Extended Unconstrained Features Model for Exploring Deep Neural Collapse

Tom Tirer and Joan Bruna

New York University Center for Data Science

The 39th International Conference on Machine Learning (ICML 2022)

The Neural Collapse Phenomenon

DNN-based classifiers can be typically represented as $\psi_{\boldsymbol{\Theta}}(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}) + \boldsymbol{b}$, where $\boldsymbol{\Theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\theta}\}$ are the learned parameters.

Common practice: keep optimizing the network's parameters after the training error vanishes to further push the training loss toward zero.

Papyan et al. (2020) empirically observed a "Terminal Phase of Training" phenomenon, dubbed "Neural Collapse" (NC).

The Neural Collapse Phenomenon

DNN-based classifiers can be typically represented as $\psi_{\boldsymbol{\Theta}}(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}) + \boldsymbol{b}$, where $\boldsymbol{\Theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\theta}\}$ are the learned parameters.

Common practice: keep optimizing the network's parameters after the training error vanishes to further push the training loss toward zero.

Papyan et al. (2020) empirically observed a "Terminal Phase of Training" phenomenon, dubbed "Neural Collapse" (NC).

The Neural Collapse Phenomenon

DNN-based classifiers can be typically represented as $\psi_{\boldsymbol{\Theta}}(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}) + \boldsymbol{b}$, where $\boldsymbol{\Theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\theta}\}$ are the learned parameters.

Common practice: keep optimizing the network's parameters after the training error vanishes to further push the training loss toward zero.

Papyan et al. (2020) empirically observed a "Terminal Phase of Training" phenomenon, dubbed "Neural Collapse" (NC).

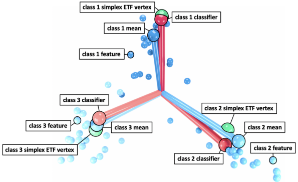
The Neural Collapse Phenomenon

DNN-based classifiers can be typically represented as $\psi_{\mathbf{\Theta}}(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}) + \boldsymbol{b}$, where $\mathbf{\Theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\theta}\}$ are the learned parameters.

Common practice: keep optimizing the network's parameters after the training error vanishes to further push the training loss toward zero.

NC is made of four (simultaneous) components:

- (NC1): The learned features $\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x})$ of within-class samples converge to their mean (i.e., the intraclass variance vanishes)
- (NC2): After centering by their global mean, the limiting means of different classes exhibit a simplex equiangular tight frame (ETF) structure
- (NC3): The limit of the last weights $\boldsymbol{W}^{\top}$ is aligned with this simplex ETF
- (NC4): The classification decision converges to the nearest class center (in feature space) rule



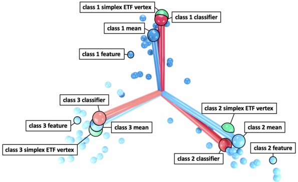
The Neural Collapse Phenomenon

DNN-based classifiers can be typically represented as $\psi_{\boldsymbol{\Theta}}(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}) + \boldsymbol{b}$, where $\boldsymbol{\Theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\theta}\}$ are the learned parameters.

Common practice: keep optimizing the network's parameters after the training error vanishes to further push the training loss toward zero.

NC is made of four (simultaneous) components:

- (NC1): The learned features $\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x})$ of within-class samples converge to their mean (i.e., the intraclass variance vanishes)
- (NC2): After centering by their global mean, the limiting means of different classes exhibit a simplex equiangular tight frame (ETF) structure
- (NC3): The limit of the last weights $\boldsymbol{W}^\top$ is aligned with this simplex ETF
- (NC4): The classification decision converges to the nearest class center (in feature space) rule



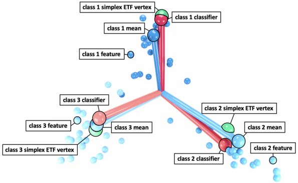
The Neural Collapse Phenomenon

DNN-based classifiers can be typically represented as $\psi_{\boldsymbol{\Theta}}(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}) + \boldsymbol{b}$, where $\boldsymbol{\Theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\theta}\}$ are the learned parameters.

Common practice: keep optimizing the network's parameters after the training error vanishes to further push the training loss toward zero.

NC is made of four (simultaneous) components:

- (NC1): The learned features $\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x})$ of within-class samples converge to their mean (i.e., the intraclass variance vanishes)
- (NC2): After centering by their global mean, the limiting means of different classes exhibit a simplex equiangular tight frame (ETF) structure
- (NC3): The limit of the last weights $\boldsymbol{W}^\top$ is aligned with this simplex ETF
- (NC4): The classification decision converges to the nearest class center (in feature space) rule



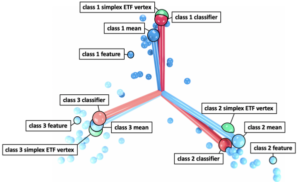
The Neural Collapse Phenomenon

DNN-based classifiers can be typically represented as $\psi_{\boldsymbol{\Theta}}(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}) + \boldsymbol{b}$, where $\boldsymbol{\Theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\theta}\}$ are the learned parameters.

Common practice: keep optimizing the network's parameters after the training error vanishes to further push the training loss toward zero.

NC is made of four (simultaneous) components:

- (NC1): The learned features $\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x})$ of within-class samples converge to their mean (i.e., the intraclass variance vanishes)
- (NC2): After centering by their global mean, the limiting means of different classes exhibit a simplex equiangular tight frame (ETF) structure
- (NC3): The limit of the last weights $\boldsymbol{W}^\top$ is aligned with this simplex ETF
- (NC4): The classification decision converges to the nearest class center (in feature space) rule



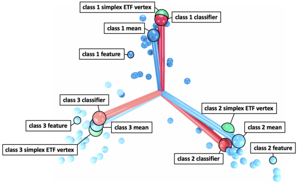
The Neural Collapse Phenomenon

DNN-based classifiers can be typically represented as $\psi_{\boldsymbol{\Theta}}(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}) + \boldsymbol{b}$, where $\boldsymbol{\Theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\theta}\}$ are the learned parameters.

Common practice: keep optimizing the network's parameters after the training error vanishes to further push the training loss toward zero.

NC is made of four (simultaneous) components:

- (NC1): The learned features $\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x})$ of within-class samples converge to their mean (i.e., the intraclass variance vanishes)
- (NC2): After centering by their global mean, the limiting means of different classes exhibit a simplex equiangular tight frame (ETF) structure
- (NC3): The limit of the last weights $\boldsymbol{W}^\top$ is aligned with this simplex ETF
- (NC4): The classification decision converges to the nearest class center (in feature space) rule



The Unconstrained Features Model

The typical way to optimize a DNN's parameters (empirical risk minim.):

$$\min_{\boldsymbol{\Theta}} \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}\left(\boldsymbol{W} \boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}_{k,i}) + \boldsymbol{b}, \boldsymbol{y}_k \right) + \mathcal{R}\left(\boldsymbol{\Theta} \right),$$

where $\boldsymbol{y}_k \in \mathbb{R}^K$ is the one-hot vector with 1 in its k-th entry, $\mathcal{L}(\cdot,\cdot)$ is a loss function (e.g., cross-entropy or MSE), and $\mathcal{R}(\cdot)$ is a regularization term (e.g., squared L_2-norm).

Mixon et al. (2020) suggested to explore NC via an Unconstrained Features Model (UFM) — The features $\{\boldsymbol{h}_{k,i} = \boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}_{k,i})\}$ are treated as free optimization variables:

$$\min_{\boldsymbol{W}, \boldsymbol{b}, \{\boldsymbol{h}_{k,i}\}} \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}\left(\boldsymbol{W} \boldsymbol{h}_{k,i} + \boldsymbol{b}, \boldsymbol{y}_k \right) + \mathcal{R}\left(\boldsymbol{W}, \boldsymbol{b}, \{\boldsymbol{h}_{k,i}\} \right).$$

The UFM rationale: Modern over-parameterized DNNs can adapt their feature mapping to almost any training data.

The Unconstrained Features Model

The typical way to optimize a DNN's parameters (empirical risk minim.):

$$\min_{\boldsymbol{\Theta}} \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}\left(\boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}_{k,i}) + \boldsymbol{b}, \boldsymbol{y}_k\right) + \mathcal{R}\left(\boldsymbol{\Theta}\right),$$

where $\boldsymbol{y}_k \in \mathbb{R}^K$ is the one-hot vector with 1 in its k-th entry, $\mathcal{L}(\cdot, \cdot)$ is a loss function (e.g., cross-entropy or MSE), and $\mathcal{R}(\cdot)$ is a regularization term (e.g., squared L_2-norm).

Mixon et al. (2020) suggested to explore NC via an Unconstrained Features Model (UFM) — The features $\{\boldsymbol{h}_{k,i} = \boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}_{k,i})\}$ are treated as free optimization variables:

$$\min_{\boldsymbol{W}, \boldsymbol{b}, \{\boldsymbol{h}_{k,i}\}} \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}\left(\boldsymbol{W}\boldsymbol{h}_{k,i} + \boldsymbol{b}, \boldsymbol{y}_k\right) + \mathcal{R}\left(\boldsymbol{W}, \boldsymbol{b}, \{\boldsymbol{h}_{k,i}\}\right).$$

The UFM rationale: Modern over-parameterized DNNs can adapt their feature mapping to almost any training data.

The Unconstrained Features Model

The typical way to optimize a DNN's parameters (empirical risk minim.):

$$\min_{\boldsymbol{\Theta}} \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}\left(\boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}_{k,i}) + \boldsymbol{b}, \boldsymbol{y}_k\right) + \mathcal{R}\left(\boldsymbol{\Theta}\right),$$

where $\boldsymbol{y}_k \in \mathbb{R}^K$ is the one-hot vector with 1 in its k-th entry, $\mathcal{L}(\cdot,\cdot)$ is a loss function (e.g., cross-entropy or MSE), and $\mathcal{R}(\cdot)$ is a regularization term (e.g., squared L_2-norm).

Mixon et al. (2020) suggested to explore NC via an Unconstrained Features Model (UFM) — The features $\{\boldsymbol{h}_{k,i} = \boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x}_{k,i})\}$ are treated as free optimization variables:

$$\min_{\boldsymbol{W},\boldsymbol{b},\{\boldsymbol{h}_{k,i}\}} \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}\left(\boldsymbol{W}\boldsymbol{h}_{k,i} + \boldsymbol{b}, \boldsymbol{y}_k\right) + \mathcal{R}\left(\boldsymbol{W}, \boldsymbol{b}, \{\boldsymbol{h}_{k,i}\}\right).$$

The UFM rationale: Modern over-parameterized DNNs can adapt their feature mapping to almost any training data.

The Unconstrained Features Model

- Most (if not all) of the theoretical works on NC consider this plain UFM.
- [Mixon et al., 2020] showed that for MSE loss and *no regularization*, a simplex EFT is (only) **a** global minimizer (yet, experiments with randomly initialized GD convergence to non-collapse global minimizers)
- Other works, e.g., [Lu et al., 2020; Fang et al., 2021; Zhu et al., 2021], considered the UFM with L_2-norm regularized CE loss w/ or w/o the bias term. They showed that *any* global minimizer has simplex EFT structure.
- Our contributions include:
  - Closing the gap for the UFM with regularized MSE loss (showing some distinction from the CE case)
  - Extending the UFM with another level of features (another layer of weights and nonlinearity) to capture depthwise NC behavior

The Unconstrained Features Model

- Most (if not all) of the theoretical works on NC consider this plain UFM.
- [Mixon et al., 2020] showed that for MSE loss and *no regularization*, a simplex EFT is (only) **a** global minimizer (yet, experiments with randomly initialized GD convergence to non-collapse global minimizers)
- Other works, e.g., [Lu et al., 2020; Fang et al., 2021; Zhu et al., 2021], considered the UFM with L_2-norm regularized CE loss w/ or w/o the bias term. They showed that *any* global minimizer has simplex EFT structure.
- Our contributions include:
  - Closing the gap for the UFM with regularized MSE loss (showing some distinction from the CE case)
  - Extending the UFM with another level of features (another layer of weights and nonlinearity) to capture depthwise NC behavior

The Unconstrained Features Model

- Most (if not all) of the theoretical works on NC consider this plain UFM.
- [Mixon et al., 2020] showed that for MSE loss and *no regularization*, a simplex EFT is (only) **a** global minimizer (yet, experiments with randomly initialized GD convergence to non-collapse global minimizers)
- Other works, e.g., [Lu et al., 2020; Fang et al., 2021; Zhu et al., 2021], considered the UFM with L_2-norm regularized CE loss w/ or w/o the bias term. They showed that *any* global minimizer has simplex EFT structure.
- Our contributions include:
  - Closing the gap for the UFM with regularized MSE loss (showing some distinction from the CE case)
  - Extending the UFM with another level of features (another layer of weights and nonlinearity) to capture depthwise NC behavior

The Unconstrained Features Model

- Most (if not all) of the theoretical works on NC consider this plain UFM.
- [Mixon et al., 2020] showed that for MSE loss and *no regularization*, a simplex EFT is (only) **a** global minimizer (yet, experiments with randomly initialized GD convergence to non-collapse global minimizers)
- Other works, e.g., [Lu et al., 2020; Fang et al., 2021; Zhu et al., 2021], considered the UFM with L_2-norm regularized CE loss w/ or w/o the bias term. They showed that *any* global minimizer has simplex EFT structure.
- Our contributions include:
  - Closing the gap for the UFM with regularized MSE loss (showing some distinction from the CE case)
  - Extending the UFM with another level of features (another layer of weights and nonlinearity) to capture depthwise NC behavior

UFM with Regularized MSE Loss

Contribution: We analyze the minima of the UFM with regularized MSE loss and show the effect of the bias term on the minimizers' structured collapse.

Theorem (The bias-free case – the factors are stated in the paper)

Let $d \geq K$ and define $c := K\sqrt{n\lambda_H\lambda_W}$. If $c \leq 1$, then any global minimizer $(\boldsymbol{W}^*, \boldsymbol{H}^*)$ of

$$\min_{\boldsymbol{W}\in\mathbb{R}^{K\times d},\, \boldsymbol{H}\in\mathbb{R}^{d\times Kn}} \frac{1}{2Kn}\|\boldsymbol{W}\boldsymbol{H}-\boldsymbol{Y}\|_F^2 + \frac{\lambda_W}{2}\|\boldsymbol{W}\|_F^2 + \frac{\lambda_H}{2}\|\boldsymbol{H}\|_F^2$$

obeys that $\boldsymbol{H}^* = \overline{\boldsymbol{H}} \otimes 1_n^\top$ for some $\overline{\boldsymbol{H}} := [\boldsymbol{h}_1^*, \ldots, \boldsymbol{h}_K^*] \in \mathbb{R}^{d\times K}$, $\boldsymbol{W}^{*\top} \propto \overline{\boldsymbol{H}}$, and

$$\boldsymbol{W}^*\overline{\boldsymbol{H}} \propto \overline{\boldsymbol{H}}^\top\overline{\boldsymbol{H}} \propto \boldsymbol{W}^*\boldsymbol{W}^{*\top} \propto \boldsymbol{I}_K.$$

If $c > 1$, then the minimizer is $(\boldsymbol{W}^*, \boldsymbol{H}^*) = (0, 0)$.

Denote the global mean $\boldsymbol{h}_G^* = \frac{1}{K}\overline{\boldsymbol{H}}1_K$, note that $\overline{\boldsymbol{H}}^\top\overline{\boldsymbol{H}} = \rho\boldsymbol{I}_K$ implies:

$$\left(\overline{\boldsymbol{H}} - \boldsymbol{h}_G^* 1_K^\top\right)^\top \left(\overline{\boldsymbol{H}} - \boldsymbol{h}_G^* 1_K^\top\right) = \rho\left(\boldsymbol{I}_K - \frac{1}{K}1_K 1_K^\top\right)$$

Extended Unconstrained Features Model

Contribution: We analyze the minima of a linear extended UFM and show its limitation in modeling (practical) depthwise NC behavior.

Theorem (Linear extended UFM)

Let $d > K$ and $(\boldsymbol{W}_2^*, \boldsymbol{W}_1^*, \boldsymbol{H}_1^*)$ be a global minimizer of

$$\min_{\boldsymbol{W}_2, \boldsymbol{W}_1, \boldsymbol{H}_1} \frac{1}{2Kn} \| \boldsymbol{W}_2 \boldsymbol{W}_1 \boldsymbol{H}_1 - \boldsymbol{Y} \|_F^2 + \frac{\lambda_{W_2}}{2} \| \boldsymbol{W}_2 \|_F^2 + \frac{\lambda_{W_1}}{2} \| \boldsymbol{W}_1 \|_F^2 + \frac{\lambda_{H_1}}{2} \| \boldsymbol{H}_1 \|_F^2 .$$

We have that $\boldsymbol{H}_1^* = \overline{\boldsymbol{H}}_1 \otimes 1_n^\top$ for some $\overline{\boldsymbol{H}}_1 \in \mathbb{R}^{d \times K}$, and

$$(\boldsymbol{W}_2^* \boldsymbol{W}_1^*) \overline{\boldsymbol{H}}_1 \propto \overline{\boldsymbol{H}}_1^\top \overline{\boldsymbol{H}}_1 \propto (\boldsymbol{W}_2^* \boldsymbol{W}_1^*)(\boldsymbol{W}_2^* \boldsymbol{W}_1^*)^\top \propto \boldsymbol{I}_K .$$

Similarly, we have that $\boldsymbol{H}_2^* := \boldsymbol{W}_1^* \boldsymbol{H}_1^* = \overline{\boldsymbol{H}}_2 \otimes 1_n^\top$ for some $\overline{\boldsymbol{H}}_2 \in \mathbb{R}^{d \times K}$, and

$$\boldsymbol{W}_2^* \overline{\boldsymbol{H}}_2 \propto \overline{\boldsymbol{H}}_2^\top \overline{\boldsymbol{H}}_2 \propto \boldsymbol{W}_2^* \boldsymbol{W}_2^{*\top} \propto \boldsymbol{I}_K .$$

Extended Unconstrained Features Model

Theorem (Linear extended UFM)

Let $d > K$ and $(\boldsymbol{W}_2^*, \boldsymbol{W}_1^*, \boldsymbol{H}_1^*)$ be a global minimizer of the linear extended model ...

We have that $\boldsymbol{H}_1^* = \overline{\boldsymbol{H}}_1 \otimes 1_n^\top$ for some $\overline{\boldsymbol{H}}_1 \in \mathbb{R}^{d \times K}$, and

$$(\boldsymbol{W}_2^* \boldsymbol{W}_1^*)\overline{\boldsymbol{H}}_1 \propto \overline{\boldsymbol{H}}_1^\top \overline{\boldsymbol{H}}_1 \propto (\boldsymbol{W}_2^* \boldsymbol{W}_1^*)(\boldsymbol{W}_2^* \boldsymbol{W}_1^*)^\top \propto \boldsymbol{I}_K.$$

Similarly, we have that $\boldsymbol{H}_2^* := \boldsymbol{W}_1^* \boldsymbol{H}_1^* = \overline{\boldsymbol{H}}_2 \otimes 1_n^\top$ for some $\overline{\boldsymbol{H}}_2 \in \mathbb{R}^{d \times K}$, and

$$\boldsymbol{W}_2^* \overline{\boldsymbol{H}}_2 \propto \overline{\boldsymbol{H}}_2^\top \overline{\boldsymbol{H}}_2 \propto \boldsymbol{W}_2^* \boldsymbol{W}_2^{*\top} \propto \boldsymbol{I}_K.$$

Limitations of this model:

- Empirically, structured collapse appears only in the deepest features, but: The theorem shows the emergence of structured (orthogonal) collapse *simultaneously* at the two levels of unconstrained features.
- Empirically, the decrease in within-class variability is depthwise gradual, but: The linear link between $\boldsymbol{H}_2$ and $\boldsymbol{H}_1$ implies (under certain conditions) that $NC_1(\boldsymbol{H}_2) \approx NC_1(\boldsymbol{H}_1)$ after random initialization and along gradient-based optimization.

Extended Unconstrained Features Model

Contribution: We analyze the minima of a ReLU-based nonlinear extended UFM and show the structured collapse of the deepest features.

Theorem (Nonlinear extended UFM)

Let $d > K$ and $(\boldsymbol{W}_2^*, \boldsymbol{W}_1^*, \boldsymbol{H}_1^*)$ be a global minimizer of

$$\min_{\boldsymbol{W}_2, \boldsymbol{W}_1, \boldsymbol{H}_1} \frac{1}{2Kn} \| \boldsymbol{W}_2 \sigma(\boldsymbol{W}_1 \boldsymbol{H}_1) - \boldsymbol{Y} \|_F^2 + \frac{\lambda_{W_2}}{2} \| \boldsymbol{W}_2 \|_F^2 + \frac{\lambda_{W_1}}{2} \| \boldsymbol{W}_1 \|_F^2 + \frac{\lambda_{H_1}}{2} \| \boldsymbol{H}_1 \|_F^2,$$

where $\sigma(\cdot) = \max(0, \cdot)$ is the element-wise ReLU function.
We have that $\boldsymbol{H}_2^* := \sigma(\boldsymbol{W}_1^* \boldsymbol{H}_1^*) = \overline{\boldsymbol{H}}_2 \otimes 1_n^\top$ for some non-negative $\overline{\boldsymbol{H}}_2 \in \mathbb{R}^{d \times K}$, and

$$\boldsymbol{W}_2^* \overline{\boldsymbol{H}}_2 \propto \overline{\boldsymbol{H}}_2^\top \overline{\boldsymbol{H}}_2 \propto \boldsymbol{W}_2^* \boldsymbol{W}_2^{*\top} \propto \boldsymbol{I}_K.$$

Our Numerical Results - Extended UFM

Setting: $K = 4, d = 20, n = 50$ and $\lambda_{W_2} = \lambda_{W_1} = \lambda_{H_1} = 0.005$ (no bias is used).
Plain gradient descent optimization with step-size 0.1.
Top: no ReLU (the features are: $\boldsymbol{W}_1\boldsymbol{H}_1$ and $\boldsymbol{H}_1$).
Bottom: with ReLU (the features are: $\sigma(\boldsymbol{W}_1\boldsymbol{H}_1)$ and $\boldsymbol{H}_1$).

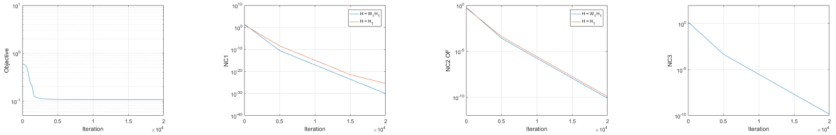


Figure 3. Verification of Theorem 4.1 (two levels of features). From left to right: the objective value, NC1 (within-class variability), NC2 (similarity of the features to OF), and NC3 (alignment between the weights and the features).

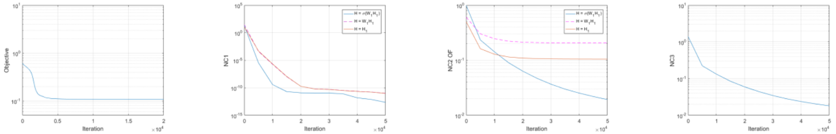


Figure 4. Verification of Theorem 4.2 (two levels of features with ReLU activation). From left to right: the objective value, NC1 (within-class variability), NC2 (similarity of the features to OF), and NC3 (alignment between the weights and the features).

Our Numerical Results - Extended UFM

Setting: $K = 5, d = 20, n = 100$, $\lambda_{W_2} = 0.005$, $\lambda_{W_1} = 0.0025$, and $\lambda_{H_1} = 0.001$ (no bias is used). Plain gradient descent optimization with step-size 0.1.
Top: no ReLU (the features are: $\boldsymbol{W}_1\boldsymbol{H}_1$ and $\boldsymbol{H}_1$).
Bottom: with ReLU (the features are: $\sigma(\boldsymbol{W}_1\boldsymbol{H}_1)$ and $\boldsymbol{H}_1$).

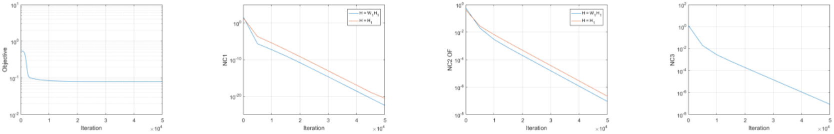


Figure 8. Verification of Theorem 4.1 (two levels of features). From left to right: the objective value, NC1 (within-class variability), NC2 (similarity of the features to OF), and NC3 (alignment between the weights and the features).

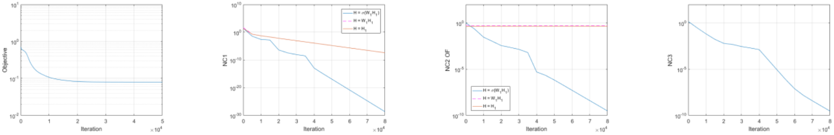


Figure 9. Verification of Theorem 4.2 (two levels of features with ReLU activation). From left to right: the objective value, NC1 (within-class variability), NC2 (similarity of the features to OF), and NC3 (alignment between the weights and the features).

Our Numerical Results - ResNet18 MNIST

Our experiments (gradual collapse across layers and structure only in final features)

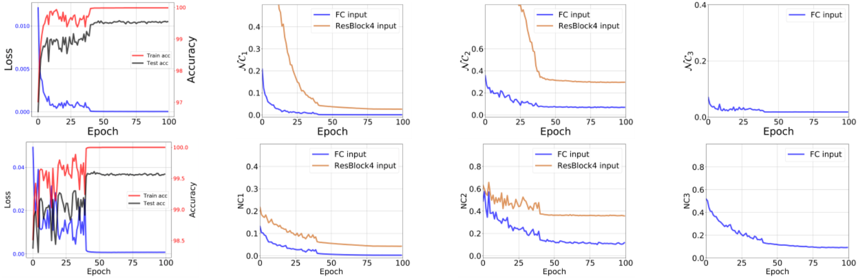


Figure 5. NC metrics for ResNet18 trained on MNIST. Top: MSE loss, weight decay, and no bias; Bottom: Cross-entropy loss and weight decay. From left to right: training's objective value and accuracy, NC1 (within-class variability), NC2 (similarity of the centered features to simplex ETF), and NC3 (alignment between the weights and the features).

Our Numerical Results - ResNet18 CIFAR10

Our experiments (gradual collapse across layers and structure only in final features)

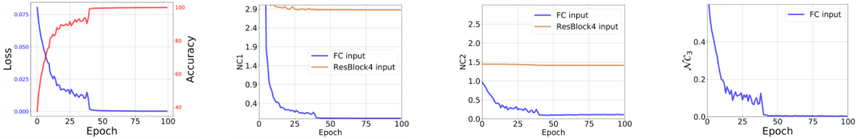


Figure 10. NC metrics for ResNet18 trained on CIFAR10 with MSE loss, weight decay, and no bias. From left to right: training's objective value and accuracy, NC1 (within-class variability), NC2 (similarity of the centered features to simplex ETF), and NC3 (alignment between the weights and the features).

Conclusion

- We characterized the (global) minimizers of the UFM for regularized MSE loss, showing some distinctions from the NC results that have been obtain for the cross-entropy loss in recent works.
- We mitigated the inability of the plain UFM to capture any NC behavior that happens across depth by adding another layer of weights as well as ReLU nonlinearity to the model and generalized our previous results.
- We empirically verified the theorems and demonstrated the usefulness of our nonlinear extended UFM in modeling the (depthwise) NC phenomenon that occurs in the training of practical networks.
- We believe that it may not be possible to show positive effects of NC on the generalization without departing from the plain UFM (Linear model on-top of features) towards the nonlinear extended UFM (shallow MLP on-top of features).

Thank You