Exploiting Local Flatness for Efficient Out-of-Distribution Detection

Seonghwan Park[1,2], Hyunji Jung[2], Dongyeop Lee[2], and Namhoon Lee[2]

[1] Korea Electronics Technology Institute (KETI)
[2] Pohang University of Science and Technology (POSTECH)
{seonghwan.park, namhoon.lee}@postech.ac.kr

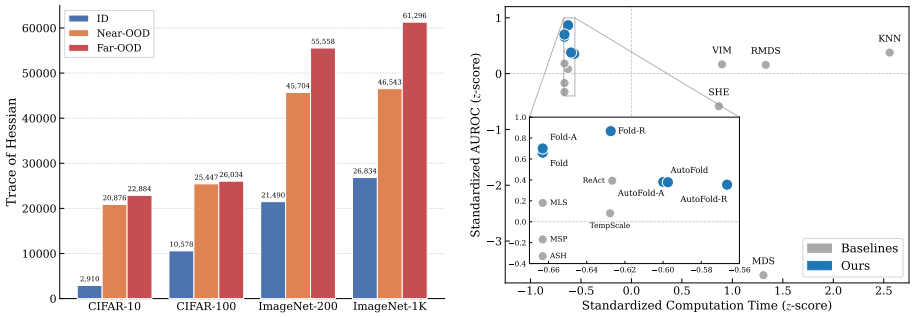
Abstract. Detecting out-of-distribution (OOD) data is crucial for reliable machine learning deployment. Among detection strategies, post-hoc methods are particularly attractive due to their efficiency, as they operate directly on pre-trained networks without requiring retraining. Within this paradigm, one promising direction exploits loss-landscape curvature to estimate model uncertainty; however, such methods incur substantial computational cost and rely on implicit assumptions about how landscape flatness differs between in-distribution (ID) and OOD data. In this work, we provide the first systematic investigation of this curvature discrepancy and show that OOD inputs exhibit larger Hessian curvature than ID data, with the gap widening under stronger distributional shifts. Motivated by these observations, we propose FOLD, a lightweight flatness-modulated OOD detector that leverages the feature Hessian and partial feature normalization to improve ID-OOD separability while avoiding costly parameter-space curvature approximations. To optimally adapt this normalization across diverse datasets, we further introduce AUTOFOLD, a self-supervised tuning scheme that synthesizes pseudo-OOD samples via ID logit masking for automatic calibration without requiring external data. Experiments on OOD benchmarks show that FOLD outperforms prior methods, improving the average AUROC by 1.63% and reducing FPR95 by 2.30%, while maintaining computational efficiency comparable to a standard forward pass. Supported by theoretical analysis and extensive ablations, FOLD provides a principled and practical solution for robust real-world deployment.

Keywords: OOD Detection · Loss Landscape Curvature · Self-Supervised Tuning

1 Introduction

Real-world machine learning systems inevitably encounter out-of-distribution (OOD) data, *i.e.*, samples drawn from distributions whose label sets are disjoint from those seen during training. Since the model has no prior knowledge of these unseen classes, it should ideally abstain from prediction or express high uncertainty. However, modern deep neural networks often produce erroneous predictions on such inputs with unwarranted overconfidence [19, 38, 42, 57]. This vulnerability makes reliable OOD identification a critical challenge and has motivated extensive research on detection methods.

Among these efforts, *post-hoc* detection methods have emerged as an appealing paradigm due to their computational efficiency and practical versatility. Rather than requiring model retraining to suppress overconfidence, these approaches extract discriminative signals directly from pre-trained networks [11, 13, 20, 31, 62]. Driven by these advantages, a rich body of work has explored post-hoc OOD detection along multiple directions, including output-level uncertainty estimation [17, 21, 29, 32, 34, 35, 48], informative intermediate activations [14, 54], and neuron- or gradient-level statistics [6, 23, 36].



(a) Sharpness across distribution shifts **(b)** Performance-efficiency trade-off

Fig. 1: (a) The trace of Hessian, serving as a metric for loss landscape sharpness, progressively increases as the data shifts from ID to near-OOD and far-OOD. (b) The trade-off between detection performance (*i.e.*, standardized AUROC) and computational cost (*i.e.*, standardized computation time). The top-left corner represents the ideal region of high accuracy and low latency. Our proposed FOLD and its automatically tuned variant, AUTOFOLD, consistently occupy this optimal region, demonstrating outstanding efficiency and performance compared to the baselines.

Within this line of work, recent studies on post-hoc uncertainty estimation exploit the curvature of a pre-trained model's loss landscape near its optimized parameters [37, 49]. Through the Laplace approximation, second-order parameter sensitivity is used to approximate the otherwise intractable Bayesian posterior, enabling a principled estimate of epistemic uncertainty for a fixed model. However, computing the exact curvature matrix scales quadratically with the number of parameters, making it infeasible for modern deep networks. Although several scalable approximations have been proposed [26, 43, 52], they remain computationally demanding and difficult to deploy in real-time settings. Furthermore, these approaches implicitly rely on the *assumption* that the loss landscape induced by in-distribution (ID) samples is flatter than that of OOD data.

In this work, we provide the first systematic investigation of this ID-OOD curvature discrepancy, culminating in a simple, computationally efficient method that exploits these structural properties.

Specifically, our empirical analysis reveals that key Hessian statistics (*e.g.*, the principal eigenvalue and the trace), are consistently smaller for ID data than for OOD samples (see Figure 1a). Furthermore, this curvature discrepancy grows with the severity of the distributional shift, becoming increasingly pronounced from semantically near- to far-OOD regimes. Motivated by these observations, we introduce FOLD, a lightweight Flatness-modulated OOD detection algorithm that leverages the *feature Hessian* for OOD discrimination while avoiding the prohibitive computational overhead typical of Bayesian curvature approximations. Our approach incorporates *partial feature normalization* as a simple structural modification that significantly amplifies the curvature gap between ID and OOD representations. To further mitigate hyperparameter sensitivity without relying on external data, we introduce AutoFold, a self-supervised tuning strategy that generates pseudo-OOD samples via logit masking using only ID data to enable automatic and robust hyperparameter selection. Extensive experiments demonstrate that Fold achieves highly competitive detection performance while maintaining the computational efficiency required for practical real-world deployment (see Figure 1b).

Our key contributions are summarized as follows:

- **Systematic curvature analysis** We present the first systematic analysis of loss landscape curvature, showing Hessian statistics are smaller for ID than for OOD data.
- **Efficient detection method** We propose FOLD, a lightweight OOD detection algorithm leveraging the *feature Hessian* and *partial feature normalization* to enhance ID–OOD separability without the overhead of Bayesian curvature-based methods.
- **Automatic hyperparameter tuning** We introduce AUTOFOLD, a self-supervised tuning strategy that synthesizes pseudo-OOD samples from ID data via logit masking, enabling automatic and reliable hyperparameter selection without external data.
- **Theoretical and empirical validation** We provide theoretical insights and extensive experiments demonstrating that FOLD achieves strong detection performance with substantial computational efficiency across diverse architectures.

2 Observations

In this section, we present a systematic empirical analysis of loss landscape curvature induced by ID and OOD samples. We first describe our experimental protocol based on scalable Hessian estimation for computing key curvature metrics, and then examine the geometric discrepancy between the two distributions at both the dataset and sample levels, establishing the empirical foundation for our approach.

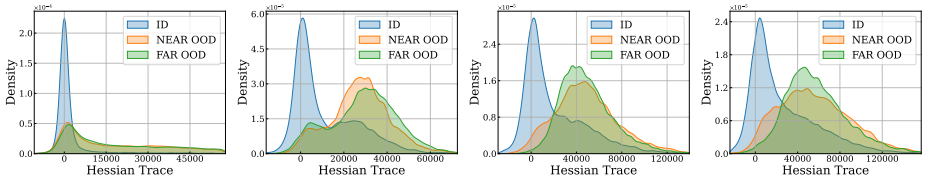
2.1 Experimental Setup

Curvature analysis protocol. We employ two metrics to characterize the curvature of the loss landscape, where H is the Hessian of the loss: the largest eigenvalue $\lambda_{\max}(H)$, capturing worst-case curvature, and the trace $\mathrm{tr}(H)$, reflecting average curvature [67]. Both quantities are estimated using Hutchinson's method, which relies on Hessian–vector products to avoid explicit construction of the full Hessian matrix [2, 3, 24]. We analyze curvature from two complementary perspectives. First, to examine our hypothesis that flatness differs across distributions, we compute Hessian statistics of the average loss over the entire dataset, which we refer to as the *dataset-level curvature*. Second, since practical OOD detection requires identifying distribution shifts per sample, we investigate whether the *sample-level curvature* provides a reliable signal for this purpose.

Experimental details. We adopt the OpenOOD framework [66, 70], which categorizes evaluation datasets into near- and far-OOD according to dataset-specific criteria. For CIFAR benchmarks [27], this distinction is defined strictly by image content and semantic similarity: near-OOD datasets [27, 30] share object-level similarities with the ID data, whereas far-OOD datasets [7, 10, 41, 72] contain numerical digits, texture patterns, or scene imagery that differ substantially in both semantics and low-level statistics [1]. In contrast, the large visual diversity of ImageNet [9] classes makes such semantic categorization less clear; thus the near-OOD [5, 61] and far-OOD [7, 28, 60] splits are determined empirically based on detection difficulty. For all experiments, we use a pretrained ResNet-50 for ImageNet-1K and ResNet-18 for CIFAR-10, CIFAR-100, and ImageNet-200; results are averaged over three runs except for ImageNet-1K, where we use the single provided pretrained checkpoint [70]. Additional implementation details and dataset specifications are provided in Appendix A.

Table 1: Comparison of the largest eigenvalue and the Hessian trace across distribution regimes (*e.g.*, ID, near-OOD, far-OOD). The reported values are averaged over datasets within each regime. ID samples exhibit lower curvature, as reflected by smaller largest eigenvalues and Hessian traces, whereas both metrics increase progressively from ID to near-OOD and further to far-OOD. This trend indicates heightened model sensitivity under increasing degrees of distributional shift.

	CIFAR-10		CIFAR-100		IMAGENET-200		IMAGENET-1K	
DATASET	$\lambda_{\max}(H)$	$\mathrm{tr}(H)$	$\lambda_{\max}(H)$	$\mathrm{tr}(H)$	$\lambda_{\max}(H)$	$\mathrm{tr}(H)$	$\lambda_{\max}(H)$	$\mathrm{tr}(H)$
ID	202.8±23.0	2910.5±717.5	316.9±3.4	10577.8±751.7	548.7±40.2	21489.6±2558.6	702.4±0.0	26834.2±0.0
NEAR-OOD	1228.1±65.2	20875.7±917.9	681.7±6.7	25447.0±704.3	941.6±24.3	45703.7±2327.0	1170.5±0.0	46543.2±0.0
FAR-OOD	3417.3±443.2	22883.5±1032.5	2347.6±34.3	26033.5±709.3	1147.9±11.7	55558.4±1435.6	1971.9±0.0	61296.1±0.0



(a) CIFAR-10 **(b)** CIFAR-100 **(c)** ImageNet-200 **(d)** ImageNet-1K

Fig. 2: Distribution of per-sample Hessian trace for ID and OOD data, encompassing both near- and far-OOD settings. For models trained on CIFAR, we use TIN [30] as the near-OOD dataset and SVHN [41] as the far-OOD dataset, while for models trained on ImageNet, we use NINCO [5] and iNaturalist [60] as the near- and far-OOD datasets, respectively. Across datasets, ID samples are concentrated at small values, whereas OOD samples shift toward larger values, exhibit broader distributions, and often display multimodality, reflecting separability under distribution shift.

2.2 Results

Dataset-level curvature. We begin by comparing dataset-level curvature between ID and OOD inputs. As shown in Table 1, ID samples consistently exhibit lower curvature than OOD samples across all datasets, with the gap generally widening as the distributional shift becomes more severe. For instance, when trained on CIFAR-100, the largest eigenvalue for near- and far-OOD inputs exceeds that of ID inputs by factors of 2.2 and 7.4, respectively. For CIFAR-10, the differences are even larger, with corresponding factors of 6.1 and 16.9. A similar pattern is observed for the Hessian trace: in ImageNet-200 and ImageNet-1K, the trace for near-OOD inputs is 2.1 and 1.7 times higher than that of ID data, respectively. Overall, these results indicate that curvature increases with stronger distributional shifts, reflecting heightened sensitivity of the model parameters.

Sample-level curvature. Next, we examine curvature at the sample level. Figure 2 shows the distributions of per-sample Hessian trace values. Across datasets, ID samples are clearly separated from OOD samples. For CIFAR-10, the ID density is concentrated near zero, whereas OOD samples shift toward larger values, with far-OOD exhibiting the heaviest tail. On CIFAR-100, the two OOD distributions exhibit pronounced bimodality, reflecting distinct curvature regimes under increasing distributional shift. A similar pattern is observed for ImageNet-trained models, where ID samples consistently produce smaller Hessian traces with narrower dispersion than OOD samples.

Overall, the sample-level analysis corroborates the dataset-level observations: curvature increases and becomes more dispersed as the distributional mismatch grows. This consistency across scales suggests that curvature is a reliable indicator of OOD behavior.

3 Proposed Method

In this section, we propose FOLD, a flatness-modulated OOD detector motivated by the observation that ID samples lie in flatter loss regions than OOD samples. We derive a feature Hessian by transporting logit-space curvature through the classifier, enabling flatness estimation in the feature space (Section 3.1). We then apply partial feature normalization and use the trace of the normalized feature Hessian as the detection score (Section 3.2). Finally, we introduce AUTOFOLD, a self-supervised scheme that selects the normalization exponent via ID logit masking without requiring OOD data (Section 3.3).

3.1 Efficient Geometric Curvature via the Feature Hessian

We consider a deep neural network $f(\cdot;\boldsymbol{\theta})$ decomposed into a feature encoder $h(\cdot;\boldsymbol{\theta}_h)$ and a linear classifier $g(\cdot;\boldsymbol{\theta}_g)$. Given an input $\mathbf{x}$, the network produces a latent representation $\mathbf{h} = h(\mathbf{x}) \in \mathbb{R}^d$ and corresponding logits $\mathbf{z} = g(\mathbf{h}) \in \mathbb{R}^C$, where C denotes the number of classes. Following prior works [54, 55], we adopt the energy function [35]

$$\mathcal{L}(\mathbf{z}) = \log\left(\sum_{i=1}^{C}\exp(z_i)\right). \tag{1}$$

As discussed in Section 2, curvature contains rich information distinguishing ID and OOD samples. However, computing the full parameter Hessian remains computationally expensive for modern deep networks, even with efficient approximations. To address this limitation, we draw on recent insights into neural network dynamics. In particular, Lee *et al*. [33] show that the logit-space Hessian $\nabla^2_{\mathbf{z}}\mathcal{L}$ exhibits strong spectral correspondence with the full Hessian, providing a tractable surrogate for curvature analysis.

Nevertheless, curvature measured purely in logit space ignores the geometric structure induced by the classifier. Since the linear mapping g determines class decision boundaries in the learned representation space, its Jacobian encodes the discriminative geometry of ID features, *i.e*., a structure known to be critical for OOD detection [63]. To jointly capture predictive uncertainty and feature-space geometry, we therefore transport the logit curvature back to the latent space through the classifier mapping.

Formally, applying the second-order chain rule to $\mathcal{L}(\mathbf{z})$ yields

$$\nabla^2_{\mathbf{h}}\mathcal{L} = (\nabla_{\mathbf{h}} g)\left(\nabla^2_{\mathbf{z}}\mathcal{L}\right)(\nabla_{\mathbf{h}} g)^\top + \sum_{i=1}^{C}\frac{\partial\mathcal{L}}{\partial z_i}\nabla^2_{\mathbf{h}} g_i. \tag{2}$$

Because g is linear, $\nabla^2_{\mathbf{h}} g_i = 0$ for all i, and the second term vanishes, yielding

$$\nabla^2_{\mathbf{h}}\mathcal{L} = (\nabla_{\mathbf{h}} g)\left(\nabla^2_{\mathbf{z}}\mathcal{L}\right)(\nabla_{\mathbf{h}} g)^\top. \tag{3}$$

Here, $\nabla_{\mathbf{h}} g \in \mathbb{R}^{C\times d}$ corresponds to the classifier weight matrix, characterizing the sensitivity of logits to perturbations in feature space.

Equation (3) indicates that the feature Hessian functions as a geometry-aware curvature operator, projecting predictive uncertainty from logit space onto the classifier-induced feature manifold. Consequently, the resulting curvature reflects directional sharpness aligned with class-discriminative subspaces, capturing geometric structures that are otherwise obscured in the purely logit-space Hessian.

3.2 Calibrated Geometric Curvature via Partial Normalization

One limitation of curvature evaluation in feature space is its dependence on the representation norm $\|\boldsymbol{h}\|$. As the feature magnitude increases, the logits scale proportionally, pushing the softmax distribution toward saturation. This effect flattens the local loss landscape and causes the feature Hessian to vanish (*i.e.*, $\nabla^2_{\boldsymbol{h}}\mathcal{L} \to \mathbf{0}$), potentially producing a degenerate curvature estimate that fails to reflect meaningful predictive uncertainty.

To mitigate this issue, we introduce partial feature normalization, transforming $\boldsymbol{h}$ into a scale-adjusted representation $\widetilde{\boldsymbol{h}}$:

$$\widetilde{\boldsymbol{h}} = \frac{\boldsymbol{h}}{\|\boldsymbol{h}\|^{\alpha}}, \qquad 0 < \alpha \leq 1, \tag{4}$$

where the hyperparameter α controls the degree of magnitude suppression and implicitly induces a sample-dependent temperature scaling of the logits. By balancing directional information with feature magnitude in this manner, the final FOLD score is defined as

$$S_{\text{FOLD}}(\boldsymbol{x}) = \operatorname{tr}\left(\nabla^2_{\widetilde{\boldsymbol{h}}}\mathcal{L}\right), \tag{5}$$

which effectively measures curvature in the normalized feature space while mitigating magnitude-induced degeneracy.

Empirically, the optimal α varies with dataset complexity. Simple datasets (*e.g.*, CIFAR) benefit from near-full normalization ($\alpha \approx 1$), whereas more diverse datasets (*e.g.*, ImageNet) favor milder suppression ($\alpha \approx 0.2$) to preserve magnitude cues (see Section 4.3). This motivates the automated calibration strategy introduced in the next section.

3.3 Self-Supervised Parameter Calibration via ID Logit Masking

As discussed in the previous section, the optimal α depends on the dataset's intrinsic feature scale. Conventional OOD detection methods often tune hyperparameters using an auxiliary OOD validation set, assuming prior knowledge of the OOD distribution and limiting realistic deployment [66, 70]. To avoid this dependency, we propose AUTOFOLD, a self-supervised framework that selects α using only a small ID validation set.

AUTOFOLD generates pseudo-OOD signals via ID logit masking by suppressing the k-th logit of a sample $\boldsymbol{x}$ with label $y = k$, simulating an unknown-class scenario:

$$\widetilde{z}_j = \begin{cases} z_j & \text{if } j \neq k, \\ -\infty & \text{if } j = k. \end{cases} \tag{6}$$

Removing the dominant class evidence forces the model to rely on residual logits, inducing predictive ambiguity similar to that observed in OOD inputs.

Unlike leave-one-class-out training [50, 62], which requires K retraining cycles for K classes, AUTOFOLD operates purely at inference time with negligible overhead. The optimal parameter α^* is obtained by maximizing the separability between the scores of ID validation samples and their masked counterparts:

$$\alpha^* = \arg\max_{\alpha} \frac{1}{K} \sum_{k=1}^{K} \text{AUROC}\left(\mathcal{S}^{(k)}_{ID}, \mathcal{S}^{(k)}_{pseudo}\right), \tag{7}$$

where $\mathcal{S}^{(k)}_{ID}$ and $\mathcal{S}^{(k)}_{pseudo}$ denote the FOLD score sets for the validation samples of class k and their logit-masked versions, respectively. This procedure calibrates α in a strictly in-distribution manner, providing a practical and deployment-friendly tuning criterion.

4 Evaluations

4.1 Experimental Setup

Baselines and implementation details. To evaluate FOLD, we compare it with a comprehensive set of post-hoc OOD detection methods operating exclusively at inference time and typically assuming classifiers trained with the standard cross-entropy loss [4, 14, 17, 18, 19, 32, 34, 35, 48, 54, 56, 63, 71]. As FOLD is orthogonal to activation thresholding and feature reshaping techniques, we further combine it with ReAct [54] and ASH [14], denoted as FOLD-R and FOLD-A, respectively. Additional implementation details are provided in Appendix A. The code is available at https://github.com/shpark97/Fold.

Evaluation metrics. We report the area under the receiver operating characteristic curve (AUROC) and the false positive rate at 95% true positive rate (FPR95) as the primary evaluation metrics. All pretrained models are kept fixed during evaluation to preserve their original in-distribution classification performance.

4.2 Main Results

Standard OOD detection benchmarks. Table 2 reports the results on CIFAR and ImageNet benchmarks, comparing FOLD with multiple existing baselines. Across all evaluations, the proposed methods achieve the best average performance in terms of both AUROC and FPR95, demonstrating the effectiveness of incorporating curvature-based information for OOD detection. In particular, the standard FOLD attains an average AUROC of 87.20%, improving upon strong prior baselines including KNN (+0.97%) and ReAct (+0.92%). For FPR95, FOLD achieves a competitive score of 42.42%, matching VIM and improving upon KNN by 0.82%.

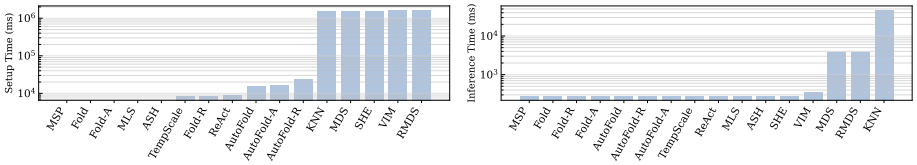
Performance further improves when FOLD is combined with feature shaping techniques. Specifically, FOLD-R achieves the best overall results, reaching an average AUROC of 87.91% and an FPR95 of 40.12%, outperforming prior methods by 1.63% and 2.30%, respectively. These gains arise because ReAct's activation thresholding stabilizes feature representations, sharpening curvature cues for more reliable uncertainty estimation. Additionally, FOLD-A performs particularly well on ImageNet benchmarks, achieving among the best results on both ImageNet-200 and ImageNet-1K.

Moreover, unlike prior approaches that often perform well on one benchmark but degrade on others, FOLD and its variants maintain consistently strong performance across both CIFAR and ImageNet benchmarks, indicating that curvature-based cues remain effective across varying dataset scales and distribution shifts.

Inference cost. To evaluate the efficiency of FOLD, we measure the setup and inference times of various baseline methods on ImageNet-1K. Here, setup time denotes the preparatory stage required before inference; for instance, KNN requires collecting feature vectors, whereas MDS and RMDS involve estimating class-conditional Gaussian distributions. Inference time refers to the time required to compute the OOD detection score for each method.

Table 2: Comparison on standard OOD detection benchmarks. AUROC (↑) and FPR95 (↓) are averaged across all OOD datasets for each ID dataset. **Bold** entries denote the best performance, and underlined entries indicate the second- and third-best results. † marks results reported in [70]. Overall, FOLD consistently outperforms the competing baselines in terms of average performance.

METHOD	CIFAR-10	CIFAR-100	IMAGENET-200	IMAGENET-1K	AVERAGE
	AUROC(↑) / FPR95(↓)	AUROC(↑) / FPR95(↓)	AUROC(↑) / FPR95(↓)	AUROC(↑) / FPR95(↓)	AUROC(↑) / FPR95(↓)
BASELINES					
OPENMAX† [4]	88.95 / 34.33	78.46 / 55.19	86.23 / 45.26	83.46 / 48.22	84.28 / 45.75
MSP† [19]	89.83 / 37.21	78.60 / 57.40	87.41 / 43.19	81.55 / 57.14	84.35 / 48.73
TEMPSCALE† [17]	90.01 / 39.31	79.46 / 56.79	87.97 / 42.33	83.39 / 53.79	85.21 / 48.05
ODIN† [34]	86.26 / 63.81	79.49 / 58.54	87.13 / 47.24	83.58 / 55.38	84.12 / 56.24
MDS† [32]	87.88 / 38.11	65.82 / 76.01	69.60 / 68.64	66.72 / 71.93	72.51 / 63.67
RMDS† [48]	91.40 / 29.86	82.00 / 53.70	85.86 / 41.08	82.63 / 50.56	85.47 / 43.80
EBO† [35]	90.00 / 48.24	80.15 / 56.26	87.51 / 45.01	84.03 / 50.46	85.42 / 49.99
REACT† [54]	89.31 / 51.12	80.52 / 54.93	88.13 / 42.10	87.15 / 42.46	86.28 / 47.65
MLS† [18]	89.90 / 48.23	80.14 / 56.31	87.83 / 44.32	84.33 / 50.06	85.55 / 49.73
VIM† [63]	91.88 / 31.65	79.46 / 54.70	86.23 / 39.99	84.44 / 43.34	85.50 / 42.42
KNN† [56]	92.19 / **27.52**	81.66 / 56.17	88.52 / 40.43	82.55 / 48.83	86.23 / 43.24
ASH† [14]	77.42 / 81.61	79.79 / 61.37	89.29 / 42.33	**88.71** / 37.02	83.80 / 55.58
SHE† [71]	84.06 / 70.87	77.59 / 62.44	85.96 / 52.02	84.07 / 54.07	82.92 / 59.85
FIXED α					
FOLD	**92.46** / 29.43	81.94 / 51.62	88.65 / 40.30	85.74 / 48.35	87.20 / 42.42
FOLD-R	92.30 / 30.65	**82.22** / **50.55**	89.07 / 38.92	88.04 / 40.36	**87.91** / **40.12**
FOLD-A	89.43 / 49.18	81.98 / 54.26	**89.68** / **37.06**	88.28 / 37.03	87.34 / 44.38
AUTO α TUNING					
AUTOFOLD	**92.46** / 29.43	79.76 / 57.52	88.37 / 40.82	84.38 / 46.93	86.24 / 43.67
AUTOFOLD-R	92.30 / 30.65	79.40 / 55.71	89.03 / 39.45	83.85 / 43.39	86.15 / 42.30
AUTOFOLD-A	89.43 / 49.18	78.44 / 63.84	89.44 / 37.97	87.60 / **36.81**	86.23 / 46.95

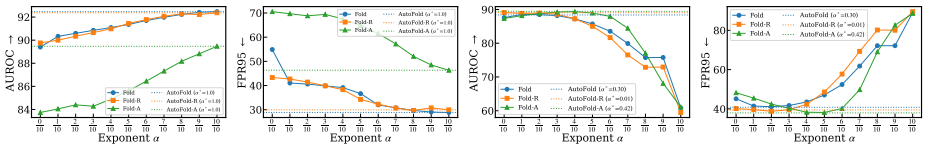


(a) Setup time (ms) **(b)** Batch-wise inference time (ms)

Fig. 3: Setup and inference times of baseline methods on ImageNet-1K, ordered left to right. All measurements are conducted on a single A100 GPU with a batch size of 256. FOLD incurs near-minimal setup and inference overhead. AUTOFOLD, which automatically determines the optimal α, introduces only a marginal increase in setup time and no additional inference cost.

As shown in Figure 3, the FOLD framework incurs minimal overhead in both setup and inference. Specifically, the base FOLD and FOLD-A require virtually no setup time, while the additional computation introduced by FOLD-R remains negligible. During inference, all FOLD variants operate with latency comparable to a standard forward pass, indicating high computational efficiency. In contrast, competing methods such as RMDS, VIM, and KNN often incur substantially higher computational costs despite achieving competitive performance in certain settings. These overheads arise either from lengthy setup procedures (*e.g.*, VIM) or significant computation during both setup and inference (*e.g.*, RMDS and KNN).

To illustrate the performance–efficiency trade-off, we plot standardized computational cost against standardized AUROC (see Figure 1b). The FOLD variants cluster in the upper-left region of the plot, indicating strong detection performance with minimal computational overhead. Overall, these results show that FOLD achieves a favorable balance between detection accuracy and computational efficiency, making it well suited for practical deployment.



(a) CIFAR-10, AUROC(↑) **(b)** CIFAR-10, FPR95(↓) **(c)** ImageNet-200, AUROC(↑) **(d)** ImageNet-200, FPR95(↓)

Fig. 4: Ablation study on the normalization parameter α. On CIFAR-10, larger α values consistently improve AUROC and FPR95 across all FOLD variants, whereas relatively smaller α values yield better performance on ImageNet-200, indicating dataset-dependent sensitivity to α. In contrast, AUTOFOLD automatically identifies a near-optimal α for each dataset, and the resulting performance (dotted line) closely matches the best achievable results.

4.3 Sensitivity Analysis and Automated Calibration

Dataset-dependent sensitivity of α. Figure 4 presents a systematic analysis of the effect of the norm parameter α. We vary α within the range $[0.0, 1.0]$ for all FOLD variants. For CIFAR-10, larger values of α lead to clear performance gains, whereas for ImageNet-200, smaller values yield superior results. This contrast suggests that the optimal influence of α is dataset-dependent, likely reflecting differences in semantic complexity and intrinsic feature distributions.

OOD-free calibration via AUTOFOLD. As indicated by the dotted lines in Figure 4, AUTOFOLD identifies a near-optimal parameter setting that closely matches the peak detection performance without manual tuning. Consistent with this observation, Table 2 shows that AUTOFOLD performs comparably to FOLD evaluated with a fixed α, and occasionally achieves slightly better results. This improvement arises from the search resolution: while the manual FOLD evaluation discretizes α into 10 intervals within $[0.0, 1.0]$, AUTOFOLD explores the parameter space with a finer resolution of 100 intervals. Consequently, AUTOFOLD achieves competitive performance without relying on any external OOD validation data.

Moreover, AUTOFOLD operates with setup times that are orders of magnitude lower (on a logarithmic scale) than those of competing methods (*e.g.*, KNN, MDS, SHE, VIM, RMDS). Notably, the reported setup costs of these baselines do not account for the additional hyperparameter tuning typically performed on auxiliary OOD validation sets. In contrast, AUTOFOLD removes this requirement entirely, highlighting its practicality and efficiency in realistic scenarios where OOD data are unavailable.

5 Further Analysis

5.1 Partial Normalization

Generalizability to existing baselines. To assess whether the proposed partial feature normalization generalizes to existing methods, we evaluate three normalization schemes (no, partial, and full) across multiple baselines on CIFAR-10 and ImageNet-200. As shown in Table 3, partial normalization consistently improves prior approaches. In contrast to previous works that adopt full feature normalization [39, 45, 48], our results reveal clear dataset-dependent behavior. On CIFAR-10, partial and full normalization yield comparable improvements. However, on more complex datasets (*i.e.*, ImageNet-200), full normalization substantially degrades detection performance by discarding

Table 3: Effects of partial feature normalization on baselines, evaluated using AUROC (↑) and FPR95 (↓) on CIFAR-10 and ImageNet-200. On CIFAR-10, normalization improves performance, with partial and full normalization both competitive. On ImageNet-200, partial normalization yields consistent gains, whereas full normalization substantially degrades detection performance.

Dataset	Norm.	Method							Average
		MSP	EBO	ReAct	ASH	Fold	Fold-R	Fold-A	
CIFAR-10	$\alpha = 0$	89.83 / 37.21	90.00 / 48.24	89.31 / 51.12	77.42 / 81.61	89.72 / 51.40	89.83 / 43.46	83.82 / 69.63	87.13 / 54.67
	$0 < \alpha < 1$	91.92 / 28.22	**92.37** / **31.61**	92.06 / **33.74**	88.02 / 56.70	92.42 / **29.22**	92.28 / **30.14**	88.82 / 51.00	91.13 / 37.23
	$\alpha = 1$	**92.01** / **27.84**	92.30 / 32.37	**92.10** / 33.75	**88.82** / **53.26**	**92.46** / 29.43	**92.29** / 31.17	**89.43** / **49.18**	**91.34** / **36.71**
ImageNet-200	$\alpha = 0$	87.41 / 43.19	**87.51** / 45.01	**88.13** / 42.10	89.29 / 42.33	87.81 / 44.10	**89.22** / 39.24	87.61 / 47.14	87.64 / 43.30
	$0 < \alpha < 1$	**88.31** / 40.62	87.38 / **44.82**	88.02 / **41.84**	**89.67** / **39.83**	**88.65** / **40.16**	89.12 / **37.90**	**89.64** / **37.47**	**89.91** / **40.38**
	$\alpha = 1$	87.65 / **40.39**	64.92 / 87.96	62.34 / 88.98	66.41 / 85.33	62.23 / 88.56	60.89 / 89.04	62.51 / 88.08	79.02 / 81.19

informative magnitude cues. In contrast, partial normalization improves performance by preserving informative feature variance while suppressing noisy magnitude effects. These findings suggest that partial normalization serves as an effective plug-and-play component for enhancing general OOD detection frameworks.

Effect of feature normalization. To further analyze the effect of partial normalization on the FOLD score, we examine the relationship between feature norm magnitude and detection score on ImageNet-200. As shown in Figure 5, partial normalization increases the mean score gap (*i.e.*, $\Delta\mu$) between ID and OOD samples from 0.90 to 1.01 while substantially reducing the variance of the OOD distribution relative to the unnormalized baseline. Consequently, Cohen's d [8], a standard measure of statistical separability, improves from 1.605 to 1.689, yielding an absolute 8.06% improvement in FPR95. In contrast, full normalization collapses score variance and severely degrades discriminability (Cohen's $d = 0.520$), rendering two distributions nearly indistinguishable (*i.e.*, FPR95 $= 92.54\%$).

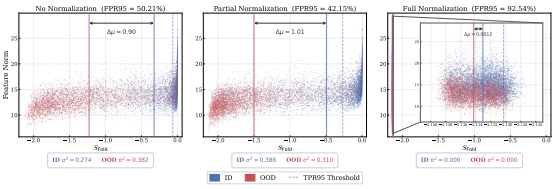


Fig. 5: Scatter plots illustrating the relationship between feature norm and FOLD score under three normalization regimes. Partial normalization enhances ID–OOD separability by increasing the mean score gap and reducing OOD variance, yielding a higher Cohen's d ($1.605 \rightarrow 1.689$) [8]. In contrast, full normalization removes magnitude information, collapsing score variance and markedly reducing discriminability (Cohen's $d = 0.520$).

Spectral analysis of feature curvature. To better understand this behavior, we analyze the top 50 absolute eigenvalues of the feature Hessian for ID and OOD samples under three normalization regimes. As shown in Figure 6, full normalization collapses the spectral gap across the spectrum, rendering the ID and OOD spectra nearly identical and thereby explaining the severe degradation in detection performance. Without normalization, the spectral difference is largely concentrated in the top 10 eigenvalues, indicating that the score primarily depends on worst-case curvature. In contrast, partial normalization slightly reduces the gap among the largest eigenvalues while enlarging the spectral separation across a broader portion of the spectrum. This shift suggests that partial normalization exploits more distributed, average-case curvature information, capturing richer geometric signals that improve the robustness of OOD detection.

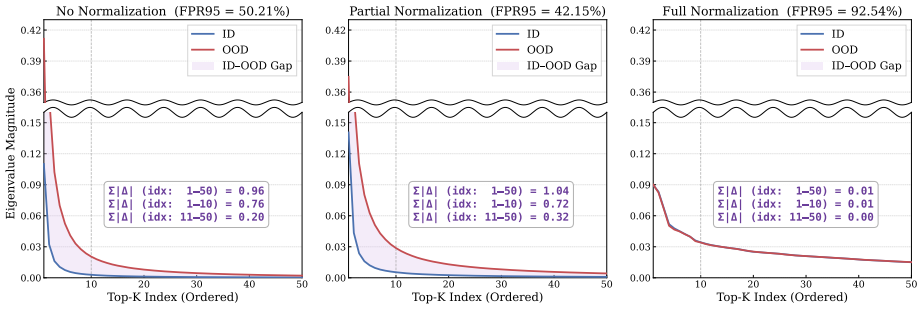


Fig. 6: Top-50 absolute eigenvalues of the feature Hessian for ID and OOD samples on ImageNet-200 under no, partial, and full normalization. Full normalization collapses the spectral gap, making the ID and OOD spectra nearly identical. Without normalization, the difference is concentrated in only the top portion of the spectrum. In contrast, partial normalization enlarges the spectral gap across a broader range of eigenvalues, aligning with its improved OOD detection performance.

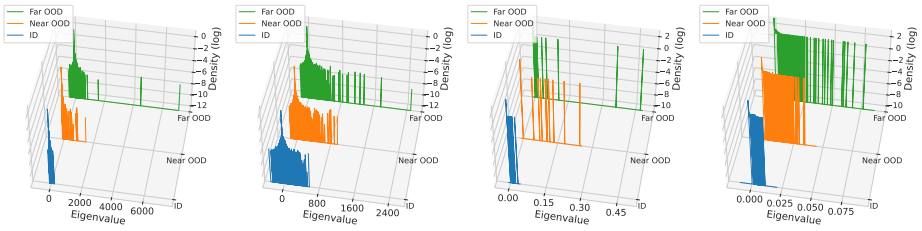
5.2 Feature Hessian

Empirical efficacy of the feature Hessian. To empirically validate the advantage of evaluating curvature in the latent space, we compare the detection performance of the feature Hessian with that of the logit Hessian. As shown in Table 4, the feature Hessian consistently improves the average AUROC across standard benchmarks. While the performance remains competitive on simpler datasets such as CIFAR-10, the gains become more pronounced on more complex domains (*e.g.*, ImageNet-200 and ImageNet-1K). These empirical findings align with our methodological motivation. By transporting curvature through the classifier mapping, the feature Hessian leverages the discriminative geometry encoded in the decision boundaries. Consequently, it captures both predictive uncertainty and intrinsic feature-space structure, yielding richer and more robust representations for OOD detection.

Table 4: Comparison of logit and feature Hessians in AUROC (↑) across datasets. The feature Hessian achieves higher performance, indicating richer signals for OOD detection.

HESSIAN	C-10	C-100	IN-200	IN-1K	AVG.
LOGIT	92.52	81.27	86.35	84.10	86.02
FEATURE	92.46	81.94	88.65	85.74	86.81

Discriminative power of the feature Hessian. To examine whether the feature Hessian exhibits curvature behavior similar to that of the parameter-space Hessian, we analyze the empirical spectral density (ESD) [67]. The ESD provides a comprehensive characterization of curvature by capturing the distribution of eigenvalues across the spectrum. As illustrated in Figure 7, although the absolute scales differ due to dimensionality and parameterization, both the spectral density and the largest eigenvalue consistently increase under distribution shifts. Notably, the feature Hessian exhibits a tightly concentrated spectrum for ID data and a more dispersed spectrum for OOD data. These observations indicate that the feature Hessian serves as a reliable and computationally efficient surrogate for the full parameter-space Hessian. Moreover, it can offer a sharper and more discriminative signal for separating ID from OOD inputs.



(a) CIFAR-10, Full **(b)** ImageNet-200, Full **(c)** CIFAR-10, Feature **(d)** ImageNet-200, Feature

Fig. 7: Empirical spectral densities of the parameter and feature Hessians for ID and OOD data on CIFAR-10 and ImageNet-200. Consistent with the full-parameter Hessian, the feature Hessian demonstrates a monotonic increase in the largest eigenvalue when transitioning from ID to OOD samples. Furthermore, the feature Hessian exhibits a broader spectral distribution than the parameter Hessian, suggesting a more pronounced separability signal for OOD detection.

Table 5: Comparison of OOD detection performance on the ImageNet-1K benchmark across diverse backbone architectures [22, 46, 65, 69]. Across architectures, FOLD yields consistent but modest improvements in average performance, indicating stable and architecture-agnostic gains.

METHOD	REGNET [46]	DENSENET [22]	WRN [69]	RESNEXT [65]	AVERAGE
	AUROC(↑) / FPR95(↓)	AUROC(↑) / FPR95(↓)	AUROC(↑) / FPR95(↓)	AUROC(↑) / FPR95(↓)	AUROC(↑) / FPR95(↓)
BASELINES					
MSP [19]	88.05 / 44.60	79.80 / 59.20	83.19 / 54.78	82.00 / 56.96	83.26 / 53.89
TEMPSCALE [17]	90.79 / 38.80	82.08 / 55.83	84.86 / 52.00	84.09 / 53.87	85.45 / 50.13
MDS [32]	88.32 / 37.66	66.44 / 75.31	71.71 / 64.41	65.85 / 71.96	73.08 / 62.33
RMDS [48]	90.66 / 34.60	79.34 / 61.99	85.31 / 43.25	83.86 / 45.81	84.79 / 46.41
EBO [48]	92.82 / 35.26	82.51 / 52.13	84.43 / 52.21	84.83 / 51.75	86.15 / 47.84
REACT [54]	84.93 / 47.75	83.80 / 46.63	85.69 / 49.32	87.40 / 42.64	85.46 / 46.59
MLS [18]	92.59 / 35.96	82.80 / 51.92	84.89 / 51.88	85.02 / 51.48	86.33 / 47.81
VIM [63]	91.50 / 31.28	80.06 / 50.06	83.62 / 45.53	84.62 / 42.90	84.95 / 42.44
KNN [56]	91.72 / 30.72	72.88 / 64.20	85.39 / 44.12	85.38 / 43.72	83.84 / 45.69
ASH [14]	78.22 / 61.88	83.00 / 50.93	87.80 / 46.40	**88.93** / 40.32	84.49 / 49.88
SHE [71]	87.23 / 50.81	81.21 / 57.61	77.67 / 65.85	79.97 / 65.03	81.52 / 59.82
FIXED α					
FOLD	**93.78** / **28.93**	83.52 / 47.21	85.53 / 45.22	85.12 / 47.00	86.99 / 42.09
FOLD-R	89.39 / 38.37	**85.41** / 45.58	86.59 / 42.95	86.38 / 42.68	86.94 / 42.40
FOLD-A	86.87 / 48.44	84.91 / **44.59**	87.95 / **37.60**	88.61 / **37.54**	**87.08** / **42.05**
AUTO α TUNING					
AUTOFOLD	92.80 / 32.76	83.69 / 47.76	85.62 / 49.05	85.28 / 48.48	86.85 / 44.51
AUTOFOLD-R	90.02 / 37.12	83.22 / 51.80	84.76 / 46.45	84.67 / 45.69	85.67 / 45.27
AUTOFOLD-A	85.59 / 51.88	85.02 / 45.50	**88.60** / 42.53	88.15 / 37.71	86.84 / 44.40

5.3 Generalization across diverse architectures

To evaluate architectural robustness, we compare OOD detection performance on the ImageNet-1K benchmark using four backbone networks: RegNet [46], DenseNet [22], WRN [69], and ResNeXt [65]. As shown in Table 5, the FOLD framework consistently demonstrates strong and stable performance across all evaluated architectures. In particular, FOLD-A achieves the highest average AUROC of 87.08% and the lowest average FPR95 of 42.05%, corresponding to a 0.75% absolute AUROC improvement over the strongest baseline (*i.e.*, MLS) and a 0.39% reduction in FPR95 compared to the most competitive baseline for that metric (*i.e.*, VIM). Moreover, the base FOLD model also attains a competitive average AUROC of 86.99%.

These results indicate that the geometric advantages of curvature-based OOD scoring are not limited to a particular architecture family but generalize across diverse models. Even without an external validation set, the AUTOFOLD variants maintain robust detection performance. Overall, these findings establish FOLD as a versatile OOD detection approach capable of achieving reliable and stable performance across architectures.

6 Theoretical Analysis

In this section, we provide theoretical justification for the proposed FOLD score by analyzing the curvature of the energy function under a simplified binary classification setting. In particular, we show that the expected trace of the feature Hessian is provably larger for OOD data than for ID data.

Let the feature representation $\widetilde{\boldsymbol{x}} = h(\boldsymbol{x}) \in \mathbb{R}^d$ follow a class-conditional distribution $\mathcal{D}_{\text{id}}$ defined as

$$\widetilde{\boldsymbol{x}} \mid y \sim \mathcal{D}_{\text{id}} = \begin{cases} \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) & \text{if } y = 1, \\ \mathcal{N}(-\boldsymbol{\mu}, \boldsymbol{\Sigma}) & \text{if } y = -1, \end{cases} \tag{8}$$

where $\boldsymbol{\Sigma}$ is a positive definite covariance matrix. Consider a linear classifier $g(\widetilde{\boldsymbol{x}}; \boldsymbol{w}) = [\widetilde{\boldsymbol{x}}^\top \boldsymbol{w}, 0]^\top$ trained by minimizing the expected logistic loss $\mathbb{E}[-\log(\sum_{i=1}^{2} \exp(yz_i))]$, where $\boldsymbol{z} = g(\widetilde{\boldsymbol{x}}; \boldsymbol{w})$ denotes the logits. To model OOD inputs, we assume features are drawn from a shifted and rotated variant of $\mathcal{D}_{\text{id}}$:

$$\widetilde{\boldsymbol{x}} \mid y \sim \mathcal{D}_{\text{ood}} = \begin{cases} \mathcal{N}(c\mathbf{R}\boldsymbol{\mu}, \boldsymbol{\Sigma}) & \text{if } y = 1, \\ \mathcal{N}(-c\mathbf{R}\boldsymbol{\mu}, \boldsymbol{\Sigma}) & \text{if } y = -1, \end{cases} \tag{9}$$

where $\mathbf{R}$ is a rotation matrix and $c > 0$ controls the magnitude of the distributional shift.

Theorem 1. *If $c|\boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1} \mathbf{R} \boldsymbol{\mu}| \leq \boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}$, then the expected trace of the feature Hessian is smaller for ID data than for OOD data,* i.e.*,*

$$\mathbb{E}_{\mathcal{D}_{\text{id}}}\left[\text{tr}(\nabla^2_{\widetilde{\boldsymbol{x}}} \mathcal{L}(\boldsymbol{w}))\right] \leq \mathbb{E}_{\mathcal{D}_{\text{ood}}}\left[\text{tr}(\nabla^2_{\widetilde{\boldsymbol{x}}} \mathcal{L}(\boldsymbol{w}))\right]. \tag{10}$$

Proof. For $\boldsymbol{z} = g(\widetilde{\boldsymbol{x}}; \boldsymbol{w}) = [\widetilde{\boldsymbol{x}}^\top \boldsymbol{w}, 0]^\top$, the trace of the logit-space Hessian satisfies

$$\text{tr}(\nabla^2_{\boldsymbol{z}} \mathcal{L}(\boldsymbol{w})) = 1 - \sum_{i=1}^{2} \left(\frac{\exp z_i}{\exp z_1 + 1} \right)^2. \tag{11}$$

Taking expectation yields

$$\mathbb{E}_{\mathcal{D}}[\text{tr}(\nabla^2_{\boldsymbol{z}} \mathcal{L}(\boldsymbol{w}))] = 2\mathbb{E}_{\mathcal{D}}\left[\frac{\exp z_1}{(\exp z_1 + 1)^2} \right]. \tag{12}$$

A fundamental property of binary classification is that the expected energy loss $\mathbb{E}[\mathcal{L}(\boldsymbol{w})]$ admits a unique minimizer whose direction aligns with that of the Bayes-optimal classifier [25, 40, 44, 53]. In particular, Oh *et al*. [44, Thm. 3.1] show that training on ID data converges to the population-risk minimizer $\boldsymbol{w}^* = \alpha \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}$ for some $\alpha > 0$. Without loss of generality, we set $\alpha = 1$.

Under $\mathcal{D}_{\text{id}}$, $\widetilde{\boldsymbol{x}}$ follows a balanced mixture of $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and $\mathcal{N}(-\boldsymbol{\mu}, \boldsymbol{\Sigma})$; under $\mathcal{D}_{\text{ood}}$, the same structure holds with shifted means. Hence

$$\mathbb{E}_{\mathcal{D}_{\text{id}}}[\text{tr}(\nabla^2_{\boldsymbol{z}} \mathcal{L}(\boldsymbol{w}))] = 2\mathbb{E}_{\widetilde{\boldsymbol{x}} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})}\left[\frac{\exp(\widetilde{\boldsymbol{x}}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu})}{(\exp(\widetilde{\boldsymbol{x}}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}) + 1)^2} \right], \tag{13}$$

$$\mathbb{E}_{\mathcal{D}_{\text{ood}}}[\text{tr}(\nabla^2_{\boldsymbol{z}} \mathcal{L}(\boldsymbol{w}))] = 2\mathbb{E}_{\widetilde{\boldsymbol{x}} \sim \mathcal{N}(c\mathbf{R}\boldsymbol{\mu}, \boldsymbol{\Sigma})}\left[\frac{\exp(\widetilde{\boldsymbol{x}}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu})}{(\exp(\widetilde{\boldsymbol{x}}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}) + 1)^2} \right]. \tag{14}$$

By the affine invariance of the Gaussian distribution, the scalar random variable $x := \widetilde{\boldsymbol{x}}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$ follows a normal distribution with variance $\sigma^2 = \boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$. Its mean is $\mu_{\text{id}} = \boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$ under $\mathcal{D}_{\text{id}}$ and $\mu_{\text{ood}} = c\boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1}\mathbf{R}\boldsymbol{\mu}$ under $\mathcal{D}_{\text{ood}}$. Therefore,

$$\mathbb{E}_{\mathcal{D}_{\text{id}}}[\text{tr}(\nabla^2_{\boldsymbol{z}}\mathcal{L}(\boldsymbol{w}))] = 2\mathbb{E}_{x\sim\mathcal{N}(\mu_{\text{id}},\sigma^2)}[\varphi(x)], \tag{15}$$

$$\mathbb{E}_{\mathcal{D}_{\text{ood}}}[\text{tr}(\nabla^2_{\boldsymbol{z}}\mathcal{L}(\boldsymbol{w}))] = 2\mathbb{E}_{x\sim\mathcal{N}(\mu_{\text{ood}},\sigma^2)}[\varphi(x)], \tag{16}$$

where $\varphi(x) = \frac{\exp x}{(\exp x+1)^2}$.

Since $\varphi(x)$ is strictly decreasing in $|x|$ for $x \geq 0$, the condition $c|\boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1}\mathbf{R}\boldsymbol{\mu}| \leq \boldsymbol{\mu}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$ implies

$$\mathbb{E}_{\mathcal{D}_{\text{id}}}[\text{tr}(\nabla^2_{\boldsymbol{z}}\mathcal{L}(\boldsymbol{w}))] \leq \mathbb{E}_{\mathcal{D}_{\text{ood}}}[\text{tr}(\nabla^2_{\boldsymbol{z}}\mathcal{L}(\boldsymbol{w}))]. \tag{17}$$

Applying the second-order chain rule,

$$\nabla^2_{\widetilde{\boldsymbol{x}}}\mathcal{L}(\boldsymbol{w}) = \boldsymbol{U}\nabla^2_{\boldsymbol{z}}\mathcal{L}(\boldsymbol{w})\boldsymbol{U}^\top, \quad \boldsymbol{U} = \begin{bmatrix}\boldsymbol{w}^* & 0\end{bmatrix}. \tag{18}$$

Using the cyclic trace identity,

$$\text{tr}(\nabla^2_{\widetilde{\boldsymbol{x}}}\mathcal{L}(\boldsymbol{w})) = \|\boldsymbol{w}^*\|^2\,\text{tr}(\nabla^2_{\boldsymbol{z}}\mathcal{L}(\boldsymbol{w}))/2. \tag{19}$$

Thus the inequality also holds in feature space:

$$\mathbb{E}_{\mathcal{D}_{\text{id}}}[\text{tr}(\nabla^2_{\widetilde{\boldsymbol{x}}}\mathcal{L})] \leq \mathbb{E}_{\mathcal{D}_{\text{ood}}}[\text{tr}(\nabla^2_{\widetilde{\boldsymbol{x}}}\mathcal{L}(\boldsymbol{w}))]. \tag{20}$$

Theorem 1 formalizes the intuition that OOD samples, which lie in regions of lower model confidence due to their absence during training, tend to exhibit larger traces of the feature Hessian. For example, when $\boldsymbol{\Sigma} = \mathbf{I}$, the inequality naturally holds for any $c < 1$ and rotation $\mathbf{R}$, corresponding to shifts toward low-confidence regions.

Overall, this analysis provides theoretical support for using the feature-Hessian trace as an OOD detection score in FOLD. Despite the simplifying assumptions on the in- and out-of-distribution data (*e.g.*, Gaussian mixtures and rotational shifts) and the omission of partial feature normalization, it offers an intuitive explanation for why OOD samples tend to exhibit larger Hessian traces than ID samples. Consistent with this intuition, our empirical results (see Table 6 in the Appendix) show substantially larger feature-Hessian traces for OOD data, supporting the practical effectiveness of the proposed approach.

7 Discussion

7.1 Feature Magnitude

The role of feature magnitude depends strongly on the complexity of the underlying recognition task. Feature magnitude reflects semantic activation intensity, but its discriminative value varies across datasets. For simpler datasets (*e.g.*, CIFAR-10), feature direction alone is often sufficient for class separation, and excessively large magnitudes mainly introduce overconfident noise, favoring strong suppression ($\alpha \approx 1$). In contrast, for more complex datasets (*e.g.*, ImageNet) with substantial intra-class variation, feature magnitude contains fine-grained semantic information that complements feature direction, making weaker normalization ($\alpha \approx 0.2$) preferable to avoid discarding useful cues. These observations explain why the optimal α varies with dataset complexity and suggest that adaptive feature calibration is a promising direction for future work.

7.2 Theoretical Limitation

Our theoretical analysis provides intuition rather than a complete characterization. It relies on a simplified setting and therefore does not fully capture the deep, multiclass regime considered in our experiments. Accordingly, it motivates the connection between the feature-Hessian trace and curvature rather than providing a formal guarantee. A tighter characterization of the ID-OOD curvature gap is left for future work.

8 Related Work

Post-hoc methods derive uncertainty scores from a fixed pre-trained model, including logit-based [17, 18, 19, 34, 35], distance-based [14, 16, 32, 48, 54, 56, 63, 73], and gradient-based methods [23, 51]. Related curvature-based methods estimate parameter-space sharpness via Laplace approximations [26, 49], but incur substantially higher computational cost. *Training-time methods* instead improve OOD detection through modified optimization, including outlier exposure [20], synthetic outlier generation [15, 59], confidence calibration [12, 31], logit normalization [64], contrastive learning [58], regularization without OOD data [21, 68], and loss-landscape regularization [47].

9 Conclusion

In this work, we investigated the curvature discrepancy between ID and OOD data to enable efficient post-hoc OOD detection, leading to three main contributions:

i. We show that OOD inputs exhibit larger Hessian curvature than ID data, with the discrepancy increasing under stronger distributional shifts.
ii. We propose FOLD, which leverages the feature Hessian and partial normalization to improve ID-OOD separability without parameter-space overhead.
iii. We introduce AUTOFOLD, a self-supervised tuning scheme based on ID logit masking for automatic normalization calibration without external data.

Together, these contributions establish a principled and efficient framework for reliable OOD detection in real-world deployment settings.

Acknowledgement

This work was partly supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (RS-2019-II191906, Artificial Intelligence Graduate School Program (POSTECH); RS-2022-II220959, (part2) Few-Shot learning of Causal Inference in Vision and Language for Decision Making; RS-2025-25441838, Development of a human foundation model for human-centric universal artificial intelligence and training of personnel), and the National Research Foundation of Korea (NRF) grants funded by the Korea government (MSIT) (RS-2023-00210466, RS-2026-25500419).

Bibliography

[1] Ahmed, F., Courville, A.: Detecting semantic anomalies. In: AAAI (2020)
[2] Avron, H., Toledo, S.: Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. Journal of the ACM (2011)
[3] Bai, Z., Fahey, G., Golub, G.: Some large-scale matrix computation problems. Journal of Computational and Applied Mathematics (1996)
[4] Bendale, A., Boult, T.E.: Towards open set deep networks. In: CVPR (2016)
[5] Bitterwolf, J., Müller, M., Hein, M.: In or out? fixing imagenet out-of-distribution detection evaluation. In: ICML (2023)
[6] Chen, C., Fu, Z., Liu, K., Chen, Z., Tao, M., Ye, J.: Optimal parameter and neuron pruning for out-of-distribution detection. In: NeurIPS (2023)
[7] Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. In: CVPR (2014)
[8] Cohen, J.: Statistical power analysis for the behavioral sciences. routledge (2013)
[9] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: CVPR (2009)
[10] Deng, L.: The mnist database of handwritten digit images for machine learning research. IEEE signal processing magazine (2012)
[11] DeVries, T., Taylor, G.W.: Learning confidence for out-of-distribution detection in neural networks. arXiv preprint arXiv:1802.04865 (2018)
[12] DeVries, T., Taylor, G.W.: Learning confidence for out-of-distribution detection in neural networks. arXiv preprint arXiv:1802.04865 (2018)
[13] Dhamija, A.R., Günther, M., Boult, T.: Reducing network agnostophobia. In: NeurIPS (2018)
[14] Djurisic, A., Bozanic, N., Ashok, A., Liu, R.: Extremely simple activation shaping for out-of-distribution detection. In: ICLR (2023)
[15] Du, X., Wang, Z., Cai, M., Li, Y.: VOS: Learning what you don't know by virtual outlier synthesis. In: ICLR (2022)
[16] Fang, K., et al.: Kernel PCA for out-of-distribution detection. In: NeurIPS (2024)
[17] Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. In: ICML (2017)
[18] Hendrycks, D., Basart, S., Mazeika, M., Mostajabi, M., Steinhardt, J., Song, D.: Scaling out-of-distribution detection for real-world settings. In: ICML (2022)
[19] Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: ICLR (2017)
[20] Hendrycks, D., Mazeika, M., Dietterich, T.: Deep anomaly detection with outlier exposure. In: ICLR (2019)
[21] Hsu, Y.C., Shen, Y., Jin, H., Kira, Z.: Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In: CVPR (2020)
[22] Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: CVPR (2017)
[23] Huang, R., Geng, A., Li, Y.: On the importance of gradients for detecting distributional shifts in the wild. In: NeurIPS (2021)

[24] Hutchinson, M.F.: A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. Communications in Statistics-Simulation and Computation (1989)
[25] Ji, Z., Telgarsky, M.: The implicit bias of gradient descent on nonseparable data. In: COLT (2019)
[26] Kristiadi, A., Hein, M., Hennig, P.: Being bayesian, even just a bit, fixes overconfidence in relu networks. In: ICML (2020)
[27] Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto (2009)
[28] Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., et al.: The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. IJCV (2020)
[29] Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In: NeurIPS (2017)
[30] Le, Y., Yang, X.: Tiny imagenet visual recognition challenge. CS 231N (2015)
[31] Lee, K., Lee, H., Lee, K., Shin, J.: Training confidence-calibrated classifiers for detecting out-of-distribution samples. In: ICLR (2018)
[32] Lee, K., Lee, K., Lee, H., Shin, J.: A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In: NeurIPS (2018)
[33] Lee, S., Park, J., Lee, J.: Implicit jacobian regularization weighted with impurity of probability output. In: ICML (2023)
[34] Liang, S., Li, Y., Srikant, R.: Enhancing the reliability of out-of-distribution image detection in neural networks. In: ICLR (2018)
[35] Liu, W., Wang, X., Owens, J., Li, Y.: Energy-based out-of-distribution detection. In: NeurIPS (2020)
[36] Liu, Y., Chris, X., Li, H., Ma, L., Wang, S.: Neuron activation coverage: Rethinking out-of-distribution detection and generalization. In: ICLR (2024)
[37] Madras, D., Atwood, J., D'Amour, A.: Detecting extrapolation with local ensembles. In: ICLR (2019)
[38] Moosavi-Dezfooli, S.M., Fawzi, A., Fawzi, O., Frossard, P.: Universal adversarial perturbations. In: CVPR (2017)
[39] Mueller, M., Hein, M.: Mahalanobis++: Improving ood detection via feature normalization. In: ICML (2025)
[40] Nacson, M.S., Srebro, N., Soudry, D.: Stochastic gradient descent on separable data: Exact convergence with a fixed learning rate. In: AISTATS (2019)
[41] Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y., et al.: Reading digits in natural images with unsupervised feature learning. In: NeurIPS Workshop on deep learning and unsupervised feature learning (2011)
[42] Nguyen, A., Yosinski, J., Clune, J.: Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In: CVPR (2015)
[43] Nguyen, A., Bertrand, A., Le Hégarat-Mascle, S., Aldea, E., FLORIN, F., EL-KORSO, M.N., LUSTRAT, R.: Fisher-rao sensitivity for out-of-distribution detection in deep neural networks. In: ICLR (2026)

[44] Oh, J., Yun, C.: Provable benefit of mixup for finding optimal decision boundaries. In: ICML (2023)
[45] Park, J., Chai, J.C.L., Yoon, J., Teoh, A.B.J.: Understanding the feature norm for out-of-distribution detection. In: ICCV (2023)
[46] Radosavovic, I., Kosaraju, R.P., Girshick, R., He, K., Dollár, P.: Designing network design spaces. In: CVPR (2020)
[47] Ravikumar, D., Soufleri, E., Roy, K.: Curvature clues: Decoding deep learning privacy with input loss curvature. In: NeurIPS (2024)
[48] Ren, J., Fort, S., Liu, J., Roy, A.G., Padhy, S., Lakshminarayanan, B.: A simple fix to mahalanobis distance for improving near-ood detection. In: ICML Workshop on Uncertainty and Robustness in Deep Learning (2021)
[49] Ritter, H., Botev, A., Barber, D.: A scalable laplace approximation for neural networks. In: ICLR (2018)
[50] Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S.A., Binder, A., Müller, E., Kloft, M.: Deep one-class classification. In: ICML (2018)
[51] Seleznova, M., et al.: GradPCA: Leveraging NTK alignment for reliable out-of-distribution detection. In: ICLR (2026)
[52] Sharma, A., Azizan, N., Pavone, M.: Sketching curvature for efficient out-of-distribution detection for deep neural networks. In: UAI (2021)
[53] Soudry, D., Hoffer, E., Nacson, M.S., Gunasekar, S., Srebro, N.: The implicit bias of gradient descent on separable data. JMLR (2018)
[54] Sun, Y., Guo, C., Li, Y.: React: Out-of-distribution detection with rectified activations. In: NeurIPS (2021)
[55] Sun, Y., Li, S.: Dice: Leveraging sparsification for out-of-distribution detection. In: ECCV (2022)
[56] Sun, Y., Ming, Y., Zhu, X., Li, Y.: Out-of-distribution detection with deep nearest neighbors. In: ICML (2022)
[57] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. In: ICLR (2014)
[58] Tack, J., Mo, S., Jeong, J., Shin, J.: Csi: Novelty detection via contrastive learning on distributionally shifted instances. In: NeurIPS (2020)
[59] Tao, L., Du, X., Zhu, X., Li, Y.: Non-parametric outlier synthesis. In: ICLR (2023)
[60] Van Horn, G., Mac Aodha, O., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., Belongie, S.: The inaturalist species classification and detection dataset. In: CVPR (2018)
[61] Vaze, S., Han, K., Vedaldi, A., Zisserman, A.: Open-set recognition: A good closed-set classifier is all you need. In: ICLR (2022)
[62] Vyas, A., Jammalamadaka, N., Zhu, X., Das, D., Kaul, B., Willke, T.L.: Out-of-distribution detection using an ensemble of self supervised leave-out classifiers. In: ECCV (2018)
[63] Wang, H., Li, Z., Feng, L., Zhang, W.: ViM: Out-of-distribution with virtual-logit matching. In: CVPR (2022)
[64] Wei, H., Xie, R., Cheng, H., Feng, L., An, B., Li, Y.: Mitigating neural network overconfidence with logit normalization. In: ICML (2022)
[65] Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: CVPR (2017)

[66] Yang, J., Wang, P., Zou, D., Zhou, Z., Ding, K., Peng, W., Wang, H., Chen, G., Li, B., Sun, Y., et al.: Openood: Benchmarking generalized out-of-distribution detection. In: NeurIPS (2022)
[67] Yao, Z., Gholami, A., Keutzer, K., Mahoney, M.W.: Pyhessian: Neural networks through the lens of the hessian. In: IEEE Big Data (2020)
[68] Yu, Q., Aizawa, K.: Unsupervised out-of-distribution detection by maximum classifier discrepancy. In: ICCV (2019)
[69] Zagoruyko, S., Komodakis, N.: Wide residual networks. In: BMVC (2016)
[70] Zhang, J., Yang, J., Wang, P., Wang, H., Lin, Y., Zhang, H., Sun, Y., Du, X., Li, Y., Liu, Z., Chen, Y., Li, H.: Openood v1.5: Enhanced benchmark for out-of-distribution detection. In: NeurIPS Workshop on Distribution Shifts (2023)
[71] Zhang, J., Fu, Q., Chen, X., Du, L., Li, Z., Wang, G., xiaoguang Liu, Han, S., Zhang, D.: Out-of-distribution detection based on in-distribution data patterns memorization with modern hopfield energy. In: ICLR (2023)
[72] Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. IEEE TPAMI (2017)
[73] Zöngür, B., et al.: Activation subspaces for out-of-distribution detection. In: ICCV (2025)