^\star_{k+1} - \alpha^\star_k, & \forall k \in [K-1] \\ \alpha^\star_1 - \alpha^\star_K + 2\pi, & k = K. \end{cases} \tag{47}$$

  For convenience, we also define $\alpha^\star_{K+1} \doteq \alpha^\star_1$ and $\theta^\star_{K+1} \doteq \theta^\star_1$.

  Geometrically, $\theta^\star_k$ is the angular distance between $\alpha^\star_k$ and α^*_{k+1}. Moreover, by summing up all terms in (47) over $k \in [K]$ we have

  $$\sum_{k\in[K]} \theta^\star_k = 2\pi. \tag{48}$$

  To prove the theorem we only need to show that $\theta^\star_k = \frac{2\pi}{K}$ for all $k \in [K]$, which implies that $\{\boldsymbol{w}^\star_k\}$ are uniformly distributed on the unit circle.

  We start by noting that the following result holds:

  $$\theta^\star_k + \theta^\star_{k+1} \geq 2 \times \frac{2\pi}{K}, \ \ \forall k \in [K]. \tag{49}$$

  To see why, let $\{\widehat{\boldsymbol{w}}_k = [\cos\widehat{\alpha}_k, \sin\widehat{\alpha}_k]\}_{k\in[K]}$ with $\widehat{\alpha}_k = \frac{k\times 2\pi}{K}, \forall k \in [K]$ be a collection of points on the unit circle that is distributed uniformly. Moreover, define $\{\widehat{\theta}_k\}_{k\in[K]}$ as

  $$\widehat{\theta}_k = \begin{cases} \widehat{\alpha}_{k+1} - \widehat{\alpha}_k, & \forall k \in [K-1] \\ \widehat{\alpha}_1 - \widehat{\alpha}_K + 2\pi, & k = K. \end{cases} \tag{50}$$

  Since $\boldsymbol{W}^\star$ is a Softmax Code, we have

  $$\rho_{\text{one-vs-rest}}(\boldsymbol{W}^\star) \geq \rho_{\text{one-vs-rest}}(\widehat{\boldsymbol{W}}), \tag{51}$$

which implies, using the definition of $\rho_{\text{one-vs-rest}}()$,

$$\min_{k\in[K]} \operatorname{dist}(\boldsymbol{w}_k^\star, \{\boldsymbol{w}_j^\star\}_{j\in[K]\setminus k}) \geq \min_{k\in[K]} \operatorname{dist}(\widehat{\boldsymbol{w}}_k, \{\widehat{\boldsymbol{w}}_j\}_{j\in[K]\setminus k}). \tag{52}$$

If there exists a $\bar{k} \in [K]$ such that $\theta_{\bar{k}}^\star + \theta_{\bar{k}+1}^\star < 2\times\frac{2\pi}{K}$, by noting that $\widehat{\theta}_{\bar{k}} + \widehat{\theta}_{\bar{k}+1} = 2\times\frac{2\pi}{K}$, it is easy to see geometrically that

$$\operatorname{dist}(\boldsymbol{w}_{\bar{k}}^\star, \{\boldsymbol{w}_j^\star\}_{j\in[K]\setminus\bar{k}}) < \operatorname{dist}(\widehat{\boldsymbol{w}}_{\bar{k}}, \{\widehat{\boldsymbol{w}}_j\}_{j\in[K]\setminus\bar{k}}) = \min_{k\in[K]} \operatorname{dist}(\widehat{\boldsymbol{w}}_k, \{\widehat{\boldsymbol{w}}_j\}_{j\in[K]\setminus k}),$$

where the last equality follows from the fact that $\{\widehat{\boldsymbol{w}}_k\}_{k\in[K]}$ are uniformly distributed. This result contradicts (52), which implies that (49) holds.

Taking the summation on both sizes of (49) over all $k \in [K]$ and divide both sides by 2, we obtain

$$\sum_{k\in[K]} \theta_k^* \geq 2\pi. \tag{53}$$

Comparing this with (48), we obtain that the inequality in (49) holds with equality for all $k \in [K]$, that is,

$$\theta_k^\star + \theta_{k+1}^\star = 2\times\frac{2\pi}{K}, \ \ \forall k \in [K]. \tag{54}$$

If there exists a $\bar{k} \in [K]$ such that $\theta_{\bar{k}}^\star \neq \theta_{\bar{k}+1}^\star$, then it is easy to see geometrically that

$$\operatorname{dist}(\boldsymbol{w}_{\bar{k}}^\star, \{\boldsymbol{w}_j^\star\}_{j\in[K]\setminus\bar{k}}) < \operatorname{dist}(\widehat{\boldsymbol{w}}_{\bar{k}}, \{\widehat{\boldsymbol{w}}_j\}_{j\in[K]\setminus\bar{k}}) = \min_{k\in[K]} \operatorname{dist}(\widehat{\boldsymbol{w}}_k, \{\widehat{\boldsymbol{w}}_j\}_{j\in[K]\setminus k}),$$

which contradicts (52). Hence, it follows that $\theta_k^\star = \frac{2\pi}{K}$ for all $k \in [K]$.

- $\boldsymbol{K \leq d+1}$**: $\{\boldsymbol{w}_k^\star\}$ forms a simplex ETF.**

We first consider optimal configuration of K unit-length vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K$. Note that

$$0 \leq \left\| \sum_{k=1}^K \boldsymbol{u}_k \right\|_2^2 = \sum_k \sum_{k'} \langle \boldsymbol{u}_k, \boldsymbol{u}_{k'} \rangle \leq K + K(K-1) \max_{k\neq k'} \langle \boldsymbol{u}_k, \boldsymbol{u}_{k'} \rangle,$$

where the first inequality achieves equality only when $\sum_{k=1}^K \boldsymbol{u}_k = 0$ and the second inequality becomes equality only when $\langle \boldsymbol{u}_k, \boldsymbol{u}_{k'} \rangle = -\frac{1}{K-1}$ for any $k \neq k'$. These two conditions mean that $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K$ form a simplex ETF. The above equation further impleis that

$$\max_{k\neq k'} \langle \boldsymbol{u}_k, \boldsymbol{u}_{k'} \rangle \geq -\frac{1}{K-1}, \ \forall \boldsymbol{u}_1, \ldots, \boldsymbol{u}_K \in \mathbb{S}^{d-1}, \tag{55}$$

and the equality holds only when $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K$ form a simplex ETF.

We will also need the following result:

$$\frac{1}{2} \sum_{k\neq k'} \|\boldsymbol{w}_k - \boldsymbol{w}_{k'}\|^2 = K^2 - \Big\| \sum_k \boldsymbol{u}_k \Big\|^2 \leq K^2, \tag{56}$$

where the last inequality becomes equality when $\sum_k \boldsymbol{u}_k = 0$.

We now prove the form of the optimal Softmax Code for $K \leq d+1$. Noting the equivalence between Softmax Code and the HardMax problem as proved in Theorem 3.2, we will analyze the HardMax problem for this case. Specifically, note that

$$\begin{aligned} &K(K-1) \max_{k\neq k'} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_k \rangle \geq \sum_{k\neq k'} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_k \rangle = \frac{1}{2} \sum_{k\neq k'} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_k - \boldsymbol{h}_{k'} \rangle \\ &\geq -\frac{1}{4} \sum_{k\neq k'} \|\boldsymbol{w}_k - \boldsymbol{w}_{k'}\|^2 - \frac{1}{4} \sum_{k\neq k'} \|\boldsymbol{h}_k - \boldsymbol{h}_{k'}\|^2 \geq -K^2, \end{aligned}$$

where the first inequality achieves equality only when $\langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_k \rangle = \langle \boldsymbol{w}_{j'} - \boldsymbol{w}_j, \boldsymbol{h}_j \rangle$ for any $k' \neq k, j' \neq j$, the second inequality follows from the Cauchy–Schwarz inequality and acheives inequality only when $\boldsymbol{w}_{k'} - \boldsymbol{w}_k = \boldsymbol{h}_{k'} - \boldsymbol{h}_k$ for any $k' \neq k$, and the third inequality follows from (56) and achieves equality only when $\sum_k \boldsymbol{w}_k = \sum_k \boldsymbol{h}_k = 0$. Assuming all these conditions hold, then $\langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_k \rangle = -\frac{K}{K-1}$, which together with the requirement $\boldsymbol{w}_{k'} - \boldsymbol{w}_k = \boldsymbol{h}_{k'} - \boldsymbol{h}_k$ implies that

$$\langle \boldsymbol{h}_{k'} - \boldsymbol{h}_k, \boldsymbol{h}_k \rangle = -\frac{K}{K-1}, \quad \Rightarrow \quad \langle \boldsymbol{h}_{k'}, \boldsymbol{h}_k \rangle = -\frac{1}{K-1}, \forall k \neq k',$$

which holds only when $\boldsymbol{H}$ forms a simplex ETF according to the derivation for (55). Using the condition $\boldsymbol{w}_{k'} - \boldsymbol{w}_k = \boldsymbol{h}_{k'} - \boldsymbol{h}_k$ which indicates $\langle \boldsymbol{w}_{k'}, \boldsymbol{w}_k \rangle = \langle \boldsymbol{h}_{k'}, \boldsymbol{h}_k \rangle$, we can obtain that $\boldsymbol{W}$ is also a simplex ETF, which completes the proof.

- **$d+1 < K \leq 2d$: $\rho_{\text{one-vs-rest}}(\boldsymbol{W}^\star) = 1$ which can be achieved when $\{\boldsymbol{w}_k^\star\}$ are some vertices of a cross-polytope.**

  We first present and prove the following result that establishes an upper bound for $\rho_{\text{one-vs-rest}}(\boldsymbol{W})$ when $K \geq d+2$.

  Lemma C.9. *Suppose $K \geq d+2$, then for any $\boldsymbol{W} \in \mathcal{OB}(d, K)$, it holds that $\rho_{\text{one-vs-rest}}(\boldsymbol{W}) \leq 1$, with equality only if $\boldsymbol{0} \in \operatorname{conv}(\boldsymbol{W})$, where $\operatorname{conv}(\boldsymbol{W})$ is the convex hull of $\{\boldsymbol{w}_j\}_{j\in[K]}$.*

  Proof of Lemma C.9. Let $\boldsymbol{v}$ denote the (unique) point in $\operatorname{conv}(\boldsymbol{W})$ of minimum l_2 norm. By Carathéodory's theorem, $\boldsymbol{v}$ resides in the convex hull of $d+1$ of the points in $\boldsymbol{W}$. Since $K \geq d+2$ by assumption, there exists $k \in [K]$ such that $\boldsymbol{v} \in \operatorname{conv}(\{\boldsymbol{w}_j\}_{j\in[K]\setminus k})$ where $\boldsymbol{v}$ is the projection of $\boldsymbol{w}_k$ to $\operatorname{conv}(\{\boldsymbol{w}_j\}_{j\in[K]\setminus k})$. The result follows by noting the following two cases.

  - **Case I: $\boldsymbol{v} = \boldsymbol{0}$.** Then $\rho(\boldsymbol{W}) \leq \operatorname{dist}(\boldsymbol{w}_k, \operatorname{conv}(\{\boldsymbol{w}_j\}_{j\in[K]\setminus k})) \leq \|\boldsymbol{w}_k - \boldsymbol{v}\|_2 = \|\boldsymbol{w}_k\| = 1$.
  - **Case II: $\boldsymbol{v} \neq \boldsymbol{0}$.** Since $\boldsymbol{v}$ is the projection of $\boldsymbol{0}$ onto $\operatorname{conv}(\boldsymbol{W})$, the supporting hyperplane at $\boldsymbol{v}$ gives $\langle \boldsymbol{x}, \boldsymbol{v} \rangle \geq \|\boldsymbol{v}\|_2^2$ for every $\boldsymbol{x} \in \operatorname{conv}(\boldsymbol{W})$. In particular, taking $\boldsymbol{x} = \boldsymbol{w}_k$ implies

    $$\|\boldsymbol{w}_k - \boldsymbol{v}\|^2 = \|\boldsymbol{w}_k\|^2 - 2\langle \boldsymbol{w}_k, \boldsymbol{v} \rangle + \|\boldsymbol{v}\|^2 \leq 1 - \|\boldsymbol{v}\|^2 < 1,$$

    and so

    $$\rho_{\text{one-vs-rest}}(\boldsymbol{W}) \leq \operatorname{dist}(\boldsymbol{w}_k, \operatorname{conv}(\{\boldsymbol{w}_j\}_{j\in[K]\setminus k})) \leq \|\boldsymbol{w}_k - \boldsymbol{v}\| < 1.$$

  □

  According to Lemma C.9, when $K \geq d+2$, for any $\boldsymbol{W} \in \mathcal{OB}(d, K)$, it holds that $\rho_{\text{one-vs-rest}}(\boldsymbol{W}) \leq 1$. Moreover, when $d+2 \leq K \leq 2d$, we can verify that any sphere code $\boldsymbol{W}$ that achieves equality in Rankin's orthoplex bound (Fickus et al., 2017) $\max_{k\neq j} \langle \boldsymbol{w}_k, \boldsymbol{w}_j \rangle \geq 0$ is a softmax code. In particular, for each k, the point $\{\boldsymbol{w}_j\}_{j\in[K]\setminus k})$ necessarily reside in the half space $H_k = \{\boldsymbol{w} : \langle \boldsymbol{w}, \boldsymbol{w}_k \rangle \leq 0\}$, and so

  $$\operatorname{dist}(\boldsymbol{w}_k, \operatorname{conv}(\{\boldsymbol{w}_j\}_{j\in[K]\setminus k})) \geq \operatorname{dist}(\boldsymbol{w}_k, H_k) = 1.$$

  By minimizing over $k \in [K]$, it follows that $\rho_{\text{one-vs-rest}}(\boldsymbol{W}) \geq 1$. The previous conclusion ($\rho_{\text{one-vs-rest}}(\boldsymbol{W}) \leq 1$ always hols when $K \geq d+2$) implies that $\boldsymbol{W}$ is a softmax code.

  Thus, $\boldsymbol{W}^\star = \{\boldsymbol{w}_k^\star\}$ as some vertices of a cross-polytope is a Softmax Code. In addition, since $\boldsymbol{W}^\star = \{\boldsymbol{w}_k^\star\}$ as some vertices of a cross-polytope and $K \geq d+2$, we have $0 \in \operatorname{conv}(\{\boldsymbol{w}_j^\star\}_{j\in[K]\boldsymbol{k}})$ for any k, and hence it also holds that $\operatorname{dist}(\boldsymbol{w}_k^\star, \{\boldsymbol{w}_j^\star\}_{j\in[K]\boldsymbol{k}}) = 1$. Thus, there is no rattler for this case.

  □

C.5. Proof of Theorem 3.5

Theorem C.10 ($\mathcal{GNC}1$)**.** *Let $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ be an optimal solution to (5). For all k that is not a rattler of $\boldsymbol{W}^\star$, it holds that*

$$\overline{\boldsymbol{h}}_k^\star \doteq \boldsymbol{h}_{k,1}^\star = \cdots = \boldsymbol{h}_{k,n}^\star = \mathcal{P}_{\mathbb{S}^{d-1}}\left(\boldsymbol{w}_k^\star - \mathcal{P}_{\{\boldsymbol{w}_j^\star\}_{j\in[K]\setminus k}}(\boldsymbol{w}_k^\star)\right). \tag{57}$$

Proof. Let $\bar{k}$ be any non-rattler of $\boldsymbol{W}^*$. We have

$$\begin{aligned}
-\rho_{\text{one-vs-rest}}(\boldsymbol{W}^*) &= -\min_{k\in[K]} \operatorname{dist}(\boldsymbol{w}_k^*, \{\boldsymbol{w}_j^*\}_{j\in[K]\setminus k}) && \text{(Definition of } \rho_{\text{one-vs-rest}}()) \\
&= -\operatorname{dist}(\boldsymbol{w}_{\bar{k}}^*, \{\boldsymbol{w}_j^*\}_{j\in[K]\setminus \bar{k}}) && \text{(Definition of rattler)} \\
&= \min_{\boldsymbol{h}\in\mathbb{S}^{d-1}} \max_{k'\neq\bar{k}} \langle \boldsymbol{w}_{k'}^* - \boldsymbol{w}_{\bar{k}}^*, \boldsymbol{h}\rangle && \text{(Lemma C.4)} \\
&\leq \max_{i\in[n]} \max_{k'\neq\bar{k}} \langle \boldsymbol{w}_{k'}^* - \boldsymbol{w}_{\bar{k}}^*, \boldsymbol{h}_{\bar{k},i}^*\rangle && \text{(Property of max)} \\
&\leq \max_{i\in[n]} \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^* - \boldsymbol{w}_{k}^*, \boldsymbol{h}_{k,i}^*\rangle && \text{(Property of max)} \\
&= \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}^*, \boldsymbol{H}^*) && \text{(Definition of } \mathcal{L}_{\text{HardMax}}) \\
&= -\rho_{\text{one-vs-rest}}(\boldsymbol{W}^{\text{SC}}) && \text{(Eq. (42))} \\
&= -\rho_{\text{one-vs-rest}}(\boldsymbol{W}^*) && \text{(Theorem 3.2)}
\end{aligned} \tag{58}$$

Since the first and last expressions are identical, all inequalities holds with equality. Hence

$$\min_{\boldsymbol{h}\in\mathbb{S}^{d-1}} \max_{k'\neq\bar{k}} \langle \boldsymbol{w}_{k'}^* - \boldsymbol{w}_{\bar{k}}^*, \boldsymbol{h}\rangle = \max_{i\in[n]} \max_{k'\neq\bar{k}} \langle \boldsymbol{w}_{k'}^* - \boldsymbol{w}_{\bar{k}}^*, \boldsymbol{h}_{\bar{k},i}^*\rangle. \tag{59}$$

By Lemma C.4, the optimal $\boldsymbol{h}$ to the optimization problem on the left is unique. Hence, all $\boldsymbol{h}_{\bar{k},i}^*, i \in [n]$ must be equal. This concludes the proof.

□

C.6. Proof of Theorem 3.7

Proof. ($\implies$) Assume that any $(\boldsymbol{W}^\star, \boldsymbol{H}^\star) \in \arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K), \boldsymbol{H}\in\mathcal{OB}(d,nK)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H})$ satisfies $\boldsymbol{h}_{k,i}^\star = \boldsymbol{w}_k^*, \forall i \in [n], \forall k \in [K]$. We show that the Tammes problem and Softmax code are equivalent. This can be established trivially from the following two claims, namely,

$$\arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \min_{\boldsymbol{H}\in\mathcal{OB}(d,nK)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) = \arg\max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}), \tag{60}$$

and

$$\arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \min_{\boldsymbol{H}\in\mathcal{OB}(d,nK)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) = \arg\max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-one}}(\boldsymbol{W}). \tag{61}$$

In the rest of the proof we show that the claims (60) and(61) hold true.

We first establish (60). From Theorem 3.2, any solutions to the HardMax problem is a Softmax Code, i.e.,

$$\arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \min_{\boldsymbol{H}\in\mathcal{OB}(d,nK)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) \subseteq \arg\max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}). \tag{62}$$

Conversely, from Theorem C.7, any Softmax Code must also be a solution to the HardMax problem, i.e.,

$$\arg\max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}) \subseteq \arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \min_{\boldsymbol{H}\in\mathcal{OB}(d,nK)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) \tag{63}$$

Combining the above two relations we readily obtain (60).

We now establish (61). Let $(\boldsymbol{W}^\star, \boldsymbol{H}^\star) \in \arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K), \boldsymbol{H}\in\mathcal{OB}(d,nK)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H})$ be any solution to the HardMax problem, which by our assumption satisfies $\boldsymbol{h}_{k,i}^\star = \boldsymbol{w}_k^*, \forall i \in [n], \forall k \in [K]$. Using this condition and plugging in the definition of the HardMax function we obtain

$$\boldsymbol{W}^* \in \arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{w}_k\rangle = \arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}, \boldsymbol{w}_k\rangle = \arg\max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-one}}(\boldsymbol{W}), \tag{64}$$

where the first equality uses the fact that $\boldsymbol{w}_k$ has unit ℓ_2 norm for all $k \in [K]$, and the second equality follows trivially from the definition of $\rho_{\text{one-vs-one}}()$. This implies

$$\arg\min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \min_{\boldsymbol{H}\in\mathcal{OB}(d,nK)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) \subseteq \arg\max_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \rho_{\text{one-vs-one}}(\boldsymbol{W}). \tag{65}$$

We argue that the converse of the set inclusion above also holds. To see that, let

$$\boldsymbol{W}^{\text{Tammes}} \in \underset{\boldsymbol{W}\in\mathcal{OB}(d,K)}{\arg\max}\ \rho_{\text{one-vs-one}}(\boldsymbol{W})$$

be any solution to the Tammes problem, and let $\boldsymbol{H}^{\text{Tammes}} \in \mathcal{OB}(d, nK)$ be such that $\boldsymbol{h}_{k,i}^{\text{Tammes}} = \boldsymbol{w}_k^{\text{Tammes}}, \forall i \in [n], \forall k \in [K]$. We have

$$\begin{aligned}
\mathcal{L}_{\text{HardMax}}(\boldsymbol{W}^{\text{Tammes}}, \boldsymbol{H}^{\text{Tammes}}) &= \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^{\text{Tammes}} - \boldsymbol{w}_k^{\text{Tammes}}, \boldsymbol{h}_{k,i}^{\text{Tammes}} \rangle && \text{(Eq. (5))} \\
&= \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^{\text{Tammes}} - \boldsymbol{w}_k^{\text{Tammes}}, \boldsymbol{w}_k^{\text{Tammes}} \rangle && \text{(Using } \boldsymbol{h}_{k,i}^{\text{Tammes}} = \boldsymbol{w}_k^{\text{Tammes}}\text{)} \\
&= \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^{\text{Tammes}}, \boldsymbol{w}_k^{\text{Tammes}} \rangle - 1 && \text{(Using } \|\boldsymbol{w}_k^{\text{Tammes}}\|_2 = 1\text{)} \\
&= \min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}, \boldsymbol{w}_k \rangle - 1 && \text{(Definition of } \boldsymbol{W}^{\text{Tammes}}\text{)} \\
&= \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_k^\star \rangle - 1 && \text{(Eq. (64))} \\
&= \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_k^\star, \boldsymbol{w}_k^\star \rangle && \text{(Using } \|\boldsymbol{w}_k^\star\|_2 = 1\text{)} \\
&= \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_k^\star, \boldsymbol{h}_{k,i}^\star \rangle && \text{(Using } \boldsymbol{h}_{k,i}^\star = \boldsymbol{w}_k^\star\text{)} \\
&= \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}^\star, \boldsymbol{H}^\star).
\end{aligned} \tag{66}$$

This implies that $(\boldsymbol{W}^{\text{Tammes}}, \boldsymbol{H}^{\text{Tammes}})$ is also a solution to the HardMax problem. Hence,

$$\underset{\boldsymbol{W}\in\mathcal{OB}(d,K)}{\arg\max}\ \rho_{\text{one-vs-one}}(\boldsymbol{W}) \subseteq \underset{\boldsymbol{W}\in\mathcal{OB}(d,K)}{\arg\min} \min_{\boldsymbol{H}\in\mathcal{OB}(d,nK)} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}). \tag{67}$$

Combining (65) and (67) we obtain (61).

($\Longleftarrow$) Assume that the Tammes problem and the Softmax Codes are equivalent. Take any optimal solution $(\boldsymbol{W}^\star, \boldsymbol{H}^\star)$ to (5), i.e.,

$$\begin{aligned}
(\boldsymbol{W}^\star, \boldsymbol{H}^\star) &= \underset{\boldsymbol{W}\in\mathcal{OB}(d,K), \boldsymbol{H}\in\mathcal{OB}(d,nK)}{\arg\max} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) \\
&= \underset{\boldsymbol{W}\in\mathcal{OB}(d,K), \boldsymbol{H}\in\mathcal{OB}(d,nK)}{\arg\max} \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_{k,i} \rangle.
\end{aligned} \tag{68}$$

We show that $\boldsymbol{h}_{k,i}^\star = \boldsymbol{w}_k^*, \forall i \in [n], \forall k \in [K]$.

From Theorem 3.2, $\boldsymbol{W}^\star$ is a Softmax Code, i.e.,

$$\boldsymbol{W}^* = \underset{\boldsymbol{W}\in\mathcal{OB}(d,K)}{\arg\max} \min_{k\in[K]} \operatorname{dist}\Big(\boldsymbol{w}_k^\star, \{\boldsymbol{w}_j^\star\}_{j\in[K]\setminus k}\Big) = \underset{\boldsymbol{W}\in\mathcal{OB}(d,K)}{\arg\min} \max_{k\in[K]} \min_{\boldsymbol{h}_k\in\mathbb{S}^{d-1}} \max_{k'\neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_k \rangle, \tag{69}$$

where the second equality follows from Lemma C.4. By the assumption that Softmax Code has no rattler, we know that

$$\min_{\boldsymbol{h}_k\in\mathbb{S}^{d-1}} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_k^\star, \boldsymbol{h}_k \rangle \tag{70}$$

is independent of $k \in [K]$. We now use this result to show that, for any $\bar{k} \in [K]$ and $\bar{i} \in [n]$ the following result holds:

$$\boldsymbol{h}_{\bar{k},\bar{i}}^* \in \underset{\boldsymbol{h}_{\bar{k}}\in\mathbb{S}^{d-1}}{\arg\min} \max_{k'\neq\bar{k}} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{h}_{\bar{k}} \rangle. \tag{71}$$

To prove (71) by constructing a contradition, we assume that (71) does not hold, i.e.,

$$\boldsymbol{h}_{\bar{k},\bar{i}}^* \notin \underset{\boldsymbol{h}_{\bar{k}}\in\mathbb{S}^{d-1}}{\arg\min} \max_{k'\neq\bar{k}} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{h}_{\bar{k}} \rangle. \tag{72}$$

Using (72) we have

$$\begin{aligned}
\mathcal{L}_{\text{HardMax}}(\boldsymbol{W}^\star, \boldsymbol{H}^\star) &= \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_k^\star, \boldsymbol{h}_{k,i}^\star \rangle && \text{(Definition of } \mathcal{L}_{\text{HardMax}}) \\
&\geq \max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle && \text{(Property of max)} \\
&> \min_{\boldsymbol{h}_{\bar{k}}\in\mathbb{S}^{d-1}} \max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{h}_{\bar{k}} \rangle && \text{(Eq. (72))} \\
&= \min_{\boldsymbol{h}_k\in\mathbb{S}^{d-1}} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_k^\star, \boldsymbol{h}_k \rangle, \forall k \in [K] && \text{(Eq. (70))} \\
&= \max_{k\in[K]} \max_{i\in[n]} \min_{\boldsymbol{h}_{k,i}\in\mathbb{S}^{d-1}} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_k^\star, \boldsymbol{h}_{k,i} \rangle && (73) \\
&= \min_{\boldsymbol{H}\in\mathbb{S}^{d-1}} \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_k^\star, \boldsymbol{h}_{k,i} \rangle \\
&\geq \min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \min_{\boldsymbol{H}\in\mathbb{S}^{d-1}} \max_{k\in[K]} \max_{i\in[n]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'} - \boldsymbol{w}_k, \boldsymbol{h}_{k,i} \rangle && \text{(Property of max)} \\
&= \min_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \min_{\boldsymbol{H}\in\mathbb{S}^{d-1}} \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}, \boldsymbol{H}) && \text{(Definition of } \mathcal{L}_{\text{HardMax}}) \\
&= \mathcal{L}_{\text{HardMax}}(\boldsymbol{W}^\star, \boldsymbol{H}^\star) && \text{(Eq. (68))}
\end{aligned}$$

which is a contradiction. Hence, we have proved that (71) holds true. Now, using (71) we have that for any $\bar{k} \in [K]$ and $\bar{i} \in [n]$, it holds that

$$\begin{aligned}
&\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle \leq \max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star - \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{w}_{\bar{k}}^\star \rangle \\
\implies &\left(\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle \right) - \langle \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle \leq \left(\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_{\bar{k}}^\star \rangle \right) - \langle \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{w}_{\bar{k}}^\star \rangle \qquad (74) \\
\implies &\left(\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle \right) \leq \left(\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_{\bar{k}}^\star \rangle \right) + \langle \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle - 1.
\end{aligned}$$

On the other hand, using the result that Tammes problem and Softmax Code are equivalent, and the fact that $\boldsymbol{W}^\star$ is a Softmax Code, we know that $\boldsymbol{W}^\star$ is also a solution to the Tammes problem, i.e.,

$$\boldsymbol{W}^\star = \operatorname*{arg\,min}_{\boldsymbol{W}\in\mathcal{OB}(d,K)} \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}, \boldsymbol{w}_k \rangle. \tag{75}$$

Hence, it must hold that for any $\bar{k} \in [K]$ and $\bar{i} \in [n]$,

$$\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_{\bar{k}}^\star \rangle \leq \max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle. \tag{76}$$

To see why (76) holds, assume for the purpose of arriving at a contradiction that it does not hold, i.e.,

$$\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_{\bar{k}}^\star \rangle > \max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle. \tag{77}$$

Then, consider $\boldsymbol{W}^0 \in \mathcal{OB}(d, K)$ with $\boldsymbol{w}_{\bar{k}}^0 := \boldsymbol{h}_{\bar{k},\bar{i}}^\star$ for $k = \bar{k}$, and $\boldsymbol{w}_k^0 := \boldsymbol{w}_k^\star$ otherwise. It can be verified that

$$\begin{aligned}
\max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^0, \boldsymbol{w}_k^0 \rangle &= \max \left(\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^0, \boldsymbol{w}_{\bar{k}}^0 \rangle, \max_{\{k,k'\}\subseteq[K]\setminus\{\bar{k}\}, k'\neq k} \langle \boldsymbol{w}_{k'}^0, \boldsymbol{w}_k^0 \rangle \right) \\
&= \max \left(\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle, \max_{\{k,k'\}\subseteq[K]\setminus\{\bar{k}\}, k'\neq k} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_k^\star \rangle \right) \\
&\leq \max \left(\max_{k'\neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle, \max_{\{k,k'\}\subseteq[K], k'\neq k} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_k^\star \rangle \right) \\
&\leq \max_{\{k,k'\}\subseteq[K], k'\neq k} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_k^\star \rangle = \max_{k\in[K]} \max_{k'\neq k} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_k^\star \rangle,
\end{aligned} \tag{78}$$

where the last inequality is obtained from using (77). This implies, using the definition of $\boldsymbol{W}^\star$ in (75), that $\boldsymbol{W}^0$ is also a solution to the Tammes problem. Moreover, because of (77) it can be seen that $\boldsymbol{w}_{\bar{k}}^0$ is a rattler, contradicting with the

assumption that the Tammes problem has no rattlers. Hence, we have proved that (76) holds true. Combining (76) with (74), we have

$$\max_{k' \neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_{\bar{k}}^\star \rangle \leq \max_{k' \neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle \leq \left(\max_{k' \neq \bar{k}} \langle \boldsymbol{w}_{k'}^\star, \boldsymbol{w}_{\bar{k}}^\star \rangle \right) + \langle \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle - 1, \tag{79}$$

hence $\langle \boldsymbol{w}_{\bar{k}}^\star, \boldsymbol{h}_{\bar{k},\bar{i}}^\star \rangle \geq 1$. Since $\boldsymbol{w}_{\bar{k}}^\star$ and $\boldsymbol{h}_{\bar{k},\bar{i}}^\star$ are of unit ℓ_2 norm, we have $\boldsymbol{h}_{\bar{k},\bar{i}}^\star = \boldsymbol{w}_{\bar{k}}^\star$, which concludes the proof. □

C.7. Proof of Theorem 3.8

The equivalence can be established by noting that the optimal solutions for the Tammes problem for the case $d = 2$ (Cohn, 2022) and $K \leq d + 1$ (Fickus et al., 2017) are the same as those for the Softmax codes in Theorem 3.3.

C.8. Proof of Theorem 3.9

We first show that one-vs-rest and one-vs-one distances are mutually bounded by each other.

Theorem C.11. *For any classifier weights $\boldsymbol{W}$ with normalization $\|\boldsymbol{w}_k\|_2 = 1, \forall\, k \in [K]$, the one-vs-rest distance and one-vs-one distance obey the following relation:*

$$\frac{\rho_{\text{one-vs-one}}^2(\boldsymbol{W})}{2} \leq \rho_{\text{one-vs-rest}}(\boldsymbol{W}) \leq \rho_{\text{one-vs-one}}(\boldsymbol{W}). \tag{80}$$

Proof. On one hand, according to the definitions of the two margins, it is clear that $\rho_{\text{one-vs-rest}}(\boldsymbol{W}) \leq \rho_{\text{one-vs-one}}(\boldsymbol{W})$. On the other hand, according to Lemma C.4, we have

$$\begin{aligned}
\rho_{\text{one-vs-rest}}(\boldsymbol{W}) &= \min_k \operatorname{dist}(\boldsymbol{w}_k, \{\boldsymbol{w}_j\}_{j \in [K] \setminus k}) = \min_k \max_{\boldsymbol{H} \in \mathcal{OB}(d,K)} \left(- \max_{j \neq k} \langle \boldsymbol{w}_j - \boldsymbol{w}_k, \boldsymbol{h}_k \rangle \right) \\
&\geq \min_k \left(1 - \max_{j \neq k} \langle \boldsymbol{w}_j, \boldsymbol{w}_k \rangle \right) = \min_k \min_{j \neq k} \frac{\|\boldsymbol{w}_j - \boldsymbol{w}_k\|^2}{2}
\end{aligned}$$

where the first inequality follows by setting $\boldsymbol{h}_k = \boldsymbol{w}_k$. The proof is completed by noting that $\rho_{\text{one-vs-one}}^2(\boldsymbol{W}) = \min_k \min_{j \neq k} \|\boldsymbol{w}_j - \boldsymbol{w}_k\|^2$. □

Combining this theorem with the existing upper bound (see (Moore, 1974)) and a lower bound (see Lemma C.13) on the one-vs-one distance, we obtain the following result.

Theorem C.12. *Assuming $K \geq \sqrt{2\pi\sqrt{ed}}$ and letting $\Gamma(\cdot)$ denote the Gamma function, we have*

$$\frac{1}{2} \left[\frac{\sqrt{\pi}}{K} \frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}+1\right)} \right]^{\frac{2}{d-1}} \leq \max_{\boldsymbol{W} \in \mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}) \leq 2 \left[\frac{2\sqrt{\pi}}{K} \frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}\right)} \right]^{\frac{1}{d-1}}. \tag{81}$$

Using the property $a^{1-s} < \Gamma(a+1)/\Gamma(a+s) < (a+1)^{1-s}$ for any $a > 0, s \in (0,1)$, we can simplify the bounds in (3.9) to

$$\frac{1}{2} \left[\frac{\sqrt{\pi}}{K\sqrt{\frac{d}{2}+1}} \right]^{\frac{2}{d-1}} \leq \max_{\boldsymbol{W} \in \mathcal{OB}(d,K)} \rho_{\text{one-vs-rest}}(\boldsymbol{W}) \leq 2 \left[\frac{2\sqrt{\pi(\frac{d+1}{2})}}{K} \right]^{\frac{1}{d-1}},$$

Lemma C.13. *For any K and d, we have*

$$\left[\frac{\sqrt{\pi}}{K} \frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}+1\right)} \right]^{\frac{1}{d-1}} \leq \max_{\boldsymbol{W} \in \mathcal{OB}(d,K)} \rho_{\text{one-vs-one}}(\boldsymbol{W}). \tag{82}$$

Proof. We prove the theorem by constructing a $\widehat{\boldsymbol{W}} \in \mathcal{OB}(d,K)$ and deriving a lower bound on $\rho_{\text{one-vs-one}}(\widehat{\boldsymbol{W}})$. Then, this lower bound is a lower bound for $\max_{\boldsymbol{W} \in \mathcal{OB}(d,K)} \rho_{\text{one-vs-one}}(\boldsymbol{W})$ as well owning to the relation

$$\rho_{\text{one-vs-one}}(\widehat{\boldsymbol{W}}) \leq \max_{\boldsymbol{W} \in \mathcal{OB}(d,K)} \rho_{\text{one-vs-one}}(\boldsymbol{W}). \tag{83}$$

The tightness of this lower bound naturally depends on the choice of $\widehat{\boldsymbol{W}}$, which we explain below.

Instead of constructing $\widehat{\boldsymbol{W}}$ directly *for a given* K, we first consider the construction of a $\boldsymbol{W}^0(\rho_0)$ *for any given* $\rho_0 > 0$. Later, we will specify a ρ_0 and construct a $\widehat{\boldsymbol{W}}$ from $\boldsymbol{W}^0(\rho_0)$. To simplify the notation, we write $\boldsymbol{W}^0(\rho_0)$ as $\boldsymbol{W}^0$.

We construct $\boldsymbol{W}^0$ using the following procedure from the proof of [Lemma 12](You, 2018):

- Set $\boldsymbol{w}_1^0$ to be an arbitrary vector in $\mathbb{S}^{d-1}$.
- For any $k > 1$, take $\boldsymbol{w}_k^0$ to be any vector in $\mathbb{S}^{d-1}$ that satisfies $\|\boldsymbol{w}_k^0 - \boldsymbol{w}_{k'}^0\|_2 > \rho_0$, for all $k' < k$.
- Terminate the process when no such point exists.

It is easy to see that this procedure terminates in finite number of steps; see [Lemma 12](You, 2018) for a rigorous argument. Assume that the total number of generated vectors is K_0. Collect these vectors into the columns of a matrix $\boldsymbol{W}^0$. By construction, $\boldsymbol{W}^0$ has the following property, which will be used later:

$$\rho_{\text{one-vs-one}}(\boldsymbol{W}^0) > \rho_0. \tag{84}$$

We now derive a lower bound for K_0. First, note that $\boldsymbol{W}^0$ provides a ρ_0-*covering* of the unit sphere $\mathbb{S}^{d-1}$. That is, for any $\boldsymbol{v} \in \mathbb{S}^{d-1}$, there must exist a $k \in \{1, \dots, K_0\}$ such that $\|\boldsymbol{v} - \boldsymbol{w}_k^0\|_2 \le \rho_0$. Otherwise, there would exist a $\boldsymbol{w} \in \mathbb{S}^{d-1}$ that satisfies $\|\boldsymbol{w} - \boldsymbol{w}_k^0\|_2 > \rho_0$ for all $k \le K_0$, contradicting the termination condition in the construction of $\boldsymbol{W}^0$.

Geometrically, a ρ_0-covering is a set of points such that, the union of Euclidean balls of radius ρ_0 centered at those points cover the entire unit sphere. Leveraging this interpretation, we provide a geometric method for bounding the number of points in a ρ_0-covering from below. Concretely, given any $\boldsymbol{w} \in \mathbb{S}^{d-1}$, we denote $\mathbb{S}_{\rho_0}^{d-1}(\boldsymbol{w}) = \{\boldsymbol{v} \in \mathbb{S}^{d-1}, \|\boldsymbol{v} - \boldsymbol{w}\|_2 \le \rho_0\}$, which is the spherical cap centered at $\boldsymbol{w}$ with radius ρ_0. Since $\boldsymbol{W}^0$ provides a ρ_0-covering, we have

$$\bigcup_{k=1}^{K_0} \mathbb{S}_{\rho_0}^{d-1}(\boldsymbol{w}_k^0) \subseteq \mathbb{S}^{d-1} \implies \sum_{k=1}^{K_0} \sigma_{d-1}(\mathbb{S}_{\rho_0}^{d-1}(\boldsymbol{w}_k^0)) \ge \sigma_{d-1}(\mathbb{S}^{d-1}), \tag{85}$$

where σ_{d-1} denotes the uniform area measure on $\mathbb{S}^{d-1}$. By noting that $\sigma_{d-1}(\mathbb{S}_{\rho_0}^{d-1}(\boldsymbol{w}_k^0))$ is independent of k, we obtain

$$K_0 \cdot \sigma_{d-1}(\mathbb{S}_{\rho_0}^{d-1}(\boldsymbol{w})) \ge \sigma_{d-1}(\mathbb{S}^{d-1}), \tag{86}$$

for an arbitrary choice of $\boldsymbol{w} \in \mathbb{S}^{d-1}$. In above, note that the quantity $\sigma_{d-1}(\mathbb{S}_{\rho_0}^{d-1}(\boldsymbol{w}))/\sigma_{d-1}(\mathbb{S}^{d-1})$ is the proportion of the area of $\mathbb{S}^{d-1}$ that lies in the spherical cap $\mathbb{S}_{\rho_0}^{d-1}(\boldsymbol{w})$. By a geometric argument, [Lemma 9](You, 2018) provides an upper bound for it which we rewrite here:

$$\frac{\sigma_{d-1}(\mathbb{S}_{\rho_0}^{d-1}(\boldsymbol{w}))}{\sigma_{d-1}(\mathbb{S}^{d-1})} \le \frac{v_{d-1}}{v_d} \left(\rho_0 \sqrt{1 - \frac{\rho_0^2}{4}}\right)^{d-1}. \tag{87}$$

In above, $v_d \doteq \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2}+1)}$ is the volumn of the unit Euclidean ball in $\mathbb{R}^d$. Combining (87) with (86), we obtain a lower bound on K_0 as

$$K_0 \ge \frac{\sigma_{d-1}(\mathbb{S}^{d-1})}{\sigma_{d-1}(\mathbb{S}_{\rho_0}^{d-1}(\boldsymbol{w}))} \ge \frac{v_d}{v_{d-1}} \frac{1}{\left(\rho_0 \sqrt{1 - \frac{\rho_0^2}{4}}\right)^{d-1}} \ge \frac{v_d}{v_{d-1}} \frac{1}{\rho_0^{d-1}}. \tag{88}$$

We are now ready to construct a $\widehat{\boldsymbol{W}} \in \mathcal{O}\mathrm{B}(d, K)$. Take

$$\rho_0 = \left(\frac{v_d}{v_{d-1} \cdot K}\right)^{\frac{1}{d-1}}. \tag{89}$$

By using (88) we have $K_0 \ge K$. Hence, we may construct $\widehat{\boldsymbol{W}}$ as the set of any K distinct columns of $\boldsymbol{W}^0$. In particular, from (83), (84) and (89), we obtain

$$\max_{\boldsymbol{W} \in \mathcal{O}\mathrm{B}(d,K)} \rho_{\text{one-vs-one}}(\boldsymbol{W}) \ge \rho_{\text{one-vs-one}}(\widehat{\boldsymbol{W}}) \ge \rho_{\text{one-vs-one}}(\boldsymbol{W}^0) > \rho_0 = \left(\frac{v_d}{v_{d-1} \cdot K}\right)^{\frac{1}{d-1}} = \left[\frac{\sqrt{\pi}}{K} \frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}+1\right)}\right]^{\frac{1}{d-1}},$$

where the second inequality follows trivially from the definition of $\rho_{\text{one-vs-one}}(\cdot)$. □