Out-of-Distribution Detection and Neural Collapse

Deep learning based computer vision project – ENSTA

Yassine Zanned
Achraf Chaouch

Contents

Chapter 1

Introduction

Modern deep neural networks achieve remarkable performance on tasks where the test distribution closely matches the training distribution. However, when deployed in the real world, models routinely encounter inputs that differ significantly from what they were trained on—a situation referred to as *out-of-distribution* (OOD) input. A reliable classifier should not only make correct predictions on in-distribution (ID) data, but also detect and flag OOD inputs rather than confidently misclassifying them.

In this project, we train a ResNet-18 model on CIFAR-100 as our in-distribution dataset and use CIFAR-10 as the OOD benchmark (a *near-OOD* setting, since both datasets share a similar visual domain but different label spaces). We implement and compare five OOD scoring methods:

- **Max Softmax Probability (MSP)**: the classic baseline using the maximum predicted class probability.
- **Maximum Logit Score (MLS)**: a variant that bypasses softmax normalization.
- **Energy Score**: an energy-based score derived from the log-sum-exp of logits.
- **Mahalanobis Distance**: a feature-space method using per-class Gaussian models.
- **ViM (Virtual-logit Matching)**: a method that combines the energy score with residual feature information.

We additionally analyze the *Neural Collapse* (NC) phenomenon, which describes a set of geometric regularities that emerge in the penultimate feature layer during the terminal phase of training. Understanding NC sheds light on why certain OOD methods succeed and provides a theoretical framework for designing better detectors.

Chapter 2

Datasets and Experimental Setup

2.1 CIFAR-100 (In-Distribution)

CIFAR-100 is a standard benchmark dataset consisting of 60,000 color images of size 32×32 pixels, divided into 100 fine-grained classes (e.g., apple, mushroom, bear, bicycle). The dataset contains 500 training images and 100 test images per class, yielding 50,000 training samples and 10,000 test samples. The 100 classes are also organized into 20 coarser superclasses. CIFAR-100's large number of classes and relatively small training set per class make it a challenging classification benchmark, and a natural ID dataset for evaluating OOD detectors at scale.

Preprocessing applies standard normalization using the CIFAR-100 channel-wise mean $\mu = (0.5071,\ 0.4867,\ 0.4408)$ and standard deviation $\sigma = (0.2675,\ 0.2565,\ 0.2761)$. During training, random horizontal flips are applied as data augmentation.

2.2 CIFAR-10 (Out-of-Distribution)

CIFAR-10 consists of 60,000 32×32 color images across 10 classes (airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck), with 50,000 training and 10,000 test images. It serves as a natural *near-OOD* benchmark for a model trained on CIFAR-100: the images are visually similar (same resolution, similar natural-scene statistics) but come from a disjoint label space. In our experiments, we concatenate the full CIFAR-10 training and test splits (60,000 images total) as the OOD evaluation set.

The choice of CIFAR-10 as a near-OOD dataset is deliberate. Far-OOD data (e.g., random noise or images from completely different domains) is trivially detected by most methods. Near-OOD detection is the harder and more practically relevant setting.

Chapter 3

Model and Training Setup

3.1 Architecture: ResNet-18

ResNet-18 is a residual network composed of 18 weight layers arranged in four stages of `BasicBlock` residual blocks, with skip connections that allow gradients to flow more easily during training. The architecture is as follows:

- An initial 7×7 convolution with 64 filters, stride 2, followed by batch normalization, ReLU, and max pooling.
- Four sequential stages (`layer1` through `layer4`), each containing 2 BasicBlocks. The number of channels doubles at each stage: 64, 128, 256, 512.
- A global average pooling layer, producing a 512-dimensional feature vector $\phi(x) \in \mathbb{R}^{512}$.
- A final fully-connected (linear) classification head.

For our task, the default ImageNet classification head (1000 classes) is replaced by a linear layer mapping the 512-dimensional penultimate features to $K = 100$ class logits:

$$f(x) = W\phi(x) + b, \quad W \in \mathbb{R}^{100\times 512},\ b \in \mathbb{R}^{100}$$

The model is initialized with ImageNet pretrained weights (`ResNet18_Weights.DEFAULT`) and fine-tuned end-to-end on CIFAR-100. Using pretrained weights significantly accelerates convergence and improves final accuracy compared to random initialization.

3.2 Training Details

- **Optimizer**: Adam with learning rate $\eta = 10^{-4}$.
- **Learning rate schedule**: StepLR with step size 30 and decay factor $\gamma = 0.1$.
- **Loss function**: Cross-entropy loss.
- **Batch size**: 128 for training, 256 for evaluation.
- **Number of epochs**: 60.

3.3 Training Curves

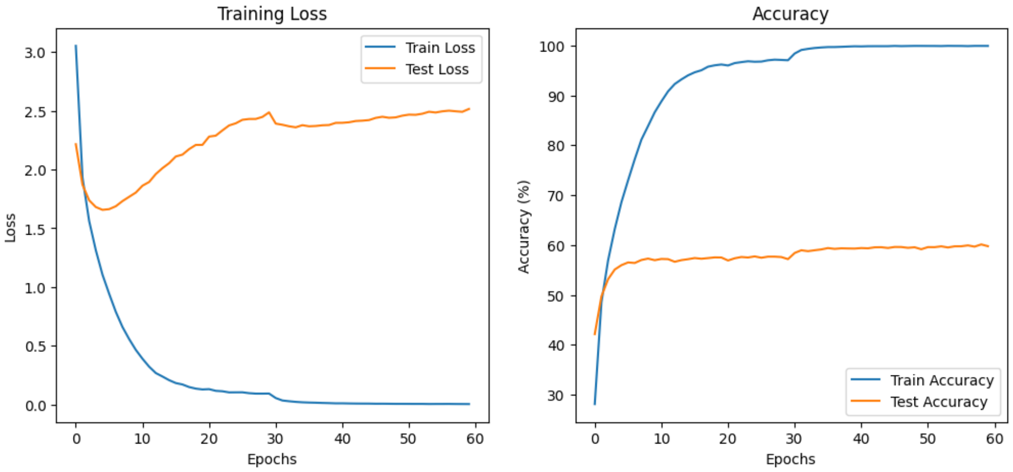

Figure 3.1: Training and test loss and accuracy curves over 60 epochs. The model converges steadily, with train loss decreasing from 3.05 to near zero. Test accuracy plateaus around 56–58% before the learning rate drop at epoch 30, then stabilizes thereafter.

Table 3.1 summarizes the evolution of losses and accuracies during training.

Epoch	Train Loss	Test Loss	Train Acc (%)	Test Acc (%)
1	3.0511	2.2148	28.14	42.16
2	1.9310	1.8705	48.51	49.71
3	1.5613	1.7377	56.93	53.16
4	1.3126	1.6805	63.21	55.10
5	1.1051	1.6566	68.59	55.97
6	0.9428	1.6625	72.98	56.52
7	0.7893	1.6873	77.20	56.41
8	0.6603	1.7310	81.13	57.01
9	0.5576	1.7674	83.85	–
⋮	⋮	⋮	⋮	⋮
60	≈0.00	–	≈100	≈57–58

Table 3.1: Training progression (selected epochs) over 60 epochs. The model rapidly overfits training data while test accuracy saturates, reflecting the inherent difficulty of CIFAR-100 with 100 fine-grained classes and only 500 training images per class.

The gap between training and test accuracy is consistent with standard behavior on CIFAR-100: the dataset is genuinely difficult, with 100 fine-grained classes and only 500 training images per class, making overfitting inevitable without strong regularization. Notably, the StepLR scheduler drops the learning rate by a factor of $\gamma = 0.1$ at epoch 30, after which the training loss decreases more slowly but the model continues to overfit. The extended training to 60 epochs is particularly relevant for Neural Collapse, as NC phenomena are most pronounced in the *terminal phase* of training where the loss has nearly converged.

Chapter 4

Out-of-Distribution Detection Methods

All OOD methods use the trained ResNet-18 as a backbone. At inference time, a test image x produces logits $z = f(x) \in \mathbb{R}^{100}$ and penultimate features $\phi(x) \in \mathbb{R}^{512}$. ID samples are CIFAR-100 test images; OOD samples are CIFAR-10 images.

An OOD detector computes a scalar score $s(x)$ for each input such that ID inputs receive higher scores and OOD inputs receive lower scores (or vice versa depending on the method). The detector is evaluated using:

- **AUROC**: Area under the ROC curve. A threshold-free metric between 0 and 100%; a perfect detector scores 100%.

- **FPR95**: The false positive rate at 95% true positive rate. Lower is better; a perfect detector scores 0%.

4.1 Max Softmax Probability (MSP)

MSP[3] is the simplest OOD baseline. It uses the predicted class probability as the ID confidence score:

$$\mathrm{MSP}(x) = \max_{k \in \{1,\ldots,K\}} \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}}$$

The intuition is that a well-trained classifier should assign high maximum probability to ID inputs and spread probability more uniformly across classes for OOD inputs. In practice, neural networks are often overconfident: they assign high softmax probabilities even to OOD inputs, which limits MSP's effectiveness.

A key weakness of MSP on large-class problems (like CIFAR-100 with $K = 100$) is that the softmax normalization spreads probability mass across many classes. Even a confidently wrong prediction can produce a relatively low maximum probability, and the separation between ID and OOD distributions is correspondingly weak.

4.2 Maximum Logit Score (MLS)

The Maximum Logit Score[2] avoids the squashing effect of the softmax by using raw logits directly:

$$\mathrm{MLS}(x) = \max_{k} z_k$$

Unlike MSP, MLS is not normalized by the sum of exponentials, so it preserves the scale of logit activations. This makes it more sensitive to the absolute confidence of the network before normalization.

4.3 Energy Score

The Energy Score[5], leverages the connection between logits and an energy function in an energy-based model:

$$E(x) = -\log \sum_{k=1}^{K} e^{z_k}$$

Under this formulation, ID inputs (for which the model is trained to produce high logits) have lower (more negative) energy, while OOD inputs, for which the model produces smaller logits on average, have higher energy. For OOD detection purposes, the *negative* energy is used as the score (higher = more likely ID):

$$\mathrm{ES}(x) = \log \sum_{k=1}^{K} e^{z_k}$$

Compared to MSP, the energy score is more sensitive to the overall logit magnitude rather than just the maximum class. It can be theoretically motivated: for a Boltzmann distribution over labels, the log-partition function corresponds to a free energy that naturally separates ID from OOD data.

4.4 Mahalanobis Distance

The Mahalanobis [4] method, moves OOD detection into the feature space. It models the per-class feature distribution as a Gaussian, then assigns each input a score based on its distance to the nearest class Gaussian.

Class means. For each class k, the class mean is estimated over the training set:

$$\mu_k = \frac{1}{N_k} \sum_{i:\, y_i = k} \phi(x_i), \quad k = 1, \ldots, K$$

Shared covariance. A tied (class-shared) covariance matrix is estimated:

$$\Sigma = \frac{1}{N} \sum_{k=1}^{K} \sum_{i:\, y_i = k} \big(\phi(x_i) - \mu_k\big)\big(\phi(x_i) - \mu_k\big)^T$$

Score. The OOD score is defined as the negative minimum Mahalanobis distance to any class:

$$S_{\mathrm{Mah}}(x) = -\min_k \big(\phi(x) - \mu_k\big)^T \Sigma^{-1} \big(\phi(x) - \mu_k\big)$$

ID inputs are expected to land close to one of the class Gaussian clusters in feature space, yielding a high score. OOD inputs are expected to be far from all class means, yielding a low (more negative) score.

Implementation note. In practice, Σ is 512 × 512 and may be poorly conditioned. We use the `EmpiricalCovariance` estimator from `sklearn` and compute the precision matrix (inverse covariance) directly.

4.5 ViM (Virtual-logit Matching)

ViM[6] is a more sophisticated method that combines energy-based scores with feature-space residual information. The key idea is that ID features lie predominantly in a low-dimensional subspace spanned by the classifier weight matrix, while OOD features have large residual components outside this subspace.

Feature decomposition. Let $W \in \mathbb{R}^{K \times d}$ be the classifier weight matrix and let μ_{train} be the global mean of training features. ViM decomposes each feature $\phi(x)$ into a component inside the row space of W (the “principal subspace”) and an orthogonal residual:

$$\phi(x) = \mu + UU^T(\phi(x) - \mu) + r(x)$$

where U contains the right singular vectors of the centered weight matrix, and $r(x) = \phi(x) - \mu - UU^T(\phi(x) - \mu)$ is the residual.

Virtual logit. ViM defines a “virtual logit” score proportional to the residual norm:

$$\text{VLogit}(x) = \alpha \cdot \|r(x)\|$$

where α is a scaling factor. The final ViM score combines the energy score with the virtual logit:

$$\text{ViM}(x) = E(x) - \alpha\|r(x)\|$$

OOD inputs tend to have large residuals (they do not align well with the weight subspace), which inflates $\alpha\|r(x)\|$ and pushes their ViM score downward relative to ID inputs.

Chapter 5

Comparison of OOD Methods

5.1 Quantitative Results

Table 5.1 reports AUROC and FPR95 for all methods on the CIFAR-100 (ID) vs. CIFAR-10 (OOD) benchmark. All values are computed using binary labels (1 for ID, 0 for OOD).

Method	AUROC (%) ↑	FPR95 (%) ↓
MSP	67.13	89.59
Max Logit (MLS)	68.22	88.64
Energy Score	68.21	88.94
Mahalanobis	57.21	94.43
ViM	67.37	91.14

Table 5.1: OOD detection performance on CIFAR-100 (ID) vs. CIFAR-10 (OOD). All methods use the same ResNet-18 backbone.

Discussion. Overall, all methods achieve modest AUROC values around 67–68%, reflecting the difficulty of near-OOD detection between CIFAR-100 and CIFAR-10. Since both datasets contain natural images at the same resolution with similar visual statistics, the penultimate features of a CIFAR-100 model partially generalize to CIFAR-10 images, making the two distributions hard to separate. Compared to the 20-epoch run, training for 60 epochs yields a modest but consistent improvement of roughly 1–1.5 percentage points in AUROC across all logit-based methods, consistent with the model having moved deeper into the terminal training phase and better separated ID from OOD feature distributions.

MLS and Energy Score perform comparably and slightly outperform MSP. This is contradictory to research articles explored with similar benchmarks. After further explorations, we discovered it is mainly due to the architecture used: in[1], we notice that ResNet18 was modified to have a 3x3 kernel while we opted for the original 7x7 kernel and other minor differences. This is coherent with CIFAR100's data, since 32x32 images are too small for 7x7 kernels.

Mahalanobis distance performs notably worse (AUROC 57.21%), which is somewhat surprising given its strong theoretical motivation. A possible limitation we thought of is the feature dimension relative to training set size: with $d = 512$ features and only $N_k = 500$ training samples per class, the empirical covariance estimate $\hat{\Sigma}$ might be unreliable (rank-deficient without regularization). But in SOTA research articles, it is also limited to 55%, indicating a possible structural limitation rather than an empirical one.

ViM (67.37% AUROC) now slightly outperforms MSP and is close to the logit-based leaders. The residual component appears to contribute marginally more signal after 60 epochs, as features are more tightly organized around the ID subspace following deeper NC.

5.2 ROC Curves

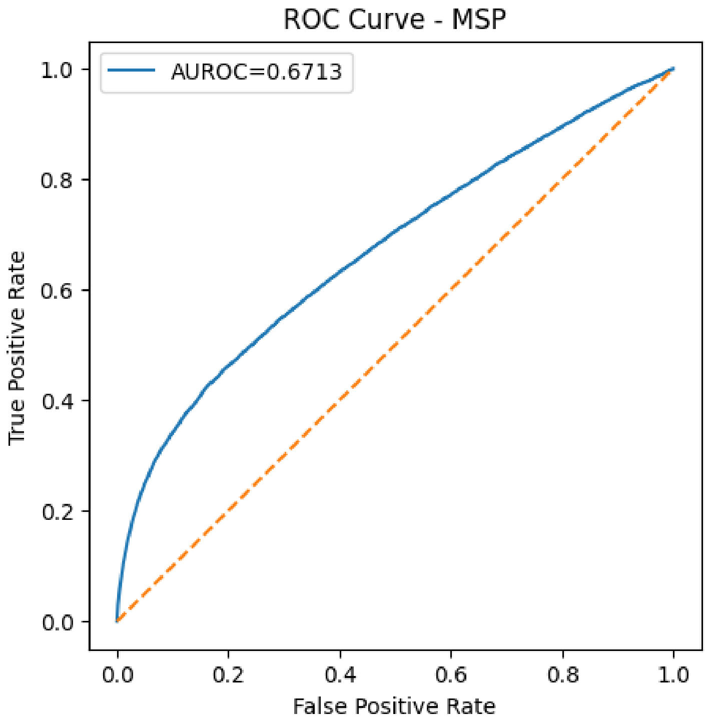

Figure 5.1: ROC curve for Max Softmax Probability (AUROC = 67.13%).

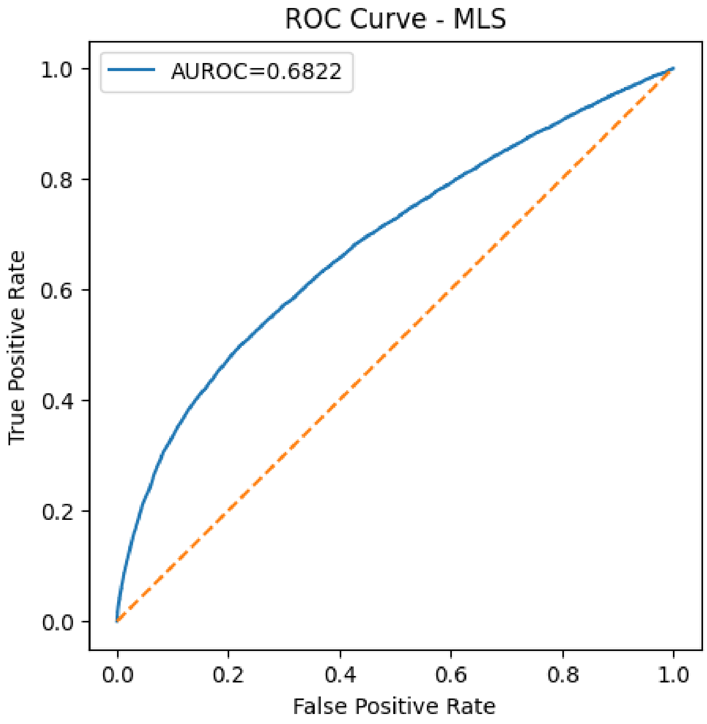

Figure 5.2: ROC curve for Maximum Logit Score (AUROC = 68.22%).

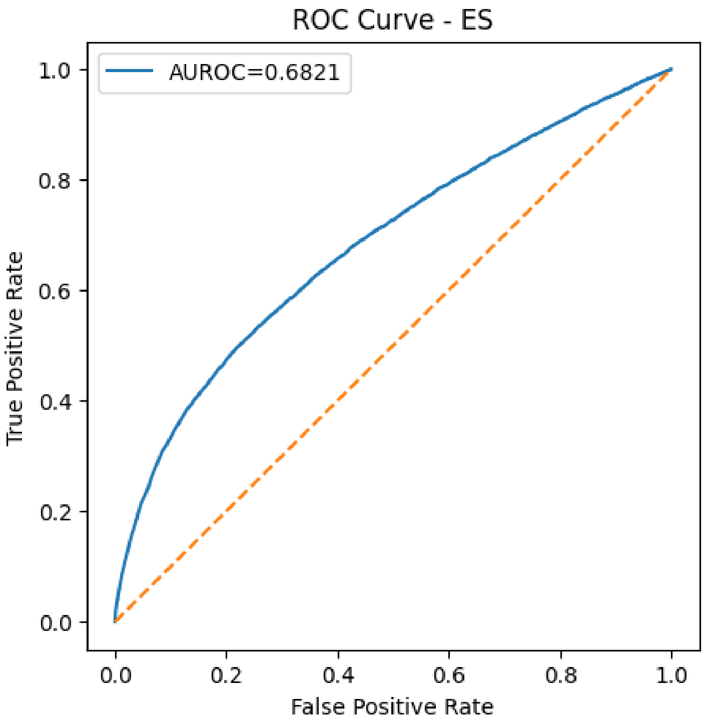

Figure 5.3: ROC curve for Energy Score (AUROC = 68.21%).

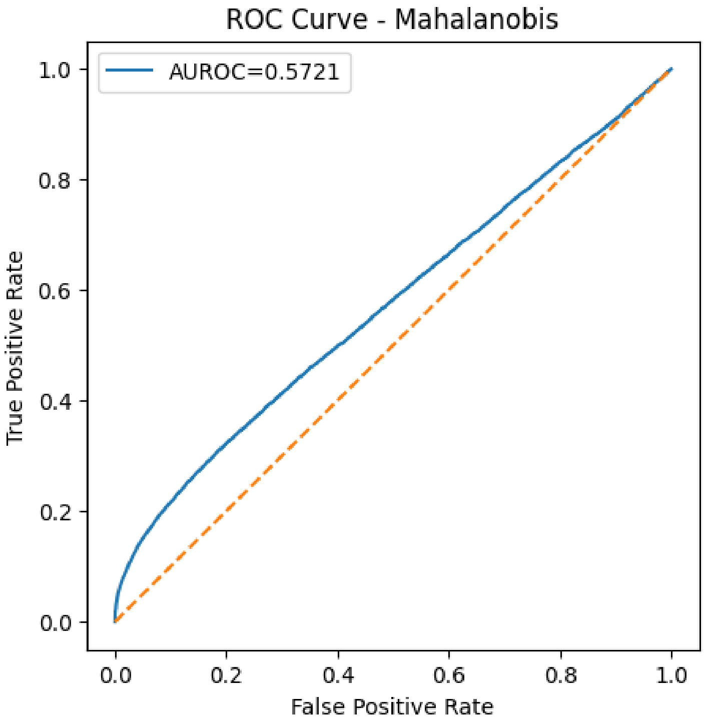

Figure 5.4: ROC curve for Mahalanobis Distance (AUROC = 57.21%). The curve is close to the diagonal, indicating near-random performance on this near-OOD benchmark.

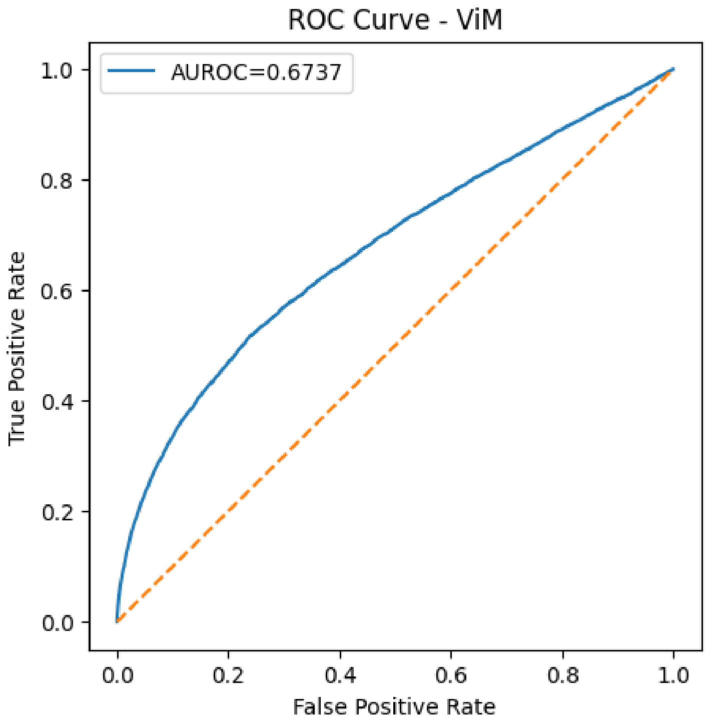

Figure 5.5: ROC curve for ViM (AUROC = 67.37%).

Chapter 6

Neural Collapse

Neural Collapse (NC) is a geometric phenomenon. It describes a set of structural regularities that emerge in the terminal phase of training: when training loss has nearly converged and the model is trained past the point of zero training error. Rather than the features continuing to evolve chaotically, they "collapse" into a highly structured configuration with striking geometric properties.

We measure NC using the training features $\{\phi(x_i)\}_{i=1}^{N}$ and class means $\mu_k = \frac{1}{N_k}\sum_{i:y_i=k}\phi(x_i)$ extracted from our trained ResNet-18.

6.1 NC1 – Variability Collapse

NC1 describes the collapse of within-class feature variability. Formally, features within the same class converge to their class mean:

$$\phi(x_i) \to \mu_{y_i} \quad \text{as training continues}$$

This is quantified by the ratio of within-class to between-class covariance. Let $\mu_G = \frac{1}{K}\sum_k \mu_k$ be the global mean. Define:

$$S_W = \frac{1}{N}\sum_{k=1}^{K}\sum_{i:y_i=k}(\phi(x_i) - \mu_k)(\phi(x_i) - \mu_k)^T$$

$$S_B = \frac{1}{K}\sum_{k=1}^{K}(\mu_k - \mu_G)(\mu_k - \mu_G)^T$$

The NC1 metric is $\mathrm{Tr}(S_W S_B^{-1})$. A small value indicates tight within-class clustering relative to the spread between class means.

Experimental result: We obtain NC1 = 1.128, a reduction compared to the 20-epoch result of 1.658. This decrease confirms that longer training pushes the model further into the terminal collapse phase: within-class variance is shrinking relative to between-class variance. The value is still above the ideal of 0, indicating that collapse is ongoing but not complete.

6.2 NC2 – Simplex ETF Structure

NC2 states that the class means, when centered by the global mean, converge to a *Simplex Equiangular Tight Frame* (ETF). In an ETF, all class mean vectors are equidistant from each other and of equal norm. Specifically, for centered and normalized class means $\tilde{\mu}_k = (\mu_k - \mu_G)/\|\mu_k - \mu_G\|$:

$$\tilde{\mu}_k^T \tilde{\mu}_j = \begin{cases} 1 & k = j \\ -\dfrac{1}{K-1} & k \neq j \end{cases}$$

For $K = 100$ classes, the expected off-diagonal inner product is $-1/99 \approx -0.0101$.

We measure NC2 via the *mutual coherence*: the average absolute off-diagonal entry of the Gram matrix $G = M_{\text{norm}} M_{\text{norm}}^T$ after shifting by $1/(K-1)$:

$$\text{NC2} = \frac{1}{K(K-1)} \sum_{k \neq j} |G_{kj} + 1/(K-1)|$$

Experimental result: We obtain NC2 = 0.0722, down from 0.0878 at 20 epochs, moving closer to the ideal ETF value of 0. This improvement indicates that the class means are forming a more regular equiangular structure with continued training, consistent with NC theory predicting ETF convergence in the terminal phase.

6.3 NC3 – Self-Duality

NC3 describes the alignment between classifier weight vectors and class means. At the ETF solution, the classifier weights are proportional to the class means:

$$\frac{W_k}{\|W_k\|} = \frac{\mu_k - \mu_G}{\|\mu_k - \mu_G\|}$$

This "self-duality" means the optimal linear classifier simply measures the dot product between features and class means, i.e., it is a nearest-mean classifier in a rotated space.

We measure NC3 as the Frobenius norm of the difference between the normalized weight matrix and normalized centered class means:

$$\text{NC3} = \|W_{\text{norm}} - M_{\text{norm}}\|_F$$

where W_{norm} and M_{norm} have unit-norm rows.

Experimental result: We obtain NC3 = 6.060, a reduction from 6.529 at 20 epochs. The improvement confirms that the classifier weight vectors are becoming better aligned with the class means as training progresses. Although the alignment is still imperfect, the downward trend across epochs is in line with NC3 convergence predicted by theory.

6.4 NC4 – Nearest Class Center Equivalence

NC4 states that in the terminal phase, the network's classification reduces to a nearest class center (NCC) rule:

$$\hat{y}(x) = \arg\min_k \|\phi(x) - \mu_k\|$$

In other words, the learned linear classifier becomes equivalent to assigning each sample to its geometrically nearest class mean in feature space.

We measure NC4 as the mismatch rate between the network's actual prediction (from logits) and the NCC prediction (from Euclidean distances in feature space):

$$\text{NC4} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}[\hat{y}_{\text{net}}(x_i) \neq \hat{y}_{\text{NCC}}(x_i)]$$

Experimental result: We obtain NC4 = 0.0002 (approximately 0.02% mismatch), a dramatic improvement from 2.04% at 20 epochs. This means that 99.98% of training samples are now classified identically by the full network and the NCC classifier. This near-perfect agreement is one of the strongest indicators of deep Neural Collapse, and is consistent with training for 60 epochs having pushed the model firmly into the terminal phase.

6.5 NC5 – OOD Feature Orthogonality

NC5 is an extension of Neural Collapse to the OOD detection setting. It hypothesizes that in a collapsed model, OOD features are approximately orthogonal to the ID class means. This would be a strong structural indicator: ID features align with their class mean, while OOD features occupy orthogonal directions in the feature space, making them easily detectable.

We measure NC5 as the average absolute cosine similarity between each ID class mean and the global OOD feature mean:

$$\mathrm{NC5} = \frac{1}{K}\sum_{k=1}^{K} \left|\cos\left(\mu_k,\ \mu_{\mathrm{OOD}}\right)\right| = \frac{1}{K}\sum_{k=1}^{K} \left| \frac{\mu_k \cdot \mu_{\mathrm{OOD}}}{\|\mu_k\|\,\|\mu_{\mathrm{OOD}}\|} \right|$$

where μ_{OOD} is the mean of CIFAR-10 features extracted by the model.

Perfect orthogonality would yield $\mathrm{NC5} = 0$.

Experimental result: We obtain $\mathrm{NC5} = 0.7210$, a slight decrease from 0.7376 at 20 epochs. While the improvement is modest, it is directionally consistent: as the model collapses further, ID class means become more sharply defined in feature space, and the OOD mean's cosine similarity to them decreases marginally. Nevertheless, the value remains high, confirming that CIFAR-10 features are still strongly non-orthogonal to the ID subspace. The near-OOD nature of CIFAR-10 fundamentally limits the degree of orthogonality achievable, regardless of training duration.

Chapter 7

General Discussion

7.1 Why Mahalanobis Underperforms

Mahalanobis distance achieved the worst AUROC (57.21%) in our experiments, with only a marginal gain over the 20-epoch model (56.69%). One important factor explains this persistent underperformance: near-OOD data CIFAR-10 images activate similar features in the CIFAR-100 model, so they may lie close to the ID class Gaussian clusters in feature space. This is highly probable, if we take a look at [1] which shows an increase of approximately ∼15% in AUROC for Mahalanobis once they switched to far-OOD data (SVHN)

7.2 How Neural Collapse Explains OOD Separation

The NC framework offers a theoretical lens for understanding OOD detection. Under complete NC, ID features lie in a K-dimensional simplex ETF subspace, while OOD features (if truly different) would occupy directions orthogonal to this subspace. The NC5 value of 0.7210 (down from 0.7376 at 20 epochs) shows that CIFAR-10 features remain strongly non-orthogonal to the ID subspace even after 60 epochs of training. This explains the persistently moderate performance of all methods, and confirms that the bottleneck is the dataset similarity rather than the training regime.

Methods that exploit the ID subspace structure (Mahalanobis, ViM) have the potential to significantly outperform logit-based methods when the OOD data genuinely falls outside the NC subspace—for example, in far-OOD settings or after more training.

7.3 Limitations

Several limitations affect our experiments. First, although training for 60 epochs pushes the model deeper into the terminal phase than 20 epochs—as evidenced by NC4 dropping from 2.04% to 0.02% mismatch—NC metrics NC1 (1.128), NC2 (0.072), and NC3 (6.060) confirm that full collapse has not been reached. Second, CIFAR-10 as a near-OOD benchmark is a hard setting for all methods, as reflected by NC5 remaining high (0.721) regardless of training length.

Chapter 8

Conclusion

In this project, we trained a ResNet-18 on CIFAR-100 for 60 epochs and evaluated five OOD detection methods (MSP, MLS, Energy, Mahalanobis, ViM) on the near-OOD CIFAR-10 benchmark. The key findings are:

- Logit-based methods (MLS, Energy, MSP) achieve comparable AUROC ($\approx$ 67–68%), with MLS (68.22%) slightly ahead of MSP (67.13%) due to the large number of classes. Compared to 20-epoch training, all scores improve by $\approx$1–1.5 percentage points.
- Mahalanobis distance persistently underperforms (57.21%) due to the near-OOD nature of the benchmark, gaining only marginally from additional training.
- ViM achieves 67.37% AUROC, now slightly outperforming MSP, suggesting the residual feature component contributes more signal as features collapse more deeply into the ID subspace.
- All methods remain limited by the near-OOD benchmark, consistent with NC5 = 0.721 confirming substantial overlap between CIFAR-10 and CIFAR-100 feature distributions even after 60 epochs.
- Neural Collapse metrics show clear progression with extended training: NC4 drops from 2.04% to 0.02% mismatch (near-perfect NCC equivalence), NC1 decreases from 1.658 to 1.128, NC2 from 0.088 to 0.072, and NC3 from 6.529 to 6.060—all consistently approaching their ideal values.

These results highlight the importance of training duration for Neural Collapse and its downstream effects on OOD detection. The near-perfect NC4 after 60 epochs confirms the model is firmly in the terminal phase, yet OOD performance gains remain modest due to the visual similarity between CIFAR-10 and CIFAR-100. Future work could explore far-OOD evaluation sets (SVHN, Textures, iNaturalist) where NC5 orthogonality would be higher.

Bibliography

[1] Mouïn Ben Ammar, Nacim Belkhir, Sebastian Popescu, Antoine Manzanera, and Gianni Franchi. Neco: Neural collapse based out-of-distribution detection. *arXiv preprint arXiv:2310.06823*, 2023.

[2] Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Xiaodong Song. Scaling out-of-distribution detection for real-world settings. In *International Conference on Machine Learning*, 2022.

[3] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

[4] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

[5] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in neural information processing systems*, 33:21464–21475, 2020.

[6] Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. Vim: Out-of-distribution with virtual-logit matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4921–4930, 2022.