IGEOOD: AN INFORMATION GEOMETRY APPROACH TO OUT-OF-DISTRIBUTION DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reliable out-of-distribution (OOD) detection is a fundamental step towards a safer implementation of modern machine learning (ML) systems. In this paper, we introduce IGEOOD, an effective method for detecting OOD samples. IGEOOD applies to any pre-trained neural network, does not require OOD samples or assumptions on the OOD data, and works under different degrees of access to the ML model. 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 outperforms competing state-of-the-art methods on a variety of network architectures and datasets, e.g., by increasing up to 8.5% the average TNR at TPR-95% across six different models and nine different OOD datasets.

1 INTRODUCTION

Deep neural networks (DNNs) achieve state-of-the-art in several classification tasks as they generalize well to data distribution similar to the training set. Whereas, in many practical applications, the training environment does not reflect well enough the open-world (Quionero-Candela et al., 2009). Indeed, the actual real-life environment is often non-stationary and sometimes with unpredictable events. As a consequence, matching the training scenario to reality can be either not possible or too complex. The failure of machine learning (ML) models to adapt to non-stationary distributions could limit their adoption in critical systems (e.g., autonomous devices, healthcare applications).

Out-Of-Distribution (OOD) or novelty detection is one of the main objectives in conceiving reliable ML systems (Amodei et al., 2016). An application of these methods arises in monitoring ML-based online services for periodically shifting 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 often only a single test example is available. Besides this, these metrics for detection are subject to potential limitations inherent to practical scenarios (e.g., only accessing the last layer or all the intermediate layers) depending on the level of information access in the ML model.

The baseline approach for OOD detection relies on the predictive uncertainty of DNNs. Hendrycks & Gimpel (2017) demonstrated that OOD samples, in general, induce DNN classifiers to output less confident softmax scores. Whereas existing state-of-the-art methods on classification problems still output high accuracy even under dataset shift. For instance, Ovadia et al. (2019) show that as the accuracy of the underlying DNN increases, the supervisors' outlier detection accuracy also improves. Unfortunately, also the variance increases. Henriksson et al. (2021) observed that small changes in model parameters that marginally impact the accuracy could have a degrading impact on the performance of the OOD discriminator. This challenge is not exclusive to discriminative models. Deep generative models also fall short in discerning OOD from in-distribution samples. Nalisnick et al. (2019) raise awareness of the fact that deep generative models also may output a higher likelihood to OOD samples. They show that, even though the samples from the in-distribution CIFAR-10 (Krizhevsky et al., 2009) dataset (e.g., cats, dogs, airplanes, ships) are conceptually and visually

different from house numbers from SVHN (Netzer et al., 2011) dataset, DNN-based classifiers may still assign a high likelihood to SVHN samples.

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 and as a distance in the context of multivariate Gaussian pdfs (Pinele et al., 2020; Strapasson et al., 2016). 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 (Bronstein et al., 2021). Our method adapts to the various scenarios depending on the level of information access of the DNN and uses only in-distribution samples but can also benefit (if available) of OOD samples.

1.1 CONTRIBUTIONS

Our work studies the problem of OOD detection and advances state-of-the-art in very different ways.

- We introduce IGEOOD, a fully flexible and effective method to perform OOD detection. To the best of our knowledge, this is the first work studying *information geometry* tools to devise a unified metric for OOD detection. We derive an explicit characterization of the Fisher-Rao distance based on the information-geometric properties of the softmax probabilities of the neural classifier and the class of multivariate Gaussian pdfs. In general terms, our Fisher-Rao distance measures the mismatch–in the geometry space–between probability density functions of the pre-trained DNN classifier conditioning on test and in-distribution samples. Section 3 shows in detail how IGEOOD is used for OOD detection.
- Experiments on BLACK-BOX and GREY-BOX setups using various datasets, architectures, and classification tasks clearly show that IGEOOD is competitive with state-of-the-art methods, i.e., by reducing the mean false-positive rate for several datasets and DNN architectures. In the BLACK-BOX setup, we assume that only the outputs, i.e., the logits of the DNN, are available. In the GREY-BOX setup, we allow access to all parameters of the network; however, the detection must be performed using only the output softmax probabilities. The latter permits input pre-processing which introduces a small (additive) noise according to the direction of the gradients w.r.t the test sample. This pre-processing allows for further discrimination between in- and out-of-distribution samples. Our benchmark contains 2 DNN architectures, 3 in-distribution datasets, and 9 OOD datasets.
- Experiments on WHITE-BOX setup based on various datasets, architectures, and classification tasks clearly show consistent improvement of IGEOOD, achieving new state-of-the-art performance on the benchmark. We increased the average TNR at 95% TPR by 6.9% on ResNet and up to 10.2% on DenseNet. In this setup, we combine the logits with the low-level features of the DNN to leverage further useful statistical information of the encoded in-distribution data. We model the pre-trained latent representations as a Gaussian mixture model with a diagonal covariance matrix (DGMM). Under this assumption, we derive a confidence score based on the Fisher-Rao distance between conditional distributions corresponding to the test and the closest in-distribution samples.

1.2 RELATED WORKS

OOD discriminators consist of a binary classifier to distinguish between in- and out-of-distribution samples. A few works (Shalev et al., 2018; Hendrycks et al., 2019; Bitterwolf et al., 2020; Mohseni et al., 2020; Winkens et al., 2020; Vyas et al., 2018; Hein et al., 2019) 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 latent representations to learn the decision boundaries 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. Other works use OOD data to improve the robustness of DNNs. Nagarajan et al. (2021) studies failure modes of OOD detection methods

to better understand how to improve them, especially how spurious features like the background can vastly degrade detection performance. Lee et al. (2021) leverage OOD data as a regularization technique to improve the generalization and robustness of current neural networks. References (Schlegl et al., 2017; Kirichenko et al., 2020; Choi & Jang, 2019; Vernekar et al., 2019; Xiao et al., 2020; Ren et al., 2019; Zhang et al., 2021; Mahmood et al., 2021) study OOD detection in the context of generative models. Open set recognition (Bendale & Boult, 2016), outlier or anomaly detection (Pimentel et al., 2014), concept drift detection (Quionero-Candela et al., 2009), and adversarial attacks detection (Goodfellow et al., 2015; Madry et al., 2018) are related topics.

BLACK-BOX and GREY-BOX scenarios. It is often the case on ML as a service (Ribeiro et al., 2015) that the model's parameters knowledge and access are restricted to the end-user, granting access only to the computation of the forward and the logits or softmax outputs. The baseline work (Hendrycks & Gimpel, 2017) for BLACK-BOX techniques considers simply the unscaled maximum value of the softmax (MSP) as OOD score. In some cases, this confidence score is enough to distinguish between in-distribution and out-of-distribution examples, but it also may assign overconfident values to OOD examples (Hein et al., 2019). ODIN's (Liang et al., 2018) method has two variations. The BLACK-BOX variation consists of temperature scaling the softmax outputs. While the GREY-BOX variation also uses an input pre-processing technique that calculates the gradient of the model parameters and adds to the input in an adversarial manner for a more effective OOD detection. Liu et al. (2020) proposes an energy-based OOD score. Without retraining, they substitute the softmax confidence score by the free energy function with a temperature parameter. They also propose a GREY-BOX variation with posterior processing for improved results. Fine-tuning is done differently across the literature and should be considered when comparing methodologies.

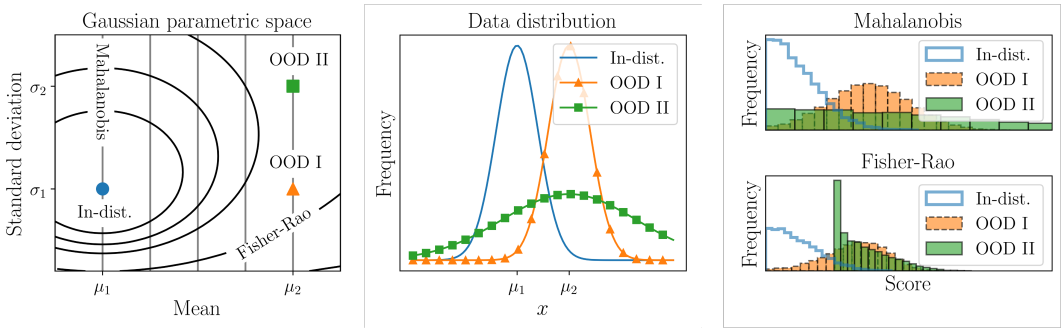
WHITE-BOX scenario. This class of OOD detectors is more flexible since discriminators have access to all intermediate layer outputs. Naturally, they have access to more information than the BLACK-BOX or GREY-BOX techniques, warranting greater detection capacity. Batch-normalization statistics between layers are used (Quintanilha et al., 2019) to fit a logistic regression that serves as OOD score for a particular OOD dataset. Lee et al. (2018) assume that latent features of DNN models trained under the softmax score follow a class-conditional Gaussian mixture distribution with tied covariance matrix and different 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 a tied covariance matrix estimated on the training set. The importance of each low-level component and hyperparameters are tuned using validation data, achieving state-of-the-art performance in training-free OOD detection. Ren et al. (2021) modifies the Mahalanobis distance-based OOD detection method (Lee et al., 2018) to improve detection of near-OOD data. They model the background distribution by a normal distribution with mean and standard calculated empirically from the entire training set and subtract it from the distance criteria, reducing the importance of features shared by in- and out-of-distribution data.

2 BACKGROUND

Let $\mathcal{X} \subseteq \mathbb{R}^d$ be the feature space (continuous) and $\mathcal{Y}$ a label space. Moreover, let p_{XY} be the underlying unknown probability density function (pdf) over $\mathcal{X} \times \mathcal{Y}$. We define the *in-distribution training dataset* as $\mathcal{D}_N \triangleq \left\{(\boldsymbol{x}_i, y_i)\right\}_{i=1}^N \sim p_{XY}$, where $\boldsymbol{x}_i \in \mathcal{X}$ is the input feature data, $y_i \in \mathcal{Y} \triangleq \{1, \ldots, C\}$ is the output class among C possible classes and N denotes the number of training samples. The training dataset is characterized by the joint pdf p_{XY} with *in-distribution marginals* $X \sim p_X$ and $Y \sim P_Y$. The predictor denoted by $f_{\mathcal{D}_N} : \mathcal{X} \to \mathcal{Y}$ is based on the inferred model $P_{\widehat{Y}|X}$, i.e., $f_{\mathcal{D}_N}(\boldsymbol{x}) \equiv f_n(\boldsymbol{x}; \mathcal{D}_N) \triangleq \arg\max_{y \in \mathcal{Y}} P_{\widehat{Y}|X}(y|\boldsymbol{x}; \mathcal{D}_N)$. In order to between distributions from samples, we introduce an artificial binary random variable $Z \in \{0, 1\}$, where $z = 1$ indicates that the test sample $\boldsymbol{x}$ is OOD. The open-world data can then be modeled as a *mixture* distribution $p_{X|Z}$ defined by

$$p_{X|Z}(\boldsymbol{x}|z = 0) \triangleq p_X(\boldsymbol{x}), \quad p_{X|Z}(\boldsymbol{x}|z = 1) \triangleq q_X(\boldsymbol{x}). \tag{1}$$

The intrinsic difficulty arises from the fact that very little can be assumed about the unknown distributions p_X and q_X, in particular for out-of-distribution.



(a) Contour lines. (b) Synthetic data distribution. (c) OOD detection score histogram.

Figure 1: Example comparing the Fisher-Rao with the Mahalanobis distance on distinguishing between 1D Gaussian distributions showcasing the motivation of using Fisher-Rao for OOD detection.

3 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 3.2), which mitigates the high-confidence scores assigned to OOD examples, and layer-wise level (Section 3.3). The key ingredient of IGEOOD is the Fisher-Rao distance that allows for effective differentiation between in-distribution and out-of-distribution samples. 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.

This measure connects information geometry and differential geometry through the R. Fisher information matrix (Fisher, 1922). Closed-form expressions of this distance are known to multivariate normal distributions under certain circumstances, among others distributions (Pinele et al., 2020). We apply the Fisher-Rao distance between the in-distribution and out-of-distribution estimated pdfs induced by the classifier network as a measure for detecting OOD examples at inference-time.

3.1 MOTIVATION FOR USING THE FISHER-RAO DISTANCE FOR OOD DETECTION

We introduce a simple example to demonstrate conceptually how Fisher-Rao distance is instrumental to OOD detection. It should be noted that this example is limited to one dimension distributions. However, we expect a similar behavior to be observed with true data that fits in a Gaussian model.

Consider the case where we are trying to distinguish between samples from distinct Gaussian distributions on 1D. We suppose that the in-distribution data follows a Gaussian $\mathcal{N}(\mu_1, \sigma_1)$ while OOD data is drawn according to either $\mathcal{N}(\mu_2, \sigma_1)$ or $\mathcal{N}(\mu_2, \sigma_2)$. These distributions are illustrated in Figures 1a and 1b. In this setup, distance-based approaches which are invariant to the variance of the distributions would have the performance limited to the information given by the difference between the means of the underlying distributions. For instance, in the case of the Mahalanobis distance, we would rely our discrimination on the difference between the sample and the in-distribution mean, rescaled by the in-distribution standard deviation only, but nothing further could be obtained. However, if we can estimate OOD standard deviations from actual or pseudo OOD data, we expect the Fisher-Rao distance between Gaussian distributions to be more effective in distinguishing between distributions. Figure 1c shows that the Fisher-Rao distance can distinguish better between "In-dist." and "OOD II" samples, while the other distances will fail.

3.2 IGEOOD SCORE USING THE SOFTMAX PROBABILITY

The Fisher-Rao distance (Atkinson & Mitchell, 1981) takes as input two probability distributions. For the classification problem, we can take the temperature T scaled softmax function (Eq. 2) as an approximation of a class-conditional probability distribution:

$$q_{\boldsymbol{\theta}}\left(y|f(\boldsymbol{x});T\right) \triangleq \frac{\exp\left(f_y(\boldsymbol{x})/T\right)}{\sum_{y'\in\mathcal{Y}}\exp\left(f_{y'}(\boldsymbol{x})/T\right)}, \tag{2}$$

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 Fisher-Rao distance $d_{\mathrm{FR-Logits}}$ between two softmax probability distributions can be shown by[1]

$$d_{\mathrm{FR-Logits}}\big(q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x})), q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x}'))\big) \triangleq 2 \arccos \left(\sum_{y \in \mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}\big(y|f(\boldsymbol{x})\big) q_{\boldsymbol{\theta}}\big(y|f(\boldsymbol{x}')\big)} \right). \tag{3}$$

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 *empirical centroid* for the logits of each class $y \in \mathcal{Y} = \{1, \ldots, C\}$ of the in-distribution training dataset $\mathcal{D}_N$ corresponding to the Fisher-Rao distance, i.e.,

$$\boldsymbol{\mu}_y \triangleq \min_{\boldsymbol{\mu} \in \mathbb{R}^C} \frac{1}{N_y} \sum_{\forall\, i \,:\, y_i = y} d_{\mathrm{FR-Logits}}\big(q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x}_i)), q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{\mu})\big), \tag{4}$$

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. This is equivalent to finding the centroid of a cluster using the Fisher-Rao distance, after each example has been assigned to a cluster. Please refer to the appendix (see Section B) for further details on this optimization.

OOD and confidence score. We propose the Fisher-Rao 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 $\boldsymbol{\mu}_y$ given by Eq. 4. By taking the sum instead of the minimal distance, we leverage useful information related to the example's confidence score for each class y. We denote it as follows:

$$\mathrm{FR}_0(\boldsymbol{x}) \triangleq \sum_{y \in \mathcal{Y}} d_{\mathrm{FR-Logits}}\big(q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x})), q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{\mu}_y)\big). \tag{5}$$

Using the softmax probability, we can define a confidence score to be the minimum of the Fisher-Rao distance between $f(\boldsymbol{x})$ and the class-conditional centroids. Thus, the estimated class $\widehat{y}_{\mathrm{FR}}$ follows:

$$\widehat{y}_{\mathrm{FR}}(\boldsymbol{x}) \triangleq \arg\min_{y \in \mathcal{Y}} d_{\mathrm{FR-Logits}}\big(q_{\boldsymbol{\theta}}(\cdot|f(\boldsymbol{x})), q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{\mu}_y)\big). \tag{6}$$

Empirically, we show in the appendix (see Section C) that this confidence score does not degrade and sometimes improves the in-distribution test classification accuracy. However, we obtained slightly better OOD detection results by using Eq. 5 instead of Eq. 6.

Input pre-processing. In consonance with the literature (Liang et al., 2018; Liu et al., 2020; Lee et al., 2018), we also perform input pre-processing to enhance the detection between in-distribution and OOD samples and potentially improve OOD detection performance for the GREY-BOX discriminator. We add small magnitude perturbations ε in a Fast Gradient-Sign Method-style (FGSM) (Goodfellow et al., 2015) to each test sample $\boldsymbol{x}$ to increase the proposed metric, that is:

$$\widetilde{\boldsymbol{x}} = \boldsymbol{x} + \varepsilon \odot \mathrm{sign}\left[\nabla_{\boldsymbol{x}} \mathrm{FR}_0(\boldsymbol{x})\right]. \tag{7}$$

The OOD detector. The detector consists of a threshold-based function for discriminating between in-distribution and OOD data. This threshold δ and parameters are set so that the true positive rate, i.e., the in-distribution samples correctly classified as in-distribution, becomes 95%. Mathematically, the BLACK-BOX OOD detector g_{BB} and the GREY-BOX OOD detector g_{GB} writes:

$$g_{\mathrm{BB}}(\boldsymbol{x}; \delta, T) = \begin{cases} 1 & \text{if } \mathrm{FR}_0\left(\boldsymbol{x}\right) \leq \delta \\ 0 & \text{if } \mathrm{FR}_0\left(\boldsymbol{x}\right) > \delta \end{cases} \quad \text{and} \quad g_{\mathrm{GB}}(\widetilde{\boldsymbol{x}}; \delta, T, \varepsilon) = \begin{cases} 1 & \text{if } \mathrm{FR}_0\left(\widetilde{\boldsymbol{x}}\right) \leq \delta \\ 0 & \text{if } \mathrm{FR}_0\left(\widetilde{\boldsymbol{x}}\right) > \delta \end{cases}. \tag{8}$$

3.3 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

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

the latent feature. We compute the empirical estimates of these parameters according to

$$\boldsymbol{\mu}_y^{(\ell)} = \frac{1}{N_y} \sum_{\forall i \,:\, y_i = y} f^{(\ell)}\left(\boldsymbol{x}_i\right), \quad \text{and} \quad \boldsymbol{\sigma}^{(\ell)} = \operatorname{diag}\left(\sqrt{\frac{1}{N} \sum_{y \in \mathcal{Y}} \sum_{\forall i \,:\, y_i = y} \left(f_j^{(\ell)}\left(\boldsymbol{x}_i\right) - \mu_{y,j}^{(\ell)}\right)^2}\right), \quad (9)$$

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 Fisher-Rao distance ρ_{FR} between two arbitrary *univariate* Gaussian pdfs $\mathcal{N}(\mu_1, \sigma_1^2)$ and $\mathcal{N}(\mu_2, \sigma_2^2)$ is given by[2]

$$\rho_{\mathrm{FR}}\left(\left(\mu_1, \sigma_1\right), \left(\mu_2, \sigma_2\right)\right) = \sqrt{2} \log \frac{\left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, -\sigma_2\right)\right| + \left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, \sigma_2\right)\right|}{\left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, -\sigma_2\right)\right| - \left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, \sigma_2\right)\right|}. \quad (10)$$

Similarly, the Fisher-Rao 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

$$d_{\mathrm{FR-Gauss}}\big((\boldsymbol{\mu}, \boldsymbol{\sigma}), (\boldsymbol{\mu}', \boldsymbol{\sigma}')\big) = \sqrt{\sum_{i=1}^{k} \rho_{\mathrm{FR}}\left(\left(\mu_i, \sigma_{i,i}\right), \left(\mu_i', \sigma_{i,i}'\right)\right)^2}, \quad (11)$$

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

Experimental support for a diagonal Gaussian mixture model. It is known that abnormal samples can be characterized better in the intermediate features of a DNN, in place of the output space of softmax posterior distributions as demonstrated by Lee et al. (2018). Nonetheless, 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. Thus, a diagonal covariance matrix will be a favorable compromise for OOD detection.

Fisher-Rao distance-based feature-wise confidence score. We derive a confidence score by applying the Fisher-Rao 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.,

$$\mathrm{FR}_\ell(\boldsymbol{x}) = \min_{y \in \mathcal{Y}} d_{\mathrm{FR-Gauss}}\big((\boldsymbol{x}, \boldsymbol{\sigma}^{(\ell)}), (\boldsymbol{\mu}_y^{(\ell)}, \boldsymbol{\sigma}^{(\ell)})\big); \quad (12)$$

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

$$\mathrm{FR}_\ell'(\boldsymbol{x}) = \min_{y \in \mathcal{Y}} d_{\mathrm{FR-Gauss}}\big((\boldsymbol{x}, \boldsymbol{\sigma}'^{(\ell)}), (\boldsymbol{\mu}_y^{(\ell)}, \boldsymbol{\sigma}^{(\ell)})\big). \quad (13)$$

This validation dataset could be obtained from a synthetic dataset, a dataset different from the testing one, or even by adversarially creating OOD data by attacking the classifier model on the training dataset. In the appendix (Section B), we include pseudo-codes for calculating the IGEOOD score for the BLACK-BOX, GREY-BOX, and WHITE-BOX settings.

Feature ensemble. To further improve performance, we combine the confidence scores of the logits and the ones from the low-level features through a linear combination. Similarly to the strategy in Lee et al. (2018), we choose the weights α_0, α_ℓ and $\alpha_\ell' \in \mathbb{R}$ by training a logistic regression detector using validation samples. Thus, we ensure that the metric emphasizes features that demonstrate a greater capacity for detecting abnormal samples. The following equation summarizes the IGEOOD score for the WHITE-BOX setting:

$$\mathrm{FR}(\boldsymbol{x}) \triangleq \alpha_0 \mathrm{FR}_0(\boldsymbol{x}) + \sum_{\ell} \alpha_\ell \cdot \mathrm{FR}_\ell(\boldsymbol{x}) + \alpha_\ell' \cdot \mathrm{FR}_\ell'(\boldsymbol{x}), \quad (14)$$

where FR_0 is given by equation 5, FR_ℓ is given by equation 12 and FR' considers a different validation diagonal covariance matrix for the test samples (equation 13). We also apply input preprocessing similarly to the GREY-BOX setting (equation 7), obtaining $\mathrm{FR}(\widetilde{\boldsymbol{x}})$ as final score.

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

Unified metric. For the three settings, the metric is the same but has different formulations given the family of the distributions. For the DNN outputs, it is under the softmax posterior probability distribution setup. For the intermediate layers, it is under the model of diagonal Gaussian pdfs. *Therefore, we have derived a unified OOD detection framework that combines a single distance for both the softmax outputs and the latent features of a neural network.* Figure 2 illustrates how each of the presented techniques contributes towards separating in-distribution and OOD samples.

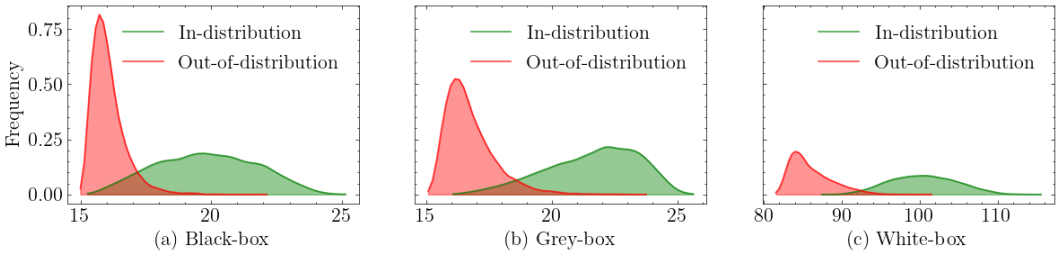


Figure 2: Probability distributions of the IGEOOD score under three different settings for a pre-trained DenseNet on CIFAR-10 for in-distribution and OOD data (Tiny-ImageNet downsampled).

4 EXPERIMENTAL RESULTS

This section shows the effectiveness of IGEOOD and compares the results to the SOTA methods. We provide detailed experimental setup[3] and additional results in the appendices (see C, D, and E).

4.1 SETUP

The experimental setup follows the setting established by Hendrycks & Gimpel (2017), Liang et al. (2018) and Lee et al. (2018). We use two *pre-trained* deep neural networks architectures for image classification tasks: a Dense Convolutional Network (DenseNet-BC-100) (Huang et al., 2017) and a Residual Neural Network (ResNet-34) (He et al., 2016). We take as *in-distribution data* images from CIFAR-10 (Krizhevsky et al., 2009), CIFAR-100 and SVHN (Netzer et al., 2011) datasets.

For *out-of-distribution data*, we use natural image examples from the datasets: Tiny-ImageNet (Le & Yang, 2015), LSUN (Yu et al., 2015), Describable Textures Dataset (Cimpoi et al., 2014), Chars74K (de Campos et al., 2009), Places365 (Zhou et al., 2017), iSUN (Xu et al., 2015) 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. We resize the images to dimension 32×32 by downsampling from their original resolution. We apply center crop when needed, and we only use test data for evaluation. Even though we ran experiments with image data, the method proposed could be applied to any neural-based classification task.

We measure the effectiveness of the OOD detectors with three standard *evaluation metrics* for a binary classification problem: the true negative rate at 95% true positive rate (TNR at TPR-95%), the area under the receiving operating curve (AUROC), and the area under the precision-recall curve (AUPR). For calculating them, we use the test set of in-distribution and OOD datasets. For the BLACK-BOX and GREY-BOX experimental settings, we *tune hyperparameters* for all of the OOD detectors only based on the DNN classifier architecture, the in-distribution dataset, and a validation dataset. The iSUN (Xu et al., 2015) dataset is chosen as a source of OOD validation data, independently from OOD test data. We choose the parameters that maximize the TNR at TPR-95% on the validation OOD dataset. For the WHITE-BOX framework, we allow the benchmark and our method to tune either on adversarially generated data from the in-distribution training dataset or a separate validation dataset containing 1,000 images from the OOD test dataset with feature ensemble described in Section 3.3. In this case, we evaluate performance on the remaining test samples.

[3]Our code is publicly available at `https://github.com/igeood/Igeood`.

4.2 RESULTS FOR THE BLACK-BOX AND THE GREY-BOX SETUPS

For comparing IGEOOD under the hypothesis of a BLACK-BOX scenario, we consider the Baseline (Hendrycks & Gimpel, 2017) method, ODIN (Liang et al., 2018) with temperature scaling only, and the free-energy-based metric (Liu et al., 2020) with temperature scaling only–without input pre-processing or re-training. The results for the BLACK-BOX setting are available in Table 1, where we show the average OOD detection performance for each of the eight OOD detection method in six different image classification contexts (couple DNN model and in-distribution dataset). The extended results for each OOD dataset can be found in Table 8. For comparison under the GREY-BOX assumption, we consider ODIN and the free-energy-based methods, both with input pre-processing. The results for the GREY-BOX setup is in the appendix (see Section E) for brevity.

Table 1: Average OOD detection performance across eight OOD datasets for each model and in-distribution dataset in a BLACK-BOX setting. IGEOOD is compared to Baseline (Hendrycks & Gimpel, 2017), ODIN (Liang et al., 2018), and Energy (Liu et al., 2020) methods. The extended results can be found in Table 8 in the appendix.

Model	In-dist.	TNR at TPR-95%	AUROC	AUPR
		Baseline / ODIN / Energy / IGEOOD (ours)		
DenseNet	CIFAR-10	52.5/**66.8**/65.3/65.6	91.7/**92.8**/92.1/92.3	90.7/**90.9**/90.3/90.6
	CIFAR-100	15.9/20.5/20.3/**20.7**	69.1/71.6/71.6/**73.2**	71.3/72.7/72.7/**74.5**
	SVHN	68.3/68.8/70.2/**72.1**	**92.3**/87.3/90.2/90.9	**95.4**/91.6/93.5/94.1
ResNet	CIFAR-10	41.7/51.9/56.3/**56.7**	89.6/90.4/90.4/**90.5**	**88.5**/88.3/87.9/88.1
	CIFAR-100	15.0/16.0/16.3/**16.4**	74.0/75.2/75.4/**75.5**	72.1/72.3/72.4/**72.6**
	SVHN	76.2/77.7/78.0/**78.4**	92.2/91.3/91.4/**91.7**	**93.4**/92.0/92.0/92.4
Average		44.9/50.3/51.1/**51.6**	84.8/84.8/85.2/**85.7**	85.2/84.6/84.8/**85.4**

Temperature scaling and input pre-processing. We observed that low values of temperature and moderate noise magnitude yield better detection performance for IGEOOD on the logits. For most models and datasets, we obtained better results for temperatures between 1 and 6 and noise magnitudes below 0.002. Detailed results and the best hyperparameters found for each configuration, as well as figures of their impact on performance, are delegated to the appendix (see Section E).

How the choice of validation dataset impacts performance. To demonstrate robustness to the choice of the validation dataset, we include in the appendix (see Section E) the average OOD detection performance for each method when we change the validation set among the nine available ones. We show that the average TNR at TPR-95% for IGEOOD ranges between 63% and 72% on a BLACK-BOX scenario and between 65% and 74% on a GREY-BOX scenario for a DenseNet pre-trained on CIFAR-10.

4.3 RESULTS FOR THE WHITE-BOX SETTING

For benchmarking IGEOOD on the WHITE-BOX setting, we compare results to the Mahalanobis (Lee et al., 2018) method with input pre-processing and feature ensemble. For both of them, we extract features from every output of the dense (or residual) block of the DenseNet (or ResNet) model and the first convolutional layer. The size of each feature is reduced by average pooling in the spatial dimensions. Thus, the initial dimension $\mathcal{F}_\ell \times \mathcal{W}_\ell \times \mathcal{H}_\ell$ is reduced to $\mathcal{F}_\ell$, where $\mathcal{F}_\ell$ is the number of channels in block ℓ. For the DenseNet, this reduction translates to features of sizes $\boldsymbol{\mathcal{F}}_1 = \{24, 108, 150, 342\}$; and for the ResNet, to features of sizes $\boldsymbol{\mathcal{F}}_2 = \{64, 64, 128, 256, 512\}$.

We consider two validation scenarios. One with OOD validation data from adversarially generated (FGSM) samples from the in-distribution training dataset. And another with validation data from the OOD test set. We tune hyperparameters for each method with generated data (pseudo OOD) and in-distribution data for the adversarial validation. While for the validation with OOD data, we tune hyperparameters with 1,000 OOD data samples and in-distribution data. We derive two methods: IGEOOD+, which is given by equation 14 and considers that we have a prior of the diagonal covariance matrix from true or pseudo OOD data as additional information; and IGEOOD, which

considers that test samples has the same covariance matrix as in-distribution data only, i.e., set $\text{FR}' = 0$ on equation 14.

Comparison with Mahalanobis. For each DNN model and in-distribution dataset pair, we report the average OOD detection performance for Mahalanobis (Lee et al., 2018), IGEOOD and IGEOOD+. Table 2 validates the contributions of our techniques. We observe substantial performance improvement in all experiments for the left-hand side of the table, where we outperform Mahalanobis on average for all test cases. IGEOOD and IGEOOD+ show improvements up to 11.4% and 17.3% on TNR at TPR-95%, respectively. To assess the robustness of IGEOOD to the choice of validation data, we measured the detection performance when all hyperparameters are tuned only using in-distribution and generated adversarial data as observed in the right-hand side of Table 2. IGEOOD record improvements up to 10.3%, and improves by 3% the average TNR at TPR-95% across all datasets and models compared to Mahalanobis.

Table 2: Average 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 10 and 11 in the appendix.

		Validation on OOD data			Validation on adversarial data		
		TNR-95%	AUROC	AUPR	TNR-95%	AUROC	AUPR
Model	In-dist.	Mahalanobis / IGEOOD (ours) / IGEOOD+ (ours)			Mahalanobis / IGEOOD (ours) / IGEOOD+ (ours)		
DenseNet	C-10	77.2/88.6/**90.1**	92.0/97.5/**97.9**	88.5/95.7/**96.6**	75.8/**80.9**/79.8	91.7/**95.1**/94.7	88.0/**95.2**/94.7
	C-100	65.7/76.5/**83.0**	87.8/93.7/**95.9**	88.0/93.7/**96.1**	60.2/**70.5/70.7**	85.2/**90.8/90.8**	85.7/88.1/**91.3**
	SVHN	95.1/93.9/**95.6**	98.8/98.7/**99.1**	99.5/99.5/**99.7**	**93.7**/93.2/90.2	**98.6**/98.5/97.9	99.3/**99.5**/99.0
ResNet	C-10	84.1/86.1/**89.4**	96.5/96.6/**97.8**	**96.7**/96.6/96.1	**78.6**/77.3/76.6	**95.3**/90.0/89.6	**95.1**/85.2/88.2
	C-100	70.8/74.0/**84.2**	92.5/93.2/**95.9**	92.7/93.3/**95.5**	57.4/64.6/**67.0**	86.9/88.5/**89.5**	86.2/88.6/**89.6**
	SVHN	97.6/96.4/**99.7**	99.4/99.2/**99.9**	99.8/99.7/**99.9**	**96.3**/93.6/91.1	**99.1**/98.4/97.2	**99.6**/99.3/98.8
Average		81.8/85.9/**90.3**	94.5/96.5/**97.8**	94.2/96.4/**97.3**	79.0/**82.0**/80.7	93.1/**94.0**/93.7	92.5/92.8/**93.8**

Ablation study. The IGEOOD score has three components, FR_0, FR_ℓ, and FR'_ℓ, that together compose the final metric given by equation 14. The outputs of the network provide limited OOD detection capacity as observed in Table 1. Always when available, the intermediate features, i.e., FR_ℓ, are a valuable resource for OOD detection. Moreover, when few reliable OOD data is available, calculating FR'_ℓ can further improve the detection performance as shown on the left-hand column of Table 2. Also, data from a source other than the in-distribution data, e.g., adversarial samples, is enough for tuning hyperparameters and combining features, as observed on the results of the hand-side column of Table 2.

5 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. First, in a (GREY-) BLACK-BOX setup, we calculate the sum of the Fisher-Rao distance between the softmax output, corresponding to the test (pre-processed) sample, and a reference probability, corresponding to the conditional-class of softmax probabilities. Similarly, in a WHITE-BOX setup, we model the low-level features of a DNN as a diagonal Gaussian mixture. 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, which reflect diverse application scenarios. Experimentally, we showed that IGEOOD significantly and consistently improves the accuracy of OOD detection on several DNN architectures across various in-distribution and OOD datasets, achieving new state-of-the-art performances for visual classifications tasks. Some perspectives for future work include studying causal factors, explainable components for OOD detection, and extensions to textual data.

REFERENCES

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul F. Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *CoRR*, abs/1606.06565, 2016. URL http://arxiv.org/abs/1606.06565.

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. ISSN 0581572X. URL http://www.jstor.org/stable/25050283.

Abhijit Bendale and Terrance E. Boult. Towards open set deep networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1563–1572, 2016. doi: 10.1109/CVPR.2016.173.

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 (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 16085–16095. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/b90c46963248e6d7aab1e0f429743ca0-Paper.pdf.

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

Hyunsun Choi and Eric Jang. Generative ensembles for robust anomaly detection, 2019. URL https://openreview.net/forum?id=B1e8CsRctX.

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.

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.

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.

R. A. Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London, A*, 222:309–368, 1922.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015. URL http://arxiv.org/abs/1412.6572.

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)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

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.

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.

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HyxCxhRcY7.

Jens Henriksson, Christian Berger, Markus Borg, Lars Tornberg, Sankar Raman Sathyamoorthy, and Cristofer Englund. Performance analysis of out-of-distribution detection on trained neural networks. *Information and Software Technology*, 130:106409, 2021. ISSN 0950-5849. doi: https://doi.org/10.1016/j.infsof.2020.106409. URL https://www.sciencedirect.com/science/article/pii/S0950584919302204.

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)*, pp. 2261–2269, 2017. doi: 10.1109/CVPR.2017.243.

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 (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 20578–20589. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/ecb9fe2fbb99c31f567e9823e884dbec-Paper.pdf.

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

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/9ef2ed4b7fd2c810847ffa5fa85bce38-Paper.pdf.

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

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 (eds.), *Advances in Neural Information Processing Systems 31*, pp. 7167–7177. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/7947-a-simple-unified-framework-for-detecting-out-of-distribution-samples-and-adve pdf.

Saehyung Lee, Changhwa Park, Hyungyu Lee, Jihun Yi, Jonghyun Lee, and Sungroh Yoon. Removing undesirable feature contributions using out-of-distribution data. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=eIHYL6fpbkA.

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. URL https://openreview.net/forum?id=H1VGkIxRZ.

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

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. URL https://openreview.net/forum?id=rJzIBfZAb.

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

Ahsan Mahmood, Junier Oliva, and Martin Andreas Styner. Multiscale score matching for out-of-distribution detection. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=xoHdgbQJohv.

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. doi: 10.1609/aaai.v34i04.5966. URL https://ojs.aaai.org/index.php/AAAI/article/view/5966.

Vaishnavh Nagarajan, Anders Andreassen, and Behnam Neyshabur. Understanding the failure modes of out-of-distribution generalization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=fSTD6NFIW_b.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=H1xwNhCcYm.

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. URL http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/8558cb408c1d76621371888657d2eb1d-Paper.pdf.

Marco Pimentel, David Clifton, Lei Clifton, and L. Tarassenko. A review of novelty detection. *Signal Processing*, 99:215–249, 06 2014. doi: 10.1016/j.sigpro.2013.12.026.

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. ISSN 1099-4300. doi: 10.3390/e22040404. URL https://www.mdpi.com/1099-4300/22/4/404.

Igor M. Quintanilha, Roberto de M. E. Filho, José Lezama, Mauricio Delbracio, and Leonardo O. Nunes. Detecting out-of-distribution samples using low-order deep features statistics, 2019. URL https://openreview.net/forum?id=rkgpCoRctm.

Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. *Dataset Shift in Machine Learning*. The MIT Press, 2009. ISBN 0262170051.

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 (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/1e79596878b2320cac26dd792a6c51c9-Paper.pdf.

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.

Mauro Ribeiro, Katarina Grolinger, and Miriam A.M. Capretz. Mlaas: Machine learning as a service. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pp. 896–902, 2015. doi: 10.1109/ICMLA.2015.152.

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

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 (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/2151b4c76b4dcb048d06a5c32942b6f6-Paper.pdf.

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)*, pp. 1–5, 2016. doi: 10.1109/SAM.2016.7569717.

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. URL https://drive.google.com/file/d/0B3mY6u_lryzdel9WOW1XTVA0aDIwazJDcG9ORlZrZWFOd0xJ/view.

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)*, pp. 560–574, 2018. URL https://doi.org/10.1007/978-3-030-01237-3_34.

Jim Winkens, Rudy Bunel, Abhijit Guha Roy, Robert Stanforth, Vivek Natarajan, Joseph R. Ledsam, Patricia MacWilliams, Pushmeet Kohli, Alan Karthikesalingam, Simon Kohl, Taylan Cemgil, S. M. Ali Eslami, and Olaf Ronneberger. Contrastive training for improved out-of-distribution detection, 2020.

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*, pp. 3485–3492, 2010. doi: 10.1109/CVPR.2010.5539970.

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 (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 20685–20696. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/eddea82ad2755b24c4e168c5fc2ebd40-Paper.pdf.

Pingmei Xu, Krista A Ehinger, Yinda Zhang, Adam Finkelstein, Sanjeev R. Kulkarni, and Jianxiong Xiao. Turkergaze: Crowdsourcing saliency with webcam based eye tracking, 2015.

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.

Yufeng Zhang, Wanwei Liu, Zhenbang Chen, Ji Wang, Zhiming Liu, Kenli Li, and Hongmei Wei. Out-of-distribution detection with distance guarantee in deep generative models, 2021.

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.

A REVIEW OF FISHER-RAO DISTANCE (FRD)

In this section, we review some results from references Atkinson & Mitchell (1981); Pinele et al. (2020). 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 $\mathcal{C} \triangleq \left\{q_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^C\right\}$.

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 (Atkinson & Mitchell, 1981):

(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

$$ds^2 = \langle d\boldsymbol{x}, d\boldsymbol{x} \rangle_{\boldsymbol{G}(\boldsymbol{x})} = d\boldsymbol{x}^{\top} \boldsymbol{G}(\boldsymbol{x}) d\boldsymbol{x}. \tag{15}$$

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:

$$d_{R,\mathcal{C}}(q_{\boldsymbol{\theta}}, q'_{\boldsymbol{\theta}}) \triangleq \inf_{\boldsymbol{\gamma}} \int_0^1 \sqrt{\frac{d\boldsymbol{\gamma}^{\top}(t)}{dt} \boldsymbol{G}(\boldsymbol{\gamma}(t)) \frac{d\boldsymbol{\gamma}(t)}{dt}}, \tag{16}$$

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 equation 16 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 equation 2 can be shown to be singular, i.e., $\text{rank}(\boldsymbol{G}(\boldsymbol{x})) \leq C - 1$, 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

$$\mathcal{P} = \left\{ q : \mathcal{Y} \to [0, 1]^C : \sum_{y \in \mathcal{Y}} q(y) = 1 \right\}. \tag{17}$$

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

$$q(y|\boldsymbol{z}) = \frac{z_y^2}{4}, \quad y \in \{1, \ldots, C\}. \tag{18}$$

From this expression, we consider the statistical manifold $\mathcal{D} = \left\{q(\cdot|\boldsymbol{z}) : \|\boldsymbol{z}\|^2 = 4, z_y \geq 0, \forall y \in \mathcal{Y}\right\}$. 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:

$$\begin{aligned}\boldsymbol{G}(\boldsymbol{z}) &= \mathbb{E}_{q(y|\boldsymbol{z})}\left[\nabla_{\boldsymbol{z}} \log q(y|\boldsymbol{z}) \nabla_{\boldsymbol{z}}^\top \log q(y|\boldsymbol{z})\right] \\ &= \sum_{y\in\mathcal{Y}} \frac{z_y^2}{4}\left(\frac{2}{z_y}\boldsymbol{e}_y\right)\left(\frac{2}{z_y}\boldsymbol{e}_y^\top\right) \\ &= \sum_{y\in\mathcal{Y}} \boldsymbol{e}_y \boldsymbol{e}_y^\top \\ &= \boldsymbol{I},\end{aligned} \tag{19}$$

where $\{\boldsymbol{e}_y\}$ are the canonical basis vectors in $\mathbb{R}^C$ and $\boldsymbol{I}$ is the identity matrix. From expression equation 19 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 $q' = q\left(\cdot|\boldsymbol{z}'\right)$ can be written as the radius of the sphere times the angle between the vectors $\boldsymbol{z}$ and $\boldsymbol{z}'$. Which leads to expression:

$$d_{R,\mathcal{D}}\left(q, q'\right) = 2\arccos\left(\frac{\boldsymbol{z}^\top \boldsymbol{z}'}{4}\right) = 2\arccos\left(\sum_{y\in\mathcal{Y}} \sqrt{q(y|\boldsymbol{z}) q\left(y|\boldsymbol{z}'\right)}\right). \tag{20}$$

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

$$d_{\mathrm{FR-Logits}}\left(q_{\boldsymbol{\theta}}, q'_{\boldsymbol{\theta}}\right) = 2\arccos\left(\sum_{y\in\mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}(y|\boldsymbol{x}) q_{\boldsymbol{\theta}}\left(y|\boldsymbol{x}'\right)}\right), \tag{21}$$

obtaining the same form of expression equation 3. 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 $\left(q_{\boldsymbol{\theta}}(1|\boldsymbol{x}), \ldots, q_{\boldsymbol{\theta}}(C|\boldsymbol{x})\right)$ and $\left(q_{\boldsymbol{\theta}}\left(1|\boldsymbol{x}'\right), \ldots, q_{\boldsymbol{\theta}}\left(C|\boldsymbol{x}'\right)\right)$ are orthogonal.

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

Consider a broader statistical manifold $\mathcal{S} \triangleq \{p_{\boldsymbol{\theta}} = p(\boldsymbol{x};\boldsymbol{\theta}) : \boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_m) \in \Theta\}$ 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:

$$\begin{aligned}g_{ij}(\boldsymbol{\theta}) &= \mathbb{E}_{\boldsymbol{\theta}}\left(\frac{\partial}{\partial\theta_i}\log p(\boldsymbol{x};\boldsymbol{\theta})\frac{\partial}{\partial\theta_j}\log p(\boldsymbol{x};\boldsymbol{\theta})\right) \\ &= \int \frac{\partial}{\partial\theta_i}\log p(\boldsymbol{x};\boldsymbol{\theta})\frac{\partial}{\partial\theta_j}\log p(\boldsymbol{x};\boldsymbol{\theta}) p(\boldsymbol{x};\boldsymbol{\theta}) dx.\end{aligned} \tag{22}$$

Next, consider a multivariate Gaussian distribution:

$$p(\boldsymbol{x};\boldsymbol{\mu},\Sigma) = \frac{(2\pi)^{-\left(\frac{n}{2}\right)}}{\sqrt{\mathrm{Det}(\Sigma)}}\exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{\mu})^\top \Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}{2}\right), \tag{23}$$

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 equation 23 in expression equation 22, we can derive the Fisher information matrix for this parametrization, obtaining:

$$g_{ij}(\boldsymbol{\theta}) = \frac{\partial\boldsymbol{\mu}^\top}{\partial\theta_i}\Sigma^{-1}\frac{\partial\boldsymbol{\mu}}{\partial\theta_j} + \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}\frac{\partial\Sigma}{\partial\theta_i}\Sigma^{-1}\frac{\partial\Sigma}{\partial\theta_i}\right), \tag{24}$$

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

$$ds^2 = d\boldsymbol{\mu}^\top \Sigma^{-1} d\boldsymbol{\mu} + \frac{1}{2}\operatorname{tr}\left[\left(\Sigma^{-1} d\Sigma\right)^2\right]. \tag{25}$$

Here, $d\boldsymbol{\mu} = (d\mu_1, \ldots, d\mu_n) \in \mathbb{R}^k$ and $d\Sigma = [d\sigma_{ij}] \in P_k(\mathbb{R})$. We observe that this metric is invariant to affine transformations (Pinele et al., 2020), 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:

$$d_{R,\mathcal{M}}\left(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2\right) = d_{R,\mathcal{M}}\left(\left(Q\boldsymbol{\mu}_1 + \boldsymbol{c}, Q\Sigma_1 Q^\top\right), \left(Q\boldsymbol{\mu}_2 + \boldsymbol{c}, Q\Sigma_2 Q^\top\right)\right). \tag{26}$$

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 equation 25 admits an additive form.

From Pinele et al. (2020), we obtain the analytical expression of the Fisher-Rao in the 2-dimensional submanifold of univariate Gaussian probability distributions $\mathcal{M}_2 = \{p_{\boldsymbol{\theta}} : \boldsymbol{\theta} = (\mu, \sigma^2) \in \mathbb{R} \times (0, +\infty)\}$:

$$\rho_{\mathrm{FR}}\left(\left(\mu_1, \sigma_1^2\right), \left(\mu_2, \sigma_2^2\right)\right) = \sqrt{2}\log\frac{\left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, -\sigma_2\right)\right| + \left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, \sigma_2\right)\right|}{\left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, -\sigma_2\right)\right| - \left|\left(\frac{\mu_1}{\sqrt{2}}, \sigma_1\right) - \left(\frac{\mu_2}{\sqrt{2}}, \sigma_2\right)\right|}, \tag{27}$$

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 $\Sigma = \mathrm{diag}\left(\sigma_1^2, \sigma_2^2, \ldots, \sigma_k^2\right)$ in the $2k$-dimensional statistical submanifold $\mathcal{M}_D = \left\{p_{\boldsymbol{\theta}} : \boldsymbol{\theta} = (\boldsymbol{\mu}, \Sigma), \Sigma = \mathrm{diag}\left(\sigma_1^2, \sigma_2^2, \ldots, \sigma_k^2\right), \sigma_i > 0, i = 1, \ldots, k\right\}$ is

$$d_{\mathrm{FR-Gauss}}\left(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2\right) = \sqrt{\sum_{i=1}^{k} d_{R,\mathcal{M}_2}\Big(\left(\mu_{1i}, \sigma_{1i}\right), \left(\mu_{2i}, \sigma_{2i}\right)\Big)^2}. \tag{28}$$

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 Pinele et al. (2020), 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 (Mahalanobis, 1936):

$$d_{R,\mathcal{M}_\Sigma}\big(\mathcal{N}(\boldsymbol{\mu}_1, \Sigma), \mathcal{N}(\boldsymbol{\mu}_2, \Sigma)\big) = \sqrt{(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \Sigma^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)}. \tag{29}$$

The Mahalanobis distance is also used for OOD detection (Lee et al., 2018) and its performance is compared to the FRD through several experiments in Section 4.

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 equation 4 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 equation 14.

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 equation 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.

Calculate the logits centroids from the training data:
$\boldsymbol{\mu}_y \triangleq \min_{\boldsymbol{\mu}\in\mathbb{R}^C} \frac{1}{N_y} \sum_{\forall\, i\,:\, y_i=y} 2 \arccos\left(\sum_{y'\in\mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}\big(y'|f(\boldsymbol{x}_i)\big) q_{\boldsymbol{\theta}}\big(y'|\boldsymbol{\mu}\big)}\right)$
Add small perturbation to $\boldsymbol{x}$:
$\widetilde{\boldsymbol{x}} \leftarrow \boldsymbol{x} + \varepsilon \odot \text{sign}\left[\nabla_{\boldsymbol{x}} \sum_y 2 \arccos\left(\sum_{y'\in\mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}(y'|f(\boldsymbol{x}))q_{\boldsymbol{\theta}}(y'|\boldsymbol{\mu}_y)}\right)\right]$
return $\text{FR}_0(\widetilde{\boldsymbol{x}}) \leftarrow \sum_y 2 \arccos\left(\sum_{y'\in\mathcal{Y}} \sqrt{q_{\boldsymbol{\theta}}(y'|f(\widetilde{\boldsymbol{x}}))q_{\boldsymbol{\theta}}(y'|\boldsymbol{\mu}_y)}\right)$

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**
  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:
  $\sigma_{jj}^{(\ell)} \leftarrow \sqrt{\frac{1}{N} \sum_{y\in\mathcal{Y}} \sum_{\forall i\,:\, y_i=y} \left(f_j^{(\ell)}(\boldsymbol{x}_i) - \mu_{y,j}^{(\ell)}\right)^2}$
  Compute the OOD score for ℓ:
  $\text{FR}_\ell(\boldsymbol{x}) \leftarrow \min_y \sqrt{\sum_{j=1}^{k} \rho_{\text{FR}}\left(\left(\mu_{y,j}^{(\ell)}, \sigma_{jj}^{(\ell)}\right), \left(f_j^{(\ell)}(\boldsymbol{x}), \sigma_{jj}^{(\ell)}\right)\right)^2}$
end
return $\big(\text{FR}_1(\boldsymbol{x}), \ldots, \text{FR}_L(\boldsymbol{x})\big)$

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 equation 4. 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 equation 4 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 equation 9. 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 (Huang et al., 2017), 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 (He et al., 2016), 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 [4]. 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%.

[4] http://github.com/igeood/Igeood

- **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.

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 (Krizhevsky et al., 2009) 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 (Krizhevsky et al., 2009) 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 (Netzer et al., 2011) 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 (Le & Yang, 2015) dataset is a subset of the large-scale natural image dataset ImageNet (Deng et al., 2009). 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 (Yu et al., 2015) 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 (Xu et al., 2015) dataset consists of selected natural scene images from the SUN (Xiao et al., 2010) 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) (Cimpoi et al., 2014) 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 (de Campos et al., 2009) 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 (Zhou et al., 2017) 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 (Hendrycks & Gimpel, 2017).

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 $\boldsymbol{x}_i^{\text{adv}}$ that maximizes the loss as given by expression equation 30. 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,

$$\boldsymbol{x}_i^{\text{adv}} = \boldsymbol{x}_i + \varepsilon^{\text{adv}} \odot \text{sign}(\nabla_{\boldsymbol{x}_i} J(\boldsymbol{\theta}, \boldsymbol{x}_i, y_i)), \tag{30}$$

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

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%

D BENCHMARK METHODS

In this section, we briefly re-introduce the benchmark OOD detection methods with a standardized notation.

D.1 BASELINE

DNNs tend to assign lower confidence for OOD samples. So, calculating the Maximum Softmax Probability (MSP) (Hendrycks & Gimpel, 2017) is a natural baseline for OOD detection. In other words, provided an input data $\boldsymbol{x}$, a pre-trained neural network $f(\cdot)$, and a confidence threshold δ, the OOD score, and the discriminator are given by

$$s(\boldsymbol{x}) = \max_{y \in \mathcal{Y}} \frac{e^{f_y(\boldsymbol{x})}}{\sum_{y' \in \mathcal{Y}} e^{f_{y'}(\boldsymbol{x})}} \quad \text{and} \quad S(\boldsymbol{x}; \delta) = \begin{cases} 1 & \text{if } s(\boldsymbol{x}) \leq \delta \\ 0 & \text{if } s(\boldsymbol{x}) > \delta \end{cases}, \tag{31}$$

respectively. Here, $f_y(\boldsymbol{x})$ indicates the y-th logits output. A limitation of this method is that unscaled softmax posterior distributions are usually spiky, i.e., softmax trained deep neural models are incorrectly calibrated, which does not favor OOD detection (Lakshminarayanan et al., 2017).

D.2 ODIN: OOD DETECTOR FOR NEURAL NETWORKS

In summary, ODIN (Liang et al., 2018) explores the weaknesses of the MSP criterion by recalibrating the output's confidence to the task of OOD detection. They improve the MSP baseline by using the temperature scaled softmax function (expression equation 2) instead. Also, ODIN adds small adversarial noise perturbation to the inputs, i.e.,

$$\widetilde{\boldsymbol{x}} = \boldsymbol{x} - \varepsilon \odot \text{sign}\left(-\nabla_{\boldsymbol{x}} \log q_{\boldsymbol{\theta}}(y|f(\boldsymbol{x}); T)\right), \tag{32}$$

where ε is the perturbation magnitude. Hyperparameters T and ε are tuned on a validation dataset without requiring prior knowledge of test OOD data. They calculate the confidence score by taking the maximum of the perturbed input temperature scaled softmax outputs.

D.3 ENERGY-BASED OOD DETECTOR

An energy-based OOD discriminator is proposed by Liu et al. (2020), where the differences of energies between in-distribution and OOD samples allow for distribution distinction. The energy-based model substitutes the softmax function with the Helmholtz free energy equation to extract a confidence score. They observed that examples with higher energy have a low likelihood of occurrence, concluding that they are likely OOD. The free energy expression is:

$$E(\boldsymbol{x}; f) = -T \cdot \log \sum_{y \in \mathcal{Y}} e^{f_y(\boldsymbol{x})/T}. \tag{33}$$

Note that, differently from ODIN and MSP, they use the information of all of the logits output values through the sum operation. Besides, they apply input pre-processing for further separating OOD data from in-distribution.

D.4 MAHALANOBIS DISTANCE-BASED CONFIDENCE SCORE

The Mahalanobis-based method in Lee et al. (2018) 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. 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:

$$M_\ell(\boldsymbol{x}) = \max_y - \left(f^{(\ell)}(\boldsymbol{x}) - \widehat{\boldsymbol{\mu}}_y^{(\ell)}\right)^\top \widehat{\Sigma}_\ell^{-1} \left(f^{(\ell)}(\boldsymbol{x}) - \widehat{\boldsymbol{\mu}}_y^{(\ell)}\right), \tag{34}$$

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 estimates. 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 $\sum_\ell \alpha_\ell M_\ell(\boldsymbol{x})$.

E ADDITIONAL OUT-OF-DISTRIBUTION DETECTION RESULTS

E.1 HYPERPARAMETERS TUNING

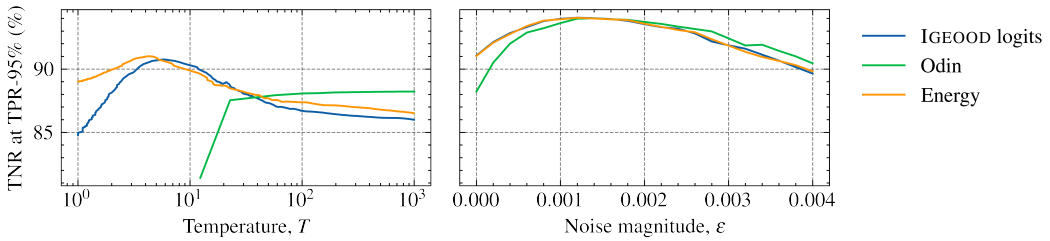
For temperature T, we ran a Bayesian optimization for 500 epochs in the interval of temperature values between 1 and 1000 where the objective function was to maximize the TNR at TPR-95% metric for the validation set. We took the best temperature among 5 runs with different random seeds. For the input pre-processing noise magnitude ε tuning, we ran a grid seach optimization with values among $\{0, 0.0005, 0.001, 0.0014, 0.0015, 0.002, 0.0025, 0.003, 0.0035, 0.0040, 0.01\}$. Table 5 shows the best hyperparametrs we found for the methods in the BLACK-BOX, GREY-BOX, and WHITE-BOX settings.

Table 5: Best temperatures T for the BLACK-BOX setup, best temperature and noise magnitude (T, ε) for the GREY-BOX setup, and best ε for the Mahalanobis score and (T, ε) for the IGEOOD and IGEOOD+ in the WHITE-BOX setup with adversarial tuning.

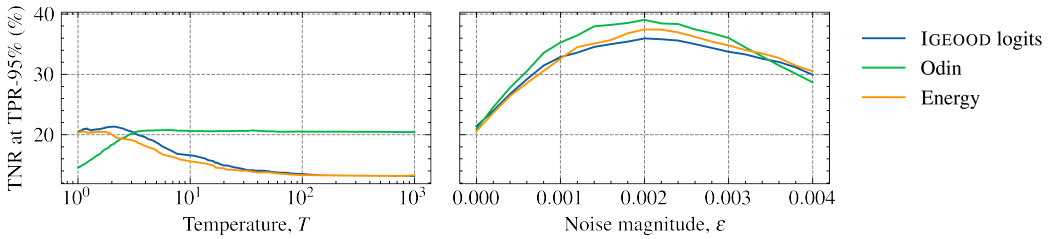
Model	In-dist. dataset	BLACK-BOX ODIN	BLACK-BOX Energy	BLACK-BOX IGEOOD	GREY-BOX ODIN	GREY-BOX Energy	GREY-BOX IGEOOD	WHITE-BOX Maha.	WHITE-BOX IGEOOD,+
DenseNet	C-10	1000	4.6	5.3	(1000, 0.0014)	(4.6, 0.0012)	(5.3, 0.0012)	0	(5, 0.0015)
	C-100	1000	1.1	2.1	(1000, 0.0020)	(1.1, 0.0020)	(2.1, 0.0020)	0	(5, 0)
	SVHN	1	1.1	1.1	(1, 0.0010)	(1.1, 0.0006)	(1.1, 0.0006)	0.001	(5, 0.0015)
ResNet	C-10	1000	5.4	5.3	(1000, 0.0014)	(5.4, 0.0012)	(5.3, 0.0012)	0.0005	(2, 0)
	C-100	1000	1	1	(1000, 0.0020)	(1, 0.0024)	(1, 0.0024)	0.0005	(1, 0)
	SVHN	1000	1.7	1	(1000, 0.0004)	(1.7, 0.0002)	(1.0, 0.0004)	0	(5, 0)

E.2 TEMPERATURE SCALING AND NOISE MAGNITUDE PLOTS

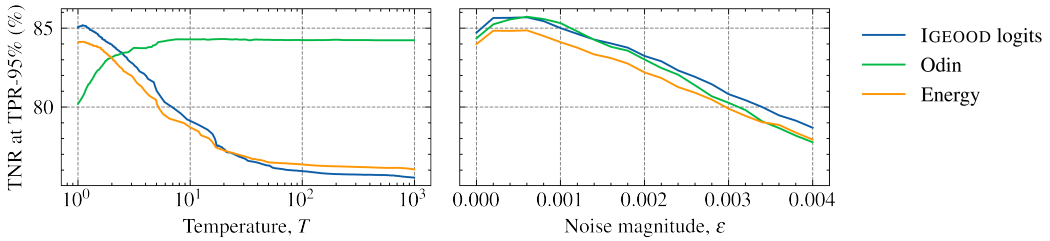
In Figure 3 and 4, we plot on the left hand side column the effect of the temperature parameter in the performance for the BLACK-BOX setup. We set the noise magnitude to zero and measured the TNR at TPR-95% for 500 different temperatures values found by a Bayesian optimization for a variety of DNN models. The performance is evaluated on the iSUN dataset. The right hand side column of Figure 3 and 4 show the effect of the noise magnitude parameter in the performance of IGEOOD score in the GREY-BOX setup. We set the temperature to the best found in the BLACK-BOX case. Then, we measured the OOD performance for 21 values of noise magnitude ε equally spaced in the interval $[0, 0.004]$. The best couple (T, ε) for each method and model is used to evaluate the GREY-BOX performances. The best hyperparameters found are detailed in Table 5.



(a) DenseNet on CIFAR-10.



(b) DenseNet on CIFAR-100.

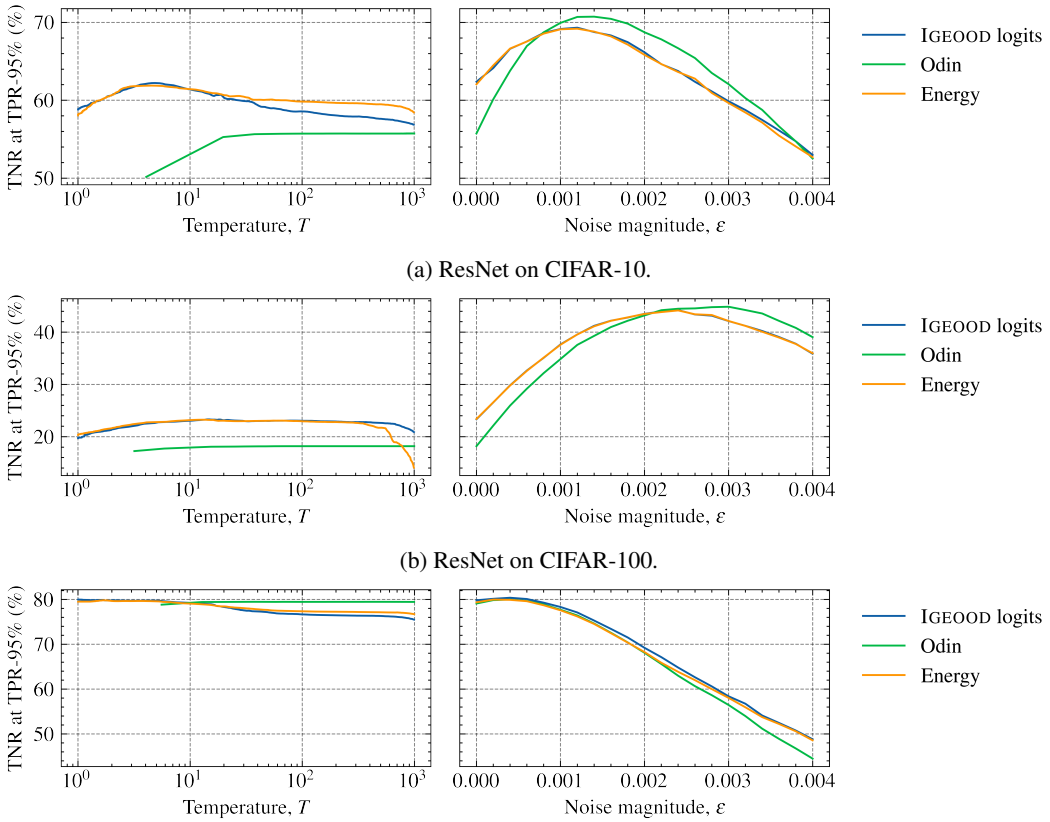


(c) DenseNet on SVHN.

Figure 3: OOD detection performance against temperature and noise magnitude parameters for ODIN (Liang et al., 2018), Energy (Liu et al., 2020) and IGEOOD (ours) on the iSUN (Xu et al., 2015) OOD dataset for a DenseNet-100 architecture.

E.3 ROBUSTNESS OF IGEOOD SCORE CONCERNING THE CHOICE OF THE VALIDATION DATA

To verify the robustness of IGEOOD and other methods to the choice of validation data, we measured the TNR at TPR-95% after tuning our method in a BLACK-BOX and GREY-BOX scenario on nine validation datasets. In Table 6, the first column shows the validation dataset, while we used the remaining OOD datasets to evaluate performance. We obtained consistent results, ranging from 63.4% to 72.0% the average TNR at TPR-95% in the BLACK-BOX case and from 65.0% to 73.4% in the GREY-BOX setting. We show that input pre-processing provides mild amelioration for our method and can be considered a fine-tuning step.



(a) ResNet on CIFAR-10.

(b) ResNet on CIFAR-100.

(c) ResNet on SVHN.

Figure 4: Temperature and noise magnitude tuning for OOD detection performance for ODIN (Liang et al., 2018), Energy (Liu et al., 2020) and IGEOOD (ours) on iSUN (Xu et al., 2015) OOD dataset for a ResNet-34 architecture.

Table 6: BLACK-BOX and GREY-BOX settings average performance across different OOD datasets for validation. The hyperparameters are tuned using one validation dataset (column 1), and evaluation is done on the remaining eight OOD test datasets. The DNN is DenseNet-BC-100 pre-trained on CIFAR-10, and the values are TNR at TPR-95% in percentage.

		BLACK-BOX			GREY-BOX		
Validation set	Baseline	ODIN	Energy	IGEOOD	ODIN	Energy	IGEOOD
iSUN	52.5	64.3	64.9	65.6	66.8	64.8	65.3
Chars	55.0	70.8	71.1	71.4	72.5	72.0	73.4
CIFAR-100	55.4	68.6	69.1	72.0	68.6	71.7	71.3
TinyImageNet	53.0	64.7	65.2	63.5	67.0	65.0	65.5
LSUN	52.1	63.9	63.7	63.6	66.6	65.3	65.0
Places365	55.3	68.5	69.0	71.8	70.0	71.5	70.9
SVHN	55.4	68.7	69.3	69.5	70.0	69.4	70.1
Textures	55.4	71.2	73.1	71.4	71.5	72.4	71.6
Gaussian	49.4	62.8	65.6	63.4	70.4	64.0	68.0
average	53.7	67.1	67.9	68.0	69.3	68.4	69.0

Table 7: Experiment using five different training seeds for DenseNet-100 on CIFAR-10 for the BLACK-BOX scenario. The average test accuracy of the 5 models is 94.58%±0.13%. All values are percentages.

	TNR at TPR-95%	AUROC
Dataset	Baseline / ODIN / Energy / IGEOOD	
Chars	34.1±20.4/54.3±20.2/59.0±14.0/59.4±13.6	88.6±4.3/90.4±3.5/90.2±3.9/90.6±3.4
C-100	37.1±0.5/47.8±1.4/44.8±2.3/45.2±2.2	88.2±0.3/88.8±0.7/88.0±1.0/88.3±0.8
Gaussian	42.7±48.6/74.0±28.3/79.6±23.3/80.2±22.9	93.7±4.3/96.6±2.4/96.7±1.8/96.7±1.8
T.I.N	50.6±4.3/76.1±4.4/78.3±5.0/78.4±5.0	92.5±1.0/95.7±1.0/96.0±1.1/96.1±1.0
LSUN	58.0±3.6/85.2±3.5/87.3±4.7/87.5±4.5	94.2±0.6/97.4±0.6/97.6±0.7/97.7±0.7
Places365	9.3±1.5/54.2±2.5/52.7±4.2/53.2±3.9	88.4±0.4/89.9±1.2/89.4±1.7/89.7±1.5
SVHN	36.0±3.0/48.1±7.1/46.1±9.9/46.5±9.6	86.8±2.0/86.6±4.8/85.9±5.7/86.4±5.0
Textures	35.6±1.7/38.2±1.4/33.3±2.8/34.0±2.6	87.2±0.6/83.3±1.0/80.8±1.9/82.1±1.3
average	41.7±10.4/59.7±8.6/60.1±8.3/60.6±8.0	90.0±1.7/91.1±1.9/90.6±2.2/90.9±1.9

E.4 ERROR BARS AND STANDARD DEVIATION

We conduct all of our experiments during inference time. Provided that we fix the DNN, the in-, and the out-of-distribution datasets, there is no source of randomness to our algorithm because the weights α of the featured ensemble are initialized deterministically. Thus, the OOD scores for the same experimental setting do not change. To confirm this, we ran the same experiment 5 times and obtained the same results in all of them. However, if we allow for retraining the DNN from scratch, we might obtain different parameters, leading to slightly different model accuracy and potentially OOD detection performance. With this in mind, we retrained a DenseNet-BC-100 model on CIFAR-10 5 times with five different random seeds. Then we ran the OOD experiments. The results can be found in table 7.

E.5 EXTENDED OOD DETECTION RESULTS

We show in Table 8 extended OOD detection results of Table 1. It contains the OOD detection performance for each model, in-distribution dataset and OOD dataset in a BLACK-BOX setting. In Table 9, we show the performance of ODIN, energy-based, and IGEOOD scores in the task of OOD detection in a GREY-BOX setup for each OOD dataset. In Table 10 and 11, we show additional results referring to the right-hand column and left-hand column of Table 2, respectively.

Table 8: Extended BLACK-BOX results for table 1. Parameter tuning on iSUN dataset.

In-dist. (model)	OOD dataset	TNR at TPR-95%	AUROC	AUPR
		Baseline / ODIN / Energy / IGEOOD		
CIFAR-10 (DenseNet)	Chars	43.5/**57.2**/54.6/55.0	90.2/**91.2**/90.4/90.5	93.0/**93.1**/92.5/92.7
	CIFAR-100	40.6/**53.1**/50.5/50.7	89.4/**90.4**/89.7/89.8	90.5/**90.7**/90.1/90.2
	Gaussian	88.1/99.8/**99.9**/**99.9**	97.6/**98.9**/98.5/98.5	98.3/**99.3**/99.1/99.1
	TinyImageNet	59.4/85.0/**88.0**/87.8	94.1/97.3/**97.6**/**97.6**	95.4/97.7/**97.9**/**97.9**
	LSUN	66.9/91.4/**93.3**/**93.3**	95.5/98.3/**98.5**/**98.5**	96.5/98.5/**98.7**/**98.7**
	Places365	40.8/**54.2**/51.5/52.0	88.8/**90.2**/89.5/89.7	74.4/**74.8**/73.9/74.3
	SVHN	40.4/**52.0**/49.6/50.1	89.9/**90.9**/90.2/90.3	**84.6**/**84.6**/83.5/83.7
	Textures	40.5/**42.1**/34.9/35.6	**88.5**/85.1/82.4/83.2	**93.1**/88.3/86.5/87.8
	average	52.5/**66.8**/65.3/65.6	91.7/**92.8**/92.1/92.3	90.7/**90.9**/90.3/90.6
CIFAR-100 (DenseNet)	Chars	15.1/**17.8**/17.0/17.2	72.8/**78.0**/77.9/77.8	79.6/83.8/**83.9**/83.8
	CIFAR-10	17.7/**18.1**/17.1/17.0	**75.6**/74.8/74.4/75.4	**78.3**/74.3/74.0/76.2
	Gaussian	0.0/0.0/0.0/0.0	30.2/19.4/19.5/**30.7**	53.2/44.0/44.1/**53.5**
	TinyImageNet	16.7/24.7/25.0/**25.7**	72.0/79.4/**79.6**/79.5	74.8/80.7/**80.9**/80.7
	LSUN	15.5/23.2/24.2/**25.4**	70.9/80.4/**80.8**/80.6	74.4/82.6/**82.9**/82.5
	Places365	18.8/21.2/**20.6**/**20.6**	75.9/**78.0**/77.7/**78.0**	54.2/54.5/54.3/**55.7**
	SVHN	25.7/36.4/36.5/**36.7**	82.8/**88.4**/**88.4**/88.2	75.4/**82.5**/82.4/82.4
	Textures	18.0/22.4/22.2/**22.8**	72.7/74.4/74.3/**75.7**	80.8/79.1/79.0/**81.5**
	average	15.9/20.5/20.3/**20.7**	69.1/71.6/71.6/**73.2**	71.3/72.7/72.7/**74.5**
SVHN (DenseNet)	Chars	46.4/27.0/45.0/**47.9**	**83.9**/52.2/79.6/80.9	**91.8**/70.2/88.9/89.6
	CIFAR-10	61.8/66.0/64.7/**67.0**	**92.3**/90.9/90.3/90.9	**96.2**/95.1/94.6/95.0
	CIFAR-100	61.3/64.4/63.0/**65.4**	**91.9**/90.3/89.5/90.3	**95.7**/94.3/93.8/94.3
	Gaussian	93.6/97.8/97.9/**98.0**	97.4/**98.0**/**98.0**/**98.0**	99.2/**99.4**/**99.4**/**99.4**
	TinyImageNet	80.4/84.4/84.1/**85.1**	**95.5**/95.3/94.9/95.3	**97.9**/97.4/97.2/97.5
	LSUN	80.1/84.4/84.3/**85.0**	**95.5**/95.3/95.1/95.3	**98.0**/97.6/97.5/97.6
	Places365	66.8/71.0/69.9/**71.9**	**93.0**/91.9/91.3/91.9	**89.1**/85.8/84.6/85.8
	Textures	56.4/55.2/52.4/**56.5**	**88.9**/84.6/82.5/84.9	**95.5**/93.3/92.2/93.4
	average	68.3/68.8/70.2/**72.1**	**92.3**/87.3/90.2/90.9	**95.4**/91.6/93.5/94.1
CIFAR-10 (ResNet)	Chars	36.8/45.8/50.7/**51.1**	89.4/90.1/90.3/**90.4**	**92.7**/92.7/92.5/**92.7**
	CIFAR-100	33.6/41.8/45.6/**45.9**	86.4/87.0/87.0/**87.1**	**87.0**/86.6/86.2/86.4
	Gaussian	81.5/**89.9**/88.2/89.0	96.9/**97.3**/96.7/96.7	97.9/**98.3**/98.0/98.0
	TinyImageNet	42.1/53.4/57.7/**58.2**	90.3/91.5/91.6/**91.7**	91.8/**92.2**/92.1/**92.2**
	LSUN	41.2/55.1/61.7/**62.0**	90.1/91.5/92.0/**92.1**	91.5/92.1/**92.3**/**92.3**
	Places365	32.9/42.4/**48.2**/**48.2**	85.8/86.6/**86.9**/**86.9**	**67.1**/66.1/65.6/65.7
	SVHN	27.7/39.9/48.6/**48.8**	89.2/90.2/90.5/**90.6**	**85.8**/85.1/84.1/84.5
	Textures	37.9/46.6/49.6/**50.3**	89.0/**89.1**/88.4/88.6	**93.9**/93.3/92.4/92.6
	average	41.7/51.9/56.3/**56.7**	89.6/90.4/90.4/**90.5**	**88.5**/88.3/87.9/88.1
CIFAR-100 (ResNet)	Chars	14.3/**15.3**/15.1/15.2	72.7/73.0/73.1/**73.4**	**77.8**/77.2/77.3/77.7
	CIFAR-10	18.1/**18.4**/17.4/18.0	76.5/76.6/76.5/**76.8**	**78.3**/77.8/77.8/78.1
	Gaussian	**1.6**/0.8/0.3/0.6	72.8/76.0/**76.7**/**76.7**	81.6/83.8/**84.4**/84.3
	TinyImageNet	17.8/20.7/**23.8**/23.0	73.3/76.6/**77.4**/76.9	76.4/78.7/**79.2**/78.8
	LSUN	15.5/18.8/**21.5**/20.6	70.8/74.1/**74.9**/74.4	72.9/75.2/**75.7**/75.2
	Places365	**17.3**/17.2/16.2/16.9	**74.1**/73.2/72.9/73.4	**44.6**/42.1/41.9/42.7
	SVHN	**14.2**/13.8/12.5/13.3	**74.8**/74.1/73.9/74.5	**59.4**/56.9/56.8/57.7
	Textures	20.9/22.8/**23.3**/**23.3**	77.0/78.0/78.2/**78.3**	85.9/86.2/86.3/**86.4**
	average	15.0/16.0/16.3/**16.4**	74.0/75.2/75.4/**75.5**	72.1/72.3/72.4/**72.6**
SVHN (ResNet)	Chars	56.9/58.2/58.4/**58.5**	**85.1**/83.7/83.7/84.0	**92.3**/91.0/91.0/91.3
	CIFAR-10	79.0/80.6/81.0/**81.3**	**93.0**/92.1/92.2/92.5	**94.7**/93.4/93.4/93.7
	CIFAR-100	78.0/79.6/79.8/**80.2**	**92.7**/91.9/91.9/92.2	**94.7**/93.4/93.4/93.7
	Gaussian	85.3/86.7/86.9/**87.3**	**95.9**/95.7/95.7/**95.9**	**97.7**/97.1/97.1/97.3
	TinyImageNet	79.8/81.4/81.8/**82.2**	**93.5**/92.9/92.9/93.2	**95.3**/94.3/94.3/94.5
	LSUN	**74.9**/76.8/77.2/77.4	**91.5**/90.5/90.6/90.9	**93.5**/92.2/92.1/92.4
	Places365	77.0/78.4/78.8/**79.0**	**92.0**/91.0/91.1/91.4	**81.4**/77.9/77.8/78.6
	Textures	78.5/80.0/80.4/**80.9**	**93.7**/93.0/93.0/93.4	**97.6**/97.0/97.0/97.2
	average	76.2/77.7/78.0/**78.4**	**92.2**/91.3/91.4/91.7	**93.4**/92.0/92.0/92.4
Average of the average values		46.0/51.6/52.1/**52.6**	85.3/85.4/85.1/**85.8**	**85.2**/84.7/84.4/85.1

Table 9: Extended GREY-BOX results. Parameter tuning on iSUN dataset.

In-dist. (model)	OOD dataset	TNR at TPR-95%	AUROC	AUPR
		ODIN / Energy / IGEOOD		
CIFAR-10 (DenseNet)	Chars	**52.5**/49.2/50.0	**89.1**/88.2/88.7	90.7/90.2/**90.8**
	CIFAR-100	48.9/49.7/**50.0**	88.3/88.5/**88.7**	88.4/88.6/**88.9**
	Gaussian	100.0/100.0/100.0	**100.0**/99.9/99.9	**100.0**/99.9/99.9
	TinyImageNet	**92.6**/92.4/92.4	98.5/98.5/98.5	**98.6**/98.5/98.5
	LSUN	96.2/96.2/96.2	**99.2**/99.1/**99.2**	**99.3**/99.2/99.2
	Places365	**52.4**/52.0/**52.4**	89.0/88.9/**89.2**	88.7/88.7/**89.0**
	SVHN	**49.3**/41.8/42.5	**89.7**/88.2/88.6	**82.1**/80.6/81.4
	Textures	**42.9**/37.0/38.7	81.4/80.8/**82.3**	84.7/84.6/**86.5**
	average	**66.8**/64.8/65.3	**91.9**/91.5/**91.9**	91.6/91.3/**91.8**
CIFAR-100 (DenseNet)	Chars	**19.8**/19.2/19.5	**76.6**/76.3/75.8	**80.8**/80.6/79.7
	CIFAR-10	15.3/16.4/**16.9**	72.4/72.8/**74.9**	72.6/72.8/**76.1**
	Gaussian	0.0/0.0/0.0	47.1/46.9/**59.9**	65.5/65.4/**74.9**
	TinyImageNet	**43.8**/42.2/40.1	**86.5**/86.2/84.8	**87.2**/87.0/85.3
	LSUN	**42.2**/40.6/38.9	**86.8**/86.4/84.7	**87.8**/87.7/85.7
	Places365	23.4/23.8/**23.9**	79.1/79.1/**79.4**	79.4/79.4/**80.0**
	SVHN	34.7/31.4/**35.7**	87.7/87.2/**88.1**	81.0/80.8/**81.8**
	Textures	**24.9**/24.7/**24.9**	76.7/76.6/**77.8**	81.7/81.6/**83.8**
	average	**25.5**/24.8/25.0	76.6/76.4/**78.2**	79.5/79.4/**80.9**
SVHN (DenseNet)	Chars	**53.3**/45.8/48.3	**81.8**/78.0/79.3	**90.4**/88.1/88.8
	CIFAR-10	**69.8**/65.1/67.1	**91.3**/89.1/89.9	**95.2**/93.7/94.2
	CIFAR-100	**69.8**/64.2/66.3	**91.1**/88.5/89.4	**94.8**/93.0/93.6
	Gaussian	**99.4**/99.0/99.1	**98.9**/98.5/98.6	**99.7**/99.5/99.6
	TinyImageNet	**88.4**/85.5/86.4	**96.1**/94.9/95.2	**97.9**/97.0/97.3
	LSUN	**88.2**/85.8/86.5	**96.2**/95.1/95.4	**98.0**/97.3/97.5
	Places365	**74.8**/69.9/71.8	**92.4**/90.3/91.0	**95.8**/94.2/94.7
	Textures	**59.4**/49.6/53.7	**84.9**/79.0/81.4	**93.4**/90.4/91.6
	average	**75.4**/70.6/72.4	**91.6**/89.2/90.0	**95.7**/94.2/94.7
CIFAR-10 (ResNet)	Chars	54.7/54.2/**54.8**	**88.3**/87.8/88.2	**90.7**/90.4/90.6
	CIFAR-100	38.9/**41.7**/41.5	83.8/83.8/**84.1**	83.7/83.5/**83.8**
	Gaussian	100.0/100.0/100.0	100.0/99.5/99.5	100.0/99.7/99.7
	TinyImageNet	**68.7**/66.6/67.0	**93.1**/92.4/92.6	**93.3**/92.7/92.9
	LSUN	**70.4**/68.8/68.9	**93.2**/92.8/92.9	**93.3**/92.9/93.0
	Places365	40.4/**43.2**/43.0	84.1/84.1/**84.2**	**84.0**/83.9/**84.0**
	SVHN	38.6/39.9/**40.0**	**84.8**/84.0/84.6	**76.4**/74.6/75.7
	Textures	46.8/47.1/**47.5**	**86.1**/85.2/85.7	**91.4**/90.7/91.0
	average	57.3/57.7/**57.8**	**89.2**/88.7/89.0	**89.1**/88.5/88.8
CIFAR-100 (ResNet)	Chars	14.8/13.8/13.8	68.0/65.1/65.0	71.0/68.5/68.4
	CIFAR-10	**14.3**/11.8/11.8	**70.8**/65.1/64.9	**70.1**/64.7/64.5
	Gaussian	**78.7**/73.8/74.3	**96.8**/95.9/96.0	**97.7**/97.4/97.4
	TinyImageNet	44.9/49.2/**49.3**	86.4/**86.5**/**86.5**	**86.1**/85.6/85.6
	LSUN	40.2/44.3/**44.4**	83.5/**83.8**/**83.8**	82.5/82.5/82.5
	Places365	**15.9**/10.2/10.1	**67.7**/59.7/59.5	**64.5**/58.2/58.0
	SVHN	10.2/**10.6**/**10.6**	64.3/**65.4**/65.3	42.2/**43.6**/43.5
	Textures	**29.5**/27.6/27.7	**77.9**/73.7/73.6	**84.4**/81.1/81.0
	average	**31.1**/30.2/30.2	**76.9**/74.4/74.3	**74.8**/72.7/72.6
SVHN (ResNet)	Chars	59.9/59.4/**60.0**	82.8/**83.2**/**83.2**	90.4/**90.7**/90.6
	CIFAR-10	80.9/81.0/**81.3**	90.9/**91.5**/91.2	92.2/**92.8**/92.5
	CIFAR-100	80.2/80.2/**80.5**	90.8/**91.4**/91.1	92.3/**92.9**/92.6
	Gaussian	**89.4**/88.7/**89.4**	95.9/95.9/95.9	**97.3**/97.2/97.2
	TinyImageNet	82.6/82.3/**82.9**	92.2/**92.6**/92.5	93.5/**93.9**/93.8
	LSUN	77.8/77.6/**78.3**	89.6/**90.1**/89.9	91.3/**91.7**/91.6
	Places365	78.7/78.8/**79.0**	89.8/**90.6**/90.2	91.2/**91.9**/91.5
	Textures	78.6/**79.8**/79.2	91.3/**92.2**/91.7	96.1/**96.6**/96.3
	average	78.5/78.5/**78.8**	90.4/**90.9**/90.7	93.0/**93.5**/93.3
Average of average values		**55.8**/54.4/54.9	**86.1**/85.2/85.7	**87.3**/86.6/87.0

Table 10: WHITE-BOX extended results. Validation on OOD data.

In-dist. (model)	OOD dataset	TNR at TPR-95%	AUROC	AUPR
		Mahalanobis (Lee et al., 2018) / IGEOOD / IGEOOD+		
CIFAR-10 (DenseNet)	Chars	91.3/97.6/**100.0**	98.0/99.3/**100.0**	98.5/99.5/**100.0**
	CIFAR-100	21.4/57.8/**57.9**	68.3/**90.9**/**90.9**	67.1/**90.7**/90.6
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	95.7/97.7/**98.9**	99.0/99.5/**99.8**	99.0/99.5/**99.7**
	iSUN	96.2/97.8/**98.9**	99.1/99.5/**99.8**	99.2/99.6/**99.8**
	LSUN	97.6/98.5/**99.6**	99.3/99.6/**99.9**	99.3/99.7/**99.9**
	Places365	18.1/58.5/**66.3**	70.2/90.9/**93.2**	39.4/74.1/**80.6**
	SVHN	90.1/98.9/**99.4**	98.1/99.7/**99.9**	96.5/99.4/**99.6**
	Textures	84.1/**90.5**/90.1	95.8/98.0/**98.2**	97.3/98.9/**99.0**
	average	77.2/88.6/**90.1**	92.0/97.5/**97.9**	88.5/95.7/**96.6**
CIFAR-100 (DenseNet)	Chars	62.9/79.2/**100.0**	94.0/96.4/**100.0**	95.8/97.6/**100.0**
	CIFAR-10	9.1/21.2/22.4	60.8/78.7/**80.5**	60.1/79.4/**82.2**
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	87.1/94.5/**98.6**	97.3/98.9/**99.7**	97.4/99.0/**99.7**
	iSUN	86.2/93.8/**98.3**	97.2/98.7/**99.6**	97.7/98.9/**99.5**
	LSUN	91.1/95.2/**99.6**	97.9/98.9/**99.9**	98.1/99.0/**99.9**
	Places365	5.6/24.5/**45.7**	54.8/75.9/**87.4**	54.7/74.1/**87.9**
	SVHN	79.0/93.3/**95.8**	96.8/98.5/**99.0**	94.1/96.9/**97.7**
	Textures	70.1/**86.9**/86.8	91.4/**97.2**/**97.2**	94.3/**98.4**/**98.4**
	average	65.7/76.5/**83.0**	87.8/93.7/**95.9**	88.0/93.7/**96.1**
SVHN (DenseNet)	Chars	78.7/80.6/**83.1**	96.1/96.3/**96.9**	98.9/98.9/**99.2**
	CIFAR-10	91.6/91.6/**94.9**	98.0/98.3/**99.0**	99.4/99.4/**99.6**
	CIFAR-100	**92.9**/88.1/92.1	98.2/97.8/**98.5**	99.4/99.2/**99.5**
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	**99.9**/98.2/98.5	**99.8**/99.5/99.7	**99.9**/99.8/**99.9**
	iSUN	**99.9**/98.3/99.3	**99.8**/99.5/**99.8**	**99.9**/99.8/**99.9**
	LSUN	**99.7**/97.1/98.1	**99.7**/99.3/99.6	99.7/99.7/**99.9**
	Places365	94.7/94.2/**97.1**	98.3/98.7/**99.3**	98.4/98.4/**99.1**
	Textures	**98.2**/97.4/97.7	99.4/99.4/**99.5**	99.9/99.9/99.9
	average	95.1/93.9/**95.6**	98.8/98.7/**99.1**	99.5/99.5/**99.7**
CIFAR-10 (ResNet)	Chars	93.6/96.9/**99.8**	98.6/99.2/**99.9**	99.1/99.5/**99.9**
	CIFAR-100	44.9/48.2/**49.2**	87.4/87.1/**89.3**	87.8/86.4/**89.3**
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	96.8/96.3/**97.8**	99.4/99.3/**99.6**	99.4/99.3/**99.5**
	iSUN	97.2/97.2/**99.0**	99.4/99.4/**99.8**	99.5/99.5/**99.8**
	LSUN	98.3/98.4/**99.6**	99.6/99.6/**99.9**	99.6/99.6/**99.9**
	Places365	45.8/48.3/**68.2**	88.1/86.9/**93.0**	**88.1**/86.4/78.0
	SVHN	96.1/98.8/**99.2**	99.0/99.7/**99.8**	98.1/99.3/**99.5**
	Textures	84.3/90.9/**92.1**	97.3/98.3/**98.6**	98.6/99.0/**99.2**
	average	84.1/86.1/**89.4**	96.5/96.6/**97.8**	**96.7**/96.6/96.1
CIFAR-100 (ResNet)	Chars	63.8/72.9/**99.8**	94.0/94.6/**99.9**	96.0/96.0/**99.9**
	CIFAR-10	18.0/20.9/**25.8**	76.6/78.7/**79.2**	76.4/**79.2**/76.9
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	90.1/90.1/**98.0**	97.9/98.0/**99.6**	98.0/97.8/**99.5**
	iSUN	89.4/93.4/**98.9**	97.7/98.7/**99.7**	98.0/98.8/**99.7**
	LSUN	92.4/94.3/**99.7**	98.3/98.8/**99.9**	98.5/98.9/**99.9**
	Places365	23.5/21.2/**51.0**	76.8/75.4/**87.9**	76.0/74.5/**86.3**
	SVHN	88.4/91.6/**98.0**	97.7/98.1/**99.6**	95.2/96.1/**98.8**
	Textures	71.6/81.3/**86.9**	93.9/96.2/**97.2**	96.6/98.0/**98.5**
	average	70.8/74.0/**84.2**	92.5/93.2/**95.9**	92.7/93.3/**95.5**
SVHN (ResNet)	Chars	84.9/76.6/**100.0**	97.0/95.2/**100.0**	99.0/98.6/**100.0**
	CIFAR-10	98.0/97.7/**99.7**	99.2/99.4/**99.8**	99.7/99.8/**99.9**
	CIFAR-100	98.3/96.9/**99.2**	99.3/99.3/**99.7**	99.8/99.8/**99.9**
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	**99.9**/99.7/**99.9**	99.9/99.9/**100.0**	100.0/99.9/**100.0**
	iSUN	**100.0**/99.8/**100.0**	99.9/99.9/**100.0**	100.0/100.0/100.0
	LSUN	**100.0**/99.7/**100.0**	99.9/99.9/**100.0**	100.0/100.0/100.0
	Places365	98.4/97.8/**99.5**	99.3/99.4/**99.8**	**99.8**/99.3/99.7
	Textures	99.0/99.3/**99.4**	99.7/99.8/**99.8**	99.9/**100.0**/**100.0**
	average	97.6/96.4/**99.7**	99.4/99.2/**99.9**	99.8/99.7/**99.9**
Average of average values		81.8/85.9/**90.3**	94.5/96.5/**97.8**	94.2/96.4/**97.3**

Table 11: WHITE-BOX extended results. Validation on adversarial (FGSM) data.

In-dist. (model)	OOD dataset	TNR at TPR-95%	AUROC	AUPR
		Mahalanobis (Lee et al., 2018) / IGEOOD / IGEOOD+		
CIFAR-10 (DenseNet)	Chars	88.5/91.7/**92.8**	97.7/98.4/**98.5**	98.3/98.8/**98.9**
	CIFAR-100	21.5/**33.5**/28.6	68.0/**81.6**/79.3	66.3/**81.1**/78.9
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	93.9/**95.0**/94.5	98.6/**99.0**/98.9	98.6/**99.0**/**99.0**
	iSUN	93.7/95.1/**95.8**	98.7/98.9/**99.1**	98.8/99.1/**99.2**
	LSUN	96.3/**97.3**/97.2	99.1/**99.3**/**99.3**	99.1/**99.4**/**99.4**
	Places365	17.8/**30.7**/27.3	70.0/**82.0**/80.3	40.1/**82.2**/80.6
	SVHN	87.0/**96.8**/95.7	97.2/**99.2**/99.0	93.7/**98.4**/98.0
	Textures	83.6/**87.9**/86.8	95.8/**97.7**/97.4	97.3/**98.8**/98.6
	average	75.8/**80.9**/79.8	91.7/**95.1**/94.7	88.0/**95.2**/94.7
CIFAR-100 (DenseNet)	Chars	52.0/58.9/**59.2**	91.9/**92.0**/**92.0**	**94.4**/94.0/93.9
	CIFAR-10	1.1/**6.7**/**6.7**	43.5/**63.7**/63.3	46.7/**64.7**/64.4
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	86.4/93.8/**94.0**	97.4/**98.8**/**98.8**	97.5/**98.8**/**98.8**
	iSUN	84.7/91.9/**92.2**	97.2/98.3/**98.4**	97.7/**98.6**/**98.6**
	LSUN	88.6/94.2/**94.4**	97.6/98.7/**98.7**	97.9/**98.8**/**98.8**
	Places365	5.5/13.8/**13.9**	56.6/71.1/**71.2**	57.5/43.6/**72.2**
	SVHN	56.1/90.3/**90.6**	91.8/**98.0**/**98.0**	85.4/96.2/**96.3**
	Textures	67.5/84.4/**85.3**	91.2/97.0/**97.1**	94.4/98.3/**98.4**
	average	60.2/70.5/**70.7**	85.2/**90.8**/**90.8**	85.7/88.1/**91.3**
SVHN (DenseNet)	Chars	72.4/**73.6**/63.3	**95.2**/95.0/91.9	**98.5**/98.4/97.1
	CIFAR-10	90.6/**91.0**/86.6	97.7/**98.0**/97.2	**99.1**/99.3/98.8
	CIFAR-100	**91.8**/89.6/84.2	**98.0**/97.9/96.8	**99.2**/**99.2**/98.7
	Gaussian	100.0/100.0/100.0	100.0/99.9/100.0	100.0/100.0/100.0
	TinyImageNet	**99.5**/98.4/96.8	**99.6**/99.5/99.3	99.5/**99.8**/99.7
	iSUN	**99.9**/98.5/98.6	**99.8**/99.4/99.6	**99.9**/99.8/**99.9**
	LSUN	**99.9**/97.8/97.0	**99.8**/99.2/99.3	**99.9**/99.7/99.7
	Places365	**94.3**/93.0/90.9	98.3/**98.4**/98.0	98.1/**99.4**/99.2
	Textures	95.3/**97.1**/94.6	98.8/**99.3**/98.9	99.6/**99.8**/99.7
	average	**93.7**/93.2/90.2	**98.6**/98.5/97.9	99.3/**99.5**/99.2
CIFAR-10 (ResNet)	Chars	82.0/**90.9**/88.7	96.9/**98.3**/98.0	97.7/**98.7**/98.5
	CIFAR-100	**36.5**/21.5/20.5	**84.5**/63.3/62.1	**84.3**/58.1/57.2
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	**96.2**/94.3/94.2	**99.2**/98.0/97.8	**99.2**/96.7/96.4
	iSUN	**96.7**/95.3/95.0	99.3/98.6/98.5	99.4/98.1/97.9
	LSUN	**98.2**/97.7/97.7	**99.5**/99.2/99.2	**99.5**/98.9/98.7
	Places365	**34.8**/15.9/15.3	**85.0**/60.1/59.0	**84.2**/24.4/55.2
	SVHN	81.0/**98.2**/97.7	96.6/**99.3**/99.1	93.7/**97.5**/96.9
	Textures	**81.7**/81.6/80.0	**96.7**/93.4/92.4	**98.2**/94.3/93.4
	average	**78.6**/77.3/76.6	**95.3**/90.0/89.6	**95.1**/85.2/88.2
CIFAR-100 (ResNet)	Chars	39.9/**55.1**/54.3	85.6/**90.4**/90.2	88.1/**92.5**/92.3
	CIFAR-10	3.0/5.0/**6.2**	61.0/59.6/**62.2**	**63.7**/60.6/63.2
	Gaussian	100.0/100.0/100.0	100.0/100.0/100.0	100.0/100.0/100.0
	TinyImageNet	88.7/86.2/**90.1**	97.6/97.3/**98.0**	97.6/97.3/**98.0**
	iSUN	87.9/84.6/**91.0**	97.3/97.0/**98.1**	97.6/97.3/**98.2**
	LSUN	91.3/88.6/**93.9**	98.0/97.8/**98.8**	98.3/98.0/**98.8**
	Places365	8.0/8.6/**10.2**	**67.9**/63.0/66.1	**66.8**/61.7/65.1
	SVHN	31.6/75.2/**78.5**	82.9/95.8/**96.1**	68.8/92.7/**92.9**
	Textures	65.9/78.1/**78.5**	91.9/95.6/**95.7**	95.2/97.6/**97.7**
	average	57.4/64.6/**67.0**	86.9/88.5/**89.5**	86.2/88.6/**89.6**
SVHN (ResNet)	Chars	**75.4**/55.2/37.4	**95.3**/89.2/79.4	**98.5**/96.1/90.9
	CIFAR-10	97.1/96.7/**97.3**	99.1/99.2/**99.3**	99.7/99.7/99.7
	CIFAR-100	**97.5**/96.2/96.0	**99.1**/**99.1**/99.0	**99.7**/99.6/99.6
	Gaussian	100.0/100.0/100.0	99.9/**100.0**/**100.0**	100.0/100.0/100.0
	TinyImageNet	**99.9**/99.6/99.6	99.9/99.9/99.9	99.9/99.9/**100.0**
	iSUN	**100.0**/99.8/99.9	99.9/99.9/99.9	100.0/100.0/100.0
	LSUN	**100.0**/99.8/100.0	99.9/99.9/**100.0**	100.0/100.0/100.0
	Places365	**98.1**/97.0/97.2	99.2/99.2/99.2	99.2/99.0/**99.7**
	Textures	**98.9**/98.4/92.3	**99.6**/**99.6**/98.4	**99.9**/**99.9**/99.6
	average	**96.3**/93.6/91.1	**99.1**/98.4/97.2	**99.6**/99.3/98.8
Average of average values		77.0/**80.0**/79.2	92.8/**93.5**/93.3	92.3/92.6/**93.6**