Bounding Worst-Case Calibration Error in OOD Detection Under Distribution Shift

Claudio César Claros-Olivares **Austin J. Brockmeier**
Department of Electrical & Computer Engineering
University of Delaware

Abstract

Reliable deployment of deep neural networks requires that predictive confidence reflects true correctness likelihood, yet models routinely produce silent failures, confident but incorrect predictions on out-of-distribution (OOD) data. Existing OOD detection scores lack probabilistic semantics, and the standard remedy of post-hoc calibration followed by empirical Expected Calibration Error (ECE) evaluation is insufficient: average-based binning attenuates rare but critical high-confidence errors on OOD inputs. We derive upper bounds on L_1 and L_2 ECE parameterized by the OOD contamination ratio α, providing a worst-case safety envelope for mixed deployment environments. The L_2 bound, grounded in the Brier score decomposition, explicitly penalizes high-magnitude confidence deviations that L_1 averaging obscures. Applying these bounds with out-of-fold calibration selection across a large-scale study of 20+ scoring functions, we demonstrate that methods ranking as optimal under empirical ECE are pruned under worst-case evaluation, exposing a pronounced architectural dichotomy: for convolutional networks trained from scratch, full-distribution entropy scores (Guessing Entropy, Predictive Entropy) yield the tightest safety guarantees, whereas for fine-tuned Vision Transformers, explicit boundary-distance methods (fDBD) dominate due to non-collapsed feature geometries inherited from pretraining.

Workshop "Towards Trustworthy Predictions: Theory and Applications of Calibration for Modern AI" at AISTATS 2026, Tangier, Morocco.

1 Introduction

A significant vulnerability of deep neural networks (DNNs) is their tendency to produce silent failures: incorrect predictions delivered with high confidence that evade standard detection mechanisms (Corbière et al., 2019; Hendrycks and Gimpel, 2016; Lee et al., 2018; Jaeger et al., 2022; Traub et al., 2024). A reliable classifier should provide calibrated confidence values that are high for correctly classified in-distribution (ID) samples and low for out-of-distribution (OOD) samples, enabling downstream systems to flag uncertain inputs and abstain from unreliable predictions.

Since the seminal work of Hendrycks and Gimpel (2016), the OOD detection literature has proposed dozens of confidence scoring functions (CSFs) to separate ID from OOD data. However, these scores generally lack probabilistic semantics, complicating downstream risk assessment. Additionally, evaluating these scores typically relies on threshold-free ranking metrics such as AUROC at 95% FPR or AUGRC (Traub et al., 2024), which assess the relative separability of ID and OOD samples but are insensitive to the absolute calibration of the scores.

A natural remedy is to apply a post-hoc calibration function to transform raw scores into probabilities suitable for risk assessment. However, this introduces its own failure mode: the calibration function can overfit to the validation distribution. In particular, non-parametric mappers such as Isotonic Regression can produce step-function mappings that assign near-zero confidence to most OOD samples but high confidence to a subset of structural outliers that happen to fall near learned thresholds. Standard empirical ECE evaluation obscures these failures because average-based binning attenuates the contribution of rare, high-magnitude confidence errors.

To address these vulnerabilities, we propose evaluating confidence scoring functions through theoretical upper bounds on L_1 and L_2 calibration error, rather than re-

lying on empirical binning. The L_1 bound decomposes into an ID residual term (K_1) and a mean OOD confidence term ($\bar{p}_{\text{OOD}}$), making the trade-off between preserving ID calibration and suppressing OOD overconfidence explicit. The L_2 bound (Root Mean Square Calibration Error) penalizes high-magnitude confidence errors quadratically, revealing failures that the linear averaging of L_1 would otherwise mask.

To ensure that our safety guarantees are not artifacts of calibration overfitting, we employ Out-of-Fold (OOF) dynamic calibration selection. Calibration mappers (Isotonic Regression (Fielding, 1974), Beta Calibration (Kull et al., 2017), Sigmoid scaling (Platt et al., 1999)) are selected based on their generalization performance on held-out validation folds, preventing data leakage from inflating the bounds.

Contributions:

1. We derive L_1 and L_2 ECE upper bounds parameterized by the contamination ratio (α), providing a mathematical safety envelope for mixed deployment environments.

2. Through a large-scale, multiple-comparison-controlled study, we demonstrate that empirical ECE obscures high-magnitude confidence errors, fundamentally altering method selection for safety-critical systems.

3. We present an assessment framework employing out-of-fold dynamic calibration selection to prevent calibration overfitting from producing overly optimistic bound estimates.

2 Related Work

OOD Benchmarking and Metrics. Recent large-scale efforts have standardized the evaluation of generalized OOD detection across semantic and covariate shifts. Frameworks such as OpenOOD (Yang et al., 2022; Zhang et al., 2023) and FD-Shifts (Jaeger et al., 2022) provide extensive suites of confidence scoring functions (CSFs) and evaluation protocols. While these benchmarks have identified that simple baselines often outperform complex scoring rules, they rely almost exclusively on threshold-free ranking metrics. The Area Under the Generalized Risk-Coverage Curve (AUGRC) (Traub et al., 2024), introduced for selective classification, similarly aggregates risk across operating points without requiring calibrated probabilities. These metrics evaluate the relative separability of ID and OOD samples but do not assess whether the scores themselves are calibrated as reliable probabilities of correctness.

Calibration Under Distribution Shift. The challenge of preserving predictive calibration when the test distribution deviates from the training data is a well-documented vulnerability in deep learning (Ovadia et al., 2019). While standard Expected Calibration Error (ECE) is the de facto metric for in-distribution evaluation (Pavlovic, 2025), its reliability under shift is limited by its average-based binning structure, which can attenuate localized, high-impact overconfidence events. Post-hoc calibration mappers such as Sigmoid (Platt) scaling, Beta calibration, and non-parametric Isotonic Regression are typically tuned on a clean validation set and may overfit to the in-distribution score distribution. Our work addresses both limitations by employing theoretical L_1 and L_2 bounds that bypass binning entirely, and by selecting calibration mappers via out-of-fold evaluation to prevent overfitting.

Scoring Rules and Feature Geometry. A separate thread of research exploits the geometric properties of learned feature spaces to improve OOD detection. Methods such as Mahalanobis distance (Lee et al., 2018) and Virtual-logit Matching (ViM) (Wang et al., 2022) use the structure of the penultimate layer to identify atypical inputs, while more recent approaches including fDBD (Liu and Qin, 2023), NNGuide (Park et al., 2023), and pNML (Bibas et al., 2021) quantify the distance between test samples and class-specific manifolds. The Neural Collapse (NC) phenomenon (Papyan et al., 2020; Zhou et al., 2025) provides a theoretical lens for understanding these methods: architectures trained from scratch exhibit strong feature concentration toward class prototypes, whereas fine-tuned models retain high-variance representational structure from pretraining. As we show in our experiments, this geometric distinction has direct consequences for worst-case calibration safety, determining which families of scoring functions maintain tight bounds under distribution shift.

3 Theoretical Framework: ECE Bounds Under Shift

To evaluate calibration safety, we avoid empirical binning and instead bound the worst-case calibration error. We define a mixed deployment environment $\mathcal{D}_{\text{mix}} = \mathcal{D}_{\text{ID}} \cup \mathcal{D}_{\text{OOD}}$, following the experimental scenario of Jaeger et al. (2022). Here, $\mathcal{D}_{\text{ID}}$ comprises correctly classified in-distribution samples (ID Hits), for which the ideal confidence is $y = 1$, and $\mathcal{D}_{\text{OOD}}$ comprises out-of-distribution samples, for which the ideal confidence is $y = 0$. This formulation isolates the calibration question from classification accuracy: given that a sample is either a correct ID prediction or an OOD input, how well does the confidence score reflect

this binary ground truth?

3.1 The L_1 and L_2 Bounds

We parameterize the mixture by the contamination ratio $\alpha = |\mathcal{D}_{\text{OOD}}|/|\mathcal{D}_{\text{ID}}|$, reflecting that the frequency of OOD encounters is unknown at deployment. Let $\hat{p} \in [0, 1]$ be the calibrated probability output of a Confidence Score Function (CSF). For a safe system, $\hat{p}$ should approach 1 for ID Hits and 0 for OOD instances.

We define the mean ID residual as $K_1 = 1 - \overline{p}_{\text{ID}} = 1 - \frac{1}{|\mathcal{D}_{\text{ID}}|}\sum_{i\in\mathcal{D}_{\text{ID}}} \hat{p}_i$, which measures how far the average ID confidence falls short of the ideal value of 1. The mean OOD confidence is $\overline{p}_{\text{OOD}} = \frac{1}{|\mathcal{D}_{\text{OOD}}|}\sum_{j\in\mathcal{D}_{\text{OOD}}} \hat{p}_j$, which measures the residual overconfidence on OOD data. Ideally, both K_1 and $\overline{p}_{\text{OOD}}$ are small. The L_1 ECE is bounded by a weighted combination of these two terms:

$$\text{ECE}_{L_1} \leq \frac{1}{1+\alpha} K_1 + \frac{\alpha}{1+\alpha}\overline{p}_{\text{OOD}}. \tag{1}$$

While the L_1 bound captures the average calibration error, it is insensitive to rare OOD instances assigned near-100% confidence. To penalize such high-magnitude errors, we derive an L_2 upper bound using the Brier Score (BS) decomposition (Blattenberger and Lad, 1985): BS = CAL + REF, where CAL $= ECE_{L_2}^2$ is the squared calibration error and REF (Refinement) measures the variance of the true labels within each probability bin. Since REF ≥ 0, the Brier Score provides a guaranteed upper envelope: $ECE_{L_2} \leq \sqrt{\text{BS}}$. Decomposing the Brier Score over the mixed dataset yields:

$$\text{ECE}_{L_2} \leq \sqrt{\frac{1}{1+\alpha} K_2 + \frac{\alpha}{1+\alpha}\overline{p^2}_{\text{OOD}}}, \tag{2}$$

where $K_2 = \frac{1}{|\mathcal{D}_{\text{ID}}|}\sum_{i\in\mathcal{D}_{\text{ID}}}(1-\hat{p}_i)^2$ and $\overline{p^2}_{\text{OOD}} = \frac{1}{|\mathcal{D}_{\text{OOD}}|}\sum_{j\in\mathcal{D}_{\text{OOD}}} \hat{p}_j^2$ are the mean squared errors on each subset. Because the L_2 norm squares individual deviations before averaging, a single OOD sample assigned $\hat{p} \approx 1$ contributes disproportionately to the bound, exposing high-confidence failures that L_1 averaging would dilute. This bound reaches equality when REF $= 0$, i.e., when every probability bin contains samples of only one type (all ID Hits or all OOD), corresponding to the worst-case scenario where the calibration mapper produces no useful probability discrimination within bins. See Appendix A for the full derivations.

3.2 Calibration Mapping

The confidence values $\hat{p}$ are obtained via a calibration function $\phi : \mathbb{R} \to [0, 1]$ such that $\hat{p} = \phi(s(x))$, where $s(x)$ is the raw CSF output. The smoothness properties of ϕ directly affect vulnerability to the ECE_{L_2} variance penalty.

Parametric mappers such as Sigmoid (Platt) scaling (Platt et al., 1999) and Beta calibration (Kull et al., 2017) enforce smooth, continuous transitions in probability space, which limits the magnitude of confidence jumps between nearby score values. In contrast, non-parametric mappers such as Isotonic Regression (Fielding, 1974) fit strictly non-decreasing piecewise constant functions. Under in-distribution conditions, Isotonic Regression achieves near-zero empirical error by fitting its step functions closely to the validation distribution. However, this close fit creates brittleness under distribution shift: if an OOD sample's raw score falls slightly above a learned threshold, the step function maps it directly to high confidence ($\hat{p} \approx 1$), producing a silent failure. Because this failure involves a single large deviation rather than many small ones, it is penalized heavily by the ECE_{L_2} bound but can be averaged away under ECE_{L_1}.

4 Experimental Setup

We adopt the FD-Shifts protocol (Jaeger et al., 2022), scaling it to evaluate calibration bounds.[1]

Models & Datasets. To ensure our results are not artifacts of a single architecture or training method, we factorially vary the backbone. For Convolutional Neural Networks (CNNs), we train VGG-13 models from scratch under three training paradigms: ConfidNet (Corbière et al., 2019), DeVries (DeVries and Taylor, 2018), and Deep Gamblers (Liu et al., 2019). For Vision Transformers (ViTs), we fine-tune a pre-trained ViT on each ID dataset. We train five independent model initializations per configuration (varying random seeds) to ensure statistical stability. Our ID datasets are CIFAR-10, CIFAR-100, SuperCIFAR-100, and TinyImageNet.

Shift Stratification. We stratify OOD datasets into near, mid, and far shifts based on their semantic distance to the ID distribution. We compute four complementary CLIP-based distance metrics: two global measures (Fréchet distance and MMD on L2-normalized CLIP image embeddings) and two class-conditional measures (image-to-centroid angular distance and image-to-text cosine similarity via prompt ensembling). All metrics are oriented so that lower

[1] Code is available at https://github.com/cesar-claros/ood_systematic

Table 1: OOD dataset stratification. For each source dataset, OOD datasets are assigned to near, mid, and far tiers via K-means clustering on CLIP-derived distances. See Appendix B for details.

Source	Near	Mid	Far
CIFAR-10	CIFAR-100, TinyImagenet	iSUN, LSUN(r), LSUN(c), SVHN	Places365, Textures
SuperCIFAR-100	CIFAR-10, TinyImagenet	iSUN, LSUN(r), LSUN(c), SVHN	Places365, Textures
CIFAR-100	CIFAR-10, TinyImagenet	iSUN, LSUN(r), LSUN(c), SVHN	Places365, Textures
Tiny-Imagenet	CIFAR-10, CIFAR-100, iSUN, LSUN(r), LSUN(c)	Places365, Textures	SVHN

values indicate closer distributions. We then apply K-means clustering (K=3) over the four-dimensional distance vectors to assign each OOD dataset to a proximity tier. Table 1 summarizes the resulting stratification; see Appendix B for the full distance values and clustering details.

Confidence Scoring Functions. We evaluate 20+ confidence scoring functions (CSFs) spanning probabilistic scores (MSR (Hendrycks and Gimpel, 2016), GEN (Liu et al., 2023), Energy (Liu et al., 2020)), entropy-based scores (Predictive Entropy, Guessing Entropy, Rényi Entropy), gradient-based scores (GradNorm (Huang et al., 2021)), geometric and manifold methods (Maha (Lee et al., 2018), NNGuide (Park et al., 2023), fDBD (Liu and Qin, 2023), CTM (Ngoc-Hieu et al., 2023), PCA RecError (Guan et al., 2023), Residual/ViM (Wang et al., 2022), pNML (Bibas et al., 2021)), and learned confidences from ConfidNet, DeVries, and Deep Gamblers. See Appendix C for the full list and definitions.

Out-of-Fold Dynamic Calibration Selection. Rather than fixing a single calibration technique, we dynamically select between Isotonic Regression, Sigmoid Scaling, and Beta Calibration. To prevent the high-capacity Isotonic mapper from minimizing the bounds by overfitting to the validation distribution, we compute selection bounds using Out-of-Fold (OOF) probabilities generated via 5-fold cross-validation on the ID validation set. This ensures that the selected mapper is evaluated on held-out data, so that the resulting bounds reflect generalization rather than in-sample fit. See Appendix D for details.

Statistical Pipeline. Because calibration bound values exhibit non-constant variance across methods and datasets, we employ a rank-based statistical pipeline. Specifically, we use the Friedman test (Demšar, 2006) with Conover (Conover, 1999)-Holm (Holm, 1979) post-hoc correction to obtain adjusted pairwise p-values. From these, we apply the Bron-Kerbosch algorithm (Bron and Kerbosch, 1973) to enumerate maximal cliques in the non-significance graph: each clique defines a set of methods that are mutually statistically indistinguishable at the 0.05 level. The *top clique* for a given evaluation regime is the maximal clique containing the best-ranked method, representing the set of methods with equivalent worst-case safety performance.

5 Results and Discussion

5.1 Empirical ECE vs. Theoretical Bounds: Statistical Clique Analysis

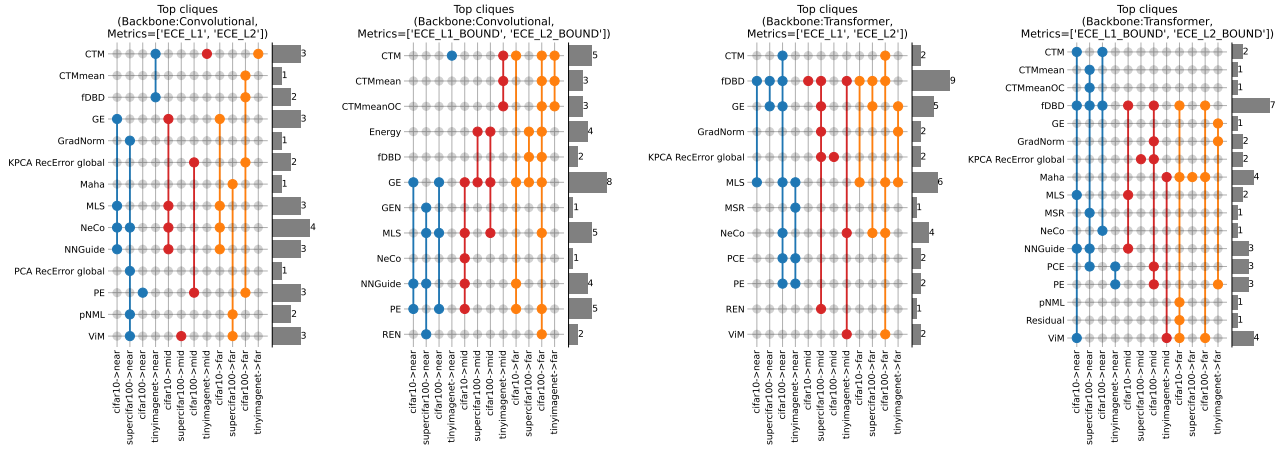
Our central result is a clear divergence between method rankings under empirical ECE and under the theoretical upper bounds. As shown in Figures 1a and 1c, evaluating CSFs using standard binned ECE_{L_1} and ECE_{L_2} yields dense statistical cliques, suggesting that under average-case analysis, a wide variety of methods, from simple baselines (MSR) to complex subspace projections (PCA), appear equivalently optimal.

Evaluation via the theoretical upper bounds (Figures 1b and 1d) systematically contracts these cliques. The L_2 variance term penalizes high-magnitude calibration deviations, filtering out methods that tolerate rare but high-impact OOD overconfidence events. This stricter criterion reveals a pronounced architecture-dependent divergence in worst-case safety behavior, analyzed below.

5.1.1 Convolutional Networks

For CNNs trained from scratch (VGG-13), entropy-based scores and unnormalized logit margins provide the tightest safety guarantees. Guessing Entropy (GE), Predictive Entropy (PE), and Maximum Logit Score (MLS) consistently anchor the top cliques across near, mid, and far shifts (Figure 1b).

We attribute this pattern to the interaction between Neural Collapse (NC) (Papyan et al., 2020) geometry and the L_2 variance penalty. NC induces strong class-wise feature concentration in CNNs trained from scratch, producing well-separated logit distributions. Under this geometry, entropy-based scores aggregate information across all class logits, making them robust to isolated logit perturbations that could produce overconfident outputs. MLS, while relying only on the top logit, benefits from NC's logit-scale separation, which maintains large margins between the predicted and runner-up classes. In contrast, scores that



(a) Convolutional, ECE (b) Convolutional, Bounds (c) ViT, ECE (d) ViT, Bounds

Figure 1: Top-clique maps for empirical ECE and ECE upper bounds. Columns represent evaluation regimes defined by source dataset (CIFAR-10, CIFAR-100, SuperCIFAR-100, TinyImageNet) and shift severity (near, mid, far). Rows represent the evaluated CSFs. Connected markers within a column denote the Conover-Holm top clique ($\alpha = 0.05$): the set of methods mutually statistically indistinguishable from the best-ranked score. Bar charts on the right aggregate total top-clique appearances per method. Under empirical ECE (1a, 1c), top cliques are dense. Under the theoretical bounds (1b, 1d), the L_2 variance penalty prunes the cliques, differentiating methods that the average-case analysis treats as equivalent.

operate on softmax probabilities (e.g., MSR) compress these margins through normalization, making them more susceptible to overconfidence on OOD samples whose logits happen to produce a peaked softmax distribution.

5.1.2 Vision Transformers

A different pattern emerges for fine-tuned Vision Transformers. Under worst-case bound evaluation (Figure 1d), probabilistic scores (MLS, GE) and prototype-matching methods (CTM) lose their top-clique memberships relative to the empirical ECE evaluation. The fast Decision Boundary Detector (fDBD) emerges as the most consistent method, appearing in 7 of 12 top cliques across shift regimes.

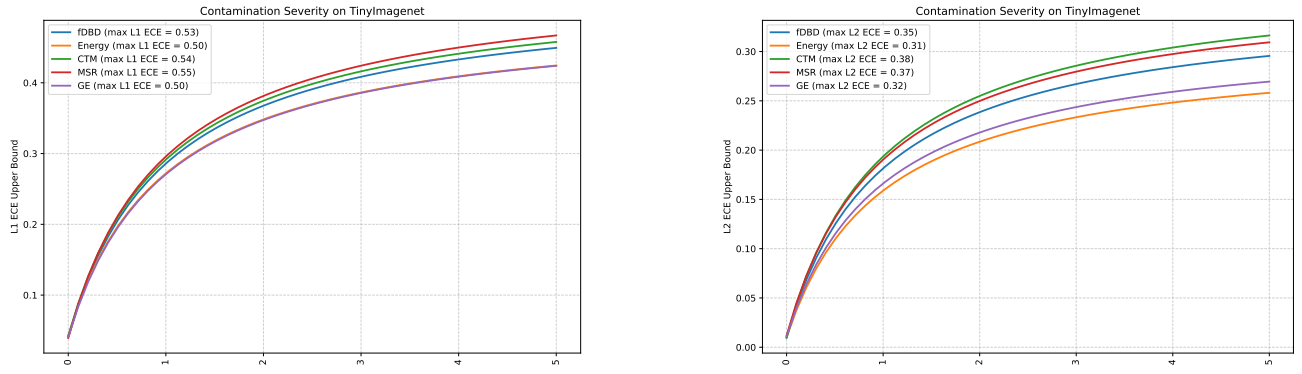
This result is consistent with evidence that fine-tuned ViTs do not exhibit feature collapse to the same extent as CNNs trained from scratch (Zhou et al., 2025). The high-dimensional feature manifold inherited from ImageNet pretraining contains directions that are not aligned with the downstream ID classes. OOD inputs can project onto these residual directions and produce confident softmax responses, a failure mode that entropy and logit-based scores cannot detect because they only observe the model's output distribution. fDBD addresses this by combining boundary-distance estimation in weight space with regularization by feature deviation from the ID mean, directly measuring whether a sample's representation lies in a well-separated region of feature space rather than relying on the model's output confidence.

5.2 The Contamination Ratio (α) and Deployment Safety

In deployment, the frequency of OOD encounters is typically unknown. To assess how the safety envelope degrades under increasing contamination, we examine the L_1 and L_2 ECE upper bounds as α increases from 0 (pure ID) to 5 (OOD-dominated). Figure 2 presents this analysis for five representative CSFs (Energy, GE, fDBD, CTM, MSR), computed for CIFAR-10 models trained under ConfidNet and evaluated on TinyImageNet. While this is a single configuration, it illustrates how the two bounds respond differently to contamination pressure.

The L_1 vs. L_2 Divergence. Under the L_1 bound (Figure 2a), Energy and Guessing Entropy (GE) appear to provide nearly indistinguishable safety guarantees, both reaching a maximum mean bound of 0.50 as α grows. The L_2 bound (Figure 2b) separates them: Energy attains a lower asymptotic bound (0.31) than GE (0.32), indicating that GE is more susceptible to individual high-magnitude confidence deviations that are amplified by the quadratic penalty.

Method Reordering under Variance Penalty. The transition from L_1 to L_2 also changes the relative ordering of weaker methods. Under L_1, MSR produces



(a) (b)

Figure 2: Contamination-ratio sensitivity analysis ($\alpha \in [0, 5]$) for CIFAR-10 models trained with ConfidNet and evaluated on TinyImageNet OOD data. Curves report mean calibration upper bounds for five confidence scoring functions (Energy, GE, fDBD, CTM, MSR): (2a) L_1 bound; (2b) L_2 bound.

the highest (worst) bound. Under L_2, CTM overtakes MSR as the weakest method (asymptotic bound of 0.38 vs. 0.37 for MSR), indicating that CTM, despite competitive average-case separability, is more vulnerable to rare high-confidence errors on OOD samples. Conversely, fDBD remains stable across both norms and outperforms both MSR and CTM under L_2. These reorderings demonstrate that the L_2 bound surfaces failure modes that L_1 averaging obscures, providing a more informative basis for method selection in safety-sensitive applications.

6 Conclusion

We have shown that standard empirical ECE, due to its average-based binning, is insensitive to rare high-confidence errors on OOD inputs. By deriving L_1 and L_2 ECE upper bounds parameterized by the contamination ratio α and evaluating them with out-of-fold calibration selection, we provide a worst-case safety assessment that reveals failure modes hidden by average-case analysis. The L_2 bound is particularly informative: its quadratic penalty on individual deviations contracts the statistical top cliques and reorders method rankings relative to both empirical ECE and the L_1 bound.

Our results establish a clear architecture-dependent prescription for calibration safety. For convolutional networks trained from scratch, where Neural Collapse produces concentrated feature geometries, entropy-based scores (GE, PE) and logit-margin scores (MLS) provide the tightest bounds. For fine-tuned Vision Transformers, where residual pretraining directions create vulnerability to output-space overconfidence, boundary-distance methods (fDBD) are required.

Limitations. The derived bounds are upper bounds and may be loose when the refinement term is large; tighter bounds that account for bin-level label variance remain an open direction. Our contamination-ratio analysis (Section 5.2) is based on a single dataset and training paradigm; while the clique analysis (Section 5.1) covers the full experimental grid, extending the α-sensitivity analysis across all configurations would strengthen the generality of the findings. Additionally, by restricting the ID set to correctly classified samples, we isolate calibration from classification accuracy but do not address the joint problem of misclassification and OOD detection.

Future work. Extending these bounds to the joint ID misclassification and OOD detection setting, and investigating how training-time interventions (e.g., mixup, outlier exposure) interact with worst-case calibration guarantees, are natural next steps.

Computing Infrastructure

All experiments were executed on our internal GPU cluster. *CNN* runs (VGG-13 trained from scratch) were scheduled on NVIDIA T4 GPUs, while *ViT* runs (fine-tuned from a large pretrained model) were scheduled on NVIDIA A100 GPUs. We did not mix GPU types within an experiment. Every training/evaluation job for a given backbone used the same GPU class to avoid hardware-induced variance.

References

Ammar, M. B., Belkhir, N., Popescu, S., Manzanera, A., and Franchi, G. (2023). Neco: Neural collapse based out-of-distribution detection. *arXiv preprint arXiv:2310.06823*.

Bibas, K., Feder, M., and Hassner, T. (2021). Single layer predictive normalized maximum likelihood for

out-of-distribution detection. *Advances in Neural Information Processing Systems*, 34:1179–1191.

Blattenberger, G. and Lad, F. (1985). Separating the brier score into calibration and refinement components: A graphical exposition. *The American Statistician*, 39(1):26–32.

Bron, C. and Kerbosch, J. (1973). Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577.

Conover, W. J. (1999). *Practical nonparametric statistics*. john wiley & sons.

Corbière, C., Thome, N., Bar-Hen, A., Cord, M., and Pérez, P. (2019). Addressing failure prediction by learning model confidence. *Advances in neural information processing systems*, 32.

Corbiere, C., Thome, N., Saporta, A., Vu, T.-H., Cord, M., and Perez, P. (2021). Confidence estimation via auxiliary models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6043–6055.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan):1–30.

DeVries, T. and Taylor, G. W. (2018). Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*.

Dowson, D. and Landau, B. (1982). The fréchet distance between multivariate normal distributions. *Journal of multivariate analysis*, 12(3):450–455.

Fang, K., Tao, Q., He, M., Lv, K., Yang, R., Hu, H., Huang, X., Yang, J., and Cao, L. (2025). Kernel pca for out-of-distribution detection: Non-linear kernel selections and approximations. *arXiv preprint arXiv:2505.15284*.

Fielding, A. (1974). Statistical inference under order restrictions. the theory and application of isotonic regression.

Fréchet, M. (1957). Sur la distance de deux lois de probabilité. In *Annales de l'ISUP*, volume 6, pages 183–198.

Granese, F., Romanelli, M., Gorla, D., Palamidessi, C., and Piantanida, P. (2021). Doctor: A simple method for detecting misclassification errors. *Advances in Neural Information Processing Systems*, 34:5669–5681.

Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., and Smola, A. (2006). A kernel method for the two-sample-problem. *Advances in neural information processing systems*, 19.

Guan, X., Liu, Z., Zheng, W.-S., Zhou, Y., and Wang, R. (2023). Revisit pca-based technique for out-of-distribution detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 19431–19439.

Hendrycks, D., Basart, S., Mazeika, M., Zou, A., Kwon, J., Mostajabi, M., Steinhardt, J., and Song, D. (2019). Scaling out-of-distribution detection for real-world settings. *arXiv preprint arXiv:1911.11132*.

Hendrycks, D. and Gimpel, K. (2016). A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics*, pages 65–70.

Huang, R., Geng, A., and Li, Y. (2021). On the importance of gradients for detecting distributional shifts in the wild. *Advances in Neural Information Processing Systems*, 34:677–689.

Jaeger, P. F., Lüth, C. T., Klein, L., and Bungert, T. J. (2022). A call to reflect on evaluation practices for failure detection in image classification. *arXiv preprint arXiv:2211.15259*.

Kull, M., Silva Filho, T., and Flach, P. (2017). Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Artificial intelligence and statistics*, pages 623–631. PMLR.

Lee, K., Lee, K., Lee, H., and Shin, J. (2018). A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31.

Liu, L. and Qin, Y. (2023). Fast decision boundary based out-of-distribution detector. *arXiv preprint arXiv:2312.11536*.

Liu, W., Wang, X., Owens, J., and Li, Y. (2020). Energy-based out-of-distribution detection. *Advances in neural information processing systems*, 33:21464–21475.

Liu, X., Lochman, Y., and Zach, C. (2023). Gen: Pushing the limits of softmax-based out-of-distribution detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23946–23955.

Liu, Z., Wang, Z., Liang, P. P., Salakhutdinov, R. R., Morency, L.-P., and Ueda, M. (2019). Deep gamblers: Learning to abstain with portfolio theory. *Advances in Neural Information Processing Systems*, 32.

Massey, J. L. (1994). Guessing and entropy. In *Proceedings of 1994 IEEE International Symposium on Information Theory*, page 204. IEEE.

Ngoc-Hieu, N., Hung-Quang, N., Ta, T.-A., Nguyen-Tang, T., Doan, K. D., and Thanh-Tung, H. (2023). A cosine similarity-based method for out-of-distribution detection. *arXiv preprint arXiv:2306.14920*.

Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J., Lakshminarayanan, B., and Snoek, J. (2019). Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32.

Papyan, V., Han, X., and Donoho, D. L. (2020). Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663.

Park, J., Jung, Y. G., and Teoh, A. B. J. (2023). Nearest neighbor guidance for out-of-distribution detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1686–1695.

Pavlovic, M. (2025). Understanding model calibration–a gentle introduction and visual exploration of calibration and the expected calibration error (ece). *arXiv preprint arXiv:2501.19047*.

Platt, J. et al. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Rényi, A. (1961). On measures of entropy and information. In *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability, volume 1: contributions to the theory of statistics*, volume 4, pages 547–562. University of California Press.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.

Traub, J., Bungert, T. J., Lüth, C. T., Baumgartner, M., Maier-Hein, K. H., Maier-Hein, L., and Jaeger, P. F. (2024). Overcoming common flaws in the evaluation of selective classification systems. *arXiv preprint arXiv:2407.01032*.

Wang, H., Li, Z., Feng, L., and Zhang, W. (2022). Vim: Out-of-distribution with virtual-logit matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4921–4930.

Yang, J., Wang, P., Zou, D., Zhou, Z., Ding, K., Peng, W., Wang, H., Chen, G., Li, B., Sun, Y., et al. (2022). Openood: Benchmarking generalized out-of-distribution detection. *Advances in Neural Information Processing Systems*, 35:32598–32611.

Zhang, J., Yang, J., Wang, P., Wang, H., Lin, Y., Zhang, H., Sun, Y., Du, X., Li, Y., Liu, Z., et al. (2023). Openood v1.5: Enhanced benchmark for out-of-distribution detection. *arXiv preprint arXiv:2306.09301*.

Zhou, J., Jiang, J., and Zhu, Z. (2025). Are all layers created equal: A neural collapse perspective. In *The Second Conference on Parsimony and Learning (Proceedings Track)*.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Yes]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Bounding Worst-Case Calibration Error in OOD Detection Under Distribution Shift: Supplementary Materials

A Derivation of the Calibration Upper Bounds Under Shift

In this section, we provide the full mathematical derivations for the L_1 and L_2 Expected Calibration Error (ECE) upper bounds parameterized by the contamination ratio α.

A.1 Definitions

We evaluate calibration in a mixed deployment environment $\mathcal{D}_{\text{mix}}$ containing two disjoint subsets:

1. $\mathcal{D}_{\text{ID}}$: The set of correctly classified in-distribution (ID) samples (Hits), with cardinality N_{ID}. For a perfectly safe system, the target probability for these samples is $y = 1$.
2. $\mathcal{D}_{\text{OOD}}$: The set of out-of-distribution samples, with cardinality N_{OOD}. The target probability for these samples is $y = 0$, representing a complete rejection by the confidence scoring function.

The total number of evaluated samples is $N = N_{\text{ID}} + N_{\text{OOD}}$. We parameterize the semantic shift severity using the contamination ratio $\alpha = \frac{N_{\text{OOD}}}{N_{\text{ID}}}$. The mixture proportions can thus be expressed strictly in terms of α:

$$\frac{N_{\text{ID}}}{N} = \frac{N_{\text{ID}}}{N_{\text{ID}} + N_{\text{OOD}}} = \frac{1}{1+\alpha}, \qquad \frac{N_{\text{OOD}}}{N} = \frac{N_{\text{OOD}}}{N_{\text{ID}} + N_{\text{OOD}}} = \frac{\alpha}{1+\alpha} \tag{3}$$

Let $\hat{p}_i \in [0, 1]$ be the mapped probability output of an OOD detection method for sample i.

A.2 The Unbinned Error as a Strict Upper Bound

Standard empirical Expected Calibration Error (ECE) relies on partitioning N predictions into M equally spaced bins $B_1, \ldots, B_M$. The L_1 ECE is defined as:

$$ECE_{L_1} = \sum_{m=1}^{M} \frac{|B_m|}{N} \left|\text{acc}(B_m) - \text{conf}(B_m)\right| \tag{4}$$

where $\text{acc}(B_m)$ is the average accuracy (target y) and $\text{conf}(B_m)$ is the average predicted probability in bin m. By the triangle inequality, the absolute difference of the averages within a bin is strictly less than or equal to the average of the absolute differences:

$$\left|\text{acc}(B_m) - \text{conf}(B_m)\right| = \left|\frac{1}{|B_m|} \sum_{i \in B_m} (y_i - \hat{p}_i)\right| \leq \frac{1}{|B_m|} \sum_{i \in B_m} |y_i - \hat{p}_i| \tag{5}$$

Substituting this inequality back into the ECE formulation yields the unbinned absolute error, which serves as a strict mathematical upper bound for the empirical ECE_{L_1} regardless of the binning scheme:

$$ECE_{L_1} \leq \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{p}_i| \tag{6}$$

This unbinned formulation allows us to evaluate the absolute worst-case calibration safety of the scoring function.

A.3 Derivation of the L_1 ECE Bound

We decompose the unbinned upper bound into its constituent parts over the mixed deployment set $\mathcal{D}_{\text{mix}} = \mathcal{D}_{\text{ID}} \cup \mathcal{D}_{\text{OOD}}$:

$$ECE_{L_1} \leq \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{p}_i| = \frac{1}{N}\left(\sum_{i\in\mathcal{D}_{\text{ID}}}|1-\hat{p}_i| + \sum_{j\in\mathcal{D}_{\text{OOD}}}|0-\hat{p}_j|\right) \tag{7}$$

Because the calibrated probabilities $\hat{p}$ are strictly bounded in $[0,1]$, we can drop the absolute value operators: $|1-\hat{p}_i| = 1-\hat{p}_i$ and $|0-\hat{p}_j| = \hat{p}_j$. We multiply and divide each summation by its respective subset cardinality to recover the mean formulations:

$$ECE_{L_1} \leq \frac{N_{\text{ID}}}{N}\left[\frac{1}{N_{\text{ID}}}\sum_{i\in\mathcal{D}_{\text{ID}}}(1-\hat{p}_i)\right] + \frac{N_{\text{OOD}}}{N}\left[\frac{1}{N_{\text{OOD}}}\sum_{j\in\mathcal{D}_{\text{OOD}}}\hat{p}_j\right] \tag{8}$$

Substituting the baseline ID uncertainty (K_1) and the mean OOD confidence ($\overline{p}_{\text{OOD}}$), we obtain:

$$ECE_{L_1} \leq \frac{N_{\text{ID}}}{N}K_1 + \frac{N_{\text{OOD}}}{N}\overline{p}_{\text{OOD}} \tag{9}$$

Finally, substituting the contamination ratio fractions yields the final parameterized L_1 bound:

$$ECE_{L_1} \leq \frac{1}{1+\alpha}K_1 + \frac{\alpha}{1+\alpha}\overline{p}_{\text{OOD}} \tag{10}$$

A.4 Derivation of the L_2 ECE Bound (RMSCE)

The L_1 bound averages out individual prediction variance. To explicitly penalize bimodal hallucination spikes, we compute the bound for the Root Mean Square Calibration Error (RMSCE), or ECE_{L_2}. Following the same logic using Jensen's inequality for convex functions, the unbinned squared error strictly bounds the binned squared error. Let the squared baseline uncertainty be $K_2 = \frac{1}{N_{\text{ID}}}\sum_{i\in\mathcal{D}_{\text{ID}}}(1-\hat{p}_i)^2$ and the squared mean OOD confidence be $\overline{p^2}_{\text{OOD}} = \frac{1}{N_{\text{OOD}}}\sum_{j\in\mathcal{D}_{\text{OOD}}}\hat{p}_j^2$. Starting from the Brier Score (BS) over the mixture:

$$\text{BS} = \frac{1}{N}\sum_{i=1}^{N}(y_i-\hat{p}_i)^2 = \frac{1}{N}\left(\sum_{i\in\mathcal{D}_{ID}}(1-\hat{p}_i)^2 + \sum_{j\in\mathcal{D}_{OOD}}(0-\hat{p}_j)^2\right) \tag{11}$$

$$= \frac{N_{\text{ID}}}{N}\left[\frac{1}{N_{\text{ID}}}\sum_{i\in\mathcal{D}_{ID}}(1-\hat{p}_i)^2\right] + \frac{N_{\text{OOD}}}{N}\left[\frac{1}{N_{\text{OOD}}}\sum_{j\in\mathcal{D}_{OOD}}\hat{p}_j^2\right] \tag{12}$$

$$= \frac{N_{\text{ID}}}{N}K_2 + \frac{N_{\text{OOD}}}{N}\overline{p^2}_{\text{OOD}} \tag{13}$$

Substituting the contamination ratio parameterization gives:

$$\text{BS} = \frac{1}{1+\alpha}K_2 + \frac{\alpha}{1+\alpha}\overline{p^2}_{\text{OOD}} \tag{14}$$

On the other hand, the Brier score decomposes into two additive components (Blattenberger and Lad, 1985): BS = CAL + REF. Given a partitioning of the data into M bins $B_1, \ldots, B_M$, the Calibration component is defined as the mean squared difference between the empirical accuracy and average confidence within each bin. This is, by definition, the squared L_2 Expected Calibration Error:

$$\text{CAL} = \sum_{m=1}^{M}\frac{|B_m|}{N}(\text{acc}(B_m) - \text{conf}(B_m))^2 = ECE_{L_2}^2 \tag{15}$$

The Refinement component (often formulated as the base uncertainty minus resolution) measures the inherent variance of the true labels within each probability bin:

$$\mathrm{REF} = \sum_{m=1}^{M} \frac{|B_m|}{N} \mathrm{acc}(B_m)(1 - \mathrm{acc}(B_m)) \tag{16}$$

Because the empirical accuracy $\mathrm{acc}(B_m)$ is a proportion in $[0, 1]$, the Refinement term is strictly non-negative ($\mathrm{REF} \geq 0$). Consequently, the Calibration term is strictly bounded from above by the total Brier Score:

$$ECE_{L_2}^2 \leq BS \tag{17}$$

Taking the square root of both sides and substituting our α-parameterized expression for the total Brier Score of the mixture perfectly recovers our derived L_2 upper bound:

$$ECE_{L_2} \leq \sqrt{BS} = \sqrt{\frac{1}{1+\alpha} K_2 + \frac{\alpha}{1+\alpha} \overline{p^2}_{\mathrm{OOD}}} \tag{18}$$

This decomposition provides theoretical insight into when the empirical ECE_{L_2} approaches our worst-case theoretical bound. The ECE_{L_2} exactly equals the bound when the Refinement term drops to zero ($\mathrm{REF} = 0$). Mathematically, $\mathrm{REF} = 0$ requires that $\mathrm{acc}(B_m) \in \{0, 1\}$ for all bins, meaning there is zero label variance within any predicted probability bin.

In the context of OOD detection, this scenario occurs when an unstable calibrator maps OOD samples (target $y = 0$) to extreme probabilities near 1.0, placing them into bins alongside correctly classified ID hits (target $y = 1$) and effectively maximizing the calibration penalty. Thus, bounding the ECE_{L_2} via the Brier Score provides a strict, mathematically guaranteed safety envelope against high-variance calibration failures.

B CLIP-based OOD Aggregation

We quantify distributional proximity to an OOD image dataset using the feature space from a CLIP model Radford et al. (2021). Concretely, we extract L2-normalized image embeddings for both ID and candidate OOD sets using a fixed CLIP encoder under identical preprocessing. We then compute two label-agnostic distances between the empirical feature distributions: Fréchet distance (FD) (Dowson and Landau, 1982; Fréchet, 1957) and Maximum Mean Discrepancy (MMD) (Gretton et al., 2006) with a polynomial kernel. Both are evaluated on CLIP embeddings, yielding global measures of how close the OOD distribution lies to the ID manifold; by construction, lower values indicate greater proximity. To capture class-aware proximity, we complement the global measures with two class-conditional distances. First, we represent each ID class by an image-embedding centroid and score a sample by its nearest-centroid angular distance in CLIP feature space. Second, we form text prototypes via prompt ensembling for each ID class (e.g., "a photo of a {class}", "a low-resolution photo of a {class}"), embed them with the CLIP text encoder, and compute the maximum image-text cosine similarity. For both class-conditional distances we compute the mean value across all classes to obtain a single metric representing distance to the ID manifold. All four metrics are oriented so that lower values indicate closer distributions. Finally, we apply K-means clustering (K=3) over the four-dimensional distance vectors to assign each OOD dataset to a near, mid, or far proximity tier. This protocol is model-agnostic with respect to the downstream OOD detector and applies unchanged to any choice of ID label space. Table 2 reports the four distance metrics and the resulting tier assignment for each source-OOD dataset pair.

C Confidence Scoring Functions

This section describes the Confidence Scoring Functions (CSF) evaluated in this work.

C.1 Class Typical Matching (CTM) and Class Typical Matching with class means prototypes (CTMmean) (Ngoc-Hieu et al., 2023)

Prototype matching in feature space consists of quantifying the similarity between a sample $\boldsymbol{x}$ and the last-layer trained weights $\{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_K\}$. Therefore the similarity to the closest trained weight is $\mathrm{CTM}(\boldsymbol{x}) =$

Table 2: CLIP-based distance metrics and tier assignments for each source dataset. Two global measures (Kernel MMD, Fréchet Distance) and two class-conditional measures (Label-Text Alignment, Image Centroid Distance) are computed on L2-normalized CLIP embeddings. Lower values indicate closer distributions. The "Group" column shows the tier assigned by K-means clustering (K=3) over the four-dimensional distance vectors. Cells are colored on a blue-to-red scale per column (blue = close, red = far).

(a) Source dataset: CIFAR-10

	Global		Class-aware		Group
	Kernel MMD	FD	Label-Text Alignment	Image Centroid Distance	
Test	-0.0000	0.0028	0.7183	0.6349	ID
CIFAR-100	0.0002	0.1592	0.7885	0.8085	Near
TinyImagenet	0.0009	0.3233	0.8060	0.9256	Near
iSUN	0.0015	0.4890	0.7943	0.8393	Mid
LSUN resize	0.0016	0.5248	0.8045	0.8634	Mid
LSUN cropped	0.0015	0.5129	0.7797	0.8168	Mid
SVHN	0.0020	0.7009	0.7744	0.8607	Mid
Places 365	0.0021	0.6379	0.8337	1.1471	Far
Textures	0.0020	0.6698	0.8231	1.0647	Far

(b) Source dataset: SuperCIFAR-100

	Global		Class-aware		Group
	Kernel MMD	FD	Label-Text Alignment	Image Centroid Distance	
Test	0.0000	0.0748	0.7581	0.7031	ID
CIFAR-10	0.0002	0.1705	0.7701	0.7511	Near
TinyImagenet	0.0008	0.2307	0.7840	0.8738	Near
iSUN	0.0012	0.3856	0.7607	0.7425	Mid
LSUN resize	0.0013	0.4244	0.7625	0.7720	Mid
LSUN cropped	0.0011	0.3999	0.7696	0.7351	Mid
SVHN	0.0017	0.6208	0.7566	0.8012	Mid
Places 365	0.0020	0.5562	0.7939	1.0964	Far
Textures	0.0016	0.5246	0.7980	1.0003	Far

(c) Source dataset: CIFAR-100

	Global		Class-aware		Group
	Kernel MMD	FD	Label-Text Alignment	Image Centroid Distance	
Test	-0.0000	0.0033	0.7043	0.6026	ID
CIFAR-10	0.0002	0.1590	0.7494	0.7268	Near
TinyImagenet	0.0008	0.2235	0.7512	0.8436	Near
iSUN	0.0012	0.3829	0.7484	0.7128	Mid
LSUN resize	0.0013	0.4204	0.7562	0.7388	Mid
LSUN cropped	0.0011	0.3999	0.7364	0.7120	Mid
SVHN	0.0017	0.6222	0.7511	0.7789	Mid
Places 365	0.0019	0.5456	0.7752	1.0568	Far
Textures	0.0016	0.5232	0.7613	0.9780	Far

(d) Source dataset: TinyImagenet

	Global		Class-aware		Group
	Kernel MMD	FD	Label-Text Alignment	Image Centroid Distance	
Test	-0.0000	0.0036	0.7141	0.6319	ID
CIFAR-100	0.0008	0.2224	0.7279	0.7956	Near
CIFAR-10	0.0009	0.3220	0.7288	0.7979	Near
iSUN	0.0012	0.3808	0.7468	0.7063	Near
LSUN resize	0.0013	0.4039	0.7500	0.7186	Near
LSUN cropped	0.0016	0.4989	0.7406	0.7503	Near
Places 365	0.0014	0.3887	0.7645	0.9846	Mid
Textures	0.0014	0.4697	0.7528	0.9317	Mid
SVHN	0.0025	0.7948	0.7409	0.8726	Far

$\max_{k\le C}\ \mathrm{sim}\big(\boldsymbol{w}_k, \boldsymbol{h}\big)$. Alternatively, we can compute class means $\boldsymbol{\mu}^c_{\text{train}}$ and score by similarity to the closest class mean, $\text{CTMmean}(\boldsymbol{x}) = \max_{k\le C}\ \mathrm{sim}\big(\boldsymbol{\mu}^k_{\text{train}}, \boldsymbol{h}\big)$. Following Ngoc-Hieu et al. (2023), we use cosine similarity in this work, where $\mathrm{sim}(\boldsymbol{u}, \boldsymbol{v}) = \frac{\boldsymbol{u}^\top \boldsymbol{v}}{\|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_2}$. CTM scores the *typicality* of $\boldsymbol{x}$ by comparing its feature $\boldsymbol{h}$ against a bank of class representatives. Higher similarity indicates greater in-distribution (ID) conformity.

C.2 Energy (Liu et al., 2020)

The energy score is defined as $\text{Energy}(\boldsymbol{x}) = -T \log \sum_{k=1}^{C} \exp\big(g(\boldsymbol{h})_k / T\big)$, with temperature $T > 0$. Higher Energy score typically indicates higher uncertainty.

C.3 Maximum Softmax Response (MSR) (Hendrycks and Gimpel, 2016) and Maximum Logit Score (MLS) (Hendrycks et al., 2019)

A baseline confidence score given by the maximum predicted probability $\text{MSR}(\boldsymbol{x}) = \max_{k\le C} \boldsymbol{p}_k$, widely used for OOD detection. Lower values indicate atypical inputs. Similarly, MLS is a confidence score measured in the logit space, $\text{MLS}(\boldsymbol{x}) = \max_{k\le C} g(\boldsymbol{h})_k$, often more stable than softmax under temperature changes.

C.4 Predictive Entropy (PE), Generalized Entropy (GEN), Renyi Entropy (REN), Guessing Entropy (GE), and Predictive Collision Entropy (PCE)

Predictive Entropy (PE) (Ovadia et al., 2019). Uncertainty via Shannon entropy (Shannon, 1948) of the predictive distribution $\text{PE}(\boldsymbol{x}) = H\big(p(\boldsymbol{x})\big) = -\sum_{k=1}^{C} \boldsymbol{p}_k \log \boldsymbol{p}_k$, with larger entropy signaling higher uncertainty.

Generalized Entropy (GEN) (Liu et al., 2023). GEN is a post-hoc OOD score that uses the softmax probabilities of a trained classifier. Let $\boldsymbol{p}_{(1)} \ge \cdots \ge \boldsymbol{p}_{(K)}$ denote the probabilities sorted in descending order for a given input $\boldsymbol{x}$. For sensitivity and numerical stability in many-class settings, GEN truncates to the top-M classes and computes a generalized entropy with exponent $\gamma \in (0, 1)$: $\text{GEN}(\boldsymbol{x}) = \sum_{k=1}^{M} \boldsymbol{p}^{\gamma}_{(k)} \big(1 - \boldsymbol{p}_{(k)}\big)^{\gamma}$. The confidence score is the *negated* generalized entropy so that a larger GEN (lower entropy) indicates more ID-like predictions.

Rényi Entropy (REN). The Rényi entropy (Rényi, 1961) of order α is a smooth generalization of Shannon entropy that emphasizes different parts of the distribution. Similar to GEN, REN is defined by a truncation paramater M and exponent $\alpha \in (0, 1)$: $\text{REN}(\boldsymbol{x}) = \frac{1}{1-\alpha} \log \sum_{k=1}^{M} \boldsymbol{p}^{\alpha}_{(k)}$.

Guessing Entropy (GE). GE (Massey, 1994) quantifies the expected number of guesses to identify the true class when labels are guessed in decreasing probability $p_k(\boldsymbol{x})$: if $p_{(1)} \ge \cdots \ge p_{(K)}$ are sorted, then $\text{GE}(\boldsymbol{x}) = \sum_{k=1}^{C} k\, \boldsymbol{p}_{(k)}$, with larger values denoting higher uncertainty.

Predictive Collision Entropy (PCE) (Granese et al., 2021). PCE measures prediction uncertainty via the *collision* (order-2 Rényi) entropy of the softmax distribution: $\text{PCE}(\boldsymbol{x}) = -\log \sum_{k=1}^{C} \boldsymbol{p}^2_k$. Since $\sum_k \boldsymbol{p}^2_k$ is the "collision probability," PCE grows as the distribution spreads (uncertain/atypical) and shrinks as it peaks (confident/ID-like). This uncertainty score uses the entire predictive distribution rather than just its maximum.

C.5 Mahalanobis Distance (Maha) (Lee et al., 2018)

Assuming Gaussian class-conditional features, score by the (negative) Mahalanobis distance to the nearest class centroid is $\text{Maha}(\boldsymbol{x}) = \max_{k\le C}\ \big(h(\boldsymbol{x}) - \boldsymbol{\mu}^k_{\text{train}}\big)^\top \boldsymbol{\Sigma}^{-1} \big(h(\boldsymbol{x}) - \boldsymbol{\mu}^k_{\text{train}}\big) = \overline{\boldsymbol{h}}_k^\top \boldsymbol{\Sigma}^{-1} \overline{\boldsymbol{h}}_k$, where $\boldsymbol{\Sigma}$ is the empirical covariance matrix.

C.6 Nearest Neighbor Guide (NNGuide) (Park et al., 2023)

NNGuide is a post-hoc wrapper that modulates any classifier-based OOD score $S_{\text{base}}(\boldsymbol{h})$ using nearest neighbors from an ID bank of features. This bank is formed by sampling $\alpha \in (0, 1)$ of ID training features $\boldsymbol{h}_i$ (L2-normalized) and their base scores $s_i = S_{\text{base}}(\boldsymbol{h}_i)$. More specifically, given an input $\boldsymbol{x}$, the corresponding feature

$\boldsymbol{h}$ (L2-normalized) defines a confidence-scaled similarity list $\{s_i \cos(\boldsymbol{h}, \boldsymbol{h}_i)\}_{i=1}^{N}$, which is sampled by taking the top-k terms, where $k = \lfloor \alpha N \rfloor$. The top-$k$ terms set the guidance $G(\boldsymbol{h}) = \frac{1}{k} \sum_{i \le k} s_i \cos(\boldsymbol{h}, \boldsymbol{h}_i)$, and the score $\text{NNGuide}(\boldsymbol{x}) = S_{\text{base}}(\boldsymbol{h}) \cdot G(\boldsymbol{h})$. In practice, S_{base} is the (negative) Energy score, but NNGuide can improve other classifier-based scores. Intuitively, $G(\boldsymbol{h})$ downscales overconfident far-OOD points where cosine similarities are small and preserves near-ID points using high-confidence neighbors.

C.7 fast Decision Boundary Detector (fDBD) (Liu and Qin, 2023)

fDBD scores a sample by how far its feature lies from the nearest class decision boundary, regularized by feature deviation from the in-distribution mean. For each non-predicted class $c \neq m(\boldsymbol{x})$, the (unknown) distance in the feature space to the c-boundary is lower bounded by $D_g(\boldsymbol{h}, k) = \frac{\left|(\boldsymbol{w}_m - \boldsymbol{w}_k)^\top \boldsymbol{h} + (b_m - b_k)\right|}{\| \boldsymbol{w}_m - \boldsymbol{w}_k \|_2}$, i.e., the Euclidean distance from $\boldsymbol{h}$ to the separating hyperplane between classes $m(\boldsymbol{x})$ and c. Averaging these distances and *regularizing* by the feature's deviation from the ID mean $\boldsymbol{\mu}_{\text{train}}$ yields the score $\text{fDBD}(\boldsymbol{x}) = \frac{1}{C-1} \sum_{k \neq m(\boldsymbol{x})}^{C} \frac{D_g(\boldsymbol{h}, k)}{\| \boldsymbol{h} - \boldsymbol{\mu}_{\text{train}} \|_2}$,.

The regularizer compares ID/OOD at equal deviation levels, empirically sharpening separation; the distance term captures that ID features tend to reside farther from decision boundaries than OOD features.

C.8 predictive Normalized Maximum Likelihood (pNML) (Bibas et al., 2021)

pNML treats a deep network as a fixed feature extractor and for each test samples computes a regret score by simulating in closed form the best last-layer update for every possible label. Given the matrix of normalized penultimate-layer training activations $\hat{\boldsymbol{H}} = [\boldsymbol{h}_1/\|\boldsymbol{h}_1\|_2, \ldots, \boldsymbol{h}_N/\|\boldsymbol{h}_N\|_2]$, the online–update direction $\boldsymbol{g}$ via the kernel–range projection is $\boldsymbol{g} = \boldsymbol{h}_\perp / \|\boldsymbol{h}_\perp\|_2^2$ if $\boldsymbol{h}_\perp = (\boldsymbol{I} - \hat{\boldsymbol{H}}^+ \hat{\boldsymbol{H}})\boldsymbol{h} \neq 0$; else $\boldsymbol{g} = \frac{\hat{\boldsymbol{H}}^+ \hat{\boldsymbol{H}}^{+\top} \boldsymbol{h}}{1 + \boldsymbol{h}^\top \hat{\boldsymbol{H}}^+ \hat{\boldsymbol{H}}^{+\top} \boldsymbol{h}}$, where $\hat{\boldsymbol{H}}^+$ is the Moore–Penrose inverse of the normalized training activations. The pNML regret is $\text{pNML}(\boldsymbol{x}) = \log \sum_{k=1}^{C} \frac{\boldsymbol{p}_k}{\boldsymbol{p}_k + \boldsymbol{p}_k^{\boldsymbol{h}^\top \boldsymbol{g}} (1 - \boldsymbol{p}_k)}$ and serves as an OOD/failure score (larger pNML $\Rightarrow$ less trustworthy prediction). Intuitively, pNML is small when $\boldsymbol{h}$ lies in the high-variance ID subspace or is far from decision boundaries (the genie's label-specific update has little effect), and large otherwise.

C.9 GradNorm (Huang et al., 2021)

Given a trained classifier with softmax $p(\boldsymbol{x})$, GradNorm defines the OOD score as the vector norm of the gradients obtained by backpropagating the Kullback–Leibler divergence from a uniform target, i.e. $\text{GradNorm}(\boldsymbol{x}) = \left\| \partial_w \text{KL}(\boldsymbol{u} \,\|\, p(\boldsymbol{x})) \right\|_p = \left\| \frac{1}{C} \sum_{k=1}^{C} \frac{\partial \mathcal{L}_{\text{CE}}(g(\boldsymbol{h}), k)}{\partial \boldsymbol{w}} \right\|_p$, typically using the L_1 norm on the *last-layer* weights. This choice is label-agnostic and exploits that in-distribution inputs produce more *peaked* predictions and thus larger gradients than OOD inputs. A simple analysis shows $\text{GradNorm}(\boldsymbol{x})$ factorizes into a feature-space term and an output-space term, capturing joint information that improves separability over output-only scores.

C.10 PCA Reconstruction Error (PCA RecError) (Guan et al., 2023)

PCA Reconstruction Error models the in-distribution feature manifold by fitting a low-dimensional principal subspace on penultimate features and scores a test example by the energy of its component orthogonal to that subspace, so larger residuals indicate atypicality. This approach computes the ID mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, then takes the top-k eigenvectors $\boldsymbol{U}_k$ of $\boldsymbol{\Sigma}$, and forms the projector $\boldsymbol{M} = \boldsymbol{U}_k \boldsymbol{U}_k^\top$. The *PCA reconstruction error* for a test point is $e(\boldsymbol{x}) = \left\| (\boldsymbol{I} - \boldsymbol{M}) \left(h(\boldsymbol{x}) - \boldsymbol{\mu} \right) \right\|_2$, i.e., the energy of the component orthogonal to the ID principal subspace. Although intuitively $e(\boldsymbol{x})$ should be smaller on ID than OOD, a detailed analysis shows that $e(\boldsymbol{x})$ (i) mixes the angle between $h(\boldsymbol{x}) - \boldsymbol{\mu}$ and the principal subspace and (ii) the norm $\|h(\boldsymbol{x}) - \boldsymbol{\mu}\|_2$, which is typically *larger* for ID than OOD; this blurs separability for vanilla PCA-OOD. To mitigate the norm effect, a simple regularized score $r(\boldsymbol{x}) = \frac{\|\boldsymbol{h} - \hat{\boldsymbol{h}}\|_2}{\|\boldsymbol{h}\|_2}$, where $\hat{\boldsymbol{h}} = \hat{h}(\boldsymbol{x}) = \boldsymbol{M}\left(h(\boldsymbol{x}) - \boldsymbol{\mu}\right) + \boldsymbol{\mu}$, improves discrimination, and can be fused multiplicatively with logit-based scores.

C.11 Kernel PCA Reconstruction Error (KPCA RecError) (Fang et al., 2025)

KPCA Reconstruction Error models the in-distribution (ID) feature manifold in a *non-linear* subspace and scores a test point by its reconstruction error in that subspace. To mitigate feature–norm imbalance and preserve useful Euclidean relations, KPCA first ℓ_2–normalizes features $\hat{\boldsymbol{h}} = \boldsymbol{h}/\|\boldsymbol{h}\|_2$ and define a Gaussian kernel on the unit sphere $k(\boldsymbol{x}, \boldsymbol{x}') = \exp\Big(-\frac{1}{2\sigma^2}\,\big\|\hat{\boldsymbol{h}} - \hat{\boldsymbol{h}}'\big\|_2^2\Big) = \exp\Big(-\frac{1}{\sigma^2}\,\big(1 - \cos(\hat{\boldsymbol{h}}, \hat{\boldsymbol{h}}')\big)\Big)$, which can be viewed as a Cosine–Gaussian composition. Given ID training points, the centered Gram matrix is defined as $\boldsymbol{K}_c = \boldsymbol{H}\boldsymbol{K}\boldsymbol{H}$ with $\boldsymbol{K}_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and $\boldsymbol{H} = \boldsymbol{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$. Using the centered Gram matrix, KPCA solves the eigenproblem $\boldsymbol{K}_c\boldsymbol{\alpha}_m = n\,\lambda_m\,\boldsymbol{\alpha}_m$, where $m = 1, \ldots, N$, and defines principal coordinates for a test point $\boldsymbol{x}$ via the centered kernel $k_c(\boldsymbol{x}, \boldsymbol{x}_i) = k(\boldsymbol{x}, \boldsymbol{x}_i) - \frac{1}{n}\sum_j k(\boldsymbol{x}, \boldsymbol{x}_j) - \frac{1}{n}\sum_j k(\boldsymbol{x}_j, \boldsymbol{x}_i) + \frac{1}{n^2}\sum_{j\ell} k(\boldsymbol{x}_j, \boldsymbol{x}_\ell)$: $\phi_m(\boldsymbol{x}) = \frac{1}{\sqrt{\lambda_m}}\sum_{i=1}^N \boldsymbol{\alpha}_{mi}\,k_c(\boldsymbol{x}, \boldsymbol{x}_i)$. The squared reconstruction error in feature space after projecting onto the top k components is $e(\boldsymbol{x}) = k_c(\boldsymbol{x}, \boldsymbol{x}) - \sum_{m=1}^k \phi_m(\boldsymbol{x})^2$. Similar to PCA Reconstruction Error, the larger $e(\boldsymbol{x})$ is, the more atypical $\boldsymbol{x}$ becomes. A norm–regularized variant $\mathrm{KPCA}(\boldsymbol{x}) = e(\boldsymbol{x})/\sqrt{k_c(\boldsymbol{x}, \boldsymbol{x})}$ further reduces residual norm confounds.

To avoid computing the full $N \times N$ kernel and $O(N^2)$ memory, we approximate the Gaussian on the sphere with an explicit map $\psi : \mathbb{R}^D \to \mathbb{R}^M$ so that $k(\boldsymbol{x}, \boldsymbol{x}') \approx \psi(\hat{\boldsymbol{h}})^\top \psi(\hat{\boldsymbol{h}}')$ with $M \ll N$. In particular, we use Nyström features with landmarks $\{\boldsymbol{x}_\ell^\star\}_{\ell=1}^M$ (e.g., low-energy ID points near the boundary), $\psi(\tilde{z}) = C\,W^{-1/2}$ where $C_\ell = k(x, x_\ell^\star)$ and W is the landmark Gram matrix. We then perform ordinary PCA on $\psi(\hat{\boldsymbol{h}})$: compute mean $\boldsymbol{\mu}$ and top-k eigenvectors U_k of the empirical covariance, and score a test point by the Euclidean reconstruction error $\tilde{e}(\boldsymbol{x}) = \big\|\big(\boldsymbol{I} - \boldsymbol{U}_k\boldsymbol{U}_k^\top\big)\big(\psi(\hat{\boldsymbol{h}}) - \boldsymbol{\mu}\big)\big\|_2^2$, and $\tilde{r}(\boldsymbol{x}) = \frac{\tilde{e}(\boldsymbol{x})}{\|\psi(\hat{\boldsymbol{h}})\|_2}$. Empirically, the Cosine–Gaussian kernel and the low-energy Nyström approximation improve separability and efficiency over linear PCA and nearest-neighbor baselines.

C.12 Residual Projection and Virtual Matching Logit (ViM) (Wang et al., 2022)

Residual Projection score. If the ID principal subspace $P \subset \mathbb{R}^N$ from training features is defined as the span of the top-D eigenvectors of $\boldsymbol{H}^\top\boldsymbol{H}$, then $R \in \mathbb{R}^{N\times(N-D)}$ have columns spanning $P^\perp$. The *residual* projection of $\boldsymbol{x}$ is $r(\boldsymbol{x}) = RR^\top\boldsymbol{h}$, and the residual projection score is $\mathrm{Residual}(\boldsymbol{x}) = \|r(\boldsymbol{x})\|_2$, which increases as the feature departs from the ID principal subspace. This score is class-agnostic and leverages feature-space geometry that is lost when projecting to logits.

ViM (Virtual-logit Matching). ViM fuses the class-agnostic residual with class-dependent logits by creating a virtual $(K+1)$-st logit from the residual and matching its scale to the real logits. ViM score is defined as the softmax probability of this virtual class: $\mathrm{ViM}(\boldsymbol{x}) = \frac{\exp\{\ell_0(\boldsymbol{x})\}}{\sum_{k=1}^K \exp\{g(\boldsymbol{x})_k\} + \exp\{\ell_0(\boldsymbol{x})\}}$, where the virtual logit $\ell_0(\boldsymbol{x}) = \alpha\,\|r(\boldsymbol{x})\|_2$, and the scaling factor $\alpha = \mathbb{E}_{\boldsymbol{x}\sim\mathrm{ID}}\big[\max_{k\le K} f_k(\boldsymbol{x})\big]/\mathbb{E}_{\boldsymbol{x}\sim\mathrm{ID}}\big[\|r(\boldsymbol{x})\|_2\big]$. Equivalently, applying the transformation $t(x) = -\ln(1/x - 1)$ yields $t(\mathrm{ViM}(\boldsymbol{x})) = \alpha\,\|r(\boldsymbol{x})\|_2 - \log\sum_{k=1}^K e^{g(\boldsymbol{h})_k}$, i.e., a residual term minus the *Energy* of the logits. Thus ViM is large when the residual is large and the (ID) logits are small.

C.13 Neural Collapse (NeCo) (Ammar et al., 2023)

This method is motivated by the Neural Collapse phenomenon (Papyan et al., 2020), which reveals geometric properties that manifest at the end of the training process. NeCo's key observation establishes ID/OOD orthogonality, which implies that OOD features concentrate near the origin after projection onto the ID subspace. This method fits PCA on ID features to obtain the top-d principal directions $\boldsymbol{P} \in \mathbb{R}^{D\times d}$ (orthonormal columns). Then it scores an input by the normalized projection of its feature onto the ID principal subspace, $\mathrm{NeCo}(\boldsymbol{x}) = \frac{\|\boldsymbol{P}^\top\boldsymbol{h}\|_2}{\|\boldsymbol{h}\|_2}$, so that ID points (well aligned with the ID subspace) yield larger scores, while OOD points (near-orthogonal to that subspace) yield smaller scores. This score is optionally calibrated by multiplying with MLS to inject class-scale information.

C.14 Learned Confidence Scores

In addition to the post-hoc scoring functions described above, we evaluate three confidence scores that are learned during training via modified loss functions. Each method augments the standard classification objective with an auxiliary mechanism that encourages the model to produce calibrated confidence estimates.

ConfidNet (Corbière et al., 2019; Corbiere et al., 2021). ConfidNet trains a small auxiliary network on top of the frozen penultimate-layer features to predict the True Class Probability (TCP), i.e., the softmax probability assigned to the ground-truth label. During training, the auxiliary network minimizes the mean squared error between its scalar output and the TCP of the main classifier. At test time, the auxiliary network's output serves as the confidence score: higher values indicate that the model predicts the sample is likely to be correctly classified.

DeVries (DeVries and Taylor, 2018). DeVries extends the classifier with an additional scalar confidence branch that shares the penultimate-layer representation. The network is trained with an interpolated loss: the predicted distribution is mixed with the ground-truth label proportionally to a learned confidence value $c \in [0, 1]$, so that low-confidence predictions are regularized toward the true label. At test time, the confidence branch output $c(\boldsymbol{x})$ is used directly as the confidence score.

Deep Gamblers (Liu et al., 2019). Deep Gamblers adds a $(K+1)$-st "abstention" class to the softmax output. The training loss rewards correct predictions and penalizes abstention via a tunable reward parameter $o > 0$: the model can "gamble" by placing probability mass on the abstention class when uncertain. The confidence score is defined as $1 - p_{K+1}(\boldsymbol{x})$, where p_{K+1} is the probability assigned to the abstention class. Lower reward values encourage more frequent abstention, while higher values push the model toward confident predictions.

D Hyperparameter and Calibration Selection

In order to rigorously evaluate the worst-case safety of each Confidence Scoring Function (CSF), all method-specific hyperparameters and post-hoc calibration mappings were selected dynamically. Rather than relying on standard threshold-free ranking metrics (such as AUROC or AUGRC) for hyperparameter tuning, we optimized the configurations by strictly minimizing the ECE_{L_1} upper bound.

To prevent high-capacity, non-parametric mappers from artificially minimizing the bounds by overfitting to the validation distribution, the selection was performed using Out-of-Fold (OOF) probabilities. Specifically, we applied 5-fold cross-validation on the in-distribution validation set, extracting the OOF predictions to compute an unbiased ECE_{L_1} bound for each configuration.

The hyperparameter search space evaluated three primary components for each CSF across the tested training paradigms:

- **Inference Stochasticity (Dropout):** We evaluated whether standard deterministic inference (`do0`) or epistemic uncertainty estimation via Monte-Carlo Dropout (`do1`, utilizing 50 stochastic forward passes) yielded a tighter, safer calibration bound.
- **Calibration Mapping:** Raw anomaly scores lack probabilistic semantics and must be mapped to the $[0, 1]$ range. We evaluated multiple mapping functions, dynamically selecting between Isotonic Regression, Beta Calibration, and Sigmoid (Platt) scaling based on which method minimized the OOF L_1 penalty.
- **Method-Specific Parameters:** For models trained under the Deep Gamblers paradigm, the abstention reward parameter (which dictates the penalty for low-confidence predictions) was explicitly tuned across a grid of candidate values.

Tables 3 and 4 detail the final optimal configurations selected via this ECE_{L_1} bound minimization strategy.

Table 3: **Optimal Hyperparameter Configurations via Validation ECE_{L_1} Bound Minimization.** This table details the selected hyperparameters for various Confidence Scoring Functions (CSFs) across four distinct training paradigms (ConfidNet, DeVries, Deep Gamblers, and fine-tuned ViT) on the CIFAR-10 and CIFAR-100 datasets. For each configuration, the table specifies whether deterministic inference (`do0`) or stochastic Monte-Carlo Dropout (`do1`) was selected, alongside the optimal probability calibration mapping (Isotonic Regression, Platt scaling or Beta Calibration). Additionally, for the Deep Gamblers paradigm, the tuned abstention reward parameter is reported. All hyperparameters were chosen dynamically by strictly minimizing the ECE_{L_1} upper bound on the in-distribution validation set.

		ConfidNet		DeVries		DeepGamblers			ViT
	CSF	Dropout	Calibration	Dropout	Calibration	Dropout	Reward	Calibration	Dropout	Calibration
CIFAR10	GradNorm	do1	Isotonic	do1	Isotonic	do1	3.0	Isotonic	do1	Isotonic
	Residual	do1	Isotonic	do1	Isotonic	do0	6.0	Isotonic	do1	Isotonic
	PCA	do1	Isotonic	do0	Isotonic	do1	10.0	Isotonic	do1	Isotonic
	Maha	do1	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do1	Isotonic
	pNML	do0	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do1	Isotonic
	PE	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do1	Isotonic
	KPCA	do0	Isotonic	do0	Isotonic	do0	2.2	Isotonic	do1	Isotonic
	ViM	do1	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do1	Isotonic
	Confidence	do1	Isotonic	do0	Isotonic	do0	2.2	Isotonic	do1	Isotonic
	REN	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do1	Isotonic
	PCE	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do1	Beta
	CTM	do0	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do1	Isotonic
	NeCo	do0	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do1	Isotonic
	GEN	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do1	Isotonic
	MSR	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do1	Beta
	CTMmeanOC	do0	Isotonic	do0	Isotonic	do0	3.0	Isotonic	do1	Isotonic
	fDBD	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do1	Isotonic
	CTMmean	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do1	Isotonic
	MLS	do0	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do1	Isotonic
	GE	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do1	Isotonic
	Energy	do1	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do1	Isotonic
	NNGuide	do1	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do1	Isotonic
CIFAR100	Residual	do0	Isotonic	do0	Isotonic	do1	6.0	Beta	do1	Isotonic
	Maha	do0	Isotonic	do0	Isotonic	do1	6.0	Beta	do1	Isotonic
	PCE	do1	Isotonic	do0	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	Energy	do1	Isotonic	do1	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	PCA	do1	Isotonic	do1	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	MSR	do0	Isotonic	do0	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	GE	do1	Isotonic	do1	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	Confidence	do1	Isotonic	do1	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	GEN	do0	Isotonic	do0	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	CTMmeanOC	do1	Isotonic	do0	Isotonic	do0	12.0	Isotonic	do1	Isotonic
	PE	do1	Isotonic	do1	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	CTMmean	do1	Isotonic	do0	Isotonic	do0	12.0	Isotonic	do1	Isotonic
	REN	do0	Isotonic	do0	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	MLS	do1	Isotonic	do1	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	KPCA	do1	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do1	Isotonic
	NNGuide	do1	Isotonic	do1	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	CTM	do0	Isotonic	do0	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	pNML	do1	Isotonic	do1	Isotonic	do1	6.0	Beta	do1	Isotonic
	NeCo	do1	Isotonic	do1	Isotonic	do1	20.0	Isotonic	do1	Isotonic
	ViM	do0	Isotonic	do0	Isotonic	do0	12.0	Isotonic	do1	Isotonic
	GradNorm	do1	Isotonic	do1	Isotonic	do1	6.0	Isotonic	do1	Isotonic
	fDBD	do0	Isotonic	do0	Isotonic	do0	12.0	Isotonic	do1	Isotonic

Table 4: **Optimal Hyperparameter Configurations via Validation ECE_{L_1} Bound Minimization.** This table details the selected hyperparameters for various Confidence Scoring Functions (CSFs) across four distinct training paradigms (ConfidNet, DeVries, Deep Gamblers, and fine-tuned ViT) on the SuperCIFAR-100 and Tiny-Imagenet datasets. For each configuration, the table specifies whether deterministic inference (`do0`) or stochastic Monte-Carlo Dropout (`do1`) was selected, alongside the optimal probability calibration mapping (Isotonic Regression, Platt scaling or Beta Calibration). Additionally, for the Deep Gamblers paradigm, the tuned abstention reward parameter is reported. All hyperparameters were chosen dynamically by strictly minimizing the ECE_{L_1} upper bound on the in-distribution validation set.

		ConfidNet		DeVries		DeepGamblers			ViT
	CSF	Dropout	Calibration	Dropout	Calibration	Dropout	Reward	Calibration	Dropout	Calibration
SuperCIFAR100	NNGuide	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	GradNorm	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	Energy	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	Confidence	do1	Isotonic	do0	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	GEN	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	MSR	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	GE	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	fDBD	do0	Isotonic	do0	Isotonic	do0	6.0	Isotonic	do0	Isotonic
	Residual	do1	Isotonic	do1	Isotonic	do1	2.2	Beta	do0	Isotonic
	NeCo	do1	Isotonic	do1	Isotonic	do1	20.0	Isotonic	do0	Isotonic
	PE	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	REN	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	MLS	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	KPCA	do1	Isotonic	do1	Isotonic	do0	6.0	Isotonic	do0	Isotonic
	CTMmeanOC	do0	Isotonic	do1	Isotonic	do0	6.0	Isotonic	do0	Isotonic
	CTMmean	do0	Isotonic	do1	Isotonic	do0	6.0	Isotonic	do0	Isotonic
	PCE	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	CTM	do0	Isotonic	do0	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	PCA	do1	Isotonic	do1	Isotonic	do1	2.2	Isotonic	do0	Isotonic
	Maha	do1	Isotonic	do1	Isotonic	do1	2.2	Beta	do0	Isotonic
	ViM	do1	Isotonic	do1	Isotonic	do0	2.2	Isotonic	do0	Isotonic
	pNML	do1	Isotonic	do1	Isotonic	do1	2.2	Beta	do0	Isotonic
TinyImagenet	Residual	do0	Beta	do0	Sigmoid	do0	10.0	Sigmoid	do0	Isotonic
	Maha	do0	Isotonic	do0	Beta	do0	10.0	Beta	do0	Isotonic
	GEN	do0	Isotonic	do0	Isotonic	do0	20.0	Isotonic	do0	Isotonic
	GradNorm	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do0	Isotonic
	PCE	do0	Isotonic	do0	Isotonic	do0	20.0	Isotonic	do0	Isotonic
	GE	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do0	Isotonic
	PE	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do0	Isotonic
	CTM	do0	Isotonic	do0	Isotonic	do0	12.0	Isotonic	do0	Isotonic
	pNML	do0	Isotonic	do0	Isotonic	do0	20.0	Isotonic	do0	Isotonic
	REN	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do0	Isotonic
	MSR	do0	Isotonic	do0	Isotonic	do0	20.0	Isotonic	do0	Isotonic
	fDBD	do0	Isotonic	do0	Isotonic	do0	12.0	Isotonic	do0	Isotonic
	ViM	do1	Isotonic	do1	Isotonic	do1	20.0	Isotonic	do0	Isotonic
	NeCo	do0	Isotonic	do0	Isotonic	do0	20.0	Isotonic	do0	Isotonic
	MLS	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do0	Isotonic
	Energy	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do0	Isotonic
	CTMmean	do0	Isotonic	do0	Isotonic	do0	20.0	Isotonic	do0	Isotonic
	KPCA	do0	Isotonic	do0	Isotonic	do0	12.0	Isotonic	do0	Isotonic
	Confidence	do1	Isotonic	do1	Isotonic	do1	10.0	Isotonic	do0	Isotonic
	CTMmeanOC	do0	Isotonic	do0	Isotonic	do0	20.0	Isotonic	do0	Isotonic
	NNGuide	do0	Isotonic	do0	Isotonic	do0	10.0	Isotonic	do0	Isotonic
	PCA	do0	Isotonic	do0	Isotonic	do0	20.0	Isotonic	do0	Isotonic