Understanding ReLU Network Robustness Through Test Set Certification Performance

Anonymous authors
Paper under double-blind review

Abstract

Neural networks might exhibit weak robustness against input perturbations within the learning distribution and become more severe for distributional shifts or data outside the distribution. For their safer use, robustness certificates provide formal guarantees to the stability of the prediction in the vicinity of the input. However, the relationship between correctness and certified robustness remains unclear. In this work, we investigate the unexpected outcomes of verification methods applied to piecewise linear classifiers for clean, perturbed, in- and out-of-distribution samples. In our experiments, we conduct a thorough analysis for image classification tasks and show that robustness certificates are strongly correlated with prediction correctness for in-distribution data. In addition, we provide a theoretical demonstration that formal verification methods robustly certify samples sufficiently far from the training distribution. These results are integrated with an experimental analysis and demonstrate their weakness compared to standard out-of-distribution detection methods.

1 Introduction

Building reliable artificial intelligence systems requires systematic methods for assessing their quality to gain confidence in their correctness or to identify possible failures. In general, neural networks are non-robust against geometric perturbations and are easily fooled by precisely calculated *adversarial attacks* (Biggio et al., 2013; Szegedy et al., 2014). For these reasons, relying solely on model's prediction is not sufficient to ensure safe results. The problem of adversarial attacks has been addressed in the literature with a variety of defense mechanisms, divided into *empirical* and *provable* defenses. Empirical defenses aim to improve the robustness of the model through training with adversarial samples (Goodfellow et al., 2015; Carlini & Wagner, 2017; Madry et al., 2017; Andriushchenko et al., 2020). However, robustness comes at the expense of accuracy (Yang et al., 2020; Jovanovic et al., 2021), and there is absolutely no guarantee that the model will behave correctly in the event of new, unseen attacks. To overcome this problem, formal verification methods, e.g. Reluplex (Katz et al., 2017), are proposed to increase the trustworthiness of a prediction by assuring its stability in the vicinity of the input. Formal verification methods are subsequently divided into *exact* or *complete* (Katz et al., 2017; Bunel et al., 2018; Lu & Kumar, 2020; Xu et al., 2020; Wang et al., 2021) and *approximate* or *incomplete* (Zhang et al., 2018; Dvijotham et al., 2018; Gehr et al., 2018; Müller et al., 2021; 2022; Wang et al., 2021). To give an intuition of how incomplete verification works, consider a convex space constructed around the input and then propagate it through the nonlinearity of the network. At the output layer, the resulting shape is certified as robust if it is entirely contained inside the same predicted class, without crossing any decision boundary.

Motivation. The aim of this work is to understand how robustness certificates can be used at operational time, i.e., when the labels are not given. In this context, robustness certificates (or formal verification methods) ensure the stability of the prediction in the vicinity of the input for a predefined perturbation. But what does this mean in practice? Can we trust the prediction or not? If not, how much does this increase our confidence in correctness? Can we use certified robustness as an additional safety metric? Why should we use robustness certificates, given that a misclassified or out-of-distribution sample can be robustly certified? In our analysis we want to explore how to relate certifiable robustness to correctness at operational time. For this reason, we evaluate robustness certificates on both in- and out-of-distribution cases.

Previous literature has mainly focused on improving robustness verification in qualitative or quantitative terms. For example, increasing the number of certified samples within the correctly classified ones, or speeding up the verification process (Wang et al., 2021; Müller et al., 2021; 2022). Another line explores the tension between adversarial robustness and accuracy from an *empirical* (Yang et al., 2020; Liu et al., 2020a) or *provable* (Jovanovic et al., 2021; Müller et al., 2022) training perspective. The purpose is to emphasize how these training methods increase robustness at the expense of network accuracy and how to mitigate this tradeoff. In this line, common benchmarks (Hendrycks & Dietterich, 2018; Croce et al., 2021; Wu et al., 2022) relate correctness with robustness only for correctly classified samples.

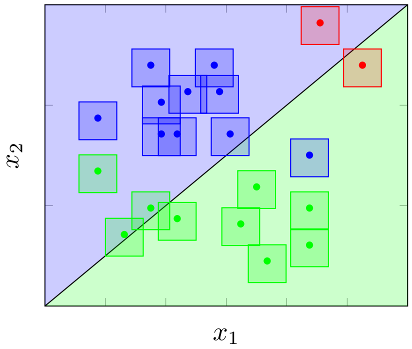


Figure 1: Bi-dimensional visualization of ℓ_∞-norm robustness certificates for ID (●, ●) & OOD (●) samples.

In this paper, we examine it through another perspective. We evaluate certified robustness independently of the classification results. Specifically, we address the following questions. The first is **certified and correctly classified** (●, ● in Figure 1). Can we guarantee that a certified test sample has been classified correctly? This question wants to clarify if it possible to verify whether a test sample is classified correctly or not. And if there exists an optimal adversarial budget for achieving a good trade-off between accuracy and robustness. The second is **In- or Out-Of-Distribution (ID or OOD)** (● in Figure 1). Given an OOD sample that gets high confidence (i.e., is not detected by standard OOD detectors), there are two possible outcomes: certified or not certified. If the number of certified OOD samples is greater than the number of certified ID samples, then we cannot safely use formal verification methods. So, the question arise: can we detect if a test sample is from a different distribution with respect to the training distribution? Is it possible to verify whether a test sample is ID or OOD?

Approach. To answer these questions, we evaluate robustness certificates with two metrics: the Area Under the Receiver Operating characteristic curve (AUROC or AUC) Davis & Goadrich (2006) and the False Positive Rate computed at 95% of true positives (FPR95). In our study, the Receiver Operating Characteristic (ROC) curve is constructed by varying the size of the convex set computed around the input sample. To give an example, if we consider an ℓ_∞-norm certificate computed around an input x, the verification will search through an adversarial example $\tilde{x}$ where the difference $\|x - \tilde{x}\|_\infty$ is at maximum ϵ, with ϵ as predefined condition. Thus, increasing ϵ will likely produce an adversarial example and fail the robustness verification. The ROC curve is than constructed by varying ϵ and considering different definitions of the True Positive Rate (TPR) and False Positive Rate (FPR) differently according to the different scenarios: in- and out-of-distribution.

Since verification methods solve a complex optimization problem, they turn out to be relatively slow. For this reason, our results are constrained to ReLU networks with convolutional or fully connected layers. On the one hand, ReLU activation allows for more accurate verification, i.e. the convex relaxation is relatively more precise (Gehr et al., 2018). On the other hand, the small size of the evaluated models allows for shorter verification times.

Main contribution. In this work, we conduct an in-depth analysis on in- and out-of-distribution data for various networks and certificate types, e.g. geometric or norm-based. To the best of our knowledge, this is the first work that investigates the practical utility of robustness certificates on ID and OOD data.

Our core contributions are summarized as follows:

- First evaluation on the relationship between correctness and certified robustness for clean and perturbed ID and OOD samples. We empirically show that the number of certified samples is directly related to the accuracy of the network and that robustness certificates are a powerful safety metric for ID data.
- Extensive evaluation of verification methods on various data sets, networks, training procedures and certification types.
- Formal proof that robustness certificates are valid for samples sufficiently far from the training distribution in case of piecewise linear classifiers, e.g. ReLU networks. In the task

of OOD detection, we show that the performance of verification methods is relatively lower than standard OOD detection approaches on normally and adversarially trained networks, and significantly lower for networks trained with OOD samples.

2 Background & Related Work

We define a neural network by a function $f : \mathbb{R}^d \to \mathbb{R}^{|\mathcal{K}|}$ which maps input samples $x \in \mathbb{R}^d$ to output $y \in \mathbb{R}^{|\mathcal{K}|}$, where $\mathcal{K} = \{1, \ldots, K\}$ is the set of K classes. We assume a feedforward architecture composed by affine transformations, $f^{(l)}(x) = W^l \sigma^{(l-1)}(x) + b^{(l)}$, for $l = 1, \ldots, L$, and followed by ReLU activation functions, $\sigma^{(k)}(x) = \max\{0, f^{(k)}(x)\}$, for $k = 1, \ldots, L-1$. In the end, the resulting classifier is obtained as composition of pre- and post-activations, i.e. $f^{(L)}(x) = W^{(L)} \sigma^{L-1}(x) + b^{(L)}$. In addition, we define all network parameters $(W^{(l)}, b^{(l)})$ as θ.

Adversarial robustness. Adversarial robustness refers to a model's ability to resist being fooled. Formally, given an input $x \in \mathbb{R}^d$, an adversary is allowed to choose any point from a convex set $\mathbb{S}(x) \subseteq \mathbb{R}^d$. The classifier is certifiably robust for this input x if the predicted class remains unchanged, i.e. $\arg\max_j f_j(\tilde{x}) = \arg\max_j f_j(x)$, $\forall \tilde{x} \in \mathbb{S}(x)$. The set $\mathbb{S}(x)$ can be defined for different specifications, e.g. ℓ_p-norm perturbation (Wong & Kolter, 2018), geometric transformations (Balunovic et al., 2019), randomized smoothing (Cohen et al., 2019) and others. To reduce the vulnerability of the network against adversarially perturbed inputs, the common approach is to train it according to the following min-max optimization problem:

$$\min_{\theta} \mathbb{E}_{(x,y) \sim \mathcal{D}_{\text{in}}} \max_{\tilde{x} \in \mathbb{S}(x)} \mathcal{L}(f(\tilde{x}), y).$$

In the other minimization, we consider training f on an ID dataset $\mathcal{D}_{\text{in}}$, while in the inner maximization we look for the maximum value of the loss function $\mathcal{L}$ that may give us an adversarial sample. As the inner maximization problem results intractable, most of the existing methods rely on approximations. For example, Projected Gradient Descent (PGD) (Madry et al., 2017) and Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2015) are commonly used techniques for improving robustness of a neural network, accomplished by generating adversarial examples and retraining the network with corrected labels.

Robustness certificate. As previously mentioned, a neural network f is certifiably robust for the input $x \in \mathbb{R}^d$ if the prediction for all perturbed versions remains unchanged. Although an adversarially-trained network is robust to attacks created during training, it may still be vulnerable to unseen new attacks. To overcome this problem, formal verification methods model the previous statement as a mathematical optimization problem:

$$\min_{\tilde{x},\, t} \left\{ f_k(\tilde{x}) - f_t(\tilde{x}) \mid \tilde{x} \in \mathbb{S}(x) \;,\; t \in \mathcal{K} \setminus \{k\} \right\},$$

where we denote by k the predicted class for x, i.e. $k = \arg\max_j f_j(x)$. We observe that optimization explores the difference by comparing the outputs of the neural network to predict any class other than the initially predicted one. If the result is positive, we certify the input sample as robust in $\mathbb{S}(x)$. Otherwise, there exists an input that misleads the prediction of the network and the certification fails.

Convex relaxation. To reduce the runtime of the verification process, convex relaxation propagates the input set $\mathbb{S}(x)$ through the network producing lower and upper bounds at every layer. This speeds up the entire verification but sacrifices exactness, resulting in a lower bound at the output layer:

$$\underline{f}_k(\tilde{x}) - \overline{f}_t(\tilde{x}) \leq f^*_k(\tilde{x}) - f^*_t(\tilde{x}),$$

where f^* denotes the optimal result of the verification, and $\underline{f}$, $\overline{f}$ the lower and upper bounds, respectively. Current state-of-the-art methods, e.g. GPUPoly (Müller et al., 2021) or β-CROWN (Wang et al., 2021), parallelize the computation and propagation of boundaries on the GPU.

Robust OOD detection. OOD detection means to determine whether a sample comes from a learned distribution or not. Formal robustness guarantees for low network confidence on OOD samples has been recently investigated. This involves verifying that a predictor assigns low values to all labels for OOD inputs within a defined neighborhood. This challenge may seem straightforward when using current methods such as softmax calibration (Hendrycks & Gimpel, 2017), ODIN (Liang et al., 2018), or Mahalanobis (Lee et al., 2018), but as highlighted in Hein et al. (2019), ReLU networks, i.e. networks with fully connected, convolutional, or residual layers, produce arbitrarily high confidence predictions far from the training data.

To overcome this problem, recent contributions have proposed mathematical guarantees to reduce the confidence far from the training distribution. An early approach of Meinke & Hein (2020) integrates the softmax layer with density estimators based on Gaussian mixture models to distinguish between ID and OOD. Although they achieve similar OOD detection performance as previous approaches, e.g. Outlier Exposure (OE) (Hendrycks et al., 2019), they can guarantee decreasing confidence far from the training distribution. In this direction, Bitterwolf et al. (2020) propose a training approach which uses interval bound propagation (IBP) to derive a provable upper bound on the maximal confidence of the network in a ℓ_∞-norm of ϵ around a given point. This procedure leads to classifiers with pointwise guarantees even for near-OOD samples, but IBP produces loose bounds that cause a drop in network accuracy. Recently, Meinke et al. (2021) combine a binary discriminator to distinguish between ID and OOD together with previous approaches to preserve high clean accuracy while providing adversarial OOD guarantees. Although they achieve state-of-the-art performance in different OOD metrics and test distributions, the results are not yet useful in practice as most are still below 50%.

Most of the existing literature focuses on improving empirical robustness to adversarial attacks inside (Goodfellow et al., 2015; Carlini & Wagner, 2017; Madry et al., 2017) and outside (Hein et al., 2019; Bitterwolf et al., 2020; Meinke & Hein, 2020; Meinke et al., 2021) the distribution or on formally demonstrating network stability in the input neighborhood (Gopinath et al., 2018; Balunovic et al., 2019; Wang et al., 2021; Müller et al., 2022; 2021). Another line of work deals with the trade-off between accuracy and robustness from a training perspective (Yang et al., 2020; Jovanovic et al., 2021; Liu et al., 2020a) or on specific benchmarks (Hendrycks & Dietterich, 2018; Croce et al., 2021; Wu et al., 2022). Unlike these, in this work we evaluate how robustness certificates relate to accuracy on ID samples and how they perform on OOD samples.

3 In-Distribution Analysis

Here, we evaluate formal verification methods for distinct networks, perturbation types and trainings. For this purpose, a benchmark analysis is conducted on clean and perturbed in-distribution data to determine the amount of samples that were successfully *Certified and Correctly classified* (CC) and those that were *Certified but Incorrectly classified* (CI). In this analysis, we want to determine if incorrectly classified samples will be robustly certified or not. In the end, if the ratio of false positives is sufficiently lower or close to zero, we can confidently rely on the robustness verification process as an indicator for correct classification, otherwise not.

Table 1: Summary of the metrics.

# of Samples	Certified Correct (CC)	Certified Incorrect (CI)
Total (N)	CCR = $^{\text{CC}}/_{N}$	CIR = $^{\text{CI}}/_{N}$
Relative	TPR = $^{\text{CC}}/_{C}$	FPR = $^{\text{CI}}/_{I}$

Assuming that we do not process inputs that cannot be verified and instead rely on a fallback strategy - such as a query to an expert - we can calculate the remaining performance and error rate for the samples actually processed by the model. Following Henne et al. (2020), in Table 1 we present similar evaluation metrics. We define the Certified Correct Ratio (CCR) and Certified Incorrect Ratio (CIR) as the number of CC and CI over the total number of samples N, respectively. Similarly, we call the ratio of CC over the total number of correctly classified samples C as TPR, and the ratio of CI over the total number of incorrectly classified samples I as FPR. Thus, with CC and CI, we graphically show the performance of robustness certificates for increasing certification range, while with TPR and FPR, we calculate AUC as an evaluation metric. To certify the robustness for geometric and norm-based perturbations we select the convex verifier GPUPoly (Müller et al., 2021).

3.1 Assessment on unperturbed samples

Geometric robustness. In this context, we utilize DeepG (Balunovic et al., 2019) to compute the linear inequality constraints around the set of geometrically transformed images. In this experiment, we consider clean (unperturbed) test samples and three geometric perturbations: rotation, shearing and scaling.

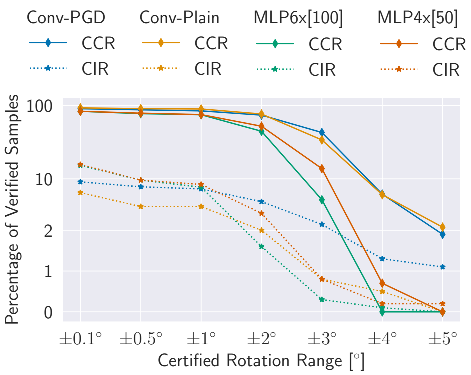


Figure 2: Comparison of network architectures and training methods for varying angles of rotation. We run robustness certificates on 1000 clean samples of the first 10 classes of the GTSRB test set.

In Figure 2, we show a comparison between architectures in terms of percentage of CCR and CIR for increasing rotation values. In the results, we obtain a comparable difference between CCR and CIR for small perturbation intervals, which remains relatively constant by increasing the range. The long computation time to generate the convex set for each rotation interval did not allow for a wide range of points. Here, we graphically show the relationship between robustly certified samples and accuracy. In the end, the goal is to find a certified interval value that decreases CIR while maintaining a high CCR and thus reliability in certification.

The non-rectilinear behavior of the curves should be attributed to the sampling procedure in DeepG (Balunovic et al., 2019). This is inherent in the way of how boundaries are constructed around the geometric perturbation. Before computing the linear inequalities, DeepG operates Monte Carlo sampling to produce geometrically manipulated images inside the given range. In the case of affine transformations, an interpolation method is needed to reinsert the manipulated pixel into the image grid, and due to its non-linear behavior, sampling is required. In the context of DeepG, the number of samples (1000) used for the LP solver and the tolerance (0.01) in Lipschitz optimization were constant for increased perturbation values. Additional experiments are presented in Appendix A.1.

Norm-based robustness. Here, we evaluate ℓ_∞-norm robustness certificates for clean test samples, i.e. $\mathbb{S}(x) = \{\tilde{x} \in \mathbb{R}^d \,|\, \|x - \tilde{x}\|_\infty \leq \epsilon, \epsilon \geq 0\}$. In Figure 3, we plot the ROC curve for each network with 400 ϵ values between zero and 0.2, where ϵ is defined as adversarial perturbation budget. The curve starts from the right hand side for ϵ equal to zero, and increasingly moves to the left hand side. We see that for small CIR∼0.02, the CCR∼0.8 remains surprisingly high. Within this range, the ROC of Conv-Plain stays mostly higher than that of Conv-PGD. We can associate this result to the fact that the plain model has higher accuracy (92.7) with respect to the adversarially trained one (90.8). In contrast, MLP6x[100], while having a slightly higher accuracy, leads to lower ROC than MLP4x[50]. This highlights that larger fully connected models are less likely to be certified and reduce the performance of robustness certificates. We show CCR and CIR curves and ROC curves of TPR & FPR in Figure 7 in Appendix A.2.

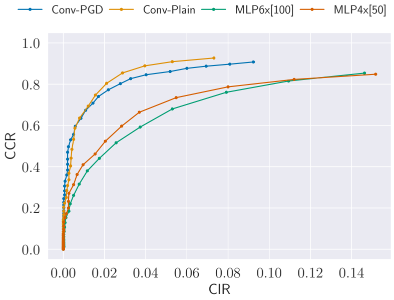


Figure 3: ROC curves of network architectures and training methods for varying ϵ of ℓ_∞-norm based robustness certificates on 4800 samples of the first 10 classes of the GTSRB test set.

3.2 Assessment on perturbed samples (distributional shift)

Here, we test geometrically manipulated in-distribution samples (or distributional shifts). To this end, we run each network on perturbed samples and evaluate the verification results for the predicted class. Testing neural networks for distributional shifts, practically assesses their use for real-world applications.

In Table 2, we show the results of AUCs for different networks and perturbations. The ROC curves are approximately estimated with 400 ϵ values between 0 and 0.02, i.e. 20% of maximum adversarial budget, which push the amount of certified samples to zero for all tests and models. Similarly to the previous section, TPR and FPR (used to generate ROC curves) are calculated with ℓ_∞-norm robustness certificates. In our analysis, we observe that the accuracy on perturbed test sets has decreased and similarly AUC decreased by the same amount. Convolutional networks maintains higher AUCs

Table 2: **AUC / ACC:** Comparison between plain and perturbed test samples. The ROCs were calculated with ℓ_∞-norm robustness certificates by varying ϵ. Random perturbation sizes inside the defined ranges are applied to the 4800 test samples of the first 10 classes of the GTSRB test set.

Perturbation		**Conv-PGD**		**Conv-Plain**		**MLP6x[100]**		**MLP4x[50]**	
Type	**Size**	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC
Unperturbed	-	87.7	90.8	85.7	92.7	75.8	85.4	81.4	84.8
Gaussian Blur	$K = 3, \sigma \in [1, 2]$	82.4	85.7	79.7	91.7	69.3	80.2	66.9	77.9
Rotation	$[-30°, +30°]$	77.2	71.9	69.0	63.8	66.5	59.7	70.4	59.0
Scaling	$[0.1, 1]$	54.4	38.6	50.9	38.6	49.6	24.9	53.1	27.4

with respect to fully connected models and Conv-PGD achieves the best results besides its lower accuracy. Since adversarial training of ℓ_∞-norm samples helps the same kind of verification, the more certified true positives, the higher the AUC will be.

Both metrics followed a similar trend, enhancing the relationship between accuracy and certified robustness. This result is intended to state that distributional shifts (or perturbed ID samples) are as difficult to verify as the network's generalization ability is lower. This is independent of the training procedure. Despite the fact that adversarially-trained networks obtain higher AUCs than plain models (mirroring the results for unperturbed samples), the AUC of adversarially-trained networks decreases analogously with respect to accuracy. Validating the fact that accuracy is correlated with robustness.

Discussion. We summarize the analysis on clean and perturbed ID samples by pointing out that robustness is strongly related to accuracy. This is visible in Figure 3, where increasing the certification range reduces both correctly and incorrectly classified samples. Luckily, we obtain more CC than CI samples for small perturbation budgets, showing that robustness certificates are a powerful safety metric for ID data. To give a numerical example, if the accuracy decreases by $\sim$11% (from 90% to 80%), the error rate drops of $\sim$75% (from 8% to 2%). We obtained similar results for both types of certification (geometric and norm-based). Therefore, similar conclusion can be derived for others verification and training methods as well, e.g. randomised smoothing. In short, the more certification-inclined a network is, the better we can use verification methods as metric to distinguish between correctly and incorrectly classified samples.

4 Out-Of-Distribution Analysis

Motivation Hein et al. (2019) demonstrated that piecewise linear classifiers held high confidence for samples outside the training distribution and post-processing techniques for softmax scores are not able to reduce this confidence. This is an inherent problem of the network architecture that further leads to the incorrect use of formal verifiers. A crucial issue on the adoption of such methods is that OOD samples not only yield high confidence, but are easily verifiable and get certified as correct. In this section, we theoretically show that robustness certificates hold for samples sufficiently far from the training distribution. Then, we support our findings with numerical results on OOD test samples.

4.1 Theoretical Analysis

Here, we formally show that robustness certificates are always valid for piecewise linear classifiers and for samples sufficiently far from the training distribution. This finding is a derivation of the more general result demonstrated in Hein et al. (2019), so let us introduce some definitions necessary for the main proof. We briefly recall the definition of continuous piecewise affine classifiers (Arora et al., 2018), which applies to feedforward neural networks with piecewise affine activation functions, e.g. ReLU, and linear at the output layer.

Definition 4.1. A function $f : \mathbb{R}^d \to \mathbb{R}$ is called piecewise affine if there exists a finite set of polytopes $\{Q_r\}_{r=1}^M$ (referred to as linear regions of f) such that $\cup_{r=1}^M Q_r = \mathbb{R}^d$ and f is an affine function when restricted to every Q_r.

This definition applies to all layers performing linear mappings, e.g. fully connected, convolutional, residual layers, skip connections and further maximum and average pooling. Specifically, given a classifier $f : \mathbb{R}^d \to \mathbb{R}^K$, where K is the number of classes, Definition 4.1 applies to each component

$f_i : \mathbb{R}^d \to \mathbb{R}$ and all K components $(f_i)_{i=1}^K$ have the same set of linear regions. We further extend the definition of ReLU networks as piecewise linear classifiers with the fact that all linear regions are polytopes and thus convex sets (Hein et al., 2019).

Lemma 4.1 (Hein et al. (2019))**.** Let $\{Q_r\}_{r=1}^R$ be the set of convex polytopes where the ReLU-classifier $f : \mathbb{R}^d \to \mathbb{R}^K$ is an affine function, meaning for every $k \in \{1, \dots, R\}$ and $x \in Q_k$ there exists $V^k \in \mathbb{R}^{K \times d}$ and $c^k \in \mathbb{R}^K$ such that $f(x) = V^k x + c^k$. Thus, for any $x \in \mathbb{R}^d \setminus \{0\}$ there exists $\alpha \in \mathbb{R}$ with $\alpha > 0$ and $r \in \{1, \dots, R\}$ such that $\beta x \in Q_r$ for all $\beta \geq \alpha$.

Given Lemma 4.1, we can state our result.

Theorem 4.1. Let $\cup_{r=1}^M Q_r = \mathbb{R}^d$ and $f(x) = V^r x + a^r$ be the piecewise affine representation of the output of a ReLU network on Q_r. If V^r does not contain identical rows for all $r = 1, \dots, R$, then for almost any $x \in \mathbb{R}^d \setminus \{0\}$, there exists $\alpha \in \mathbb{R}$ with $\alpha > 0$ and a predicted class $k \in \mathcal{K}$ such that:

$$\min_{z,t} \{f_k(z) - f_t(z) \mid z \in \mathbb{S}(\alpha x),\ t \in \mathcal{K} \setminus \{k\}\} > 0,$$

holds for $\mathbb{S}(\alpha x) \subset Q_r$.

The proof is given in Appendix B. The main consequence of this theorem consists in the fact that the surrounding of infinitely many samples far enough from the training distribution would be easily certified as robust. As already noted in Hein et al. (2019), the constraint on V^r is very weak. On the other side, the fact that $\mathbb{S}(\alpha x) \subset Q_r$ is not straightforward, since the definition of $\mathbb{S}(x)$ may vary depending on the type of certificate one is interested in.

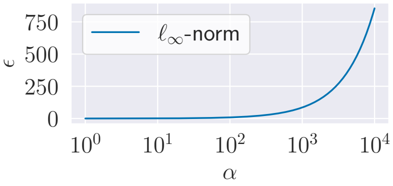


Figure 4: Given a single input αx, we compute the robustness certificate based on the ℓ_∞-norm for increasing α. As noted, α and ϵ are linearly correlated.

In Figure 4, we show the relationship between α and ϵ for ℓ_∞-norm robustness certificates, where ϵ is the adversarial budget, i.e. $\mathbb{S}(\alpha x) = \{\tilde{x} \in \mathbb{R}^d \mid \|\alpha x - \tilde{x}\|_\infty \leq \epsilon,\ \epsilon \geq 0\}$. We display the maximum ϵ value for which the certificate holds for increasing α. One can note that this settings is unlikely in practice as all the images are normalized to be inside the interval $[0, 1]^d$ and therefore $\epsilon \in [0, 1]$. Besides this, we theoretically demonstrate that samples far enough from the training distribution are expected to be certified and that the certification range augments with the distance.

This is a major problem for the practical use of robustness certificates, since samples are likely to be certified quite far from the training distribution. Hence, the use of formal verification methods suggests integration with OOD detectors.

4.2 Experimental Analysis

In this section, we conduct experiments on OOD samples for different datasets, networks and training methods. The aim is to evaluate the performance of robustness certificates in detecting whether a sample is in- or out-of-distribution. To this end, we compare the convex verifier *GPUPoly* (Müller et al., 2021) against standard OOD detection methods: *MaxSoftmax* (Hendrycks & Gimpel, 2017), *ODIN* (Liang et al., 2018), *Mahalanobis* (Lee et al., 2018) and *Energy* (Liu et al., 2020b). We ran all methods on the entire test set except GPUPoly, which was executed only on the first 1000 test samples. This is due to the incredibly long run time of validating a large amount of samples for a large range of ϵ values. The ROC curve for GPUPoly has been computed by varying the adversarial budget ϵ. Here, we consider as true positives all certified samples from the ID test set, and as false positives all certified samples from the OOD test set. We define 4 000 ϵ values equidistant between 0 and 0.2, i.e. 20% of maximum adversarial budget, which push the amount of certified samples to zero for all tests and models. As an example, verifying 1 000 images on our largest network (31360 neurons) with GPUPoly takes about 20 minutes per single ϵ. We conduct our experiments on a Nvidia GPU RTX 3090.

Grayscale Category. In Table 3, we show the results on grayscale datasets, where we use MNIST as ID dataset. Given the limited size of the models, PGD and FGSM attacks prevent convergence during training, so we evaluate only Plain, OE and Randomized trained networks in this analysis. In case of OE, we consider two datasets: OrganAMNIST and FMNIST. As might be expected,

Table 3: **Grayscale Category:** Comparison between standard OOD detection methods and robustness certificates of ℓ_∞-norm: GPUPoly(ℓ_∞) (Müller et al., 2021). We report the clean accuracy (ACC) on the in-distribution (ID) dataset: MNIST. All methods were executed on all samples in the test set except GPUPoly, which was executed on the first 1000 test samples. In the context of GPUPoly, AUC and FPR95 are computed by varying the adversarial budget ϵ.

Network/ Training	**ID: MNIST ACC**	**Method**	**EMNIST (letters)**		**KMNIST**		**FMNIST**	
			AUC↑	**FPR95↓**	**AUC↑**	**FPR95↓**	**AUC↑**	**FPR95↓**
MLP6x[200]/ Plain	98.0	MaxSoftmax (Hendrycks & Gimpel, 2017)	87.3	43.8	91.8	32.1	88.8	40.5
		ODIN (Liang et al., 2018)	89.1	**35.6**	**93.0**	**26.0**	91.2	**32.5**
		Mahalanobis (Lee et al., 2018)	**89.2**	47.4	91.8	38.8	**92.8**	40.6
		Energy (Liu et al., 2020b)	87.3	45.0	92.6	32.4	89.6	40.5
		GPUPoly(ℓ_∞) (Müller et al., 2021)	81.3	61.9	86.4	51.0	76.1	62.0
MLP6x[200]/ OE (OrganAMNIST)	97.9	MaxSoftmax (Hendrycks & Gimpel, 2017)	90.6	38.6	**98.2**	8.7	97.9	12.4
		ODIN (Liang et al., 2018)	**90.7**	**36.2**	98.1	**8.4**	**98.2**	**10.1**
		Mahalanobis (Lee et al., 2018)	90.5	39.7	97.0	15.1	97.3	12.2
		Energy (Liu et al., 2020b)	90.5	39.0	98.1	9.3	97.4	14.1
		GPUPoly(ℓ_∞) (Müller et al., 2021)	82.3	55.0	92.1	31.0	87.7	45.0
MLP6x[200]/ OE (FMINST)	98.2	MaxSoftmax (Hendrycks & Gimpel, 2017)	**97.0**	**11.7**	**99.8**	**0.8**	-	-
		ODIN (Liang et al., 2018)	96.5	13.7	**99.8**	0.9	-	-
		Mahalanobis (Lee et al., 2018)	96.1	15.7	99.7	1.4	-	-
		Energy (Liu et al., 2020b)	96.9	12.3	**99.8**	0.9	-	-
		GPUPoly(ℓ_∞) (Müller et al., 2021)	89.7	29.3	94.1	25.2	-	-
ConvSmall/ Plain	98.8	MaxSoftmax (Hendrycks & Gimpel, 2017)	79.3	61.1	85.5	51.4	85.7	58.1
		ODIN (Liang et al., 2018)	80.0	60.1	85.3	51.8	85.0	59.1
		Mahalanobis (Lee et al., 2018)	**91.4**	**38.9**	**92.0**	43.1	**91.9**	**43.2**
		Energy (Liu et al., 2020b)	80.5	57.7	85.3	51.9	83.5	64.2
		GPUPoly(ℓ_∞) (Müller et al., 2021)	81.9	48.3	87.1	**39.4**	78.3	64.6
ConvSmall/ Randomized ($\sigma = 0.1$)	98.7	MaxSoftmax (Hendrycks & Gimpel, 2017)	73.5	73.0	87.4	48.0	81.3	65.9
		ODIN (Liang et al., 2018)	73.5	72.5	86.9	49.7	79.6	67.7
		Mahalanobis (Lee et al., 2018)	**91.6**	**38.7**	89.7	54.1	**82.7**	61.0
		Energy (Liu et al., 2020b)	75.2	68.6	87.3	47.8	79.9	69.0
		GPUPoly(ℓ_∞) (Müller et al., 2021)	84.8	50.3	**90.2**	**39.6**	82.4	**58.9**

networks trained with OE perform significantly better than those trained with Plain or Randomized. The results with FMNIST as OOD training set compared to OrganAMNIST are surprisingly close to optimum in KMNIST for all standard OOD detection methods.

In the context of GPUPoly, we observe better results compared to other methods for convolutional networks in the KMNIST dataset, and definitely lower results for fully connected models. On the one hand, GPUPoly struggles to certify samples in distribution, leading to inferior results than standard OOD detection methods. On the other hand, for FPR at 95% of true positives, we obtain more certified OOD samples, empirically validating the hypothesis that verification methods easily certified samples far enough from the training distribution. The hardness of verifying OE trained networks should be related to the slightly thinner decision boundaries induced during the training procedure. Surprisingly, the randomized trained convolutional network performed slightly better than its plain counterpart. In appendix C.1, we report the ROC curves for convolutional networks.

Table 4: **RGB Category:** Comparison between standard OOD detection methods and GPUPoly(ℓ_∞) (Müller et al., 2021) for different training methods of the ConvMed network. We report the clean accuracy (ACC) on the in-distribution (ID) dataset: GTSRB. In the context of GPUPoly, the AUC and FPR95 are computed by varying the adversarial budget ϵ of the ℓ_∞-norm based robustness.

Training	**ID: GTSRB ACC**	**Method**	**CIFAR10**		**CIFAR100**		**SVHN**	
			AUC↑	**FPR95↓**	**AUC↑**	**FPR95↓**	**AUC↑**	**FPR95↓**
OE (ImageNet (C))	83.3	MaxSoftmax (Hendrycks & Gimpel, 2017)	97.9	0.6	**97.9**	1.0	**97.7**	2.5
		ODIN (Liang et al., 2018)	**99.9**	**0.4**	**97.9**	**0.8**	**97.7**	2.4
		Mahalanobis (Lee et al., 2018)	97.8	0.5	97.7	1.1	97.3	**1.7**
		Energy (Liu et al., 2020b)	97.7	0.6	97.7	0.9	97.5	2.7
		GPUPoly(ℓ_∞) (Müller et al., 2021)	18.9	99.3	20.1	99.2	34.2	97.3
FGSM ($\epsilon = 1/255$)	84.1	MaxSoftmax (Hendrycks & Gimpel, 2017)	61.4	94.7	64.0	93.0	77.1	81.2
		ODIN (Liang et al., 2018)	**66.9**	**81.9**	**69.5**	**78.4**	80.9	64.0
		Mahalanobis (Lee et al., 2018)	62.8	83.2	63.6	82.9	**81.9**	**61.9**
		Energy (Liu et al., 2020b)	62.2	95.8	65.1	94.1	76.2	87.3
		GPUPoly(ℓ_∞) (Müller et al., 2021)	57.9	95.1	60.5	95.0	70.4	90.7
PGD ($\epsilon = 1/255$)	81.4	MaxSoftmax (Hendrycks & Gimpel, 2017)	58.1	96.0	58.7	92.9	83.7	69.5
		ODIN (Liang et al., 2018)	64.0	85.1	63.2	80.4	88.2	47.7
		Mahalanobis (Lee et al., 2018)	**73.8**	**75.1**	**68.1**	**79.1**	**89.0**	**44.7**
		Energy (Liu et al., 2020b)	54.1	97.9	55.2	95.6	80.0	78.1
		GPUPoly(ℓ_∞) (Müller et al., 2021)	55.4	95.5	58.2	93.8	70.3	90.5
Randomized ($\sigma = 0.1$)	83.7	MaxSoftmax (Hendrycks & Gimpel, 2017)	61.6	94.8	62.6	92.6	83.5	71.0
		ODIN (Liang et al., 2018)	**67.1**	82.9	**67.6**	**79.9**	87.4	51.8
		Mahalanobis (Lee et al., 2018)	65.6	**80.6**	64.0	83.1	**88.2**	**47.1**
		Energy (Liu et al., 2020b)	60.9	95.3	62.3	93.2	81.7	76.4
		GPUPoly(ℓ_∞) (Müller et al., 2021)	60.4	92.1	62.7	91.3	72.0	86.9

RGB Category. Here, we test two convolutional networks of different sizes: ConvSmall and ConvMed, trained on two ID datasets: GTSRB and CIFAR10. In general, the clean accuracy is remarkably low compared to state-of-the-art models, and slightly lower for adversarially trained networks than for plain models, but it is aligned with related work on verification methods (see, e.g., (Müller et al., 2021; 2022)).

In Table 4, we show the results for the ConvMed model. In this setting, we trained each network on all 43 classes of the GTSRB dataset. As a consequence, we obtain lower accuracy with respect to the models of section 3 trained on just the first 10 classes. Similarly to the grayscale category, standard OOD detection methods perform likewise. In the case of OE, GPUPoly certifies more OOD than ID samples, drawing the AUC below the random guess value of 0.5. On the one side, adversarial training procedures, such as PGD, FGSM and randomized, do not seem to help the verification process, resulting in substandard performance for GPUPoly. On the other side, standard OOD detection methods are slightly affected.

In Figure 5, we show the ROC curve of ConvMed trained with OE on the GTSRB as ID and ImageNet cropped as OOD training sets and tested on SVHN. As noted, GPUPoly certifies more OOD than ID samples. We discuss this behavior and affiliate it with a couple of reasons. First, OE induces an irregular gradient that leads the verification process to fail in the case of both ID and OOD samples. This decreases the number of robustness certified samples, and affects TPR and FPR equally. Second, OE reduces the accuracy on ID samples, leaving more stable gradients and a larger prediction space for OOD samples. This augments the number of OOD certified samples, and increases the FPR. In the end, this experiment empirically validates the theoretical results discussed above. Additional results are shown in Appendix C.

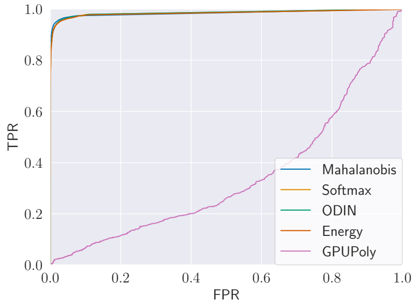


Figure 5: Comparison of ROC curves for standard OOD detection methods and GPUPoly on SVHN dataset. We consider the ConvMed model trained with OE on the GTSRB as ID and ImageNet cropped as OOD training sets.

Discussion. In this analysis, we have shown that robustness certificates perform similarly than standard OOD detection methods for adversarially-trained networks, both on grayscale and RGB images. The results are completely different for networks trained with OE, where the ROC is lower than the random guess. This highlights the problematic nature of formal verification methods in easily certifying OOD samples for networks trained to be OOD aware. Hence, the use of robustness certificates on piecewise linear classifiers needs to be complemented with additional safety measures.

5 Conclusion

In this paper, we explored the robustness of ReLU networks through an in-depth analysis of clean and perturbed samples inside and outside the distribution. We used convex verification methods to robustly certify the network prediction. By varying the adversarial perturbation budget ϵ, we constructed ROC curves. In our in-distribution analysis, we showed that there is a strong correlation between certified robustness and accuracy for clean and perturbed samples. In this context, formal verification methods prove to be a useful error-reduction metric leading to an overall increase in reliability. Completely different are the results for the OOD analysis, where robustness certificates demonstrate their unreliability compared to standard OOD detection methods. In such analysis, we proved theoretically that samples far enough from the training distribution are easily certified for ReLU classifiers. In our experiments, we validate the theoretical findings through an extensive analysis. The results suggest the addition of OOD detection measures for the practical use of robustness certificates in real applications.

In the end, verification methods can be a piece of the puzzle toward trustworthy AI. As a future perspective, it would be interesting to combine these methods with standard OOD detection methods in a useful way. However, it is still unclear how to distinguish between the two for a given sample.

REFERENCES

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: A query-efficient black-box adversarial attack via random search. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision – ECCV 2020*, pp. 484–501, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58592-1.

Raman Arora, Amitabh Basu, Poorya Mianjy, and Anirbit Mukherjee. Understanding deep neural networks with rectified linear units. In *International Conference on Learning Representations*, 2018.

Mislav Balunovic, Maximilian Baader, Gagandeep Singh, Timon Gehr, and Martin Vechev. Certifying geometric robustness of neural networks. 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/f7fa6aca028e7ff4ef62d75ed025fe76-Paper.pdf`.

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 387–402. Springer, 2013.

Julian Bitterwolf, Alexander Meinke, and Matthias Hein. Certifiably adversarially robust detection of out-of-distribution data. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL `https://proceedings.neurips.cc/paper/2020/hash/b90c46963248e6d7aab1e0f429743ca0-Abstract.html`.

Rudy R Bunel, Ilker Turkaslan, Philip Torr, Pushmeet Kohli, and Pawan K Mudigonda. A unified view of piecewise linear neural network verification. *Advances in Neural Information Processing Systems*, 31, 2018.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pp. 39–57. Ieee, 2017.

Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical japanese literature. *arXiv preprint arXiv:1812.01718*, 2018.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. Emnist: Extending mnist to handwritten letters. In *2017 international joint conference on neural networks (IJCNN)*, pp. 2921–2926. IEEE, 2017.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1310–1320. PMLR, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/cohen19c.html`.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pp. 233–240, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933832. doi: 10.1145/1143844.1143874. URL `https://doi.org/10.1145/1143844.1143874`.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Krishnamurthy Dvijotham, Robert Stanforth, Sven Gowal, Timothy A Mann, and Pushmeet Kohli. A dual approach to scalable verification of deep networks. In *UAI*, volume 1, pp. 3, 2018.

Timon Gehr, Matthew Mirman, Dana Drachsler-Cohen, Petar Tsankov, Swarat Chaudhuri, and Martin Vechev. Ai2: Safety and robustness certification of neural networks with abstract interpretation. In *2018 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, 2018. doi: 10.1109/SP.2018.00058.

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.

Divya Gopinath, Guy Katz, Corina S. Păsăreanu, and Clark Barrett. Deepsafe: A data-driven approach for assessing robustness of neural networks. In Shuvendu K. Lahiri and Chao Wang (eds.), *Automated Technology for Verification and Analysis*, pp. 3–19, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01090-4.

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. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 41–50. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00013. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Hein_Why_ReLU_Networks_Yield_High-Confidence_Predictions_Far_Away_From_the_CVPR_2019_paper.html.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2018.

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. URL https://openreview.net/forum?id=Hkg4TI9xl.

Dan Hendrycks, Mantas Mazeika, and Thomas G. Dietterich. Deep anomaly detection with outlier exposure. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=HyxCxhRcY7.

Maximilian Henne, Adrian Schwaiger, Karsten Roscher, and Gereon Weiss. Benchmarking uncertainty estimation methods for deep learning with safety-related metrics. In *SafeAI@ AAAI*, pp. 83–90, 2020.

Sebastian Houben, Johannes Stallkamp, Jan Salmen, Marc Schlipsing, and Christian Igel. Detection of traffic signs in real-world images: The german traffic sign detection benchmark. In *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2013. doi: 10.1109/IJCNN.2013.6706807.

Nikola Jovanovic, Mislav Balunovic, Maximilian Baader, and Martin T. Vechev. Certified defenses: Why tighter relaxations may hurt training? *CoRR*, abs/2102.06700, 2021. URL https://arxiv.org/abs/2102.06700.

Guy Katz, Clark Barrett, David L Dill, Kyle Julian, and Mykel J Kochenderfer. Reluplex: An efficient smt solver for verifying deep neural networks. In *International conference on computer aided verification*, pp. 97–117. Springer, 2017.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images.(2009), 2009.

Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp.

7167–7177, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/abdeb6f575ac5c6676b747bca8d09cc2-Abstract.html.

Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=H1VGkIxRZ.

Chen Liu, Mathieu Salzmann, Tao Lin, Ryota Tomioka, and Sabine Süsstrunk. On the loss landscape of adversarial training: Identifying challenges and how to overcome them. *Advances in Neural Information Processing Systems*, 33:21476–21487, 2020a.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 33:21464–21475, 2020b.

Jingyue Lu and M. Pawan Kumar. Neural network branching for neural network verification. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=B1evfa4tPB.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Alexander Meinke and Matthias Hein. Towards neural networks that provably know when they don't know. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=ByxGkySKwH.

Alexander Meinke, Julian Bitterwolf, and Matthias Hein. Provably robust detection of out-of-distribution data (almost) for free. *arXiv preprint arXiv:2106.04260*, 2021. URL https://arxiv.org/abs/2106.04260.

Christoph Müller, François Serre, Gagandeep Singh, Markus Püschel, and Martin Vechev. Scaling polyhedral neural network verification on gpus. In A. Smola, A. Dimakis, and I. Stoica (eds.), *Proceedings of Machine Learning and Systems*, volume 3, pp. 733–746, 2021. URL https://proceedings.mlsys.org/paper/2021/file/ca46c1b9512a7a8315fa3c5a946e8265-Paper.pdf.

Mark Niklas Müller, Franziska Eckert, Marc Fischer, and Martin Vechev. Certified training: Small boxes are all you need. *arXiv preprint arXiv:2210.04871*, 2022.

Mark Niklas Müller, Gleb Makarchuk, Gagandeep Singh, Markus Püschel, and Martin Vechev. Prima: General and precise neural network certification via scalable convex hull approximations. *Proc. ACM Program. Lang.*, 6(POPL), jan 2022. doi: 10.1145/3498704. URL https://doi.org/10.1145/3498704.

Pierre Sermanet, Soumith Chintala, and Yann LeCun. Convolutional neural networks applied to house numbers digit classification. In *Proceedings of the 21st international conference on pattern recognition (ICPR2012)*, pp. 3288–3291. IEEE, 2012.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In Yoshua Bengio and Yann LeCun (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6199.

Shiqi Wang, Huan Zhang, Kaidi Xu, Xue Lin, Suman Jana, Cho-Jui Hsieh, and J. Zico Kolter. Beta-crown: Efficient bound propagation with per-neuron split constraints for neural network robustness verification. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 29909–29921. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/fac7fead96dafceaf80c1daffeae82a4-Paper.pdf.

Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5286–5295. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/wong18a.html`.

Haoze Wu, Teruhiro Tagomori, Alexander Robey, Fengjun Yang, Nikolai Matni, George Pappas, Hamed Hassani, Corina Pasareanu, and Clark Barrett. Toward certified robustness against real-world distribution shifts. *arXiv preprint arXiv:2206.03669*, 2022.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Kaidi Xu, Huan Zhang, Shiqi Wang, Yihan Wang, Suman Jana, Xue Lin, and Cho-Jui Hsieh. Fast and complete: Enabling complete neural network verification with rapid and massively parallel incomplete verifiers. In *International Conference on Learning Representations*, 2020.

Jiancheng Yang, Rui Shi, and Bingbing Ni. Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis. In *IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pp. 191–195, 2021.

Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Russ R Salakhutdinov, and Kamalika Chaudhuri. A closer look at accuracy vs. robustness. *Advances in neural information processing systems*, 33:8588–8601, 2020.

Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, and Luca Daniel. Efficient neural network robustness certification with general activation functions. *Advances in neural information processing systems*, 31, 2018.

A Extension to the in-distribution analysis.

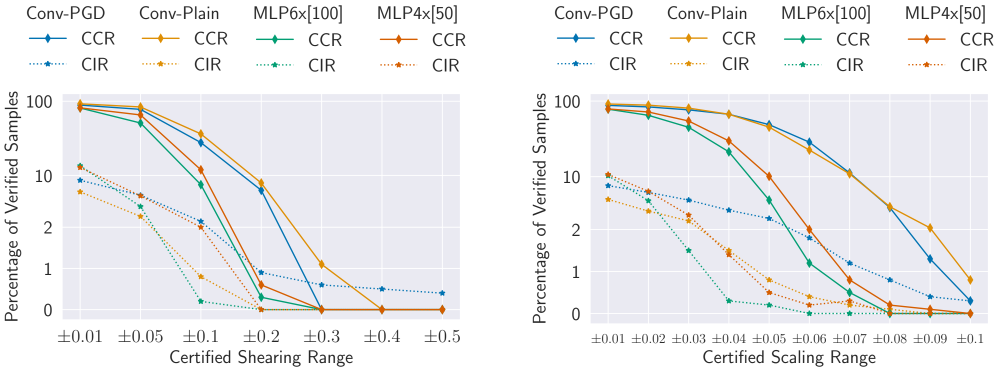
Table 5: Networks trained on the first ten classes of the GTSRB dataset. The accuracy is computed on 4800 test samples.

Network	Architecture	Activation	Training	ACC	# Neurons
MLP4x[50]	4 FC	ReLU	Plain	84.8	210
MLP6x[100]	6 FC	ReLU	Plain	85.4	610
Conv	2 Conv. & 2 FC	ReLU	Plain/PGD	92.7/90.8	4852

Architectures & Training We train a total of four neural networks on the first ten classes of the GTSRB dataset (Houben et al., 2013). Two Fully Connected (FC) Multi-layer Perceptron (MLP): MLP4x[50] and MLP6x[100] normally trained (plain), and two convolutional neural networks: one trained with PGD (Madry et al., 2017) attacks (ℓ_∞-norm attacks with $\epsilon = 0.01$ for a maximum of 40 steps), and the other normally trained, which are denoted as Conv-PGD and Conv-Plain, respectively. The clean accuracy (ACC) and other parameters are reported in Table 5. To achieve higher accuracy with such small networks we reduced the number of classes to 10 to decrease the amount of features to be learned by the network.

A.1 Geometric Robustness

Here, we report further geometric robustness certificates of the GTSRB test set. In Figure 6, we show results for shearing and scaling. In this experiment, we fed each network with clean test samples. As previously discussed, the ratio between CCR and CIR reflects the network's accuracy. Differently from rotations, shearing and scaling result to be less prone to be certified. The lower part of the graph highlights the erratic behavior of FPR, attributable to DeepG's sampling procedure.



(a) Shearing

(b) Scaling

Figure 6: Comparison of network architectures and training methods for shearing and scaling. The robustness certificates are computed on 1000 samples of the first 10 classes of the GTSRB test set.

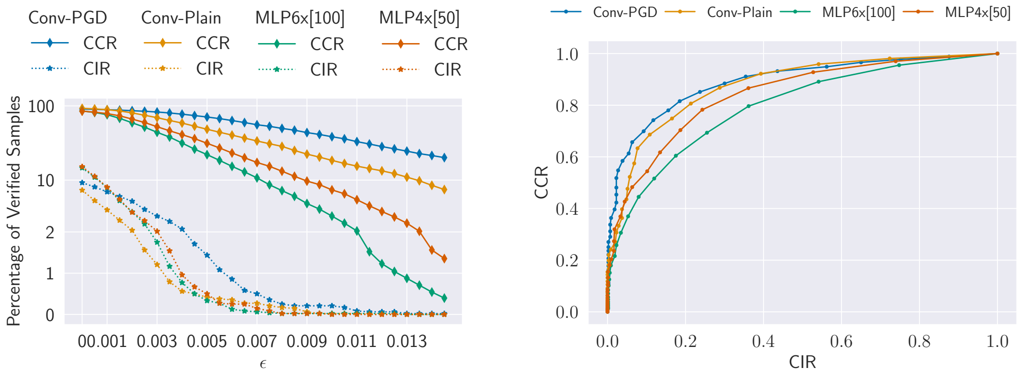
A.2 Norm-based Robustness

In Figure 7a, we show the percentage of verified samples for increasing ϵ of the ℓ_∞-norm, where ϵ is defined as adversarial perturbation budget. As noted, convolutional neural networks yield better results with respect to fully connected ones and Conv-PGD turns out to be the best. In addition, we see that for very small $\epsilon \sim 0.003$, the $\sim 1\%$ of CI is comparatively very small respect to the $\sim 75\%$ of CC for Conv-Plain. Instead, at $\sim 2\%$ of CI we have $\sim 50\%$ of CC for fully connected models.

In Figure 7b, we plot the ROC curves for TPR and FPR. We observe that the ROC of Conv-PGD remains mostly higher than that of Conv-Plain. We attribute this result to the fact that adversarially trained networks are more easily certifiable than simple models, which leads to generally higher TPR results. Similarly to the case of CCR & CIR curves, MLP6x[100] demonstrates to be less prone to be certified and results in lower AUC with respect to MLP4x[50].

B Proof of Theorem 4.1

Theorem 4.1 Let $\cup_{r=1}^{M} Q_r = \mathbb{R}^d$ and $f(x) = V^l x + a^l$ be the piecewise affine representation of the output of a ReLU network, then for almost any $x \in \mathbb{R}^d$, there exists $\alpha \in \mathbb{R}$ with $\alpha > 0$ and a



(a) CCR & CIR curves.

(b) ROC curves of TPR & FPR.

Figure 7: Comparison of network architectures and training methods for varying ϵ of ℓ_∞-norm based robustness certificates on 4800 samples of the first 10 classes of the GTSRB test set.

predicted class $k \in \mathcal{K}$ such that:

$$\min_{z,t} \{f_k(z) - f_t(z) \mid z \in \mathbb{S}(\alpha x),\ t \in \mathcal{K} \setminus \{k\}\} > 0,$$

holds for $\mathbb{S}(\alpha x) \subset Q_r$.

Proof. By Lemma 4.1, there exists a region Q_r, with $r \in \{1, \dots, R\}$ and $\beta > 0$ such that for all $\alpha \geq \beta$ we have $\alpha x \in Q_r$. Given that $\mathbb{S}(\alpha x) \subset Q_r$ and since $z \in \mathbb{S}(\alpha x)$ we have that $z \in Q_r$. Let $f(z) = V^r z + a^r$ be the affine form of the ReLU classifier f on Q_r. Let $k^* = \arg\max_k \langle v_k^r, z \rangle$, where v_k^r is the k-th row of V^r. Given the fact that V^r does not has identical rows, i.e. $v_l^r \neq v_m^r$ for $l \neq m$, the maximum is unique up to zero. If the maximum is unique, it holds for sufficiently large $\alpha \geq \beta$:

$$\langle v_{k^*}^r, z \rangle + a_{k^*}^r - \langle v_t^r, z \rangle - a_t^r > 0, \quad \forall t \in \mathcal{K} \setminus \{k^*\}.$$

□

C Extension to the out-of-distribution experimental analysis.

Datasets. In order to obtain a more comprehensive and fair evaluation, we consider two categories of image datasets: *grayscale* and *RGB*. In the grayscale category we place datasets with grayscale images of size 28x28: MNIST (LeCun, 1998), EMNIST (Cohen et al., 2017), KMNIST (i.e. Kuzushiji-MNIST Clanuwat et al. (2018)) and FMNIST (i.e., Fashion-MNIST Xiao et al. (2017)). In the RGB, we have RGB images of size 32x32: CIFAR10/100 (Krizhevsky et al., 2009), SVHN (Sermanet et al., 2012) and GTSRB (Houben et al., 2013). In addition, we utilize OrganAMNIST from MedMNIST (Yang et al., 2021) and ImageNet Cropped (C) (Deng et al., 2009) for training OOD aware models. For each category, we normalise all datasets by the same mean and standard deviation of the ID training set.

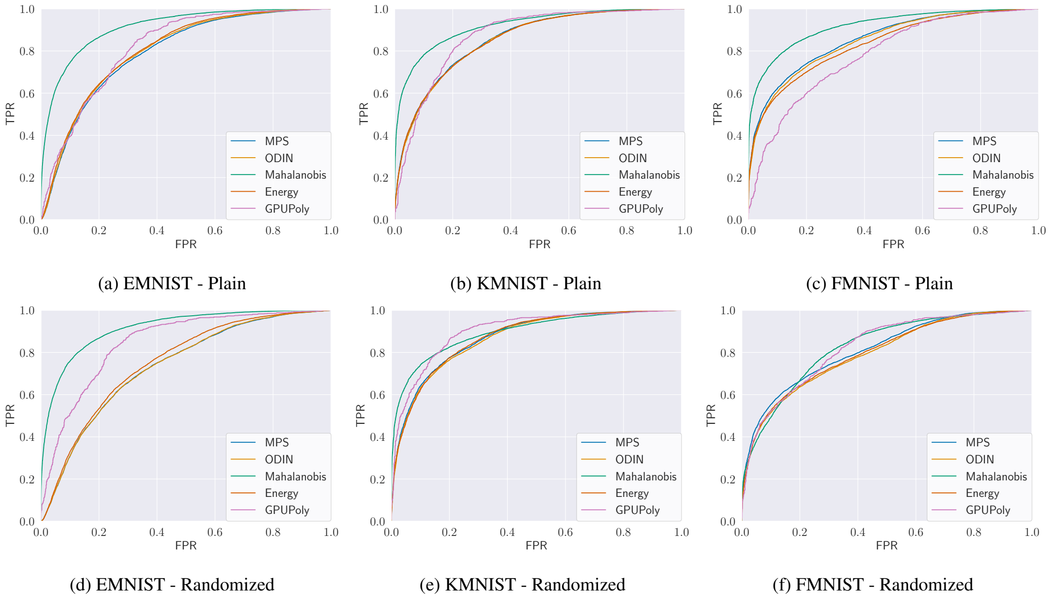
Architectures & Training. In Table 6, we describe the network architectures, activation type and number of neurons for each dataset category. Evaluation is carried out on different training procedures. Networks trained only with clean training data are called **Plain**. Adversarially trained networks are **PGD** (Madry et al., 2017) or **FGSM** (Goodfellow et al., 2015), where ϵ is the adversarial perturbation budget. **Randomized** are networks trained with randomized smoothing (Cohen et al., 2019) where σ is the standard deviation. Lastly, **OE** stands for Outlier Exposure (Hendrycks et al., 2019), where we insert the OOD training set in parentheses.

Table 6: Networks architectures for each dataset category.

Category	Input	Network	Architecture	Activation	# Neurons
Grayscale	28x28x1	MLP6x[200]	6 FC	ReLU	1 000
		ConvSmall	2 Conv. & 2 FC	ReLU	3 604
RGB	32x32x3	ConvSmall	2 Conv. & 2 FC	ReLU	4 852
		ConvMed	5 Conv. & 3 FC	ReLU	6 756

Analysis. Here, we extend our evaluation of robustness certificates for OOD detection. Below, we show ROC curves and other results for different trainings, networks and datasets. As previously mentioned, GPUPoly is used with ℓ_∞-norm robustness certificates and a range of 4 000 values of ϵ between 0 and 0.2. Instead, all other methods uses a range of $10e5$ for the threshold.

C.1 Grayscale Category Extension



(a) EMNIST - Plain (b) KMNIST - Plain (c) FMNIST - Plain

(d) EMNIST - Randomized (e) KMNIST - Randomized (f) FMNIST - Randomized

Figure 8: **ConvSmall - ID:MNIST** Comparison of ROC curves for OOD detection methods on KMNIST, EMNIST datasets. GPUPoly is used with ℓ_∞ robustness certificates and a range of 4000 values of ϵ between 0 and 0.2. All other methods uses a range of $10e5$ for the threshold.

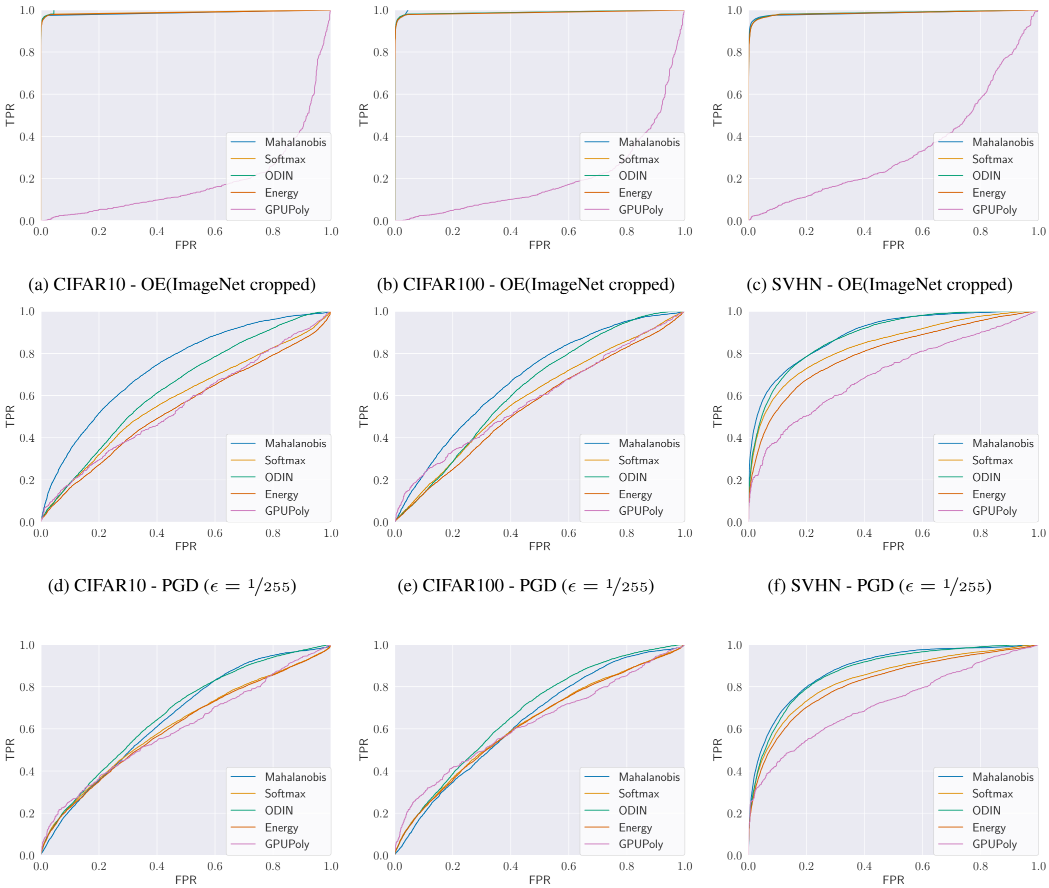
In Figure 8, we report the ROC curves for convolutional networks trained normally and with randomized smoothing on the MNIST dataset. We observe that GPUPoly(ℓ_∞) perform relatively better than standard OOD detection methods on EMNIST and KMNIST.

C.2 Colored Category Extension

In Figure 9, we show the ROC curves for the ConvMed model trained with OE (ImageNet cropped), PGD and randomized smoothing. In the context of OE, we clearly see that the ROC curves of GPUPoly are below the average value of 0.5. Besides his average performance on PGD and randomized trained networks, the results graphically demonstrate that OOD samples are more likely to be certified than ID samples for OOD aware networks.

Table 7: **Colored Datasets:** Comparison between standard OOD detection methods and robustness certificates of ℓ_∞-norm: GPUPoly(ℓ_∞) (Müller et al., 2021). We report the clean accuracy (ACC) on the in-distribution (ID) dataset: CIFAR10. In the context of GPUPoly, the AUC and FPR95 are computed by varying the adversarial power ϵ.

Network/ Training	ID: CIFAR10 ACC	Method	CIFAR100 AUC↑	CIFAR100 FPR95↓	GTSRB AUC↑	GTSRB FPR95↓	SVHN AUC↑	SVHN FPR95↓
ConvSmall/ Plain	58.0	Mahalanobis	54.9	93.2	**82.4**	**51.4**	**89.8**	**41.9**
		Softmax	63.0	**90.3**	74.1	75.6	75.4	85.8
		ODIN	63.3	**90.3**	75.6	68.6	82.4	70.0
		Energy	**64.5**	90.4	69.6	91.4	69.2	96.1
		GPUPoly(ℓ_∞)	58.3	91.3	62.1	91.4	65.8	91.2
ConvMed/ FGSM ($\epsilon = 1/255$)	57.3	Mahalanobis	57.7	92.4	**77.7**	**65.5**	**88.0**	**55.2**
		Softmax	64.8	89.7	70.1	80.9	69.9	90.2
		ODIN	**65.3**	**89.1**	68.0	86.2	77.2	78.5
		Energy	64.3	90.5	70.3	85.6	64.5	96.2
		GPUPoly(ℓ_∞)	58.8	95.1	64.4	92.2	66.0	93.3
ConvMed/ PGD ($\epsilon = 1/255$)	56.1	Mahalanobis	57.6	92.4	**80.4**	**57.4**	**91.5**	**41.3**
		Softmax	64.7	**89.8**	75.6	77.5	66.3	92.1
		ODIN	**64.9**	89.9	75.2	69.1	79.2	73.9
		Energy	64.1	90.8	70.8	89.9	59.0	98.1
		GPUPoly(ℓ_∞)	58.4	92.1	66.2	89.7	68.0	90.3



(a) CIFAR10 - OE(ImageNet cropped)

(b) CIFAR100 - OE(ImageNet cropped)

(c) SVHN - OE(ImageNet cropped)

(d) CIFAR10 - PGD ($\epsilon = 1/255$)

(e) CIFAR100 - PGD ($\epsilon = 1/255$)

(f) SVHN - PGD ($\epsilon = 1/255$)

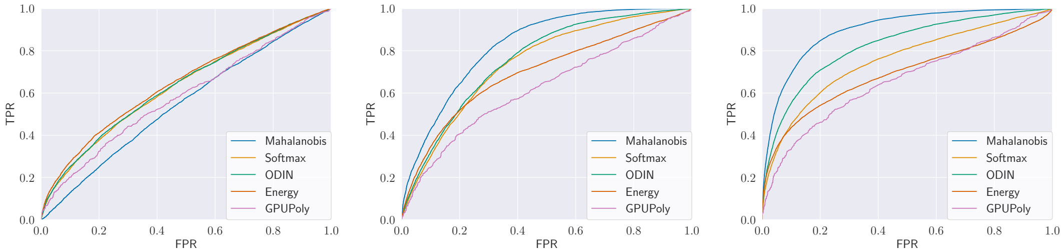
(g) CIFAR10 - Randomized ($\sigma = 0.1$)

(h) CIFAR100 - Randomized ($\sigma = 0.1$)

(i) SVHN - Randomized ($\sigma = 0.1$)

Figure 9: **ConvMed - ID: GTSRB.** Comparison of ROC curves for OOD detection methods on CIFAR10/100 and SVHN datasets. GPUPoly is used with ℓ_∞ robustness certificates and a range of $4\,000$ values of ϵ between zero and 0.2. All other methods uses a range of $10e5$ for the threshold.

In Table 7, we report the results for the RGB category with CIFAR10 as ID dataset. In this evaluation, we compare PGD, FGSM and normally trained convolutional networks. Similarly to the networks trained on the GTSRB dataset, we observe similar performances as standard OOD detection methods. As previously mentioned, the accuracy is generally low compared to state-of-the-art networks, but is aligned with related work on verification methods (Müller et al., 2021; 2022).



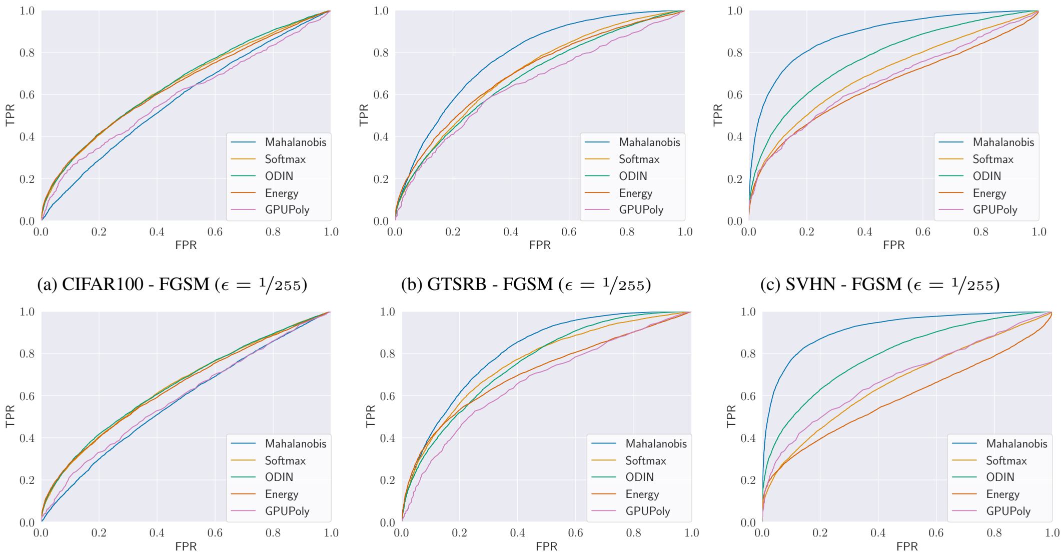
(a) CIFAR100 - Plain

(b) GTSRB - Plain

(c) SVHN - Plain

Figure 10: **ConvSmall - ID: CIFAR10.** Comparison of ROC curves for standard OOD detection methods and GPUPoly on CIFAR100, GTSRB and SVHN datasets.

In Figure 10 and Figure 11, we report the ROC curves for ConvSmall and ConvMed, respectively. All networks are trained with CIFAR10 as ID dataset.



(a) CIFAR100 - FGSM ($\epsilon = 1/255$)

(b) GTSRB - FGSM ($\epsilon = 1/255$)

(c) SVHN - FGSM ($\epsilon = 1/255$)

(d) CIFAR100 - PGD ($\epsilon = 1/255$)

(e) GTSRB - PGD ($\epsilon = 1/255$)

(f) SVHN - PGD ($\epsilon = 1/255$)

Figure 11: **ConvMed - ID: CIFAR10.** Comparison of ROC curves for standard OOD detection methods and GPUPoly on CIFAR100, GTSRB and SVHN datasets.