Landscape Analysis of Neural Collapse

Qing Qu
Dept. of EECS, University of Michigan
May 21, 2021
• Analyzes the **global landscape** of the training loss based on the **unconstrained feature model**

• Explains the ubiquity of **Neural Collapse** of the learned representations of the network

Understanding Deep Neural Networks

\[
\psi_\Theta(x) = W_L \sigma (W_{L-1} \cdots \sigma (W_1 x + b_1) + b_{L-1}) + b_L
\]

\[
\Theta := \{W_\ell, b_\ell\}_{\ell=1}^L \quad \sigma(\cdot): \text{nonlinear activations}
\]
Understanding Deep Neural Networks

\[
\min_{\Theta} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}_{CE}(\psi_{\Theta}(x_{k,i}), y_k) + \lambda \| \Theta \|_F^2
\]

- \( i \)-th input in the \( k \)-th class
- One-hot vector for the \( k \)-th class
Fundamental Challenges: Optimization

Landscape in Classical Optimization (abundant algorithms & theory)

Landscape of Modern Deep Neural Networks Credited to [Li’17]
Optimization: Existing Results

Existing analysis are based on various simplifications:

- **Go Linear**: deep linear networks [Kawaguchi’16], deep matrix factorizations [Arora’19], etc.
- **Go Shallow**: Two-layer neural networks [Safran’18, Liang’18], etc.
- **Go Wide**: Neural tangent kernels [Jacot’18, Allen-Zhu’18, Du’19], mean-field analysis [Mei’19, Sirignano’19], etc.

Most of results *hardly* provide much insights for practical neural networks.
Features – What NNs (Conceptually) Designed to Learn

Wishful Design: NNs learn rich feature representations across different levels?
Neural Collapse in Classification

\[ \psi_{\Theta}(x) = W_L \sigma(W_{L-1} \cdots \sigma(W_1 x + b_1) + b_{L-1}) + b_L \]

Data in the Input Space

Last-layer classifier

\[ \phi_{\theta}(x) = h \]

Last-layer feature

Neural Collapse in the Feature Space

Simplex Equiangular Tight Frames (Simplex ETF)
Neural Collapse in Classification

Prevalence of neural collapse during the terminal phase of deep learning training

Vardan Papyan, X. Y. Han, and David L. Donoho

See all authors and affiliations

PNAS October 6, 2020 117 (40) 24652-24663; first published September 21, 2020;
https://doi.org/10.1073/pnas.2015509117

Contributed by David L. Donoho, August 18, 2020 (sent for review July 22, 2020; reviewed by Helmut Boelscke and Stéphane Mallat)
Neural Collapse: Symmetry and Structures

Balanced training dataset with $n = n_1 = n_2 = \cdots = n_K$, and

$$W := W_L, \quad H := [h_{1,1} \cdots h_{K,n}].$$

Neural Collapse (NC) means that

1) **Within-Class Variability Collapse on $H$:** features of each class collapse to class-mean with zero variability;

$$h_{k,i} \rightarrow \overline{h}_k, \quad \forall k \in [K], \ i \in [n].$$

2) **Convergence to Simplex ETF on $H$:** the class means are linearly separable, and maximally distant;

$$M^\top M = \frac{K}{K-1} \left( I_K - \frac{1}{K} 1_K 1_K^\top \right), \quad M = \alpha U \overline{H}$$
Neural Collapse: Symmetry and Structures

Balanced training dataset with \( n = n_1 = n_2 = \cdots = n_K \), and
\[
W := W_L, \quad H := \begin{bmatrix} h_{1,1} & \cdots & h_{K,n} \end{bmatrix}.
\]

Neural Collapse (NC) means that

3) **Convergence to Self-Duality** \((W,H)\): the last-layer classifiers are **perfected matched** with the class-means of features.

\[
\mathbf{w}^k = \beta \overline{\mathbf{h}}_k, \quad \forall \ k \in [K].
\]

3) **Simple Decision Rule** via Nearest Class-Center decision.
Simplification: Unconstrained Features

\[ \psi_\Theta(x) = W_L \sigma(W_{L-1} \cdots \sigma(W_1 x + b_1) + b_{L-1}) + b_L \]

\[ \phi_\theta(x) = h \]

Last-layer classifier \( \rightarrow \) \( \phi_\theta \) \( \rightarrow \) Last-layer feature

Treat \( H = \begin{bmatrix} h_{1,1} & \cdots & h_{K,n} \end{bmatrix} \) as a **free** optimization variable
Simplification: Unconstrained Features

\[ \psi_\Theta(x) = W_L \sigma(W_{L-1} \cdots \sigma(W_1 x + b_1) + b_{L-1}) + b_L \]

**Last-layer classifier** \( \phi_\theta(x) = h \)  **Last-layer feature**

Treat \( H = [h_{1,1} \cdots h_{K,n}] \) as a **free** optimization variable

\[
\min_{W,H,b} \frac{1}{K_n} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}_{CE}(W h_{k,i} + b, y_k) + \frac{\lambda_W}{2} \| W \|_F^2 + \frac{\lambda_H}{2} \| H \|_F^2 + \frac{\lambda_b}{2} \| b \|_2^2
\]
Simplification: Unconstrained Features

\[ \psi_\Theta(x) = W_L \sigma (W_{L-1} \cdots \sigma (W_1 x + b_1) + b_{L-1}) + b_L \]

Treat \( H = [h_{1,1} \cdots h_{K,n}] \) as a free optimization variable

\[
\min_{W,H,b} \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}_{CE}(Wh_{k,i} + b, y_k) + \frac{\lambda_W}{2} \|W\|_F^2 + \frac{\lambda_H}{2} \|H\|_F^2 + \frac{\lambda_b}{2} \|b\|_2^2
\]

- **Validity:** Modern network are highly overparameterized, that can approximate any point in the feature space [Shaham’18];
- **State-of-the-Art:** also called Layer-Peeled Model [Fang’21], existing work [E’20, Lu’20, Mixon’20, Fang’21] only studied global optimality conditions.
Main Theoretical Results

**Theorem (Informal)** Consider the nonconvex loss with unconstrained feature model with $K < d$ and balanced data

$$\min_{W,H,b} \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}_{CE}(Wh_{k,i} + b, y_k) + \frac{\lambda W}{2} \|W\|_F^2 + \frac{\lambda H}{2} \|H\|_F^2 + \frac{\lambda b}{2} \|b\|_2^2$$

- **(Global Optimality)** Any global solution $(W_*, H_*)$ satisfies the NC properties (1-4).
- **(Benign Global Landscape)** The function has no spurious local minimizer and is a strict saddle function, with negative curvature for non-global critical point.
Main Theoretical Results

Theorem (Informal) Consider the nonconvex loss with unconstrained feature model with $K < d$ and balanced data.

- **(Global Optimality)** Any global solution $(W_*, H_*)$ satisfies the NC properties (1-4).
- **(Benign Global Landscape)** The function has no spurious local minimizer and is a strict saddle function, with negative curvature for nonglobal critical point.

Message: deep networks always learn Neural Collapse features and classifiers, provably.
Experiment: NC is Algorithm Independent

CIFAR-10 Dataset, ResNet18, with different training algorithms

Measure of Within-Class Variability
Measure of Between-Class Separation
Measure of Self-Duality Collapse
Generalization is Algorithm dependent

MINST

CIFAR-10
Experiment: NC Occurs for Random Labels

CIFAR-10 Dataset, MLP, random labels with varying network width

Validity of Unconstrained Feature Model: Learned last-layer features and classifiers seems to be independent of input!
Experiment: NC Occurs for Random Labels

CIFAR-10 Dataset, ResNet18, random labels with varying network width

Measure of Within-Class Variability

Measure of Self-Duality Collapse

Training Error

Validity of Unconstrained Feature Model: Learned last-layer features and classifiers seems to be independent of input!
Experiment: NC with Different Weight Decays

CIFAR-10 Dataset, ResNet18, different weight decay

Test Accuracy: 99.57% vs. 99.60% (MINST); 77.92% vs. 78.42% (CIFAR-10)
Implications for Practical Network Training

Observation: For NC features, when $K \leq d$ the best classifier is given by the Simplex ETF

$$W_* = [\mu_1 \cdots \mu_K]^\top.$$
Implications for Practical Network Training

Observation: For NC features, when $K \leq d$ the best classifier is given by the Simplex ETF

$$W_\star = [\mu_1 \cdots \mu_K]^\top.$$  

- Implication 1: No need to learn the classifier
  - Just fix them as a Simplex ETF
  - Save 8%, 12%, and 53% parameters for ResNet50, DenseNet169, and ShuffleNet!
Implications for Practical Network Training

Observation: For NC features, when $K \leq d$ the best classifier is given by the Simplex ETF

$$W_\star = [\mu_1 \cdots \mu_K]^\top.$$  

- **Implication 1:** No need to learn the classifier
  - Just fix them as a Simplex ETF
  - Save 8%, 12%, and 53% parameters for ResNet50, DenseNet169, and ShuffleNet!

- **Implication 2:** No need of large feature dimension $d$
  - Just use feature dim $d = \#\text{class } K$ (e.g., $d=10$ for CIFAR10)
  - Further saves 21% and 4.5% parameters for ResNet18 and ResNet50!
**Experiment: Fixed Classifier with** \( d = K \)

ResNet50, CIFAR10, Comparison of **Learned vs. Fixed Classifiers of** \( W \)

Measure of Between-Class Separation

Training Accuracy

Testing Accuracy

Training with fixed last-layer classifiers achieves **on-par performance** with learned classifiers.
Summary and Discussion


• Through landscape analysis under unconstrained feature model, we provide a complete characterization of learned representation of deep networks.

• The understandings of learned representations could shed lights on generalization, robustness, and transferability.
Future Directions

• Study Deeper Networks

  • Fix the last layer classifier $W$ as the Simplex ETF, and conduct NTK analysis for the learning dynamics of features $H$?
  • Recursively study the features of each layer from output?

[Paypan’19]
Future Directions

• **Study the settings** $K > d$ (self-supervised learning)
  • The solution with weight decay loss, and constrained loss are different. What is the optimal solution configuration?

$$W^\top W = \beta I$$

$$\|w^1\|_2 = \cdots = \|w^K\|_2 = 1$$

$$\max_{i,j} \langle w^i, w^j \rangle \leq \mu$$
Future Directions

• Study generalization through the representation?

• Study robustness via tradeoff?

• Other Practical Implications?
Thank You!