Prevalence of Neural Collapse during the terminal phase of deep learning training

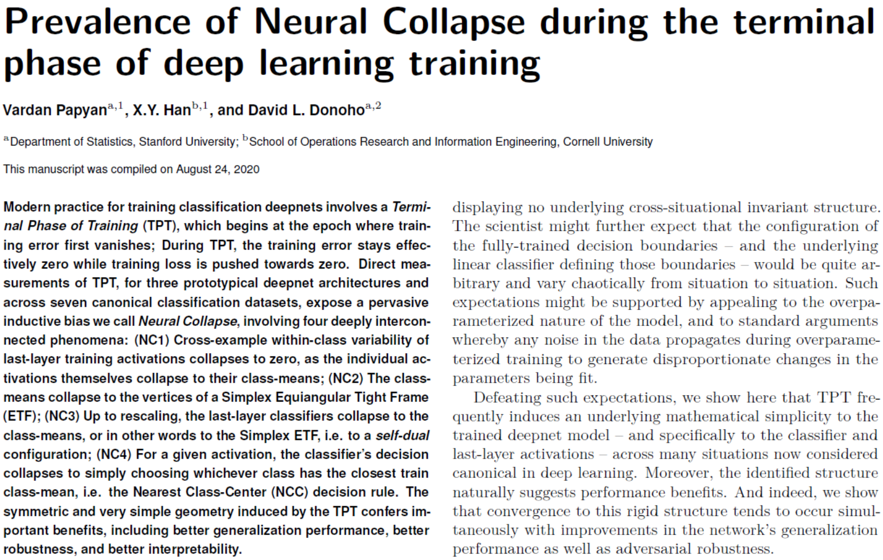
Zhigang Huang

Nankai University

School of Statistics and Data Science

July 26, 2025

PNAS 2020



Prevalence of Neural Collapse during the terminal phase of deep learning training

Vardan Papyan[a,1], X.Y. Han[b,1], and David L. Donoho[a,2]

[a]Department of Statistics, Stanford University; [b]School of Operations Research and Information Engineering, Cornell University

This manuscript was compiled on August 24, 2020

Modern practice for training classification deepnets involves a *Terminal Phase of Training* (TPT), which begins at the epoch where training error first vanishes; During TPT, the training error stays effectively zero while training loss is pushed towards zero. Direct measurements of TPT, for three prototypical deepnet architectures and across seven canonical classification datasets, expose a pervasive inductive bias we call *Neural Collapse*, involving four deeply interconnected phenomena: (NC1) Cross-example within-class variability of last-layer training activations collapses to zero, as the individual activations themselves collapse to their class-means; (NC2) The class-means collapse to the vertices of a Simplex Equiangular Tight Frame (ETF); (NC3) Up to rescaling, the last-layer classifiers collapse to the class-means, or in other words to the Simplex ETF, i.e. to a *self-dual* configuration; (NC4) For a given activation, the classifier's decision collapses to simply choosing whichever class has the closest train class-mean, i.e. the Nearest Class-Center (NCC) decision rule. The symmetric and very simple geometry induced by the TPT confers important benefits, including better generalization performance, better robustness, and better interpretability.

displaying no underlying cross-situational invariant structure. The scientist might further expect that the configuration of the fully-trained decision boundaries – and the underlying linear classifier defining those boundaries – would be quite arbitrary and vary chaotically from situation to situation. Such expectations might be supported by appealing to the overparameterized nature of the model, and to standard arguments whereby any noise in the data propagates during overparameterized training to generate disproportionate changes in the parameters being fit.

Defeating such expectations, we show here that TPT frequently induces an underlying mathematical simplicity to the trained deepnet model – and specifically to the classifier and last-layer activations – across many situations now considered canonical in deep learning. Moreover, the identified structure naturally suggests performance benefits. And indeed, we show that convergence to this rigid structure tends to occur simultaneously with improvements in the network's generalization performance as well as adversarial robustness.

Contents

1. Introduction
2. Overview of Neural Collapse
3. Evidence of Neural Collapse
4. Benefits of Neural Collapse
5. Inception of Neural Collapse
6. Other Related Works
7. Summary

Contents

1. Introduction
2. Overview of Neural Collapse
3. Evidence of Neural Collapse
4. Benefits of Neural Collapse
5. Inception of Neural Collapse
6. Other Related Works
7. Summary

Background & Motivation

Over the last decade

- One might expect the trained networks to exhibit many **particularities**...
- A scientist might anticipate impossible to find any empirical regularities across **different datasets and architectures**....
- Researchers regarded them as **blackboxes** with little that could be understood...

In this paper

- Present extensive measurements across image-classification datasets and architectures, exposing **a common empirical pattern**....
- Observe the emergence of **a simple and highly symmetric geometry** of the deepnet features and of the deepnet classifier....

Deep Net Training Paradigm

- Networks trained iteratively for 300 epochs.
- Networks trained beyond **zero misclassification error**.
- Training continues towards zero **cross-entropy loss**.
- Networks are massively overparameterized.

We call this **Terminal Phase of Training (TPT)**.

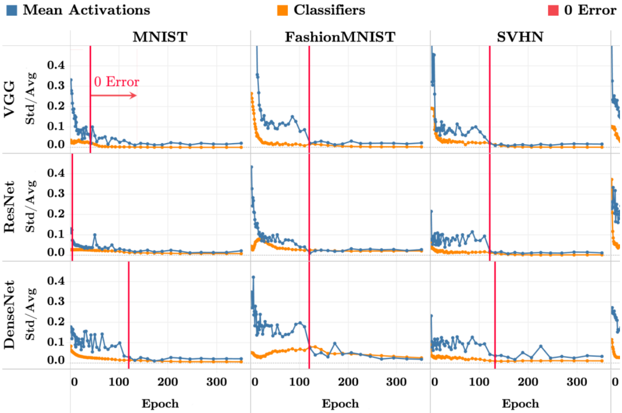
Definition (TPT)

The phase that begins at the epoch where the training error first vanishes. During TPT, the training error remains effectively zero while the training loss is pushed towards zero .

Notice

In TPT, the loss is non-zero even if the classification error is zero

Example of TPT



Remark

The red vertical line marks the begining of the effective beginning of TPT, i.e, the epoch when the training accuracy reaches **99.6%** for ImageNet and **99.9%** for the remaining datasets. [2] [6]

Deep Net Terminology

- **Labels**: c $=$ 1 , 2, ..., C, where C is the number of classes.
    - C $=$ 10 digits for MNIST, Fashion-MNIST, SVHN and CIFAR-10
    - C $=$ 100 classes for CIFAR-100
    - C $=$ 1000 classes for ImageNet
- **Observations**: $x_{i,c}$
- **Features**: $h^{\ell}_{i,c} = h^{\ell}(x_{i,c})$
- **Last layer features**: $h^{L}_{i,c} = h(x_{i,c})$
- **Classifiers**: w_c
- **Estimated class**: $\hat{c}(x) = argmax_c \langle w_c, h(x) \rangle$

Deep Net Terminology

- Implicitly weights and features depend on training epoch t.
- Training error at epoch t:

$$\text{TrainError}_t \equiv \text{Ave}_{i,c}\, \mathbf{1}_{\{\hat{c}(x_{i,c};w^t;h^t_{i,c}) \neq c\}}$$

- Training loss at epoch t:

$$\text{TrainLoss}_t \equiv \text{Ave}_{i,c}\left\{-\log(p_c(x_{i,c}))\right\},$$

where

$$p_c(x_{i,c}) = \frac{\exp(\langle w_c, h(x_{i,c})\rangle)}{\sum_{c'} \exp(\langle w_{c'}, h(x_{i,c})\rangle)}.$$

Terminology

- Global mean

$$\mu_G \triangleq \text{Ave}_{i,c} h_{i,c}$$

- Class means

$$\mu_c \triangleq \text{Ave}_i h_{i,c}, c = 1, 2, \ldots, C$$

- Within-class covariances: $\sum_W \in \mathbb{R}^{p \times p}$

$$\sum_W \triangleq \text{Ave}_{i,c} \left\{ (h_{i,c} - \mu_c)(h_{i,c} - \mu_c)^\top \right\}$$

- Between-class covariances: $\sum_B \in \mathbb{R}^{p \times p}$

$$\sum_B \triangleq \text{Ave}_c \left\{ (\mu_c - \mu_G)(\mu_c - \mu_G)^\top \right\}$$

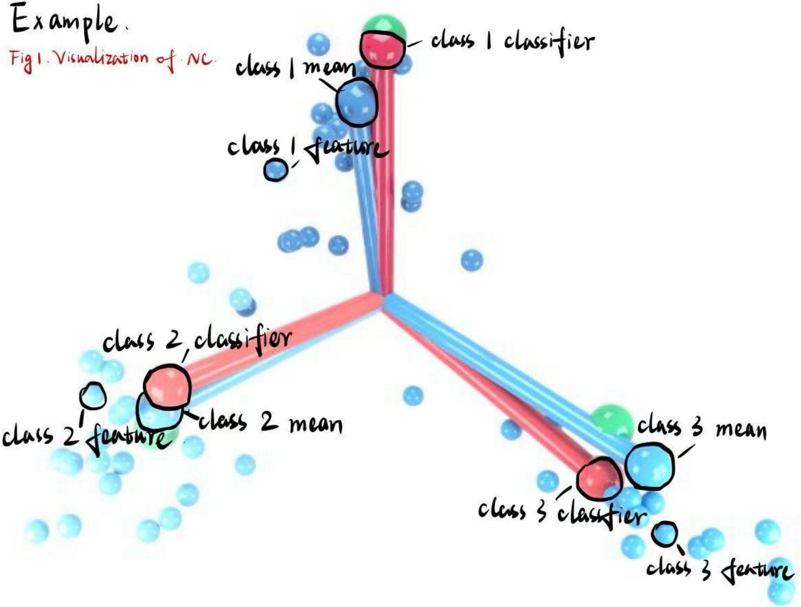
- Train total covariance: $\sum_T = \sum_B + \sum_W \in \mathbb{R}^{p \times p}$

$$\sum_T \triangleq \text{Ave}_{i,c} \left\{ (h_{i,c} - \mu_G)(h_{i,c} - \mu_G)^\top \right\}$$

Terminology Visualization

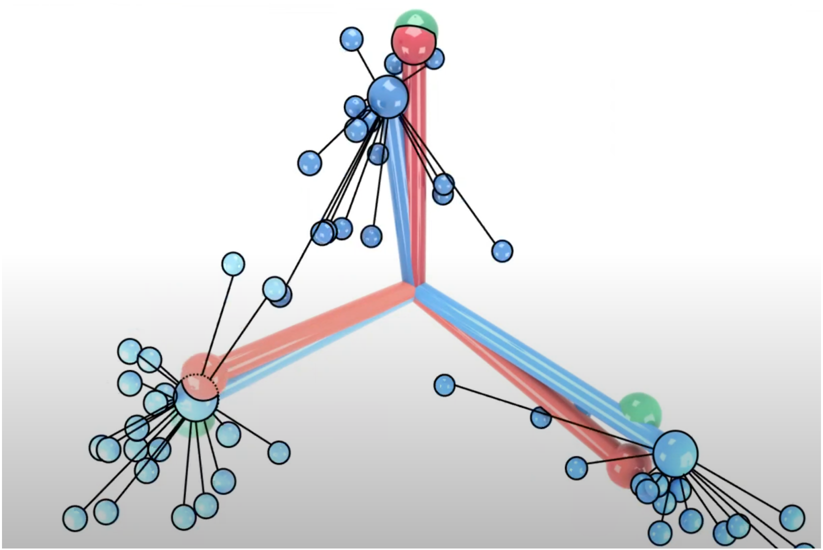
Example.

Fig 1. Visualization of NC



Terminology Visualization

Within-class covariances(figure from here)



Contents

1 Introduction

2 Overview of Neural Collapse

3 Evidence of Neural Collapse

4 Benefits of Neural Collapse

5 Inception of Neural Collapse

6 Other Related Works

7 Summary

What is Neural Collapse?

Definition (Neural Collapse (NC))

Neural Collapse is a phenomenon observed during the terminal phase of training deep neural networks, characterized by the following four properties:

- NC1: Variability collapse
- NC2: Convergence to Simplex ETF
- NC3: Convergence to self-duality
- NC4: Simplification to Nearest Class-Center (NCC)

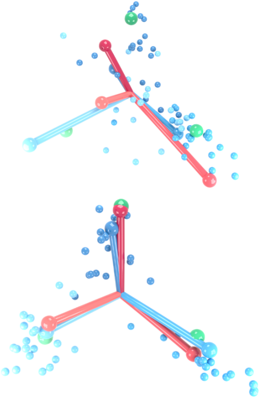
These properties emerge as the network approaches zero cross-entropy loss, leading to a highly symmetric and simplified geometry of the features and classifiers.

What is Neural Collapse?

It refers to a certain simplification in the structure of the learned representations and the classifier's weights that occurs as training progresses, i.e. the convergence of certain patterns in the feature space of neural networks as they approach the optimal solution during training.

Example

Training a network to classify cats, dogs, birds (3 classes): early on, their features overlap. In TPT, Neural Collapse occurs: cat features cluster around a "cat mean", dogs around a "dog mean", birds around a "bird mean". Within-class features grow similar, cross-class distinct, letting the network classify via nearest class mean. [5]

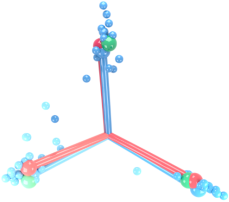


Variability collapse

- As training progresses, the within-class variation of the activations becomes negligible as these activations collapse to their class-means.
- $\sum_W \to 0$

Neural Collapse(NC)

NC2:Convergence to Simplex ETF



Class mean convergence to Simplex ETF(Equiangular Tight Frame)

- $\left|\|\mu_c - \mu_G\|_2 - \|\mu_{c'} - \mu_G\|_2\right| \to 0 \quad \forall\, c, c'$
- $\langle \tilde{\mu}_c, \tilde{\mu}_{c'} \rangle \to \frac{C}{C-1}\delta_{c,c'} - \frac{1}{C-1} \quad \forall\, c, c'$
- $\delta_{c,c'}$ is the Kronecker delta symbol

Neural Collapse(NC)

NC2:Convergence to Simplex ETF

Simplex ETF Definition

A **Simplex ETF** is a collection of points in $\mathbb{R}^p$ specified by the columns of matrix $M = \alpha U M^*$, where:

- $M^* = \sqrt{\frac{C}{C-1}}\left(I - \frac{1}{C}\mathbf{1}\mathbf{1}^\mathsf{T}\right) \in \mathbb{R}^{C\times C}$
- $\alpha \in \mathbb{R}_+$ is a scale factor
- $U \in \mathbb{R}^{p\times C}$ $(p \geq C)$ is a partial orthogonal matrix $(U^\mathsf{T} U = I)$

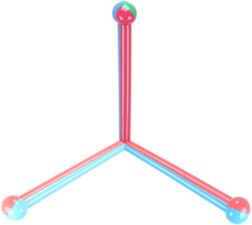
Why cosine is $-\frac{1}{C-1}$?

Assumption: $\|\mu_c'\| = s; \mu_c' \cdot \mu_{c'}' = t \; (\forall c \neq c'); \sum_{c=1}^{C} \mu_c' = 0$

$$\sum_{c=1}^{C} \mu_c' = 0 \implies \mu_1' \cdot \mu_1' + \sum_{c=2}^{C} \mu_1' \cdot \mu_c' = 0 \iff \langle \tilde{\mu}_c, \tilde{\mu}_{c'} \rangle = \frac{t}{s^2} \to -\frac{1}{C-1}$$

Neural Collapse(NC)

NC3:Convergence to self-duality



Convergence to self-duality

- The class-means and linear classifiers – although mathematically quite different objects, living in dual vector spaces – converge to each other, up to rescaling.
- Combined with (NC2), this implies a complete symmetry in the network classifiers' decisions
- $$\left\| \frac{\mu_c^t}{\|\mu_c^t\|} - \frac{w_c^t}{\|w_c^t\|} \right\| \to 0$$

Simplification to Nearest Class-Center (NCC)

- For a given deepnet activation, the network classifier converges to choosing whichever class has the nearest train class-mean (in standard Euclidean distance)

NC4 Formula

The classifier decision rule evolves as:

$$\arg\max_{c'} \langle \boldsymbol{w}_{c'}, \boldsymbol{h} \rangle + b_{c'} \rightarrow \arg\min_{c'} \|\boldsymbol{h} - \boldsymbol{\mu}_{c'}\|_2$$

where $\tilde{\boldsymbol{\mu}}_c = (\boldsymbol{\mu}_c - \boldsymbol{\mu}_G)/\|\boldsymbol{\mu}_c - \boldsymbol{\mu}_G\|_2$ are the renormalized the class-means, $\dot{\boldsymbol{M}} = [\boldsymbol{\mu}_c - \boldsymbol{\mu}_G, c = 1, \ldots, C] \in \mathbb{R}^{p \times C}$ is the matrix obtained by stacking the class-means into the columns of a matrix, and $\delta_{c,c'}$ is the Kronecker delta symbol.

Neural Collapse Phenomena (NC1-NC4)

As t increases:

[NC1] **Within-class variability collapse:**

$$\Sigma_W^t \to 0$$

[NC2] **Class mean convergence to simplex ETF:**

$$\frac{\langle \mu_c^t, \mu_{c'}^t \rangle}{\|\mu_c^t\| \cdot \|\mu_{c'}^t\|} \to \begin{cases} 1, & \text{for } c = c' \\ \dfrac{-1}{C-1}, & \text{for } c \neq c' \end{cases}$$

[NC3] **Convergence to self-duality:**

$$\left\| \frac{\mu_c^t}{\|\mu_c^t\|} - \frac{w_c^t}{\|w_c^t\|} \right\| \to 0$$

[NC4] **Behavioral convergence to nearest class center:**

$$\hat{c}^t(X) \to \arg\min_{c'} \|h(x) - \mu_{c'}^t\|$$

Contents

1. Introduction
2. Overview of Neural Collapse
3. Evidence of Neural Collapse
4. Benefits of Neural Collapse
5. Inception of Neural Collapse
6. Other Related Works
7. Summary

Setting and Methodology

We consider deep classifiers following a typical paradigm:

$$\text{Data } \boldsymbol{x} \xrightarrow{\text{Network Layers}} \boldsymbol{h}(\boldsymbol{x}) \xrightarrow{\text{Linear Classifier}} \text{Class Prediction}$$

where:

- $\boldsymbol{h}(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}^p$ is the "last layer activation" of the network's forward pass, mapping the original d-dimensional input to a p-dimensional feature space.
- The linear classifier is defined by weights $\boldsymbol{W} \in \mathbb{R}^{C \times p}$ and biases $\boldsymbol{b} \in \mathbb{R}^C$, with the prediction logic:

$$\hat{y} = \arg\max_{c'} \langle \boldsymbol{w}_{c'}, \boldsymbol{h}(\boldsymbol{x}) \rangle + b_{c'}$$

where $\boldsymbol{w}_{c'}$ is the c'-th column of $\boldsymbol{W}$, corresponding to the classification weights for the c'-th class.

Setting and Methodology

Training Objective and Loss Function

For a balanced dataset (with N samples per class $\{\boldsymbol{x}_{i,c}\}_{i=1}^{N}$), the model is trained by minimizing the cross-entropy loss:

$$\min_{\boldsymbol{\theta},\boldsymbol{W},\boldsymbol{b}} \sum_{c=1}^{C} \sum_{i=1}^{N} \mathcal{L}\big(\boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(x_{i,c}) + \boldsymbol{b}, \boldsymbol{y}_c\big)$$

- $\boldsymbol{\theta}$: Parameters of the network layers (excluding the linear classifier).
- $\mathcal{L} : \mathbb{R}^C \times \mathbb{R}^C \to \mathbb{R}^+$: Cross-entropy loss, measuring the discrepancy between predictions and one-hot labels $\boldsymbol{y}_c$.

Datasets and Architectures

- MNIST, FashionMNIST, CIFAR10, CIFAR100, SVHN, STL10, and ImageNet
- VGG, ResNet, and DenseNet

Optimization methodology SGD + momentum

Some questions

Neural Collapse in Imbalanced Datasets?

Imbalanced datasets may affect neural collapse, such as ”minority collapse” in imbalanced training. How does imbalance impact NC1-NC4? Can a unified framework describe collapse in both balanced and imbalanced cases? [3]

Neural Collapse Under MSE Loss?

The recently discovered Neural Collapse (NC) phenomenon occurs pervasively in today’s deep net training paradigm of driving cross-entropy (CE) loss towards zero. In contrast, as a preliminary, the authors of this paper empirically establish that NC emerges in such MSE [4] - trained deep nets as well through experiments on three canonical networks and five benchmark datasets.

Evidence

NC1

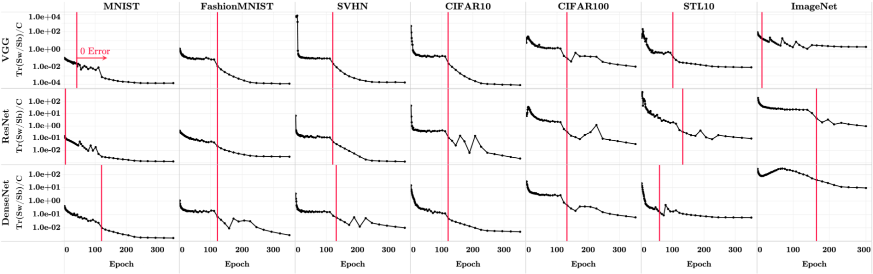


Fig. 6. Training within-class variation collapses: The formatting and technical details are as described in Section 3. In each array cell, the vertical axis (log-scaled) shows the magnitude of the between-class covariance compared to the within-class covariance of the train activations . Mathematically, this is represented by $\mathrm{Tr}\left\{\boldsymbol{\Sigma}_W \boldsymbol{\Sigma}_B^{\dagger}\right\}/C$ where $\mathrm{Tr}\{\cdot\}$ is the trace operator, $\boldsymbol{\Sigma}_W$ is the within-class covariance of the last-layer activations of the training data, $\boldsymbol{\Sigma}_B$ is the corresponding between-class covariance, C is the total number of classes, and $[\cdot]^{\dagger}$ is Moore-Penrose pseudoinverse. This value decreases as a function of training – indicating collapse of within-class variation.

Evidence

NC2’

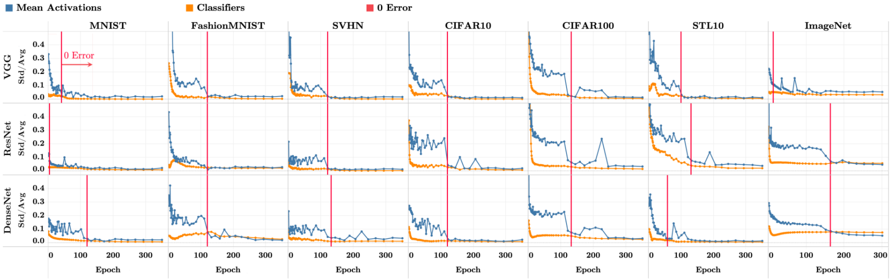


Fig. 2. Train class-means become equinorm: The formatting and technical details are as described in Section 3. In each array cell, the vertical axis shows the coefficient of variation of the centered class-mean norms as well as the network classifiers norms. In particular, the **blue** line shows Std$_c(\|\boldsymbol{\mu}_c - \boldsymbol{\mu}_G\|_2)$/Avg$_c(\|\boldsymbol{\mu}_c - \boldsymbol{\mu}_G\|_2)$ where $\{\boldsymbol{\mu}_c\}$ are the class-means of the last-layer activations of the training data and $\boldsymbol{\mu}_G$ is the corresponding train global-mean; the **orange** line shows Std$_c(\|\boldsymbol{w}_c\|_2)$/Avg$_c(\|\boldsymbol{w}_c\|_2)$ where $\boldsymbol{w}_c$ is the last-layer classifier of the c-th class. As training progresses, the coefficients of variation of both class-means and classifiers decreases.

Evidence

NC2”

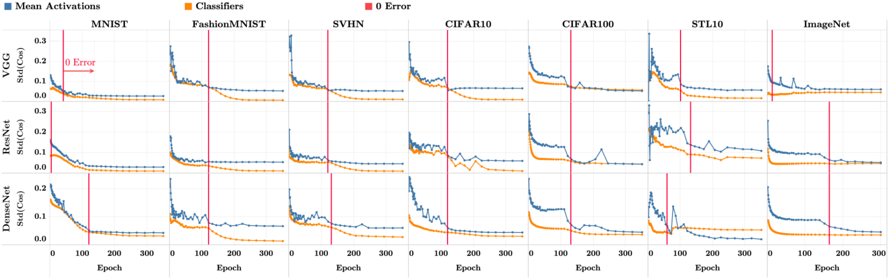


Fig. 3. Classifiers and train class-means approach equiangularity: The formatting and technical details are as described in Section 3. In each array cell, the vertical axis shows the standard deviation of the cosines between pairs of centered class-means and classifiers across all distinct pairs of classes c and c'. Mathematically, denote $\cos_{\boldsymbol{\mu}}(c, c') = \langle \boldsymbol{\mu}_c - \boldsymbol{\mu}_G, \boldsymbol{\mu}_{c'} - \boldsymbol{\mu}_G \rangle / (\|\boldsymbol{\mu}_c - \boldsymbol{\mu}_G\|_2 \|\boldsymbol{\mu}_{c'} - \boldsymbol{\mu}_G\|_2$ and $\cos_{\boldsymbol{w}}(c, c') = \langle \boldsymbol{w}_c, \boldsymbol{w}_{c'} \rangle / (\|\boldsymbol{w}_c\|_2 \|\boldsymbol{w}_{c'}\|_2)$ where $\{\boldsymbol{w}_c\}_{c=1}^{C}$, $\{\boldsymbol{\mu}_c\}_{c=1}^{C}$, and $\boldsymbol{\mu}_G$ are as in Figure 2. We measure $\text{Std}_{c,c' \neq c}(\cos_{\boldsymbol{\mu}}(c, c'))$ (**blue**) and $\text{Std}_{c,c' \neq c}(\cos_{\boldsymbol{w}}(c, c'))$ (**orange**). As training progresses, the standard deviations of the cosines approach zero indicating equiangularity.

Evidence

NC2"'

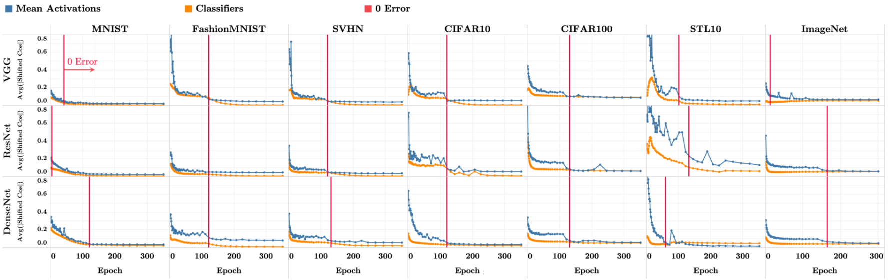


Fig. 4. Classifiers and train class-means approach maximal-angle equiangularity: The formatting and technical details are as described in Section 3. We plot in the vertical axis of each cell the quantities Avg$_{c,c'}$ $|\cos_{\boldsymbol{\mu}}(c,c') + 1/(C-1)|$ (**blue**) and Avg$_{c,c'}$ $|\cos_{\boldsymbol{w}}(c,c') + 1/(C-1)|$ (**orange**), where $\cos_{\boldsymbol{\mu}}(c,c')$ and $\cos_{\boldsymbol{w}}(c,c')$ are as in Figure 3. As training progresses, the convergence of these values to zero implies that all cosines converge to $-1/(C-1)$. This corresponds to the maximum separation possible for globally centered, equiangular vectors.

Evidence

NC3

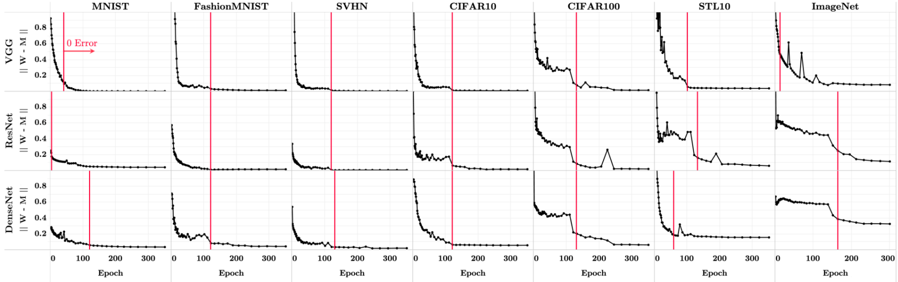


Fig. 5. Classifier converges to train class-means: The formatting and technical details are as described in Section 3. In the vertical axis of each cell, we measure the distance between the classifiers and the centered class-means, both rescaled to unit-norm. Mathematically, denote $\widetilde{\boldsymbol{M}} = \dot{\boldsymbol{M}}/\|\dot{\boldsymbol{M}}\|_F$ where $\dot{\boldsymbol{M}} = [\boldsymbol{\mu}_c - \boldsymbol{\mu}_G : c = 1, \dots, C] \in \mathbb{R}^{p \times C}$ is the matrix whose columns consist of the centered train class-means; denote $\widetilde{\boldsymbol{W}} = \boldsymbol{W}/\|\boldsymbol{W}\|_F$ where $\boldsymbol{W} \in \mathbb{R}^{C \times p}$ is the last-layer classifier of the network. We plot the quantity $\|\widetilde{\boldsymbol{W}}^\top - \widetilde{\boldsymbol{M}}\|_F^2$ on the vertical axis. This value decreases as a function of training, indicating the network classifier and the centered-means matrices become proportional to each other (self-duality).

Evidence

NC4

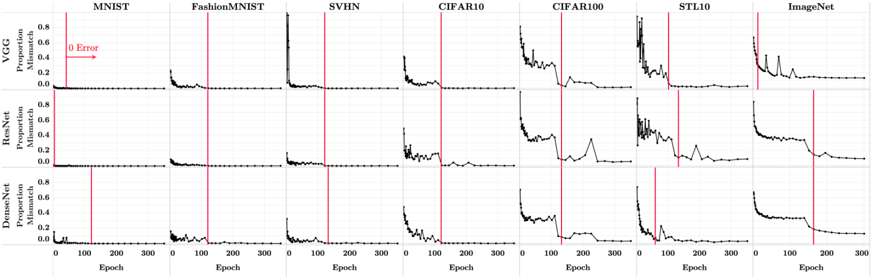


Fig. 7. Classifier behavior approaches that of Nearest Class-Center: The formatting and technical details are as described in Section 3. In each array cell, we plot the proportion of examples (vertical axis) in the *testing* set where network classifier disagrees with the result that would have been obtained by choosing $\arg\min_c \|\boldsymbol{h} - \boldsymbol{\mu}_c\|_2$ where $\boldsymbol{h}$ is a last-layer test activation, and $\{\boldsymbol{\mu}_c\}_{c=1}^{C}$ are the class-means of the last-layer train activations. As training progresses, the disagreement tends to zero, showing the classifier's behavioral simplification to the nearest train class-mean decision rule.

Contents

1 Introduction

2 Overview of Neural Collapse

3 Evidence of Neural Collapse

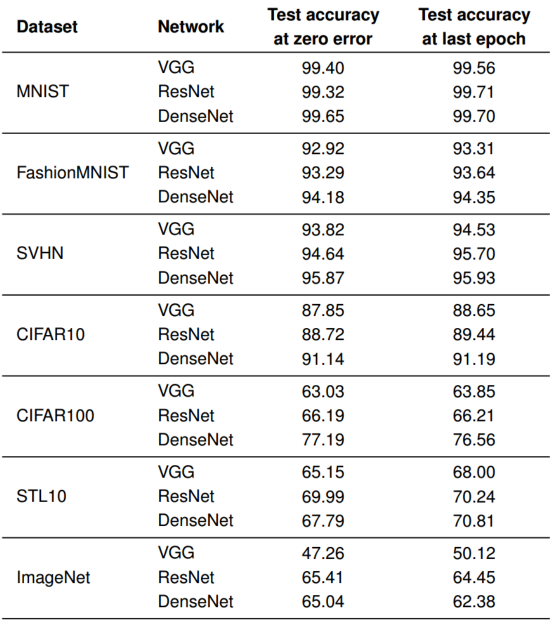
4 Benefits of Neural Collapse

5 Inception of Neural Collapse

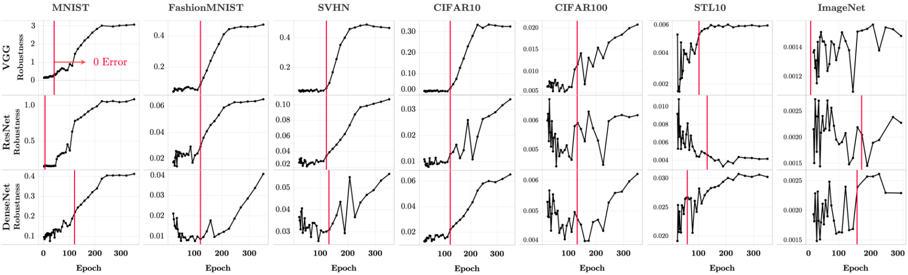
6 Other Related Works

7 Summary

Table 1. Training beyond zero-error improves test performance.



Dataset	Network	Test accuracy at zero error	Test accuracy at last epoch
MNIST	VGG	99.40	99.56
	ResNet	99.32	99.71
	DenseNet	99.65	99.70
FashionMNIST	VGG	92.92	93.31
	ResNet	93.29	93.64
	DenseNet	94.18	94.35
SVHN	VGG	93.82	94.53
	ResNet	94.64	95.70
	DenseNet	95.87	95.93
CIFAR10	VGG	87.85	88.65
	ResNet	88.72	89.44
	DenseNet	91.14	91.19
CIFAR100	VGG	63.03	63.85
	ResNet	66.19	66.21
	DenseNet	77.19	76.56
STL10	VGG	65.15	68.00
	ResNet	69.99	70.24
	DenseNet	67.79	70.81
ImageNet	VGG	47.26	50.12
	ResNet	65.41	64.45
	DenseNet	65.04	62.38



Training beyond zero-error improves adversarial robustness

Summary of NC Benefits

1. Stronger generalization with higher test accuracy
2. Greater robustness against adversarial attacks
3. Improved interpretability through geometric regularity
4. Potential for more efficient network architectures

These benefits highlight the importance of TPT and NC in deep learning optimization.

Contents

1. Introduction
2. Overview of Neural Collapse
3. Evidence of Neural Collapse
4. Benefits of Neural Collapse
5. Inception of Neural Collapse
6. Other Related Works
7. Summary

Inception of Neural Collapse: Understanding the Driving Forces

- Neural Collapse (NC) describes the geometric regularities observed in feature space during the terminal phase of deep learning training.
- Beyond empirical observation, understanding *why* NC emerges is crucial for a complete theoretical picture.
- This section explores potential theoretical underpinnings, drawing connections to:
  - Implicit Bias of Gradient Descent (Soudry et al., 2018)
  - Information Theory Perspective (Webb & Lowe's principles, broader information-theoretic views)

Implicit Bias of Gradient Descent: A Driver for NC

- **Key Idea (Soudry et al., 2018):** For linearly separable data, Gradient Descent (GD) on homogeneous, unregularized linear models converges to a **maximum margin classifier**, even without explicit regularization.
- **Mechanism:** GD pushes the decision boundary away from the training points, maximizing the margin until the solution saturates (weights tend to infinity while maintaining direction).
- This phenomenon is particularly relevant in **overparameterized deep networks**, where exact interpolation of training data is possible, and GD continues to optimize beyond zero training error.

Implicit Bias of Gradient Descent: A Driver for NC

- **Relevance to Neural Collapse (NC):** The maximum margin behavior directly explains the geometric properties of NC.
- To maximize margins, features belonging to the same class are driven to a single point (or a very compact cluster), minimizing within-class variance (**NC1**).
- To ensure maximal separation between classes, the class means are pushed to be as far apart as possible, yet constrained by their "boundary" in the feature space. For C classes in a p-dimensional space, the most "efficient" way to achieve maximal, symmetric separation is the **Simplex ETF (NC2)**.
- The resulting maximum margin classifier's weights naturally align with the directions defined by these separated class means, leading to the **Nearest Class Center (NCC)** decision rule (**NC3** & **NC4**).

Information Theory Perspective: Efficient Coding & Neural Collapse

- **Core Idea:** Neural networks, particularly their feature extractors, can be viewed as trying to learn an **efficient representation** of the input data.
- **Efficient Coding:** This involves preserving essential information for the task (e.g., class identity) while discarding irrelevant variability or noise.
- **Webb and Lowe's principles (1990):** Early work showed that optimized internal representations in multi-layer classifiers perform non-linear discriminant analysis. This aligns with the idea of learning discriminative and compact features.

Information Theory Perspective: Efficient Coding & Neural Collapse

- **Core Idea:** Neural networks learn an **efficient representation** of the input data.
- **Efficient Coding:** Preserving essential information (e.g., class identity) while discarding irrelevant variability.
- **Relevance to Neural Collapse (NC):** NC achieves an optimal balance between compression and discrimination.
- **NC1:** Minimizing within-class variability compresses irrelevant variations.
- **NC2:** Maximizing between-class separability ensures distinct codes in the feature space.
- **Related Frameworks:** Rate-Distortion Theory and Information Bottleneck principle suggest optimal representations compress irrelevant details while retaining task-relevant information.

Contents

1. Introduction
2. Overview of Neural Collapse
3. Evidence of Neural Collapse
4. Benefits of Neural Collapse
5. Inception of Neural Collapse
6. Other Related Works
7. Summary

Other Related Works

- The Prevalence of Neural Collapse in Neural Multivariate Regression [1]
- Linguistic collapse: Neural collapse in (large) language models [7]
- A geometric analysis of neural collapse with unconstrained features [10]

Contents

1. Introduction
2. Overview of Neural Collapse
3. Evidence of Neural Collapse
4. Benefits of Neural Collapse
5. Inception of Neural Collapse
6. Other Related Works
7. Summary

Key Findings of Neural Collapse

- Occurs during the **Terminal Phase of Training (TPT)**: training continues after zero error, minimizing cross-entropy loss.
- Characterized by four interconnected properties:
    - **NC1**: Within-class variability of last-layer activations collapses to zero ($\sum_W \to 0$).
    - **NC2**: Class means converge to vertices of a Simplex Equiangular Tight Frame (ETF) (equinorm, equiangular).
    - **NC3**: Last-layer classifiers align with class means (self-duality, up to rescaling).
    - **NC4**: Classifier decisions simplify to the Nearest Class-Center (NCC) rule.
- Observed across 7 canonical datasets (e.g., MNIST, ImageNet) and 3 architectures (VGG, ResNet, DenseNet).

Implications and Significance

- **Performance Benefits**:
  - Improved generalization (test accuracy gains, Table 1).
  - Enhanced adversarial robustness (larger perturbations required, Figure 8).
- **Theoretical Insights**:
  - Reveals rigid, symmetric geometry in deep networks, challenging the ”black box” view.
  - Explains order in overparameterized models; TPT is critical for structural optimization.
- **Practical Value**: Potential to design efficient architectures (e.g., fixed Simplex ETF classifiers).

Thank you for listening!

References I

[1] George Andriopoulos, Zixuan Dong, Li Guo, Zifan Zhao, and Keith Ross.
The prevalence of neural collapse in neural multivariate regression.
Advances in Neural Information Processing Systems, 37:126417–126451, 2024.

[2] Rajmadhan Ekambaram, Dmitry B. Goldgof, and Lawrence O. Hall.
Finding label noise examples in large scale datasets.
In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2420–2424, 2017.

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

References II

[4] X.Y. Han, Vardan Papyan, and David L. Donoho.
Neural collapse under MSE loss: Proximity to and dynamics on the central path.
In *International Conference on Learning Representations*, 2022.

[5] Na Lei, Zhongxuan Luo, Shing-Tung Yau, and David Xianfeng Gu.
Geometric understanding of deep learning, 2018.

[6] Nicolas M. Muller and Karla Markert.
Identifying mislabeled instances in classification datasets.
In *2019 International Joint Conference on Neural Networks (IJCNN)*, page 1–8. IEEE, July 2019.

[7] Robert Wu and Vardan Papyan.
Linguistic collapse: Neural collapse in (large) language models.
Advances in Neural Information Processing Systems, 37:137432–137473, 2024.

References III

[8] Enhao Zhang, Chaohua Li, Chuanxing Geng, and Songcan Chen.
All-around neural collapse for imbalanced classification.
IEEE Transactions on Knowledge and Data Engineering, 37(8):4460–4470, 2025.

[9] Zhisheng Zhong, Jiequan Cui, Yibo Yang, Xiaoyang Wu, Xiaojuan Qi, Xiangyu Zhang, and Jiaya Jia.
Understanding imbalanced semantic segmentation through neural collapse.
In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19550–19560, June 2023.

References IV

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