IGEOOD: An Information Geometry Approach to Out-of-Distribution Detection

Eduardo D. C. Gomes, Florence Alberge, Pierre Duhamel and Pablo Piantanida

Laboratoire des signaux et systèmes (L2S)
Université Paris-Saclay CNRS CentraleSupélec
91190, Gif-sur-Yvette, France.

Abstract

Reliable out-of-distribution (OOD) detection is a fundamental step towards a safer implementation of modern machine learning (ML) systems under distribution shift. In this paper, we introduce IGEOOD, an effective method for detecting OOD samples. IGEOOD applies to any pre-trained neural network, works under different degrees of access to the ML model, does not require OOD samples or assumptions on the OOD data, but can also benefit (if available) from OOD samples. By building on the geodesic (Fisher-Rao) distance between the underlying data distributions, our discriminator combines confidence scores from the logits outputs and the learned features of a deep neural network. Empirically, we show that IGEOOD is competitive and often outperforms state-of-the-art methods by a large margin on a variety of networks architectures and datasets.

1 Introduction

Out-Of-Distribution (OOD) or novelty detection is one of the main objectives in conceiving reliable ML systems [1]. An application of these methods arises in monitoring ML-based online services for drifting distributions. Tracking changes in the underlying data distribution is challenging as they contain unusual (irregular or unexpected) events and have large dimensions. For instance, the detection of such anomalous observations will have to rely on the intrinsic properties of the ML models, methods, and algorithms based on their statistical behavior in the presence of in-distribution data. Classic approaches to OOD detection consist of deriving metrics for detecting those abnormalities from the lens of ML models (e.g., softmax output, latent representations across layers), provided that the data is high dimensional and often only a single test example is available.

1.1 Contributions

In this paper, we propose IGEOOD, a new unified and effective method to perform OOD detection by rigorously exploring the information-geometric properties of the feature space on various depths of a DNN. IGEOOD provides a flexible framework that applies to any pre-trained softmax neural classifier. A key ingredient of IGEOOD is the Fisher-Rao distance. This distance is used as an effective differential geometry tool for clustering in the context of multivariate Gaussian pdfs [30, 37]. In our context, we measure the dissimilarity between probability (in and out) distributions, as the length of the shortest path within the manifold induced by the underlying class of distributions (i.e., the softmax probabilities of the neural classifier or the densities modeling the learned representations across the layers). By doing so, we can explore statistical invariances of the geometric properties of the learned features [5]. Our method adapts to various scenarios depending on the level of information access of the DNN, uses only in-distribution samples but can also benefit (if available) of OOD samples.

Workshop on Distribution Shifts, 35th Conference on Neural Information Processing Systems (NeurIPS 2021).

1.2 Related works

OOD detectors are binary classifiers that discriminate in- and out-of-distribution samples. A few works [36, 14, 4, 26, 40, 39, 12] propose retraining the base (or an auxiliary) model with synthetic or ground truth OOD samples to serve as a classifier and as an OOD discriminator. Disposing of both OOD and in-distribution samples during training enables the hidden layers to learn representations to facilitate OOD detection. These methods will not be compared to ours in this work, as they entail retraining or modifying the base neural network by using OOD data to further train parameters. Moreover, this assumes that OOD samples are stationary, which is an unrealistic assumption in practical scenarios. The work [27] demonstrates failure modes of OOD detection methods to better understand how to improve them, especially how spurious features like image background can vastly degrade detection performance. references [35, 17, 6, 38, 42, 33, 45, 25] study OOD detection in the context of generative models. Open set recognition [3], outlier or anomaly detection [29], concept drift detection [31], and adversarial attacks detection [10, 23] are related topics.

WHITE-BOX scenario. This class of OOD detectors allows discriminators to have access to all intermediate layer outputs. Naturally, they have access to more information than BLACK-BOX or GREY-BOX techniques, which provide detection based only on the network's outputs, i.e., MSP [13], ODIN [21], and the free-energy [22] based methods to name a few. reference [34] proposes high order Gram matrices to perform OOD detection, by computing class-conditional pairwise feature correlations across the hidden layers of the network. The work [20] models the latent features' outputs of DNN models as a class-conditional Gaussian mixture distribution with tied covariance matrix and class-conditional mean vectors. They calculate the Mahalanobis distance between an OOD sample as a single estimator of the mean of a class-conditional Gaussian distribution with covariance matrix estimated on the entire training set. The importance of each feature component and hyperparameters are tuned using validation data. The work [32] modifies the Mahalanobis distance-based OOD detector [20] to improve near-OOD detection by reducing the importance of features shared by in- and out-of-distribution data.

2 IGEOOD: OOD Detection using the Fisher-Rao Distance

In this section, we introduce IGEOOD, a flexible framework for OOD detection. IGEOOD is implemented in two ways: at the level of the logits using temperature scaling (Section 2.1), and layer-wise level (Section 2.2). The key ingredient of IGEOOD is the Fisher-Rao (F-R) distance [2] This distance measures the dissimilarity between two probability models within a class of probability distributions by calculating the geodesic distance between two points on the learned manifold.

2.1 IGEOOD score using the softmax probability

For the classification problem, we can take the temperature T scaled softmax function (Eq. (1)) as an approximation of a class-conditional probability distribution:

$$q_{\boldsymbol{\theta}}\left(y|f(\boldsymbol{x});T\right) \triangleq \frac{\exp\left(f_y(\boldsymbol{x})/T\right)}{\sum_{y'\in\mathcal{Y}}\exp\left(f_{y'}(\boldsymbol{x})/T\right)}, \tag{1}$$

where $f:\mathcal{X}\to\mathbb{R}^C$ is a vectorial function with $f \triangleq \big(f_1, f_2, \ldots, f_C\big)$ and $f_y(\cdot)$ denotes the y-th logits output value of the DNN classifier. The F-R distance $d_{\mathrm{FR-Logits}}$ between two softmax probability distributions can be shown by[1]

$$d_{\mathrm{FR-Logits}}\big(q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x})), q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x}'))\big) \triangleq 2 \arccos\left(\sum_{y\in\mathcal{Y}}\sqrt{q_{\boldsymbol{\theta}}\big(y|f(\boldsymbol{x})\big)q_{\boldsymbol{\theta}}\big(y|f(\boldsymbol{x}')\big)}\right). \tag{2}$$

Class conditional centroid estimation. We model the training dataset class-conditional posterior distribution by calculating the centroid of the logits representations of this set. Precisely, we compute the *centroid* for the logits of each class y of the in-distribution training dataset $\mathcal{D}_N$ corresponding to

[1] We refer the reader to the appendix (see Section A).

the F-R distance, i.e.,

$$\boldsymbol{\mu}_y \triangleq \min_{\boldsymbol{\mu}\in\mathbb{R}^C} \frac{1}{N_y} \sum_{\forall\, i\,:\, y_i=y} d_{\mathrm{FR-Logits}}\big(q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x}_i)), q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{\mu})\big), \tag{3}$$

where N_y is the amount of training examples with label y. We optimize this expression using SGD algorithm, where the parameter to be tuned is $\boldsymbol{\mu}$ in the logits space.

OOD detection score. We propose the F-R distance-based OOD detection score $\mathrm{FR}_0(\boldsymbol{x})$ on the space of the logits to be the sum of the distances between $f(\boldsymbol{x})$ and each individual class conditional centroid $\boldsymbol{\mu}_y$ calculated by Eq. (3). We denote it as follows:

$$\mathrm{FR}_0(\boldsymbol{x}) \triangleq \sum_{y\in\mathcal{Y}} d_{\mathrm{FR-Logits}}\big(q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x})), q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{\mu}_y)\big). \tag{4}$$

We obtained better performance by taking the sum instead of the minimal distance. A likely explanation for this would be that Eq. (4) leverages useful information related to the example's confidence score for each class y.

2.2 IGEOOD score leveraging latent features

For each layer, we define a set of class-conditional Gaussian distributions with diagonal standard deviation matrix $\boldsymbol{\sigma}^{(\ell)}$ and class-conditional mean $\boldsymbol{\mu}_y^{(\ell)}$, where $y \in \{1,\ldots,C\}$ and ℓ is the index of the latent feature. We compute the empirical estimates of these parameters according to

$$\boldsymbol{\mu}_y^{(\ell)} = \frac{1}{N_y}\sum_{\forall i\,:\, y_i=y} f^{(\ell)}\left(\boldsymbol{x}_i\right), \quad \text{and} \quad \boldsymbol{\sigma}^{(\ell)} = \operatorname{diag}\left(\sqrt{\frac{1}{N}\sum_{y\in\mathcal{Y}}\sum_{\forall i\,:\, y_i=y}\left(f_j^{(\ell)}\left(\boldsymbol{x}_i\right)-\mu_{y,j}^{(\ell)}\right)^2}\right), \tag{5}$$

where $j\in\{1,\ldots,k\}$, k is the size of feature ℓ, and $f^{(\ell)}(\cdot)$ is the output of the network for feature ℓ. The F-R distance ρ_{FR} between two *univariate* Gaussian pdfs $\mathcal{N}(\mu_1,\sigma_1^2)$ and $\mathcal{N}(\mu_2,\sigma_2^2)$ is given by[2]

$$\rho_{\mathrm{FR}}\left(\left(\mu_1,\sigma_1\right),\left(\mu_2,\sigma_2\right)\right) = \sqrt{2}\log\frac{\left|\left(\frac{\mu_1}{\sqrt{2}},\sigma_1\right)-\left(\frac{\mu_2}{\sqrt{2}},-\sigma_2\right)\right| + \left|\left(\frac{\mu_1}{\sqrt{2}},\sigma_1\right)-\left(\frac{\mu_2}{\sqrt{2}},\sigma_2\right)\right|}{\left|\left(\frac{\mu_1}{\sqrt{2}},\sigma_1\right)-\left(\frac{\mu_2}{\sqrt{2}},-\sigma_2\right)\right| - \left|\left(\frac{\mu_1}{\sqrt{2}},\sigma_1\right)-\left(\frac{\mu_2}{\sqrt{2}},\sigma_2\right)\right|}. \tag{6}$$

where (μ_i,σ_i) is a 2-dimensional vector with components μ_i and σ_i and $|\cdot|$ is the 2-norm. Similarly, the F-R distance $d_{\mathrm{FR-Gauss}}$ between two *multivariate* Gaussian pdfs with diagonal standard deviation matrix is derived from the univariate case and is given by

$$d_{\mathrm{FR-Gauss}}\big((\boldsymbol{\mu},\boldsymbol{\sigma}),(\boldsymbol{\mu}',\boldsymbol{\sigma}')\big) = \sqrt{\sum_{i=1}^{k}\rho_{\mathrm{FR}}\left(\left(\mu_i,\sigma_{i,i}\right),\left(\mu_i',\sigma_{i,i}'\right)\right)^2}, \tag{7}$$

where k is the cardinality of the distributions $\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\sigma}^2)$ and $\mathcal{N}(\boldsymbol{\mu}',\boldsymbol{\sigma}'^2)$, μ_i is the i-th component of the vector $\boldsymbol{\mu}$, and $\sigma_{i,i}$ is the entry of index i,i of the standard deviation matrix $\boldsymbol{\sigma}$.

Experimental support for a diagonal Gaussian mixture model. We observed that the latent features covariance matrices are often *ill-conditioned* and are diagonal dominant. In other words, the condition number of the covariance matrix often diverges, and the magnitude of the diagonal entry in a row is greater than or equal to the sum of all the other entries in that row for most rows.

Fisher-Rao distance-based feature-wise confidence score. We derive a confidence score by applying the F-R distance between the test sample $\boldsymbol{x}$ and the closest class-conditional diagonal Gaussian distribution. We can consider two situations: **(i)** we do not have access to any validation OOD data whatsoever. In this case, the natural choice is to model the test samples as Gaussian distribution with the same diagonal standard deviation as the learned representation, i.e.,

$$\mathrm{FR}_\ell(\boldsymbol{x}) = \min_{y\in\mathcal{Y}} d_{\mathrm{FR-Gauss}}\big((\boldsymbol{x},\boldsymbol{\sigma}^{(\ell)}),(\boldsymbol{\mu}_y^{(\ell)},\boldsymbol{\sigma}^{(\ell)})\big); \tag{8}$$

and **(ii)**, we dispose of a validation OOD dataset on which the features' diagonal standard deviation matrices $\boldsymbol{\sigma}'^{(\ell)}$ and $\boldsymbol{\mu}'^{(\ell)}$ can be estimated, as well as the quantity

$$\mathrm{FR}_\ell'(\boldsymbol{x}) = \min_{y\in\mathcal{Y}} d_{\mathrm{FR-Gauss}}\big((\boldsymbol{x},\boldsymbol{\sigma}^{(\ell)}),(\boldsymbol{\mu}'^{(\ell)},\boldsymbol{\sigma}'^{(\ell)})\big). \tag{9}$$

[2]The reader can be refereed to the appendix (Section A) for the derivation of this distance.

Feature ensemble. Similarly to [20], we combine the confidence scores of the logits and low-level features through a linear combination to emphasize features that demonstrate a greater capacity for detecting abnormal samples. The following equation summarizes the IGEOOD score:

$$\mathrm{FR}(\boldsymbol{x}) \triangleq \alpha_0 \mathrm{FR}_0(\boldsymbol{x}) + \sum_{\ell} \alpha_\ell \cdot \mathrm{FR}_\ell(\boldsymbol{x}) + \alpha'_\ell \cdot \mathrm{FR}'_\ell(\boldsymbol{x}). \tag{10}$$

3 Experimental Results

The experimental setup follows the setting established by [13], [21] and [20]. We use two *pre-trained* deep neural networks architectures for image classification tasks: DenseNet-BC-100 [16] and a ResNet-34 [11]. We take as *in-distribution data* images from CIFAR-10 [18], CIFAR-100 and SVHN [28] datasets. For *out-of-distribution data*, we use natural image examples from the datasets: Tiny-ImageNet [19], LSUN [44], Describable Textures Dataset [7], Chars74K [8], Places365 [46], iSUN [43] and a synthetic dataset generated from Gaussian noise. For models pre-trained on CIFAR-10, data from CIFAR-100 and SVHN are also considered OOD; for models pre-trained on CIFAR-100, data from CIFAR-10 and SVHN are considered OOD, and for models pre-trained on SVHN, the CIFAR-10 and CIFAR-100 datasets are considered OOD. Even though we ran experiments with image data, IGEOOD could be applied to any neural-based classification task.

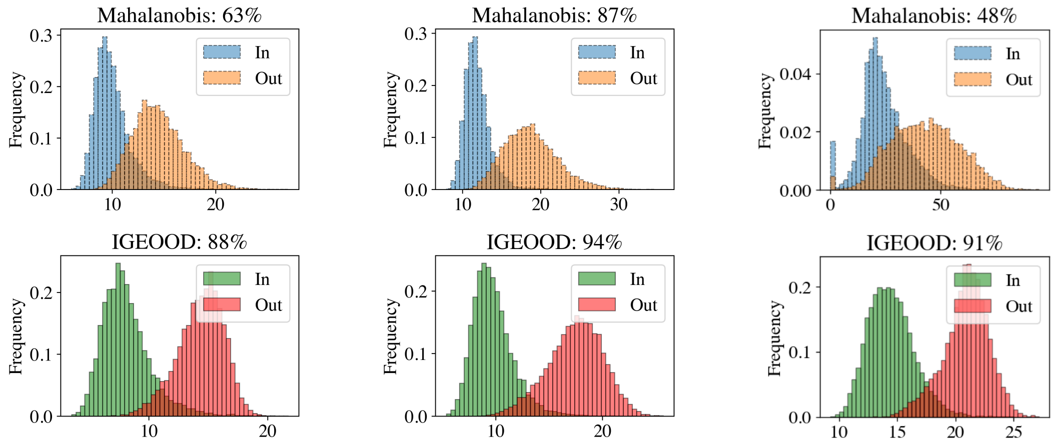
We consider two tuning scenarios: one with data from adversarially generated [10] (FGSM) samples from the training dataset, and another with data from the OOD test set. For the former, we tune hyperparameters for each method with generated data (pseudo OOD) and in-distribution data. While for the latter, we tune hyperparameters with 1,000 OOD data samples and in-distribution data. We derive two methods: IGEOOD+, which is given by Eq. (10) and considers that we have an estimate on the diagonal covariance matrix and mean vector from OOD data as additional information; and IGEOOD, which doesn't consider any prior on OOD data, i.e., set $\alpha'_\ell = 0$ on Eq. (10).

Comparison with Mahalanobis. For each DNN model and in-distribution dataset pair, we report the average OOD detection performance for Mahalanobis [20], IGEOOD and IGEOOD+. Table 1 validates the contributions of our techniques. We observe substantial performance improvement in all experiments for the left-hand side of the table, where IGEOOD+ outperforms Mahalanobis on average for all test cases, recording an improvement up to 23% on TNR at TPR-95%. To assess the consistency of IGEOOD to the choice of validation data, we measured the detection performance when hyperparameters are tuned only using in-distribution and generated adversarial data as observed in the right-hand side of Table 1. In this setup, IGEOOD improves by 2.5% the average TNR at TPR-95% across all datasets and models, but is sometimes outperformed by 2-3%.

Table 1: Average and standard deviation of OOD detection performance for the WHITE-BOX settings. The abbreviation TNR-95%, C-10 and C-100 stands for TNR at TPR-95%, CIFAR-10 and CIFAR-100, respectively. The extended results can be found in Tables 6 and 7 in the appendix.

		Tuning on OOD data		Tuning on adversarial data	
		TNR-95%	AUROC	TNR-95%	AUROC
Model	In-dist.	Mahalanobis / IGEOOD+ (ours)		Mahalanobis / IGEOOD (ours)	
DenseNet	C-10	76.6±31/**92.6**±14	92.1±12/**98.4**±3.0	75.9±30/**77.9**±29	91.7±12/**94.0**±9.0
	C-100	67.2±28/**90.2**±21	90.2±13/**97.7**±5.0	60.4±34/**70.9**±35	85.3±19/**90.8**±13
	SVHN	93.3±8.0/**98.0**±2.0	98.6±1.0/**99.6**±0.1	**93.7**±10/92.2±9.0	**98.6**±2.0/98.4±1.0
ResNet	C-10	82.5±23/**91.6**±16	96.5±4.0/**98.4**±3.0	**78.6**±24/77.3±32	**95.3**±6.0/90.0±15
	C-100	70.4±30/**86.4**±23	91.9±10/**97.1**±5.0	57.4±36/**65.1**±33	86.9±13/**88.6**±15
	SVHN	96.8±6.0/**98.9**±2.0	99.2±1.0/**99.7**±0.1	**96.3**±8.0/93.6±14	**99.1**±1.0/98.4±3.0
Average and Std.		81.1±11/**92.9**±4.0	94.8±4.0/**98.5**±1.0	77.0±15/**79.5**±10	92.8±5.4/**93.4**±3.9

Ablation study. The IGEOOD score has three components, FR_0, FR_ℓ, and FR'_ℓ, that together compose the final metric given by Eq. (10). The outputs of the network provide limited OOD detection capacity. Always when available, the intermediate features, i.e., FR_ℓ, are a valuable resource for OOD detection. Moreover, when few reliable OOD data are available, calculating FR'_ℓ can further improve the detection performance as shown on the left side column of Table 1. Also, data from a source other than in-distribution, e.g., adversarial samples, is enough for tuning hyperparameters and combining features, as observed on the right side column of Table 1. Figure 1 shows the detection performance for each hidden feature of a DenseNet as well as the scores histograms for a particular task for both Mahalanobis and IGEOOD scores.



(a) Block 1. (b) Block 2. (c) Block 3.

Figure 1: Histograms of the Mahalanobis and IGEOOD scores for the outputs of each hidden block of a DenseNet model on CIFAR-10 (in-dstribution) and SVHN (out-of-distribution) datasets.

IGEOOD compared to other WHITE-BOX methods. Even though reference [20] shares the closest setup to ours, recent literature also proposes OOD detection in a WHITE-BOX setting, achieving state-of-the-art in a few benchmarks. Notably, [34, 15, 47] achieve great performance in a range of benchmarks. Thus, we report the results from the original references in Table 2. This setup considers that a few OOD samples are available for tuning. We refer the reader to the appendix (see Section D) where we provide a table of results with validation on adversarial data.

Table 2: TNR at TPR-95% (%) performance comparison in a WHITE-BOX setting considering the original results from [20, 34, 15, 47]. Methods with an (*) are tuned only with in-distribution data.

	OOD dataset	CIFAR-10	CIFAR-100	SVHN
		Mahalanobis [20] / Gram Matrix* [34] / DeConf-C* [15] / Res-Flow [47] / IGEOOD / IGEOOD+		
DenseNet	iSUN	95.3/99.0/ - / - /97.7/**99.8**	87.0/95.9/ - / - /93.8/**99.7**	**99.9**/99.4/ - / - /98.3/**99.9**
	LSUN	97.2/99.5/99.4/98.2/98.5/**99.9**	91.4/97.2/98.7/96.3/95.2/**99.9**	**99.9**/99.5/ - /**100**/97.1/**99.9**
	TinyImgNet	95.0/98.8/99.1/96.4/95.7/**99.8**	86.6/95.7/98.6/93.0/94.5/**99.5**	**99.9**/99.1/ - /**100**/98.2/**99.9**
	SVHN/C-10	90.8/96.1/98.8/94.9/98.9/**99.9**	82.5/89.3/95.9/84.9/93.3/**99.6**	96.8/80.4/ - /**99.0**/91.6/98.3
ResNet	iSUN	97.8/99.3/ - / - /97.2/**99.9**	89.9/94.8/ - / - /93.4/**99.8**	99.7/99.4/ - / - /99.8/**100**
	LSUN	98.8/99.6/ - /99.0/98.4/**100**	90.9/96.6/ - /96.2/94.3/**100**	**99.9**/99.6/ - /**100**/99.7/**99.9**
	TinyImgNet	97.1/98.7/ - /97.8/96.3/**99.6**	90.9/94.8/ - /94.6/90.1/**99.6**	**99.9**/99.3/ - /**100**/99.7/**99.9**
	SVHN/C-10	87.8/97.6/ - /96.5/98.8/**99.8**	91.9/80.8/ - /93.0/91.6/**99.7**	98.4/85.8/ - /99.4/97.7/**99.7**

4 Summary and Concluding Remarks

In this work, we introduced IGEOOD, an effective and flexible method for Out-Of-Distribution (OOD) detection that applies to any pre-trained neural network. The main feature of IGEOOD relies on the geodesic distance of the probabilistic manifold of the learned latent representations that induces an effective measure for OOD detection. The Fisher-Rao distance between pdf of the latent feature, corresponding to the test sample, and a reference pdf, corresponding to the conditional-class of pdfs, provides an effective confidence score. We consider diverse testing environments where prior knowledge of OOD data may or may not be available. Experimentally, we showed that IGEOOD can significantly and consistently improves the accuracy of OOD detection on several DNN models and across various OOD datasets, achieving new state-of-the-art performances on a few benchmarks.

Acknowledgments and Disclosure of Funding

This work has been supported by the project PSPC AIDA: 2019-PSPC-09 funded by BPI-France.

References

[1] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul F. Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *CoRR*, abs/1606.06565, 2016.

[2] Colin Atkinson and Ann F. S. Mitchell. Rao's distance measure. *Sankhyā: The Indian Journal of Statistics, Series A (1961-2002)*, 43(3):345–365, 1981.

[3] Abhijit Bendale and Terrance E. Boult. Towards open set deep networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1563–1572, 2016.

[4] Julian Bitterwolf, Alexander Meinke, and Matthias Hein. Certifiably adversarially robust detection of out-of-distribution data. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 16085–16095. Curran Associates, Inc., 2020.

[5] Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges, 2021.

[6] Hyun-Jae Choi and Eric Jang. Generative ensembles for robust anomaly detection. *ArXiv*, abs/1810.01392, 2018.

[7] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[8] T. E. de Campos, B. R. Babu, and M. Varma. Character recognition in natural images. In *Proceedings of the International Conference on Computer Vision Theory and Applications, Lisbon, Portugal*, February 2009.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[10] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[12] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 41–50, 2019.

[13] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017.

[14] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019.

[15] Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10948–10957, 2020.

[16] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017.

[17] Polina Kirichenko, Pavel Izmailov, and Andrew G Wilson. Why normalizing flows fail to detect out-of-distribution data. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 20578–20589. Curran Associates, Inc., 2020.

[18] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009.

[19] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. 2015.

[20] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 7167–7177. Curran Associates, Inc., 2018.

[21] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018.

[22] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 2020.

[23] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[24] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2:49–55, 1936.

[25] Ahsan Mahmood, Junier Oliva, and Martin Andreas Styner. Multiscale score matching for out-of-distribution detection. In *International Conference on Learning Representations*, 2021.

[26] Sina Mohseni, Mandar Pitale, JBS Yadawa, and Zhangyang Wang. Self-supervised learning for generalizable out-of-distribution detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):5216–5223, Apr. 2020.

[27] Vaishnavh Nagarajan, Anders Andreassen, and Behnam Neyshabur. Understanding the failure modes of out-of-distribution generalization. In *International Conference on Learning Representations*, 2021.

[28] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011.

[29] Marco Pimentel, David Clifton, Lei Clifton, and L. Tarassenko. A review of novelty detection. *Signal Processing*, 99:215–249, 06 2014.

[30] Julianna Pinele, João E. Strapasson, and Sueli I. R. Costa. The fisher–rao distance between multivariate normal distributions: Special cases, bounds and applications. *Entropy*, 22(4), 2020.

[31] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. *Dataset Shift in Machine Learning*. The MIT Press, 2009.

[32] Jie Ren, Stanislav Fort, Jeremiah Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. A simple fix to mahalanobis distance for improving near-ood detection, 2021.

[33] Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[34] Chandramouli Shama Sastry and Sageev Oore. Detecting out-of-distribution examples with Gram matrices. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8491–8501. PMLR, 13–18 Jul 2020.

[35] Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Margarethe Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *IPMI*, 2017.

[36] Gabi Shalev, Yossi Adi, and Joseph Keshet. Out-of-distribution detection using multiple semantic label representations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[37] João E. Strapasson, Julianna Pinele, and Sueli I. R. Costa. Clustering using the fisher-rao distance. In *2016 IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pages 1–5, 2016.

[38] Sachin Vernekar, Ashish Gaurav, Vahdat Abdelzad, Taylor Denouden, Rick Salay, and Krzysztof Czarnecki. Out-of-distribution detection in classifiers via generation. In *Neural Information Processing Systems (NeurIPS 2019), Safety and Robustness in Decision Making Workshop*. https://sites.google.com/view/neurips19-safe-robust-workshop, https://sites.google.com/view/neurips19-safe-robust-workshop, 12/2019 2019.

[39] Apoorv Vyas, Nataraj Jammalamadaka, Xia Zhu, Dipankar Das, Bharat Kaul, and Theodore L. Willke. Out-of-distribution detection using an ensemble of self supervised leave-out classifiers. In *ECCV (8)*, pages 560–574, 2018.

[40] Jim Winkens, Rudy Bunel, Abhijit Guha Roy, Robert Stanforth, Vivek Natarajan, Joseph R. Ledsam, Patricia MacWilliams, Pushmeet Kohli, Alan Karthikesalingam, Simon A. A. Kohl, taylan. cemgil, S. M. Ali Eslami, and Olaf Ronneberger. Contrastive training for improved out-of-distribution detection. *ArXiv*, abs/2007.05566, 2020.

[41] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492, 2010.

[42] Zhisheng Xiao, Qing Yan, and Yali Amit. Likelihood regret: An out-of-distribution detection score for variational auto-encoder. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 20685–20696. Curran Associates, Inc., 2020.

[43] Pingmei Xu, Krista A. Ehinger, Yinda Zhang, Adam Finkelstein, Sanjeev R. Kulkarni, and Jianxiong Xiao. Turkergaze: Crowdsourcing saliency with webcam based eye tracking. *ArXiv*, abs/1504.06755, 2015.

[44] F. Yu, Y. Zhang, Shuran Song, Ari Seff, and J. Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *ArXiv*, abs/1506.03365, 2015.

[45] Yufeng Zhang, Wanwei Liu, Zhenbang Chen, Ji Wang, Zhiming Liu, Kenli Li, and Hongmei Wei. Towards out-of-distribution detection with divergence guarantee in deep generative models. 2020.

[46] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[47] Ev Zisselman and Aviv Tamar. Deep residual flow for out of distribution detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

A Review of Fisher-Rao Distance (FRD)

In this section, we review some results from references [2, 30]. We intend to clarify some basic concepts surrounding the Fisher-Rao distance while motivating the use of this measure in the context of OOD detection.

In few words, the Fisher-Rao's distance is given by the geodesic distance, i.e., the shortest path between points in a Riemannian space induced by a parametric family. Consider the family $\mathcal{C}$ of probability distributions over the class of discrete concepts or labels: $\mathcal{Y} = \{1, \ldots, C\}$, denoted by $\mathcal{C} \triangleq \{q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^C\}$.

We are interested in measuring the distance between probability distributions $q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x})$ with respect to the testing input $\boldsymbol{x}$ and a population of inputs drawn accordingly to the in-distribution data set. To this end, we first need to characterize the Fisher-Rao distance for two inputs or for two probability distributions $q_{\boldsymbol{\theta}}, q'_{\boldsymbol{\theta}} \in \mathcal{C}$.

Assume that the following regularity conditions hold [2]:

(i) $\nabla_{\boldsymbol{x}}\, q_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ exists for all $\boldsymbol{x}, y$ and $\boldsymbol{\theta} \in \Theta$;

(ii) $\sum\limits_{y \in \mathcal{Y}} \nabla_{\boldsymbol{x}}\, q_{\boldsymbol{\theta}}(y|\boldsymbol{x}) = 0$ for all $\boldsymbol{x}$ and $\boldsymbol{\theta} \in \Theta$;

(iii) $\boldsymbol{G}(\boldsymbol{x}) = \mathbb{E}_{Y \sim q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x})}\left[\nabla_{\boldsymbol{x}}\, \log q_{\boldsymbol{\theta}}(Y|\boldsymbol{x}) \nabla_{\boldsymbol{x}}^{\top}\, \log q_{\boldsymbol{\theta}}(Y|\boldsymbol{x})\right]$ is positive definite for any $\boldsymbol{x}$ and $\boldsymbol{\theta} \in \Theta$.

Notice that if (i) holds, (ii) also holds immediately for discrete distributions over finite spaces (assuming that $\sum_{y \in \mathcal{Y}}$ and $\nabla_{\boldsymbol{x}}$ are interchangeable operations) as in our case. When (i)-(iii) are met, the variance of the differential form $\nabla_{\boldsymbol{x}}^{\top} \log q_{\boldsymbol{\theta}}(Y|\boldsymbol{x}) d\boldsymbol{x}$ can be interpreted as the square of a differential arc length ds^2 in the space $\mathcal{C}$, which yields

$$ds^2 = \langle d\boldsymbol{x}, d\boldsymbol{x} \rangle_{\boldsymbol{G}(\boldsymbol{x})} = d\boldsymbol{x}^{\top} \boldsymbol{G}(\boldsymbol{x}) d\boldsymbol{x}. \tag{11}$$

Thus, $\boldsymbol{G}$, which is the Fisher Information Matrix (FIM), can be adopted as a metric tensor. We now consider a curve $\boldsymbol{\gamma} : [0,1] \to \mathcal{X}$ connecting a pair of arbitrary points $\boldsymbol{x}$, $\boldsymbol{x}'$ in the input space $\mathcal{X}$, i.e., $\boldsymbol{\gamma}(0) = \boldsymbol{x}$ and $\boldsymbol{\gamma}(1) = \boldsymbol{x}'$. Notice that any curve $\boldsymbol{\gamma}$ induces a curve $q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{\gamma}(t))$ for $t \in [0,1]$ in the space $\mathcal{C}$. The Fisher-Rao distance between the distributions $q_{\boldsymbol{\theta}} = q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x})$ and $q'_{\boldsymbol{\theta}} = q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}')$ will be denoted as $d_{R,\mathcal{C}}(q_{\boldsymbol{\theta}}, q'_{\boldsymbol{\theta}})$ and is formally defined by the expression:

$$d_{R,\mathcal{C}}(q_{\boldsymbol{\theta}}, q'_{\boldsymbol{\theta}}) \triangleq \inf_{\boldsymbol{\gamma}} \int_0^1 \sqrt{\frac{d\boldsymbol{\gamma}^{\top}(t)}{dt} \boldsymbol{G}(\boldsymbol{\gamma}(t)) \frac{d\boldsymbol{\gamma}(t)}{dt}}, \tag{12}$$

where the infimum is taken over all piecewise smooth curves. This means that the FRD is the length of the *geodesic* between points $\boldsymbol{x}$ and $\boldsymbol{x}'$ using the FIM as the metric tensor. In general, the minimization of the functional in Eq. (12) is a problem that can be solved using the well-known Euler-Lagrange differential equation.

A.1 Derivation of Fisher-Rao distance for the class of Softmax probability distributions

The direct computation of the FIM of the family $\mathcal{C}$ with $q_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ in the form of the softmax probability distribution function Eq. (1) can be shown to be singular, i.e., $\operatorname{rank}(\boldsymbol{G}(\boldsymbol{x})) \leq C - 1$, where $C - 1$ is the number of degrees of freedom of the manifold $\mathcal{C}$. To overcome this issue, we introduce the probability simplex $\mathcal{P}$ defined by

$$\mathcal{P} = \left\{ q : \mathcal{Y} \to [0,1]^C : \sum_{y \in \mathcal{Y}} q(y) = 1 \right\}. \tag{13}$$

Next, we consider the following parametrization for any distribution $q \in \mathcal{P}$:

$$q(y|\boldsymbol{z}) = \frac{z_y^2}{4}, \quad y \in \{1, \ldots, C\}. \tag{14}$$

From this expression, we consider the statistical manifold $\mathcal{D} = \{q(\cdot|\boldsymbol{z}) : \|\boldsymbol{z}\|^2 = 4, z_y \geq 0, \forall y \in \mathcal{Y}\}$. Note that the parameter vector $\boldsymbol{z}$ belongs to the positive portion of a sphere of radius 2 and centered at the origin in $\mathbb{R}^C$. The computation of the FIM for $\boldsymbol{z}$ on $\mathcal{D}$ yields:

$$\begin{aligned} \boldsymbol{G}(\boldsymbol{z}) &= \mathbb{E}_{q(y|\boldsymbol{z})}\left[\nabla_{\boldsymbol{z}} \log q(y|\boldsymbol{z}) \nabla_{\boldsymbol{z}}^{\top} \log q(y|\boldsymbol{z})\right] \\ &= \sum_{y \in \mathcal{Y}} \frac{z_y^2}{4} \left(\frac{2}{z_y} \boldsymbol{e}_y\right)\left(\frac{2}{z_y} \boldsymbol{e}_y^{\top}\right) \\ &= \sum_{y \in \mathcal{Y}} \boldsymbol{e}_y \boldsymbol{e}_y^{\top} \\ &= \boldsymbol{I}, \end{aligned} \tag{15}$$

where $\{\boldsymbol{e}_y\}$ are the canonical basis vectors in $\mathbb{R}^C$ and $\boldsymbol{I}$ is the identity matrix. From expression Eq. (15) we can conclude that the Fisher-Rao metric in this parametric space is equal to the Euclidean metric. Also, since the parameter vector lies on a sphere, the FRD between the distributions $q = q(\cdot|\boldsymbol{z})$ and $q' = q(\cdot|\boldsymbol{z}')$ can be written as the radius of the sphere times the angle between the vectors $\boldsymbol{z}$ and $\boldsymbol{z}'$. Which leads to expression:

$$d_{R,\mathcal{D}}\left(q, q'\right) = 2 \arccos\left(\frac{\boldsymbol{z}^\top \boldsymbol{z}'}{4}\right) = 2\arccos\left(\sum_{y\in\mathcal{Y}} \sqrt{q(y|\boldsymbol{z}) q\left(y|\boldsymbol{z}'\right)}\right). \tag{16}$$

Finally, we can compute the FRD for softmax distributions in $\mathcal{C}$ as

$$d_{\mathrm{FR-Logits}}\left(q_{\boldsymbol{\theta}}, q'_{\boldsymbol{\theta}}\right) = 2\arccos\left(\sum_{y\in\mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}(y|\boldsymbol{x}) q_{\boldsymbol{\theta}}\left(y|\boldsymbol{x}'\right)}\right), \tag{17}$$

obtaining the same form of expression Eq. (2). Notice that $0 \leq d_{\mathrm{FR-Logits}}\left(q_{\boldsymbol{\theta}}, q'_{\boldsymbol{\theta}}\right) \leq \pi$ for all $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X} \subseteq \mathbb{R}^C$, being zero when $q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}) = q_{\boldsymbol{\theta}}\left(\cdot|\boldsymbol{x}'\right)$ and maximum when the vectors $\big(q_{\boldsymbol{\theta}}(1|\boldsymbol{x}), \dots, q_{\boldsymbol{\theta}}(C|\boldsymbol{x})\big)$ and $\big(q_{\boldsymbol{\theta}}\left(1|\boldsymbol{x}'\right), \dots, q_{\boldsymbol{\theta}}\left(C|\boldsymbol{x}'\right)\big)$ are orthogonal.

A.2 Derivation of Fisher-Rao distance for multivariate Gaussian distributions

Consider a broader statistical manifold $\mathcal{S} \triangleq \{p_{\boldsymbol{\theta}} = p(\boldsymbol{x};\boldsymbol{\theta}) : \boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_m) \in \Theta\}$ of multivariate differential probability density functions. The Fisher information matrix $\boldsymbol{G}(\boldsymbol{\theta}) = [g_{ij}(\boldsymbol{\theta})]$ in this parametric space is provided by:

$$\begin{aligned} g_{ij}(\boldsymbol{\theta}) &= \mathbb{E}_{\boldsymbol{\theta}}\left(\frac{\partial}{\partial\theta_i}\log p(\boldsymbol{x};\boldsymbol{\theta})\frac{\partial}{\partial\theta_j}\log p(\boldsymbol{x};\boldsymbol{\theta})\right) \\ &= \int \frac{\partial}{\partial\theta_i}\log p(\boldsymbol{x};\boldsymbol{\theta})\frac{\partial}{\partial\theta_j}\log p(\boldsymbol{x};\boldsymbol{\theta}) p(\boldsymbol{x};\boldsymbol{\theta}) dx. \end{aligned} \tag{18}$$

Next, consider a multivariate Gaussian distribution:

$$p(\boldsymbol{x};\boldsymbol{\mu},\Sigma) = \frac{(2\pi)^{-\left(\frac{n}{2}\right)}}{\sqrt{\mathrm{Det}(\Sigma)}}\exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{\mu})^\top\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}{2}\right), \tag{19}$$

where $\boldsymbol{x} \in \mathbb{R}^k$ is the variable vector, $\boldsymbol{\mu} \in \mathbb{R}^k$ is the mean vector, $\Sigma \in P_k(\mathbb{R})$ is the covariance matrix, and $P_k(\mathbb{R})$ is the space of k positive definite symmetric matrices. We can define the statistical manifold composed by these distributions as $\mathcal{M} = \{p_{\boldsymbol{\theta}}; \boldsymbol{\theta} = (\boldsymbol{\mu}, \Sigma) \in \mathbb{R}^k \times P_k(\mathbb{R})\}$. By substituting Eq. (19) in expression Eq. (18), we can derive the Fisher information matrix for this parametrization, obtaining:

$$g_{ij}(\boldsymbol{\theta}) = \frac{\partial\boldsymbol{\mu}^\top}{\partial\theta_i}\Sigma^{-1}\frac{\partial\boldsymbol{\mu}}{\partial\theta_j} + \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}\frac{\partial\Sigma}{\partial\theta_i}\Sigma^{-1}\frac{\partial\Sigma}{\partial\theta_i}\right), \tag{20}$$

which induces the following square differential arc length in $\mathcal{M}$:

$$ds^2 = d\boldsymbol{\mu}^\top\Sigma^{-1}d\boldsymbol{\mu} + \frac{1}{2}\operatorname{tr}\left[\left(\Sigma^{-1}d\Sigma\right)^2\right]. \tag{21}$$

Here, $d\boldsymbol{\mu} = (d\mu_1, \dots, d\mu_n) \in \mathbb{R}^k$ and $d\Sigma = [d\sigma_{ij}] \in P_k(\mathbb{R})$. We observe that this metric is invariant to affine transformations [30], i.e., for any $(\boldsymbol{c}, Q) \in \mathbb{R}^k \times GL_k(\mathbb{R})$, with $GL_k(\mathbb{R})$ the space of non-singular order k matrices, the map $(\boldsymbol{\mu}, \Sigma) \mapsto \left(Q\boldsymbol{\mu} + \boldsymbol{c}, Q\Sigma Q^\top\right)$ is an isometry in $\mathcal{M}$. Thus, the Fisher-Rao distance between two multivariate normal distributions with parameters $\boldsymbol{\theta}_1 = (\boldsymbol{\mu}_1, \Sigma_1)$ and $\boldsymbol{\theta}_2 = (\boldsymbol{\mu}_2, \Sigma_2)$ in $\mathcal{M}$ satisfies:

$$d_{R,\mathcal{M}}\left(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2\right) = d_{R,\mathcal{M}}\left(\left(Q\boldsymbol{\mu}_1 + \boldsymbol{c}, Q\Sigma_1 Q^\top\right), \left(Q\boldsymbol{\mu}_2 + \boldsymbol{c}, Q\Sigma_2 Q^\top\right)\right). \tag{22}$$

Unfortunately, a closed-form solution for the Fisher-Rao distance remains unknown. This is still an open problem for an arbitrary covariance matrix Σ and mean vector μ. Fortunately, the FRD is known for the univariate case and hence, for the submanifold where Σ is diagonal. Notice that in this case Eq. (21) admits an additive form.

From [30], we obtain the analytical expression of the Fisher-Rao in the 2-dimensional submanifold of univariate Gaussian probability distributions $\mathcal{M}_2 = \{p_{\boldsymbol{\theta}} : \boldsymbol{\theta} = (\mu, \sigma^2) \in \mathbb{R} \times (0, +\infty)\}$:

$$\rho_{\mathrm{FR}}\left(\left(\mu_1, \sigma_1^2\right), \left(\mu_2, \sigma_2^2\right)\right) = \sqrt{2}\log\frac{\left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, -\sigma_2\right)\right| + \left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, \sigma_2\right)\right|}{\left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, -\sigma_2\right)\right| - \left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, \sigma_2\right)\right|}, \tag{23}$$

where $|\cdot|$ is the Euclidian norm in $\mathbb{R}^2$ and σ denotes the standard deviation. Consequently, the FRD for Gaussian distributions with diagonal covariance matrix $\Sigma = \operatorname{diag}\left(\sigma_1^2, \sigma_2^2, \dots, \sigma_k^2\right)$ in the $2k$-dimensional statistical submanifold $\mathcal{M}_D = \left\{p_{\boldsymbol{\theta}} : \boldsymbol{\theta} = (\boldsymbol{\mu}, \Sigma), \Sigma = \operatorname{diag}\left(\sigma_1^2, \sigma_2^2, \dots, \sigma_k^2\right), \sigma_i > 0, i = 1, \dots, k\right\}$ is

$$d_{\mathrm{FR-Gauss}}\left(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2\right) = \sqrt{\sum_{i=1}^k d_{R,\mathcal{M}_2}\Big(\left(\mu_{1i}, \sigma_{1i}\right), \left(\mu_{2i}, \sigma_{2i}\right)\Big)^2}. \tag{24}$$

A.3 Fisher-Rao vs. Mahalanobis distance

There is an intricate relationship between the FRD for multivariate Gaussian distributions and the Mahalanobis distance. We borrow the result from [30], which states that in the k-dimensional submanifold $\mathcal{M}_\Sigma$ of $\mathcal{M}$ where Σ is constant, i.e., $\mathcal{M}_\Sigma = \{p_{\boldsymbol{\theta}} : \boldsymbol{\theta} = (\boldsymbol{\mu}, \Sigma), \Sigma = \Sigma_0 \in P_k(\mathbb{R})\}$, the Fisher-Rao distance $d_{R,\mathcal{M}_\Sigma}$ between two distributions is given by the Mahalanobis distance [24]:

$$d_{R,\mathcal{M}_\Sigma}\big(\mathcal{N}(\boldsymbol{\mu}_1, \Sigma), \mathcal{N}(\boldsymbol{\mu}_2, \Sigma)\big) = \sqrt{(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \Sigma^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)}. \tag{25}$$

The Mahalanobis distance is also used for OOD detection [20] and its performance is compared to the FRD through several experiments in Section 3.

B IGEOOD Algorithms

In this section, we provide pseudo-code for calculating the IGEOOD score from the logits (Algorithm 1) and from the latent features (Algorithm 2). The BLACK-BOX IGEOOD score is obtained with Algorithm 1 by setting $\varepsilon = 0$, while the GREY-BOX IGEOOD score is obtained with $\varepsilon > 0$. We calculated the centroid of the logits for the in-distribution training set by optimizing the objective function given by Eq. (3) through a gradient descent algorithm for each DNN. We used a constant learning rate of 0.01 and a batch size of 128 for 100 epochs. Finally, the WHITE-BOX IGEOOD score is obtained by combining the outputs of Algorithms 1 and 2 through fitting the multiplicative weights α through a logistic function classifier on a labeled mixture dataset composed from in- and out-of-distribution data according to a validation dataset, which leads to expression Eq. (10).

Algorithm 1: Evaluating IGEOOD score based on the logits.

Input : Test sample $\boldsymbol{x}$, temperature T and noise magnitude ε parameters, and training set $\mathcal{D}_N = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$.
Output : FR_0: IGEOOD score in the logits level.

`// Offline computation`

Calculate the logits centroids from the training data:

$\boldsymbol{\mu}_y \triangleq \min_{\boldsymbol{\mu}\in\mathbb{R}^C} \frac{1}{N_y} \sum_{\forall\, i\,:\, y_i = y} 2 \arccos\left(\sum_{y'\in\mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}\big(y'|f(\boldsymbol{x}_i)\big) q_{\boldsymbol{\theta}}\big(y'|\boldsymbol{\mu}\big)}\right)$

`// Online computation`

Add small perturbation to $\boldsymbol{x}$:

$\widetilde{\boldsymbol{x}} \leftarrow \boldsymbol{x} + \varepsilon \odot \operatorname{sign}\left[\nabla_{\boldsymbol{x}} \sum_y 2 \arccos\left(\sum_{y'\in\mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}(y'|f(\boldsymbol{x})) q_{\boldsymbol{\theta}}(y'|\boldsymbol{\mu}_y)}\right)\right]$

return $\mathrm{FR}_0(\widetilde{\boldsymbol{x}}) \leftarrow \sum_y 2 \arccos\left(\sum_{y'\in\mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}(y'|f(\widetilde{\boldsymbol{x}})) q_{\boldsymbol{\theta}}(y'|\boldsymbol{\mu}_y)}\right)$

Algorithm 2: Evaluating feature-wise IGEOOD score.

Input : Test sample $\boldsymbol{x}$ and training set $\mathcal{D}_N = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$.
Output : FR_ℓ: feature-wise IGEOOD scores.

for *each feature* $\ell \in \{1, \ldots, L\}$ **do**

`// Offline computation`

Calculate the means: $\boldsymbol{\mu}_y^{(\ell)} \leftarrow \frac{1}{N_y} \sum_{i: y_i = y} f^{(\ell)}(\boldsymbol{x}_i)$

Calculate the diagonal standard deviation matrix:

$\sigma_{jj}^{(\ell)} \leftarrow \sqrt{\frac{1}{N} \sum_{y\in\mathcal{Y}} \sum_{\forall i\,:\, y_i = y} \left(f_j^{(\ell)}(\boldsymbol{x}_i) - \mu_{y,j}^{(\ell)}\right)^2}$

`// Online computation`

Compute the OOD score for ℓ:

$\mathrm{FR}_\ell(\boldsymbol{x}) \leftarrow \min_y \sqrt{\sum_{j=1}^k \rho_{\mathrm{FR}}\left(\left(\mu_{y,j}^{(\ell)}, \sigma_{jj}^{(\ell)}\right), \left(f_j^{(\ell)}(\boldsymbol{x}), \sigma_{jj}^{(\ell)}\right)\right)^2}$

end

return $\big(\mathrm{FR}_1(\boldsymbol{x}), \ldots, \mathrm{FR}_L(\boldsymbol{x})\big)$

Algorithm 3: Evaluating feature-wise IGEOOD+ score.

Input : Test sample $\boldsymbol{x}$, training set $\mathcal{D}_N = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ and M OOD samples $\mathcal{O}_M = \{\boldsymbol{x}'_i\}_{i=1}^M$.

Output : FR_ℓ and FR'_ℓ: feature-wise IGEOOD+ scores.

for *each feature* $\ell \in \{1, \ldots, L\}$ **do**

```
// Offline computation
```

Calculate class conditional means: $\boldsymbol{\mu}_y^{(\ell)} \leftarrow \frac{1}{N_y} \sum_{i:y_i=y} f^{(\ell)}(\boldsymbol{x}_i)$

Calculate OOD samples mean: $\boldsymbol{\mu}^{(\ell)\prime} \leftarrow \frac{1}{M} \sum_{i=1}^M f^{(\ell)}(\boldsymbol{x}'_i)$

Calculate the diagonal standard deviation matrix from training data:

$\sigma_{jj}^{(\ell)} \leftarrow \sqrt{\frac{1}{N} \sum_{y \in \mathcal{Y}} \sum_{\forall i\,:\, y_i = y} \left(f_j^{(\ell)}(\boldsymbol{x}_i) - \mu_{y,j}^{(\ell)} \right)^2}$

Calculate the diagonal standard deviation matrix from OOD data:

$\sigma_{jj}^{(\ell)\prime} \leftarrow \sqrt{\frac{1}{M} \sum_{i=i}^M \left(f_j^{(\ell)}(\boldsymbol{x}'_i) - \mu_j^{(\ell)\prime} \right)^2}$

```
// Online computation
```

Compute the OOD scores for ℓ:

$\mathrm{FR}_\ell(\boldsymbol{x}) \leftarrow \min_y \sqrt{\sum_{j=1}^k \rho_{\mathrm{FR}} \left(\left(\mu_{y,j}^{(\ell)}, \sigma_{jj}^{(\ell)} \right), \left(f_j^{(\ell)}(\boldsymbol{x}), \sigma_{jj}^{(\ell)} \right) \right)^2}$

$\mathrm{FR}'_\ell(\boldsymbol{x}) \leftarrow \min_y \sqrt{\sum_{j=1}^k \rho_{\mathrm{FR}} \left(\left(\mu_j^{(\ell)\prime}, \sigma_{jj}^{(\ell)\prime} \right), \left(f_j^{(\ell)}(\boldsymbol{x}), \sigma_{jj}^{(\ell)} \right) \right)^2}$

end

return $\left(\mathrm{FR}_1(\boldsymbol{x}), \mathrm{FR}'_1(\boldsymbol{x}) \ldots, \mathrm{FR}_L(\boldsymbol{x}), \mathrm{FR}'_L(\boldsymbol{x})\right)$

Note that the calculation of the training logits centroids $\boldsymbol{\mu}_y$, as well as the latent representations mean vectors $\boldsymbol{\mu}_y^{(\ell)}$ and standard covariance matrices $\boldsymbol{\sigma}^{(\ell)}$ is performed beforehand, prior to inference. In this way, we retrieve the objects from memory at inference time. Also, we define k as the cardinality of feature ℓ, or $|f^{(\ell)}|$ and ρ_{FR} as the Fisher-Rao distance between univariate Gaussian distribution given by expression Eq. (6).

B.1 Logits centroids estimation details

In order to obtain the logits centroids given the Fisher-Rao distance in the space of softmax probability distributions, we designed a simple optimization problem. This problem aims to minimize the average distance between the class conditional training samples and the centroids as given by Eq. (3). We initialized the C centroids, where C is the number of classes of a given model, with the identity matrix of size $C \times C$. Note that the initial centroid for class i is given by the matrix's line number i. We minimized the expression in Eq. (3) with a gradient descent optimizer for 100 epochs with a fixed learning rate equal to 0.01 for every DNN model and in-distribution dataset.

B.2 Feature importance regression details

For both Mahalanobis and IGEOOD methods, we fitted a logistic regression model with cross-validation using 1,000 OOD and 1,000 in-distribution data samples. Each regression parameter multiplies the layer scores outputs with the objective function of maximizing the TNR at TPR-95%. We set the maximum number of iterations to be 100.

B.3 Covariance matrix estimation details

We model the latent output probability distributions as Gaussian distributions with diagonal covariance matrix calculated with expression Eq. (5). We chose this model motivated by a closed form for the FRD and by observing that the standard covariance matrix for the latent features is often ill-conditioned. The condition number of a matrix correlates to its numerical stability, i.e., a small rounding error in its estimation may cause a large difference in its values. So, a matrix with a low condition number is said to be well-conditioned, while a matrix with a high condition number is said to be ill-conditioned. We calculate the condition number of the covariance matrices with the formula $\kappa(\Sigma) = \left\|\Sigma^{-1}\right\|_\infty \|\Sigma\|_\infty$, where $\|\cdot\|_\infty$ is the infinity norm. For each of the four dense blocks outputs of a DenseNet trained on CIFAR-10, we obtained the condition numbers $\kappa_\Sigma = \{2.8\text{e}10, 3.5\text{e}6, 3.1\text{e}5, 3.5\text{e}21\}$. While for the *diagonal* covariance matrix, we obtained smaller values of condition numbers: $\kappa_{\Sigma_D} = \{1.0\text{e}3, 3.0\text{e}1, 1.4\text{e}1, 7.6\text{e}20\}$. We associate the high value for the last feature due to the last feature being high dimensional and coarse, i.e., most of the values in the diagonal are close to zero.

C Detailed Experimental Setup

C.1 DNN models and training details

We describe the DNN models used in the experiments:

- **DenseNet.** Densely Connected Convolutional Networks [16], or DenseNet for short, are compositions of dense blocks, which are composed of multiple layers directly connected to every other layer in a feed-forward fashion. In this work, we use the DenseNet-BC-100 architecture. The BC stands for a model with 1x1 convolutional bottleneck (B) layers and channel number compression (C) of 0.5. The models have depth $L = 100$ and growth rate $k = 12$. We consider the outputs of each dense block after the transition layer (3 in total) and the first convolutional layer output as the latent features. After an averaging pooling, the latent features have dimensions $\boldsymbol{\mathcal{F}}_1 = \{24, 108, 150, 342\}$.
- **ResNet.** Residual Networks [11], or ResNet, are deep neural networks composed of residual blocks. Each residual block is composed of layers connected in a feed-forward manner plus a skip connection. We use the ResNet with 34 layers pre-trained on CIFAR-10, CIFAR-100, and SVHN datasets. We take the output of every residual block (4 in total) and the first convolutional layer for calculating the score on the WHITE-BOX setting. After an averaging pooling, the latent features have dimensions $\boldsymbol{\mathcal{F}}_2 = \{64, 64, 128, 256, 512\}$.

We train each model by minimizing the cross-entropy loss using SGD with Nesterov momentum equal to 0.9, weight decay equal to 0.0001, and a multi-step learning rate schedule starting at 0.1 for 300 epochs. The pre-trained models is available at [3]. We report their test set accuracy in Table 3 with the softmax function and by replacing it with the Fisher-Rao distance between the training class-conditional centroids and the test sample outputs. Also, it is worth noting that one high-end GPU is sufficient for running every experiment presented in this work.

Table 3: Test set accuracy in percentage for ResNet and DenseNet architectures pre-trained on CIFAR-10, CIFAR-100 and SVHN.

	ResNet-34		DenseNet-BC-100	
In-Dataset	Softmax	Fisher-Rao	Softmax	Fisher-Rao
CIFAR-10	93.52	**93.53**	95.20	95.20
CIFAR-100	**77.11**	77.09	77.62	**77.63**
SVHN	96.61	96.61	95.16	95.16

C.2 Evaluation metrics

We introduce below standard binary classification performance metrics used to evaluate the OOD discriminators.

- **True Negative Rate at 95% True Positive Rate (TNR at TPR-95% (%)).** This metric measures the true negative rate (TNR) at a specific true positive rate (TPR). The operating point is chosen such that the TPR of the in-distribution test set is fixed to some value, 95% in this case. Mathematically, let TP, TN, FP, and FN denote true positive, true negative, false positive and false negative, respectively. We measure $\text{TNR} = \text{TN}/(\text{FP} + \text{TN})$, when $\text{TPR} = \text{TP}/(\text{TP} + \text{FN})$ is 95%.
- **Area Under the Receiver Operating Characteristic curve (AUROC (%)).** The ROC curve is constructed by plotting the true positive rate (TPR) against the false positive rate $= \text{FP}/(\text{FP} + \text{TN})$ at various threshold values. The area under this curve tells how much the OOD discriminator can distinguish in-distribution and OOD data in a threshold-independent manner.
- **Area Under the Precision-Recall curve (AUPR (%)).** The PR curve plots the precision $= \text{TP}/(\text{TP} + \text{FP})$ against the recall $= \text{TP}/(\text{TP} + \text{FN})$ by varying a threshold. For the experiments, in-distribution data are specified as positives while OOD data as negative.

[3] `http://github.com/igeood/Igeood`

Note that the TNR at TPR-95% is especially important because we want to identify OOD data and preserve a sufficiently good performance on identifying in-distribution data, which is not the case for the other metrics.

C.3 Datasets

In our experiments, we use natural image examples from the following image classification and synthetic datasets. We normalize the test samples with the in-distribution dataset statistics.

- **CIFAR-10.** The CIFAR-10 [18] dataset is composed of 32×32 natural images of 10 different classes, e.g., airplane, ship, bird, etc. The training set is composed of 50,000 images, and the test set is composed of 10,000 images. The classes are approximately equally distributed (5,000 examples each label). The CIFAR-10 dataset is under the MIT license.
- **CIFAR-100.** The CIFAR-100 [18] dataset contains similar natural images to the CIFAR-10 dataset, but with 90 additional categories. Its set repartition is also 50,000 for training and 10,000 for the test set. We expect around 500 samples for each class of the training set. It is also under the MIT license.
- **SVHN.** The SVHN [28] dataset collects street house numbers for digit classification. It contains 73,257 training and 26,032 test RGB images of size 32×32 of printed digits (from 0 to 9). We take only the first 10,000 examples of the test set for evaluating the methods to have a balanced dataset of in-distribution and out-of-distribution data. This dataset is subject to a non-commercial license.
- **Tiny-ImageNet.** The Tiny-ImageNet [19] dataset is a subset of the large-scale natural image dataset ImageNet [9]. It contains 200 different classes and 10,000 test examples. We downsize the images from their original resolution to images of dimension $32 \times 32 \times 3$.
- **LSUN.** The LSUN [44] dataset, which has equally 10,000 test examples, is used for the large-scale scene classification of different scene categories (e.g., bedroom, bridge, kitchen, etc.). Similarly, we resize the images following the same procedure for the Tiny-ImageNet dataset. LSUN is under the Apache 2.0 license.
- **iSUN.** The iSUN [43] dataset consists of selected natural scene images from the SUN [41] dataset. The test set has 8925 images, which we downsample to $32 \times 32 \times 3$. We use this dataset as a source of OOD for validation purposes as an independent dataset from the test OOD data.
- **Textures.** The Describable Textures Dataset (DTD) [7] is a collection of textural pattern images observed in nature. It contains 47 categories totaling 5640 images of various sizes, which are resized and center cropped to fit into the input size of 32×32.
- **Chars74K.** The Chars74K dataset [8] contains 74,000 samples of 62 classes of characters found in natural images, handwritten text, and synthesized from computer fonts. We used as OOD data only the *EnglishImg* dataset split, which contains 7705 characters from natural scenes. We resized and center-cropped the images.
- **Places365.** The Places365 dataset [46] contains images of 365 natural scenes categories. We used the small images validation split as OOD data in our experiments. It contains 36,500 RGB images which were downsampled from 256×256 to 32×32.
- **Gaussian.** For the Gaussian dataset, we generated 10,000 synthetic RGB images from 2D Gaussian noise, where each RGB pixel is sampled from an i.i.d Gaussian distribution with mean 0.5 and variance 1.0. The pixel values are clipped to $[0, 1]$ interval. This synthetic data was introduced in previous work as an easy benchmark [13].

C.4 Adversarial data generation

We generate adversarial samples from the in-distribution dataset using the fast gradient sign method (FGSM). This method works by exploiting the gradients of the neural network to create a non-targeted adversarial attack. For an input image $\boldsymbol{x}_i$, the method computes the sign of the gradients of the loss function J with respect to the input image to create a new image $\boldsymbol{x}_i^{\mathrm{adv}}$ that maximizes the loss as given by expression Eq. (26). This fabricated image is called an adversarial image, which we use for tuning the hyperparameters of the OOD detection methods in the WHITE-BOX case. Mathematically,

$$\boldsymbol{x}_i^{\mathrm{adv}} = \boldsymbol{x}_i + \varepsilon^{\mathrm{adv}} \odot \operatorname{sign}(\nabla_{\boldsymbol{x}_i} J(\boldsymbol{\theta}, \boldsymbol{x}_i, y_i)), \tag{26}$$

where $\varepsilon^{\mathrm{adv}} > 0$ is the additive noise magnitude parameter. Table 4 shows the resulting L_∞ mean perturbation and classification accuracy on adversarial samples.

C.5 Mahalanobis distance-based confidence score

The Mahalanobis-based method in [20] fits the DNN training data features as class-conditional Gaussian distributions. These use the outputs of every DNN latent block to leverage useful information for discrimination.

Table 4: The L_∞ mean perturbation used to generate adversarial data with FGSM algorithm and classification accuracy on adversarial samples for the DNN models and in-distribution datasets.

	CIFAR-10		CIFAR-100		SVHN	
	L_∞	Acc.	L_∞	Acc.	L_∞	Acc.
DenseNet-100	0.21	19.5%	0.20	4.45%	0.32	54.7%
ResNet-34	0.21	23.7%	0.20	12.49%	0.25	50.0%

For a test sample $\boldsymbol{x}$, the confidence score from the ℓ-th feature is calculated based on the Mahalanobis distance between $f^{(\ell)}(\boldsymbol{x})$ and the closest class-conditional distribution:

$$M_\ell(\boldsymbol{x}) = \max_y - \left(f^{(\ell)}(\boldsymbol{x}) - \widehat{\boldsymbol{\mu}}_y^{(\ell)}\right)^\top \widehat{\Sigma}_\ell^{-1} \left(f^{(\ell)}(\boldsymbol{x}) - \widehat{\boldsymbol{\mu}}_y^{(\ell)}\right), \tag{27}$$

where $f^{(\ell)}(\boldsymbol{x})$ is the ℓ-th latent feature output, and $\widehat{\boldsymbol{\mu}}_y^{(\ell)}$ and $\widehat{\Sigma}_\ell$ are, respectively, the empirical class mean and covariance matrix estimates. The covariance matrix is often not full rank, so the pseudo-inverse is calculated instead of the inverse. In addition, input pre-processing and feature ensemble are also used to boost performance. A logistic regression model learns the multiplicative weights α_ℓ for each layer score, which predicts 1 for in-distribution and 0 for OOD examples from a mixture validation dataset. Finally, the Mahalanobis-based discriminator is given by thresholding expression $\sum_\ell \alpha_\ell M_\ell(\boldsymbol{x})$.

D Additional Out-Of-Distribution detection results

Table 5 shows compares IGEOOD to the current literetaure in the setup where no OOD data is available for tuning. We show in Tables 6 and 7 additional results referring to the right-hand column and left-hand column of Table 1, respectively.

Table 5: TNR at TPR-95% (%) performance in a WHITE-BOX setting considering the original results from [20, 34, 15, 47] without access to OOD samples for hyperparameter tuning.

	OOD dataset	CIFAR-10	CIFAR-100	SVHN
		Mahalanobis / Gram Matrix / DeConf-C / Res-Flow / IGEOOD / IGEOOD+		
DenseNet	iSUN	94.3/99.0/**99.4**/ - /94.5/95.8	84.8/95.9/**98.4**/ - /93.8/92.2	**99.9**/99.4/ - / - /98.2/98.6
	LSUN	97.2/**99.5**/99.4/98.1/96.4/97.2	91.4/97.2/**98.7**/95.8/95.1/94.4	**100**/99.5/ - /**100**/97.3/97.0
	TinyImgNet	94.9/98.8/**99.1**/96.1/93.4/94.5	87.2/95.7/**98.6**/91.5/94.3/94.0	**99.9**/99.1/ - /**99.9**/98.1/96.8
	SVHN/C-10	89.9/96.1/**98.8**/86.1/94.3/95.7	62.2/89.3/**95.9**/48.9/90.1/90.6	**90.0**/80.4/ - /**90.0**/89.5/86.6
	average	94.1/98.3/**99.2**/93.4/94.6/95.8	81.4/94.5/**97.9**/78.7/93.3/92.8	**97.4**/94.6/ - /96.6/95.8/94.8
ResNet	iSUN	96.8/**99.3**/88.8/ - /95.3/95.0	87.9/**94.8**/75.3/ - /89.4/91.0	**100**/99.4/ - / - /**99.8**/**99.9**
	LSUN	98.1/**99.6**/90.9/99.1/97.7/97.7	56.6/**96.6**/76.8/70.4/88.6/93.9	**99.9**/99.6/ - /**100**/**99.8**/**100**
	TinyImgNet	95.5/**98.7**/81.4/98.0/94.3/94.2	70.3/**94.8**/76.5/77.5/86.2/90.1	99.2/99.3/ - /**99.9**/**99.6**/**99.6**
	SVHN/C-10	75.8/97.6/89.5/91.0/**98.2**/97.7	41.9/**80.8**/55.1/74.1/75.2/78.5	94.1/85.8/ - /96.6/96.7/**97.3**
	average	91.5/**98.8**/87.6/96.0/96.3/96.2	64.2/**91.7**/71.0/74.0/84.8/88.4	98.3/96.0/ - /98.8/**99.0**/**99.2**

E Histograms

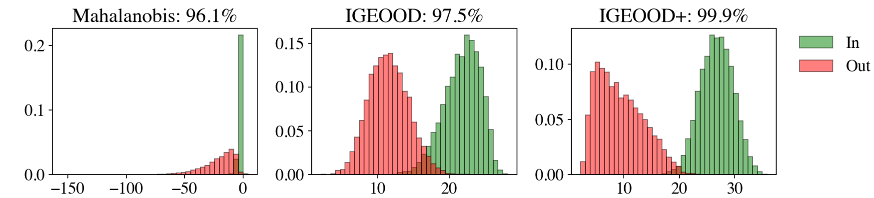
Figures 3 and 2 display histograms for the OOD detection score for IGEOOD and [20] in the WHITE-BOX with adversarial validation and WHITE-BOX with OOD data validation, respectively.

Table 6: WHITE-BOX extended results. Validation on OOD data.

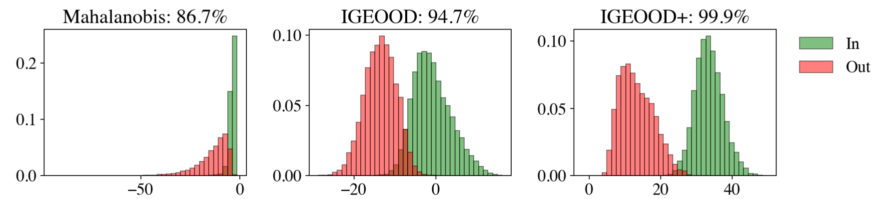
In-dist. (model)	OOD dataset	TNR at TPR-95%	AUROC	AUPR
		Mahalanobis [20] / IGEOOD+		
CIFAR-10 (DenseNet)	Chars	91.3/**99.4**	97.5/**99.9**	97.7/**99.9**
	CIFAR-100	21.4/**56.6**	67.3/**90.7**	64.4/**90.8**
	TinyImgNet	96.9/**99.8**	99.3/**99.9**	99.3/**99.9**
	LSUN	98.2/**99.9**	99.5/**100**	99.5/**100**
	Places365	18.1/**80.2**	72.7/**95.7**	72.8/**95.4**
	SVHN	90.1/**99.9**	97.3/**100**	97.3/**100**
	Textures	84.1/**97.4**	95.6/**99.5**	94.7/**99.5**
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	97.3/**99.8**	99.4/**100**	99.4/**100**
	average	77.5±31/**92.6**±14	92.1±12/**98.4**±3.0	91.7±13/**98.4**±3.0
CIFAR-100 (DenseNet)	Chars	62.9/**97.5**	94.0/**99.4**	95.8/**99.4**
	CIFAR-10	9.1/**22.7**	60.8/**80.7**	60.1/**83.0**
	TinyImgNet	87.1/**99.5**	97.4/**99.9**	97.4/**99.9**
	LSUN	91.1/**99.9**	97.8/**100**	98.1/**100**
	Places365	5.9/**58.2**	54.8/**90.0**	54.7/**89.2**
	SVHN	79.0/**99.6**	96.8/**99.9**	94.1/**99.9**
	Textures	70.3/**90.2**	91.4/**98.1**	94.3/**98.2**
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	86.4/**99.7**	96.8/**99.9**	97.7/**99.9**
	average	67.7±28/**90.2**±21	87.8±13/**97.7**±5.0	88.0±12/**97.8**±5.0
SVHN (DenseNet)	Chars	78.7/**92.2**	96.1/**98.4**	**98.9**/98.5
	CIFAR-10	91.6/**98.3**	98.0/**99.6**	99.4/**99.6**
	CIFAR-100	92.9/**95.3**	98.2/**99.1**	**99.4**/99.2
	TinyImgNet	99.9/**99.9**	99.8/**99.9**	**99.9/99.9**
	LSUN	**99.9/99.9**	99.8/**100**	99.7/**100**
	Places365	94.7/**98.3**	98.3/**99.6**	98.4/**99.7**
	Textures	98.2/**98.5**	99.4/**99.6**	**99.9**/99.6
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	**99.9/99.9**	99.8/**99.9**	**99.9/99.9**
	average	95.1±8.0/**98.0**±2.0	98.8±1.0/**99.6**±0.1	99.5±1.0/**99.6**±0.1
CIFAR-10 (ResNet)	Chars	93.6/**99.3**	98.6/**99.8**	99.1/**99.8**
	CIFAR-100	44.9/**51.3**	87.4/**90.9**	87.8/**91.7**
	TinyImgNet	96.8/**99.6**	99.4/**99.9**	99.4/**99.9**
	LSUN	98.3/**99.9**	99.6/**100**	99.6/**100**
	Places365	45.8/**77.6**	88.1/**95.6**	88.1/**95.5**
	SVHN	96.1/**99.8**	99.0/**99.9**	98.1/**99.9**
	Textures	84.3/**97.0**	97.3/**99.4**	98.6/**99.4**
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	97.2/**99.9**	99.4/**100**	99.5/**100**
	average	84.1±23/**91.6**±16	96.5±4.0/**98.4**±3.0	96.7±4.0/**98.5**±3.0
CIFAR-100 (ResNet)	Chars	63.8/**97.8**	94.0/**99.5**	96.0/**99.5**
	CIFAR-10	18.0/**30.8**	76.6/**85.3**	76.4/**87.8**
	TinyImgNet	90.1/**99.6**	97.9/**99.9**	98.0/**99.9**
	LSUN	92.4/**100**	98.3/**100**	98.5/**100**
	Places365	23.5/**59.1**	76.8/**91.2**	76.0/**91.4**
	SVHN	88.4/**99.7**	97.7/**99.9**	95.2/**99.9**
	Textures	71.6/**90.7**	93.9/**98.2**	96.6/**98.1**
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	89.4/**99.8**	97.7/**99.9**	98.0/**99.9**
	average	70.8±30/**86.4**±23	92.5±10/**97.1**±5.0	92.7±10/**97.4**±4.0
SVHN (ResNet)	Chars	84.9/**92.4**	97.0/**98.4**	**99.0**/98.5
	CIFAR-10	98.0/**99.7**	99.2/**99.9**	99.7/**99.9**
	CIFAR-100	98.3/**99.1**	99.3/**99.7**	**99.8/99.8**
	TinyImgNet	**99.9/99.9**	99.9/**100**	**100/100**
	LSUN	**99.9/99.9**	99.9/**100**	**100/100**
	Places365	98.4/**99.6**	99.3/**99.9**	99.8/**99.9**
	Textures	99.0/**99.9**	99.7/**99.9**	**99.9/99.9**
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	**100/100**	99.9/**100**	**100/100**
	average	97.6±6.0/**98.9**±2.0	99.4±1.0/**99.7**±0.1	**99.8**±1.0/**99.8**±0.1
Avg. and std. of avg. values		82.1±11/**92.9**±4.0	94.5±4.0/**98.5**±1.0	94.7±4.0/**98.6**±1.0

Table 7: WHITE-BOX extended results. Validation on adversarial (FGSM) data.

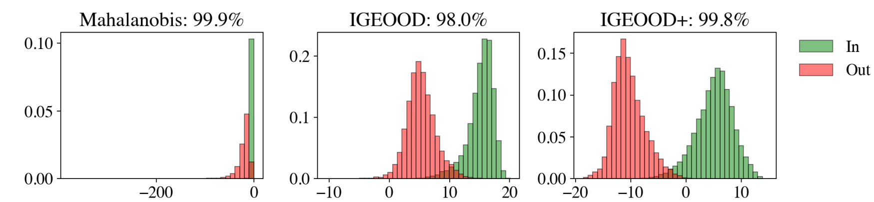
In-dist. (model)	OOD dataset	TNR at TPR-95%	AUROC	AUPR
		Mahalanobis [20] / IGEOOD		
CIFAR-10 (DenseNet)	Chars	**88.5**/87.3	**97.7**/97.7	**98.3**/98.3
	CIFAR-100	21.5/**26.4**	68.0/**77.7**	66.3/**75.5**
	TinyImageNet	**93.9**/93.4	98.6/**98.7**	98.6/**98.7**
	LSUN	96.3/**96.4**	99.1/**99.2**	99.1/**99.2**
	Places365	17.8/**23.2**	70.0/**77.9**	40.1/**76.3**
	SVHN	87.0/**94.3**	97.2/**98.7**	93.7/**97.3**
	Textures	83.6/**86.0**	95.8/**97.2**	97.3/**98.2**
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	94.3/**94.5**	98.8/**98.9**	98.9/**99.0**
	average	75.9±30/**77.9**±29	91.7±12/**94.0**±9.0	88.0±20/**93.6**±10
CIFAR-100 (DenseNet)	CIFAR-10	1.1/**5.7**	43.5/**62.6**	46.7/**62.6**
	Chars	53.9/**59.6**	**92.2**/92.0	**94.5**/93.9
	TinyImageNet	86.4/**94.3**	97.4/**98.8**	97.5/**98.9**
	LSUN	88.6/**95.1**	97.6/**98.9**	97.9/**98.9**
	Places365	5.5/**13.0**	56.6/**71.0**	57.5/**71.0**
	SVHN	56.1/**90.1**	91.8/**98.0**	85.4/**96.2**
	Textures	67.5/**86.7**	91.2/**97.4**	94.4/**98.4**
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	84.8/**93.8**	97.2/**98.7**	97.6/**98.8**
	average	60.4±34/**70.9**±35	85.3±19/**90.8**±13	85.7±19/**91.0**±13
SVHN (DenseNet)	CIFAR-10	**90.6**/89.5	97.7/**97.8**	99.1/**99.2**
	CIFAR-100	**91.8**/88.4	**98.0**/97.7	**99.2**/99.1
	Chars	**72.3**/70.5	**95.2**/94.5	**98.5**/98.3
	TinyImageNet	**99.5**/98.1	**99.6**/99.3	99.5/**99.8**
	LSUN	**99.9**/97.3	**99.8**/99.1	**99.9**/99.7
	Places365	**94.3**/91.9	**98.3**/98.2	98.1/**99.3**
	Textures	95.3/**97.1**	98.8/**99.3**	99.6/**99.8**
	Gaussian	**100/100**	**100/99.9**	**100/100**
	iSUN	**99.9**/98.2	**99.8**/99.3	**99.9**/99.8
	average	**93.7**±8.0/92.3±9.0	**98.6**±1.0/98.3±2.0	**99.3**±1.0/99.4±0
CIFAR-10 (ResNet)	CIFAR-100	**36.5**/21.5	**84.5**/63.3	**84.3**/58.1
	Chars	82.0/**90.9**	96.9/**98.3**	97.7/**98.7**
	TinyImageNet	**96.2**/94.3	**99.2**/98.0	**99.2**/96.7
	LSUN	**98.2**/97.7	**99.5**/99.2	**99.5**/98.9
	Places365	**34.8**/15.9	**85.0**/60.1	**84.2**/24.4
	SVHN	81.0/**98.2**	96.6/**99.3**	93.7/**97.5**
	Textures	**81.7**/81.6	**96.7**/93.4	**98.2**/94.3
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	**96.8**/95.3	**99.3**/98.6	**99.3**/98.1
	average	**78.6**±24/77.3±32	**95.3**±6.0/90.0±15	**95.1**±6.0/85.2±25
CIFAR-100 (ResNet)	CIFAR-10	3.0/**5.0**	**61.0**/59.6	**63.7**/60.6
	Chars	39.9/**55.1**	85.6/**90.4**	88.1/**92.5**
	TinyImageNet	**88.7**/86.2	**97.6**/97.3	**97.6**/97.3
	LSUN	**91.3**/88.6	**98.0**/97.8	**98.3**/98.0
	Places365	8.0/**8.6**	**67.9**/63.0	**66.8**/61.7
	SVHN	31.6/**75.2**	82.9/**95.8**	68.8/**92.7**
	Textures	65.9/**78.1**	91.9/**95.6**	95.2/**97.6**
	Gaussian	**100/100**	**100/100**	**100/100**
	iSUN	87.9/**89.4**	97.4/**97.8**	97.6/**97.7**
	average	57.4±36/**65.1**±33	**86.9**±13/88.6±15	86.2±14/**88.7**±15
SVHN (ResNet)	CIFAR-10	**97.1**/96.7	99.1/**99.2**	**99.7**/99.7
	CIFAR-100	**97.5**/96.2	**99.1**/99.1	**99.7**/99.6
	Chars	**75.4**/55.1	**95.3**/89.1	**98.5**/96.0
	TinyImageNet	**99.9**/99.6	**99.9/99.9**	**99.9/99.9**
	LSUN	**100**/99.8	**99.9/99.9**	**100/100**
	Places365	**98.1**/97.0	**99.2**/99.2	**99.2**/99.0
	Textures	**98.9**/98.4	**99.6**/99.6	**99.9/99.9**
	Gaussian	**100/100**	**99.9/100**	**100/100**
	iSUN	**100**/99.8	99.8/**99.9**	**99.9/100**
	average	**96.3**±8.0/93.6±14	**99.1**±1.0/98.4±3.0	**99.6**±0/99.3±1.0
Avg. and std. of avg. values		77.0±15/**79.5**±10	92.8±5.4/**93.4**±3.9	92.3±5.9/**92.9**±5.2



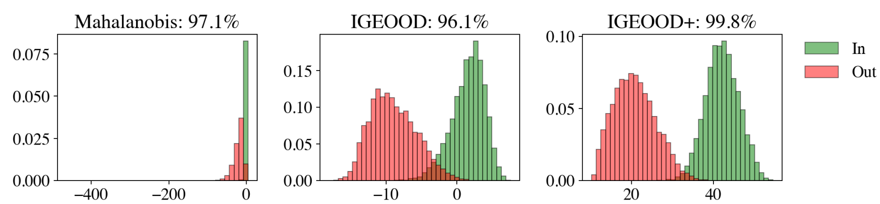
(a) DenseNet on CIFAR-10.



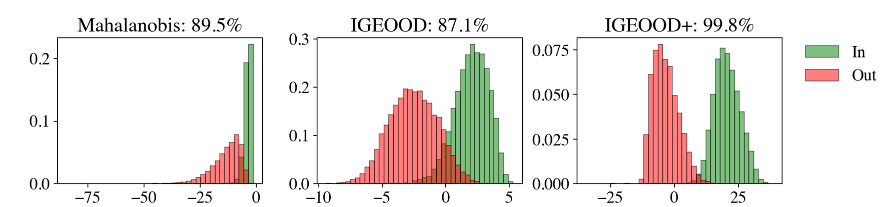
(b) DenseNet on CIFAR-100.



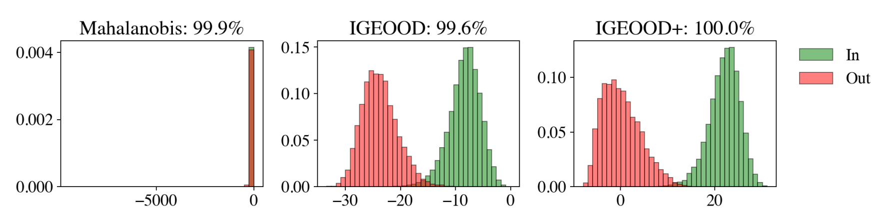
(c) DenseNet on SVHN.



(d) ResNet on CIFAR-10.

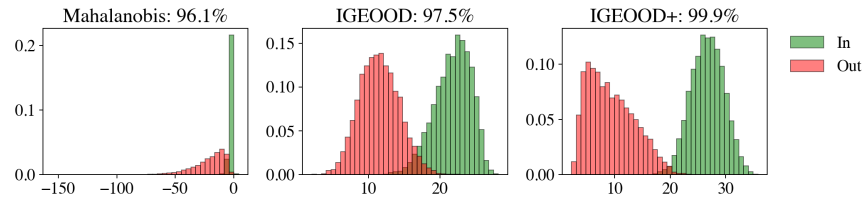


(e) ResNet on CIFAR-100.

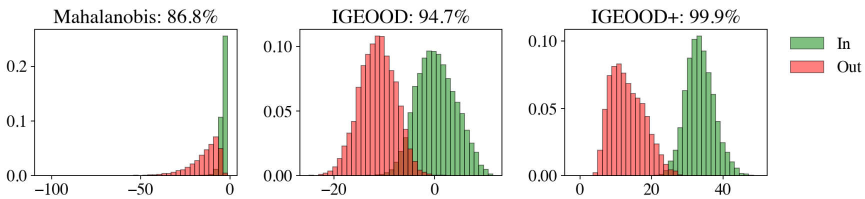


(f) ResNet on SVHN.

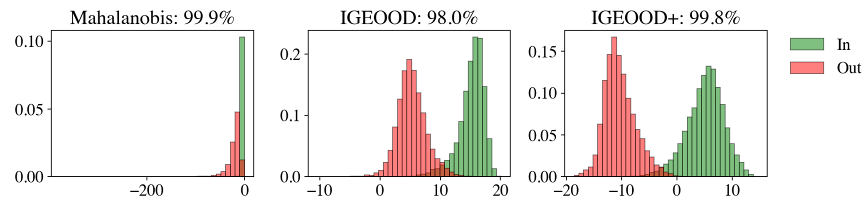
Figure 2: WHITE-BOX setup with adversarial data validation. TinyImageNet as OOD dataset.



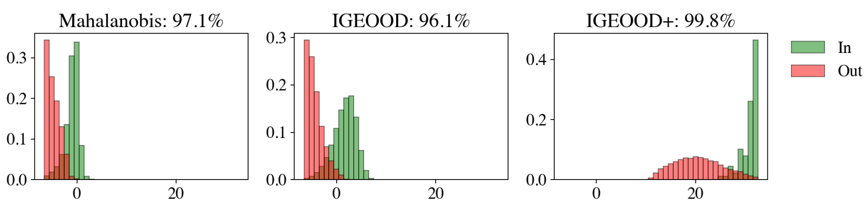
(a) DenseNet on CIFAR-10.



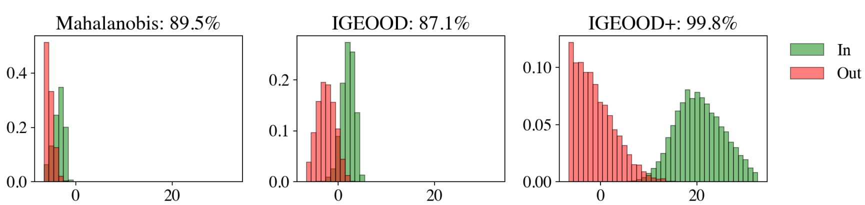
(b) DenseNet on CIFAR-100.



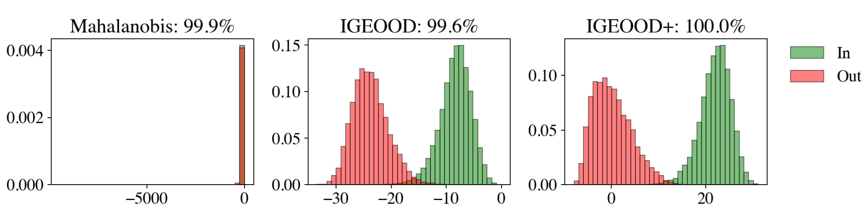
(c) DenseNet on SVHN.



(d) ResNet on CIFAR-10.



(e) ResNet on CIFAR-100.



(f) ResNet on SVHN.

Figure 3: WHITE-BOX setup with validation on OOD data. TinyImageNet as OOD dataset.