Tone the Highs Down: Out-of-distribution Detection With Rectified Activations

Anonymous Author(s)
Affiliation
Address
email

Abstract

Out-of-distribution (OOD) detection has received much attention lately due to its practical importance in enhancing the safe deployment of neural networks. One of the primary challenges is that models often produce very confident predictions on OOD data, which undermines the driving principle in OOD detection that the model should only be confident about in-distribution samples. In this work, we propose **ReAct**—a simple technique for reducing model overconfidence on OOD data. Our method is motivated by novel analysis on internal activations of neural networks, which displays highly distinctive signature patterns for most OOD distributions. We empirically demonstrate that ReAct achieves a new *state-of-the-art* detection performance on a comprehensive suite of benchmark datasets, and give theoretical explication for our method's efficacy. On the ImageNet benchmark, ReAct reduces the false positive rate (FPR95) by 25.05% compared to the previous best method.

1 Introduction

Neural networks deployed in real-world systems often encounter out-of-distribution (OOD) inputs—anomalous samples that the network has not been exposed to during training. Identifying and handling these OOD inputs can be paramount in safety-critical applications such as autonomous driving [7] and health care [1]. For example, an autonomous vehicle may fail to recognize objects on the road that do not appear in its detection model's training set, potentially leading to a crash. This can be prevented if the system identifies the unrecognized object as OOD and warns the driver in advance.

A driving idea behind OOD detection is that the model should be much more uncertain (*i.e.*, less confident) about samples outside of its training distribution. However, recent studies have shown that modern neural networks can, contrary to popular belief, produce overconfident predictions on OOD inputs as well [40]. This phenomenon renders the separation of in-distribution (ID) and OOD data a very daunting task. Indeed, much of the prior work on OOD detection focused on defining more suitable measures of uncertainty with limited success [15, 19, 31–33]. Consequently, it is arguable that continued progress in OOD detection requires insights into the fundamental cause and mitigation of model overconfidence on OOD data.

In this paper, we start by revealing an important observation that OOD data can trigger unit activation patterns that are significantly different from ID data. Figure 1(b) shows the distribution of activations in the penultimate layer of ResNet-50 trained on ImageNet [4]. Each point on the horizontal axis corresponds to a single unit. The mean and standard deviation are shown by the solid line and shaded area, respectively. The mean activation for ID data (blue) is well-behaved with a near-constant mean and standard deviation. In contrast, for OOD data (gray), the mean activation has significantly larger variations across units and is biased towards having sharp positive values (i.e., positively skewed). As a result, such high unit activation can undesirably manifest in model output, producing overconfident predictions on OOD data. A similar distributional property holds for other OOD datasets as well.

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

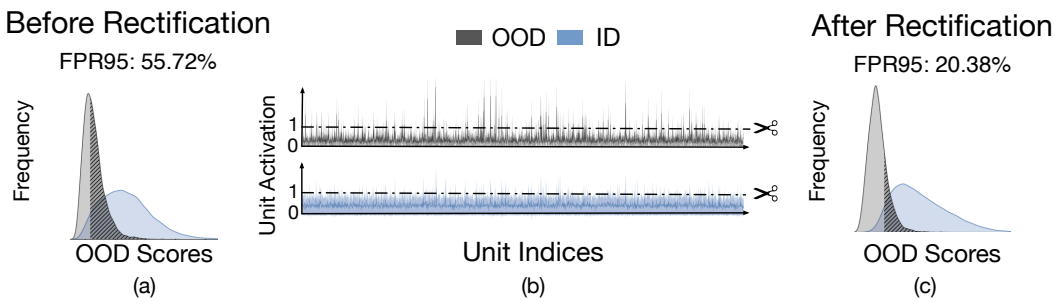

Figure 1: Plots showing (a) the distribution of ID (ImageNet [4]) and OOD (iNaturalist [18]) uncertainty scores before truncation, (b) the distribution of per-unit activations in the penultimate layer for ID and OOD data, and (c) the distribution of OOD uncertainty scores [33] after rectification. Applying ReAct drastically improves the separation of ID and OOD data. See text for details.

The above observation naturally inspires a simple yet surprisingly effective method **Re**ctified **Act**ivations (dubbed **ReAct**) for OOD detection. In particular, the outsized contribution of a few selected hidden units on OOD confidence can be attenuated by rectifying the activations at an upper limit $c > 0$. Conveniently, this can be done on a pre-trained model without any modification to training. The dashed horizontal line in Figure 1(b) shows the cutoff point c, and its effect on the OOD uncertainty score is shown in Figure 1(c). After rectification, the output distributions for ID and OOD data become much more well-separated and the FPR is significantly reduced from 55.72% to 20.38%. Importantly, this truncation largely preserves the activation for in-distribution data, and therefore ensures the classification accuracy on the original task is largely comparable.

We provide both empirical and theoretical insights, characterizing and explaining the mechanism by which ReAct improves OOD detection. We perform extensive evaluations and establish state-of-the-art performance on a suite of common OOD detection benchmarks, including CIFAR-10 and CIFAR-100, as well as a large-scale ImageNet dataset [4]. ReAct outperforms the best baseline by a large margin, reducing the average FPR95 by up to 25.05%. We further analyze our method theoretically and show that ReAct is most beneficial when OOD activations are more chaotic (*i.e.*, having a larger variance) and positively-skewed compared to ID activations, a behavior that is typical of many OOD datasets (*cf.* Figure 1). In summary, our **key results and contributions** are:

- We introduce ReAct —a simple yet effective approach for OOD detection utilizing activation truncation. We show that one can flexibly incorporate ReAct post hoc for any pre-trained neural classifier for improved OOD detection.
- We extensively evaluate ReAct on a suite of OOD detection tasks and establish a state-of-the-art performance. Compared to the previous best method, ReAct achieves an FPR95 reduction of 25.05% on a large-scale evaluation task with the ImageNet pre-trained model.
- We provide empirical ablation and theoretical analysis, revealing important insights that abnormally high activations on OOD data can harm the detection and how ReAct effectively mitigates the issue. We hope that our insights inspire future research to further examine the internal mechanisms of neural networks for OOD detection. Code and dataset will be released for reproducible research.

2 Background

The problem of *out-of-distribution detection* for classification is defined using the following setup. Let $\mathcal{X}$ (resp. $\mathcal{Y}$) be the input (resp. output) space and let $\mathcal{P}$ be a distribution over $\mathcal{X} \times \mathcal{Y}$, and let $f : \mathcal{X} \to \mathcal{Y}$ be a neural network trained on samples drawn from $\mathcal{P}$ to predict the label of an input sample. Denote by $\mathcal{D}_{\text{in}}$ the marginal distribution of $\mathcal{P}$ for $\mathcal{X}$, which represents the distribution of in-distribution data. At test time, the environment can incur an out-of-distribution $\mathcal{D}_{\text{out}}$ over $\mathcal{X}$. The goal of OOD detection is to define a decision function G for a given test input $\mathbf{x} \in \mathcal{X}$ such that:

$$G(\mathbf{x}; f) = \begin{cases} 0 & \text{if } \mathbf{x} \sim \mathcal{D}_{\text{out}}, \\ 1 & \text{if } \mathbf{x} \sim \mathcal{D}_{\text{in}}. \end{cases}$$

The difficulty of OOD detection greatly depends on the separation between $\mathcal{D}_{\text{in}}$ and $\mathcal{D}_{\text{out}}$. In the extreme case, if $\mathcal{D}_{\text{in}} = \mathcal{D}_{\text{out}}$ then the optimal decision function is a random coin flip. In practice, $\mathcal{D}_{\text{out}}$ is often defined by a distribution that simulates anomalies encountered during deployment time, such as samples from an irrelevant distribution whose label set has no intersection with $\mathcal{Y}$.

3 Method

We introduce a simple and surprisingly effective technique, **Re**ctified **Act**ivations (ReAct), for improving OOD detection performance. Our key idea is to perform *post hoc* modification to the unit activation, so to bring the overall activation pattern closer to the well-behaved case. Specifically, we consider a pre-trained neural network parameterized by θ, which encodes an input $\mathbf{x} \in \mathbb{R}^d$ to a feature space with dimension m. We denote by $h(\mathbf{x}) \in \mathbb{R}^m$ the feature vector from the penultimate layer of the network. A weight matrix $\mathbf{W} \in \mathbb{R}^{m \times K}$ connects the feature $h(\mathbf{x})$ to the output $f(\mathbf{x})$. K is the total number of classes in $\mathcal{Y} = \{1, 2, ..., K\}$.

ReAct: Rectified Activation We propose an `ReAct` operation, applied on the penultimate layer of a network:

$$\bar{h}(\mathbf{x}) = \texttt{ReAct}(h(\mathbf{x}); c), \tag{1}$$

where $\texttt{ReAct}(x; c) = \min(x, c)$ and is applied element-wise to the feature vector $h(\mathbf{x})$. In effect, this operation truncates activations above c to limit the effect of noise. The model output after *rectified activation* is given by:

$$f^{\text{ReAct}}(\mathbf{x}; \theta) = \mathbf{W}^\top \bar{h}(\mathbf{x}) + \mathbf{b}, \tag{2}$$

where $\mathbf{b} \in \mathbb{R}^K$ is the bias vector. A higher c indicates a larger threshold of activation truncation. When $c = \infty$, the output becomes equivalent to the original output $f(\mathbf{x}; \theta)$ without rectification, where $f(\mathbf{x}; \theta) = \mathbf{W}^\top h(\mathbf{x}) + \mathbf{b}$. Ideally, the rectification parameter of c should be adaptive based on the in-distribution data's activation pattern. A proper c should sufficiently preserve the activations for ID data while rectifying that of OOD data. In practice, we set c based on the p-th percentile of activations estimated on the ID data. For example, when $p = 90$, it indicates that 90% percent of the ID activations are less than the threshold c. We discuss the effect of percentile in detail in Section 4.

OOD Detection with Rectified Activation During inference time, our method ReAct can be flexibly leveraged by downstream OOD scoring function relying on $f^{\text{ReAct}}(\mathbf{x}; \theta)$:

$$G_\lambda(\mathbf{x}; f^{\text{ReAct}}) = \begin{cases} \text{in} & S(\mathbf{x}; f^{\text{ReAct}}) \geq \lambda \\ \text{out} & S(\mathbf{x}; f^{\text{ReAct}}) < \lambda \end{cases}, \tag{3}$$

where a thresholding mechanism is exercised to distinguish between ID and OOD during test time. To align with the convention, samples with higher scores $S(\mathbf{x}; f)$ are classified as ID and vice versa. The threshold λ is typically chosen so that a high fraction of ID data (*e.g.,* 95%) is correctly classified. ReAct can be compatible with several commonly used OOD scoring functions derived from the model output $f(\mathbf{x}; \theta)$, including the softmax confidence [15], ODIN score [32], Generalized ODIN score [19], and the energy score [33]. We provide an expansive description on these methods in Appendix B. In Section 4, we default to using the energy score (since it is hyperparameter-free and fine-tuning free), but demonstrate the benefit of using ReAct with other OOD scoring functions too.

4 Experiments

In this section, we evaluate our method on a suite of OOD detection tasks. In Section 4.1, we begin with the large-scale image classification networks trained on ImageNet [4]. We continue by looking at CIFAR benchmarks [27] that are routinely used in literature (Section 4.2).

4.1 Evaluation on Large-scale ImageNet Classification Networks

We first evaluate ReAct on a large-scale classification benchmark ImageNet [4]. Compared to the CIFAR benchmarks that are routinely used in literature, the ImageNet model is more challenging due to a larger label space ($K = 1,000$). Moreover, such large-scale evaluation is more relevant to real-world applications, where the deployed models often operate on images that have high resolution and contain more class than the CIFAR benchmarks.

Model	Methods	OOD Datasets								Average	
		iNaturalist		SUN		Places		Textures			
		FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
ResNet	MSP [15]	54.99	87.74	70.83	80.86	73.99	79.76	68.00	79.61	66.95	81.99
	ODIN [32]	47.66	89.66	60.15	84.59	67.89	81.78	50.23	85.62	56.48	85.41
	Mahalanobis [31]	97.00	52.65	98.50	42.41	98.40	41.79	55.80	85.01	87.43	55.47
	Energy [33]	55.72	89.95	59.26	85.89	64.92	82.86	53.72	85.99	58.41	86.17
	Generalized ODIN⋆ [19]	61.91	85.40	60.83	85.60	63.70	83.81	77.85	73.27	66.07	82.02
	ReAct (Ours)	**20.38**	**96.22**	**24.20**	**94.20**	**33.85**	**91.58**	**47.30**	**89.80**	**31.43**	**92.95**
MobileNet	MSP [15]	64.29	85.32	77.02	77.10	79.23	76.27	73.51	77.30	73.51	79.00
	ODIN [32]	55.39	87.62	54.07	85.88	57.36	84.71	49.96	85.03	54.20	85.81
	Mahalanobis [31]	62.11	81.00	47.82	86.33	52.09	83.63	92.38	33.06	63.60	71.01
	Energy [33]	59.50	88.91	62.65	84.50	69.37	81.19	58.05	85.03	62.39	84.91
	Generalized ODIN ⋆ [19]	63.24	83.26	**47.42**	**88.27**	52.14	85.62	59.82	79.53	55.66	84.17
	ReAct (Ours)	**42.40**	**91.53**	47.69	88.16	**51.56**	**86.64**	**38.42**	**91.53**	**45.02**	**89.47**

Table 1: **Main results.** Comparison with competitive *post hoc* out-of-distribution detection methods. All methods are based on a model trained on **ID data only** (ImageNet-1k), without using any auxiliary outlier data. ↑ indicates larger values are better and ↓ indicates smaller values are better. All values are percentages. ⋆ indicates model fine-tuning is needed.

Setup We consider models pre-trained on ImageNet-1k, using ResNet architecture [13] with depth 50. At test time, all images are resized to 224×224. We evaluate on four test OOD datasets from (subsets of) `Places365` [59], `Textures` [3], `iNaturalist` [18], and `SUN` [53] with non-overlapping categories w.r.t ImageNet. Our evaluations span a diverse range of domains including fine-grained images, scene images, and textural images (see Appendix C for details). We use a validation set of Gaussian noise. The optimal p is selected from $\{10, 65, 80, 85, 90, 95, 99\}$, which is 90 for all.

Comparison with competitive OOD detection methods. In Table 1, we compare ReAct with OOD detection methods that are competitive in the literature, where ReAct establishes the state-of-the-art performance. For a fair comparison, all methods use the pre-trained networks *post hoc*. We report performance for each OOD test dataset, as well as the average of the four. ReAct outperforms baselines, including Maximum Softmax Probability [15], ODIN [32], Mahalanobis distance [31], Generalized ODIN [19], and energy score [33]. Noticeably, ReAct reduces the FPR95 by **25.05**% compared to the best baseline [32] on ResNet. Note that Mahalanobis requires training a separate binary classifier, and displays limiting performance since the increased size of label space makes the class-conditional Gaussian density estimation less viable. In contrast, ReAct is much easier to use in practice, and can be implemented through simple *post hoc* activation rectification.

Rectification percentile	FPR95 ↓	AUROC ↑	AUPR ↑	ID ACC. ↑	Rectification threshold c
No ReAct [33]	58.41	86.17	96.88	75.08	∞
$p = 99$	44.57	90.45	97.96	75.12	2.25
$p = 95$	35.39	92.39	98.37	74.76	1.50
$p = 90$	31.43	92.95	98.50	73.75	1.00
$p = 85$	34.08	92.05	98.35	72.91	0.84
$p = 80$	41.51	89.54	97.91	71.93	0.72
$p = 65$	74.62	74.14	94.39	67.14	0.50
$p = 10$	74.70	57.55	86.06	1.22	0.06

Table 2: Effect of rectification threshold for inference. Model is trained on ImageNet using ResNet-50 [12]. All numbers are percentages and are averaged over 4 OOD test datasets.

Effect of rectification threshold. We now characterize the effect of the rectification parameter c, which can be modulated by the percentile p described in Section 3. For example, when $p = 90$, it indicates that 90% percent of the ID activations are smaller than the threshold c. In Table 2, we summarize the OOD detection performance, where we vary $p = \{10, 65, 80, 85, 90, 95, 99\}$. A larger p leads to a higher rectification threshold. This ablation confirms that over-activation does compromise the OOD detection ability, and ReAct can effectively alleviate the problem. Moreover, when p is sufficiently large, ReAct can improve OOD detection while maintaining comparable ID classification accuracy. Alternatively, once a sample is detected to be ID, one can always use the original activation $h(\mathbf{x})$, *which is guaranteed to given the identical classification accuracy*. When p is too small, the OOD performance starts to degrade as expected.

Effect on alternative neural network architecture. We show that ReAct is effective on a different architecture in Table 1. In particular, we consider a lightweight model MobileNet-v2 [45], which can be suitable for OOD detection in on-device mobile applications. Same as before, we apply ReAct on

Method	CIFAR-10			CIFAR-100			ImageNet		
	FPR95 ↓	AUROC ↑	AUPR ↑	FPR95 ↓	AUROC ↑	AUPR ↑	FPR95 ↓	AUROC ↑	AUPR ↑
Softmax score [15]	56.71	91.17	79.11	80.72	76.83	78.41	66.95	81.99	95.76
Softmax score + ReAct	53.81	91.70	92.11	75.45	80.40	84.28	58.28	87.06	97.22
ODIN [32]	31.10	93.79	94.95	66.21	82.88	86.25	56.48	85.41	96.61
ODIN+ReAct	28.81	94.04	94.82	59.91	85.23	87.53	44.10	90.70	98.04
Generalized ODIN [19]	32.74	93.78	95.14	74.27	77.95	81.84	66.07	82.02	95.97
Generalized ODIN+ReAct	24.54	93.76	94.27	44.84	89.68	91.17	43.15	89.70	97.55
Energy [33]	35.60	93.57	95.01	71.93	82.82	86.28	58.41	86.17	96.88
Energy+ReAct	32.91	94.27	95.53	59.61	87.48	89.63	31.43	92.95	98.50

Table 3: **Ablation results.** ReAct is compatible with different OOD scoring functions. For each ID dataset, we use the same model and compare the performance with and without ReAct respectively. ↑ indicates larger values are better and ↓ indicates smaller values are better. All values are percentages and are averaged over multiple OOD test datasets.

the output of the penultimate layer, with rectification threshold chosen based on the 90-th percentile. Our method reduces the FPR95 by **9.18**% compared to the best baseline [32].

What about applying ReAct on other layers? Our results suggest that applying ReAct on the penultimate layer is the most effective, since the activation signature patterns are most distinctive. To see this, we provide the activation patterns for intermediate layers in the Appendix (see Figure 5). Interestingly, early layers display less distinctive signatures between ID and OOD data. This is expected because neural networks generally capture lower-level features in early layers (such as Gabor filters [58] in layer 1), whose activations can be very similar between ID and OOD. The semantic-level features only emerge as with deeper layers, where ReAct is the most effective.

4.2 Evaluation on CIFAR Benchmarks

Datasets We use CIFAR-10 [27], and CIFAR-100 [27] datasets as in-distribution data. We use the standard split with 50,000 training images and 10,000 test images. We evaluate the model on six common OOD benchmark datasets: `Textures` [3], `SVHN` [39], `Places365` [59], `LSUN-Crop` [57], `LSUN-Resize` [57], and `iSUN` [55].

Experimental details. We use ResNet-18 architecture [12] and train on in-distribution datasets. The feature dimension of the penultimate layer is 512. For both CIFAR-10 and CIFAR-100, the models are trained for 100 epochs. The start learning rate is 0.1 and decays by a factor of 10 at epochs 50, 75, and 90. For rectification threshold c, we use the 90-th percentile of activations estimated on the ID data.

ReAct is compatible with various OOD scoring functions. We show in Table 3 that ReAct is a flexible method that is compatible with alternative scoring functions $S(\mathbf{x}; f^{\text{ReAct}})$. To see this, we consider commonly used scoring functions, and compare the performance both with and without using ReAct respectively. In particular, we consider softmax confidence [15], Generalized ODIN [19], ODIN score with calibrated softmax confidence [32] as well as energy score [33]—all of which derive OOD scores directly from the logit output $f(\mathbf{x})$. Note that Mahalanobis [31] estimates OOD score using feature representations instead of the model output $f(\mathbf{x})$, hence are less compatible with ReAct. On all three datasets, using ReAct consistently outperforms the counterpart without rectification.

ReAct outperforms generative-based methods. In Table 4, we also compare state-of-the-art methods using generative modeling [6, 9, 24, 46]. We follow the same evaluations as in prior literature. Using CIFAR-10 as ID data, ReAct outperforms the best method [46] by 6% on average. It is also important to note that generative models can be prohibitively challenging to train and optimize on

Method	SVHN	CIFAR-100	CelebA
Class-conditional Glow [24]	0.64	0.65	0.54
IGEBM [6]	0.43	0.54	0.69
JEM-softmax [9]	0.89	0.87	0.79
JEM-likelihood [9]	0.67	0.67	0.75
Input complexity [46]	0.95	0.74	0.86
ReAct (ours)	**0.96**	**0.90**	**0.87**

Table 4: Comparison with generative-based models for OOD detection on CIFAR-10 using the AUROC metric.

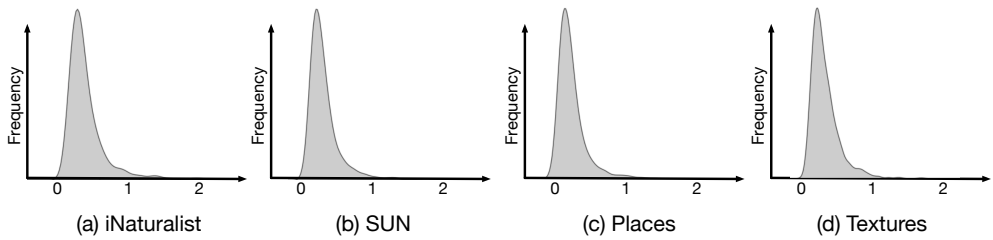

Figure 2: Positively skewed distribution of μ_i (mean of each unit) in the penultimate layer for four OOD datasets (iNaturalist [18], SUN [53], Places [59], Textures [3]). Model is trained on ImageNet.

large-scale datasets (and therefore are primarily evaluated on CIFAR-10 in literature). ReAct does not suffer from this issue and performs competitively on ImageNet-1k, as we have shown in Section 4.1.

5 Theoretical Analysis

To better understand the effect of ReAct, we mathematically model the ID and OOD activations as rectified Gaussian distributions and derive their respective distributions after applying ReAct. These modeling assumptions are based on the activation statistics observed on ImageNet in Figure 1. In the following analysis, we show that ReAct reduces mean OOD activations more than ID activations due to the fact that OOD activations are more chaotic and positively skewed. Derivations of the results are given in Appendix A.

ID activations. Let $h(\mathbf{x}) = (z_1, \dots, z_m) =: \mathbf{z}$ be the activations for the penultimate layer. We assume each $z_i \sim \mathcal{N}^R(\mu, \sigma_{\text{in}}^2)$ for some $\sigma_{\text{in}} > 0$. Here $\mathcal{N}^R(\mu, \sigma_{\text{in}}^2) = \max(0, \mathcal{N}(\mu, \sigma_{\text{in}}^2))$ denotes the rectified Gaussian distribution, which reflects the fact that activations after ReLU have no negative components. Before truncation with ReAct, the expectation of z_i is given by:

$$\mathbb{E}_{\text{in}}[z_i] = \left[1 - \Phi\left(\frac{-\mu}{\sigma_{\text{in}}}\right)\right] \cdot \mu + \phi\left(\frac{-\mu}{\sigma_{\text{in}}}\right) \cdot \sigma_{\text{in}},$$

where Φ and ϕ denote the cdf and pdf of the standard normal distribution, respectively. After rectification with ReAct, the expectation of $\bar{z}_i = \min(z_i, c)$ is:

$$\mathbb{E}_{\text{in}}[\bar{z}_i] = \left[\Phi\left(\frac{c-\mu}{\sigma_{\text{in}}}\right) - \Phi\left(\frac{-\mu}{\sigma_{\text{in}}}\right)\right] \cdot \mu + \left[1 - \Phi\left(\frac{c-\mu}{\sigma_{\text{in}}}\right)\right] \cdot c + \left[\phi\left(\frac{-\mu}{\sigma_{\text{in}}}\right) - \phi\left(\frac{c-\mu}{\sigma_{\text{in}}}\right)\right] \cdot \sigma_{\text{in}},$$

The reduction in activation after ReAct is:

$$\mathbb{E}_{\text{in}}[z_i - \bar{z}_i] = \phi\left(\frac{c-\mu}{\sigma_{\text{in}}}\right) \cdot \sigma_{\text{in}} - \left[1 - \Phi\left(\frac{c-\mu}{\sigma_{\text{in}}}\right)\right] \cdot (c - \mu) \tag{4}$$

OOD activations. We model OOD activations as being generated by a two-stage process: Each OOD distribution defines a set of μ_i's that represent the mode of the activation distribution for unit i, and the activations z_i given μ_i is represented by $z_i|\mu_i \sim \mathcal{N}^R(\mu_i, \tau^2)$ with $\tau > 0$. For instance, the dark gray line in Figure 1 shows the set μ_i's on the iNaturalist dataset, and the light gray area depicts the distribution of $z_i|\mu_i$. One commonality across different OOD datasets is that the distribution of μ_i is *positively-skewed*. Indeed, Figure 2 shows the empirical distribution of μ_i on an ImageNet pre-trained model for four OOD datasets, all of which display strong positive-skewness, *i.e.*, the right tail has a much higher density than the left tail.

Utilizing the positive-skewness property of μ_i, we analyze the distribution of z_i after marginalizing out μ_i, which corresponds to averaging across different μ_i's induced by various OOD distributions. Let $x_i|\mu_i \sim \mathcal{N}(\mu_i, \tau^2)$ so that $z_i|\mu_i = \max(x_i|\mu_i, 0)$. Since $x_i|\mu_i$ is symmetric and μ_i is positively-skewed, the marginal distribution of x_i is also positively-skewed[1], which we model with the epsilon-skew-normal (ESN) distribution [36]. Specifically, we assume that $x_i \sim \text{ESN}(\mu, \sigma_{\text{out}}^2, \epsilon)$, which has

[1]This can be argued rigorously using Pearson's mode skewness coefficient if the distribution of μ_i is unimodal.

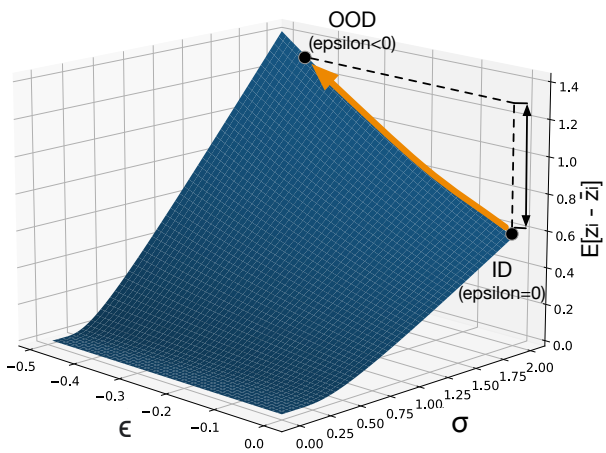

Figure 3: Plot showing the relationship between the skewness parameter ϵ and the chaotic-ness parameter σ on activation reduction after applying ReAct. The function $\mathbb{E}[z_i - \bar{z}_i]$ is increasing in both $-\epsilon$ and σ, which suggests that ReAct has a greater reduction effect for activation distributions with positive skewness ($\epsilon < 0$) and chaotic-ness—two signature characteristics of OOD activation.

the following density function:

$$q(x) = \begin{cases} \phi((x-\mu)/\sigma_{\text{out}}(1+\epsilon))/\sigma_{\text{out}} & \text{if } x < \mu, \\ \phi((x-\mu)/\sigma_{\text{out}}(1-\epsilon))/\sigma_{\text{out}} & \text{if } x \geq \mu. \end{cases} \tag{5}$$

with $\epsilon \in [-1, 1]$ controlling the skewness. In particular, the ESN distribution is positively-skewed when $\epsilon < 0$. It follows that $z_i = \max(x_i, 0)$, with expectation:

$$\mathbb{E}_{\text{out}}[z_i] = \mu - (1+\epsilon)\Phi\left(\frac{-\mu}{(1+\epsilon)\sigma_{\text{out}}}\right) \cdot \mu + (1+\epsilon)^2\phi\left(\frac{-\mu}{(1+\epsilon)\sigma_{\text{out}}}\right) \cdot \sigma_{\text{out}} - \frac{4\epsilon}{\sqrt{2\pi}} \cdot \sigma_{\text{out}}. \tag{6}$$

Expectation after applying ReAct becomes:

$$\begin{aligned} \mathbb{E}_{\text{out}}[\bar{z}_i] = \mu - (1+\epsilon)\Phi\left(\frac{-\mu}{(1+\epsilon)\sigma_{\text{out}}}\right) \cdot \mu + (1-\epsilon)\left[1 - \Phi\left(\frac{c-\mu}{(1-\epsilon)\sigma_{\text{out}}}\right)\right] \cdot (c-\mu) \\ + \left[(1+\epsilon)^2\phi\left(\frac{-\mu}{(1+\epsilon)\sigma_{\text{out}}}\right) - (1-\epsilon)^2\phi\left(\frac{c-\mu}{(1-\epsilon)\sigma_{\text{out}}}\right) - \frac{4\epsilon}{\sqrt{2\pi}}\right] \cdot \sigma_{\text{out}}, \end{aligned} \tag{7}$$

Hence:

$$\mathbb{E}_{\text{out}}[z_i - \bar{z}_i] = (1-\epsilon)^2\phi\left(\frac{c-\mu}{(1-\epsilon)\sigma_{\text{out}}}\right) \cdot \sigma_{\text{out}} - (1-\epsilon)\left[1 - \Phi\left(\frac{c-\mu}{(1-\epsilon)\sigma_{\text{out}}}\right)\right] \cdot (c-\mu), \tag{8}$$

which recovers Equation 4 when $\epsilon = 0$ and $\sigma_{\text{out}} = \sigma_{\text{in}}$.

Remark 1 (Activation reduction on OOD is more than ID) Figure 3 shows a plot of $\mathbb{E}[z_i - \bar{z}_i]$ for $\mu = 0.5$ and $c = 1$. Observe that decreasing ϵ (more positive-skewness) or increasing σ (more chaotic-ness) leads to a larger reduction in the mean activation after applying ReAct. For example, under the same σ, a larger $\mathbb{E}_{\text{out}}[z_i - \bar{z}_i] - \mathbb{E}_{\text{in}}[z_i - \bar{z}_i]$ can be observed by the gap of z-axis value between $\epsilon = 0$ and $\epsilon < 0$ (e.g., $\epsilon = -0.4$). This suggests that rectification on average affects OOD activations more severely compared to ID activations.

Remark 2 (Output reduction on OOD is more than ID) To derive the effect on the distribution of model output, consider output logits $f(\mathbf{z}) = W\mathbf{z} + \mathbf{b}$ and assume without loss of generality that $W\mathbf{1} > 0$ element-wise. This can be achieved by adding a positive constant to W without changing the output probabilities or classification decision. Let $\delta = \mathbb{E}_{\text{out}}[\mathbf{z} - \bar{\mathbf{z}}] - \mathbb{E}_{\text{in}}[\mathbf{z} - \bar{\mathbf{z}}] > 0$. Then:

$$\begin{aligned} \mathbb{E}_{\text{out}}[f(\mathbf{z}) - f(\bar{\mathbf{z}})] &= \mathbb{E}_{\text{out}}[W(\mathbf{z} - \bar{\mathbf{z}})] = W\mathbb{E}_{\text{out}}[\mathbf{z} - \bar{\mathbf{z}}] = W\left(\mathbb{E}_{\text{in}}[\mathbf{z} - \bar{\mathbf{z}}] + \delta\mathbf{1}\right) \\ &= \mathbb{E}_{\text{in}}[W(\mathbf{z} - \bar{\mathbf{z}})] + \delta W\mathbf{1} \\ &> \mathbb{E}_{\text{in}}[f(\mathbf{z}) - f(\bar{\mathbf{z}})]. \end{aligned}$$

Hence the increased separation between OOD and ID activations transfers to the output space as well. Note that the condition of $W\mathbf{1} > 0$ is sufficient but not necessary for this result to hold. In fact, our experiments in Section 4 do not require this condition. However, we verified empirically that ensuring $W\mathbf{1} > 0$ by adding a positive constant to W and applying ReAct does confer benefits to OOD detection, which validates our theoretical analysis.

6 Discussion and Further Analysis

Why does OOD data trigger abnormal unit activation patterns? So far we have shown that OOD data can trigger unit activation patterns that are significantly different from ID data, and that ReAct can effectively alleviate this issue (empirically in Section 4 and theoretically in Section 5). Yet a question left in mystery is *why* does such a pattern occur in modern neural networks? Answering this question requires carefully examining the internal mechanism by which the network is trained and evaluated. Here we provide one plausible explanation for the activation patterns observed in Figure 1, with the hope of shedding light for future research.

Intriguingly, our analysis reveals an important insight that batch normalization (BatchNorm) [22]—a common technique employed during model training—is in fact both a blessing (for ID classification) and a curse (for OOD detection). Specifically, for a unit activation denoted by z, the network estimates the running mean $\mathbb{E}_{\text{in}}(z)$ and variance $\text{Var}_{\text{in}}(z)$, over the entire ID training set during training. During inference time, the network applies BatchNorm statistics $\mathbb{E}_{\text{in}}(z)$ and $\text{Var}_{\text{in}}(z)$, which helps normalize the activations for the test data with the same distribution $\mathcal{D}_{\text{in}}$:

$$\text{BatchNorm}(z; \gamma, \beta, \epsilon) = \frac{z - \mathbb{E}_{\text{in}}[z]}{\sqrt{\text{Var}_{\text{in}}[z] + \epsilon}} \cdot \gamma + \beta \tag{9}$$

However, our key observation is that using *mismatched* BatchNorm statistics—that are estimated on $\mathcal{D}_{\text{in}}$ yet blindly applied to the OOD $\mathcal{D}_{\text{out}}$—can trigger abnormally high unit activations. As a thought experiment, we instead apply the *true* BatchNorm statistics estimated on a batch of OOD images, and surprisingly, we observe well-behaved activation patterns with near-constant mean and standard deviations—just like the ones observed on the ID data (see Figure 6 in Appendix). After applying the true statistics (estimated on OOD), the output distributions between ID and OOD data become much more separable. While this thought experiment has shed some guiding light, the solution of estimating BatchNorm statistics on a batch of OOD data is not at all satisfactory and realistic. Arguably, it poses a strong and impractical assumption of having access to a batch of OOD data during test time, as previously challenged in [47]. Despite its limitation, we view this method as an oracle, which serves as an *upper bound on performance* for our approach.

In particular, results in Table 5 suggest that our method favorably matches the oracle performance using the ground truth BN statistics. This is encouraging as our method does not impose any batch assumption and can be feasible for *single-input* testing scenarios.

Method	iNaturalist	Places	SUN	Textures
Oracle (batch OOD for estimating BN statistics)	99.59	99.09	98.32	91.43
ReAct (single OOD)	96.22	94.20	91.58	89.80
No ReAct [33]	89.95	85.89	82.86	85.99

Table 5: Comparison with oracle using OOD BN statistics. Model is trained on ImageNet (see Section 4.1). Values are AUROC.

What about networks trained with other normalization mechanisms? Going beyond batch normalization [22], we further investigate (1) whether networks trained with alternative normalization approaches exhibit similar activation patterns, and (2) whether ReAct is helpful there. To answer this question, we additionally evaluate networks trained with WeightNorm [44] and GroupNorm [52]—two other well-known normalization methods. As shown in Figure 4, the unit activations also display highly distinctive signature patterns between ID and OOD data, with more chaos on OOD data. In all cases, the networks are trained to adapt to the ID data, resulting in abnormal activation signatures on OOD data in testing. Unlike BatchNorm, there is no easy oracle (*e.g.*, re-estimating the statistics on OOD data) to counteract the ill-fated normalizations.

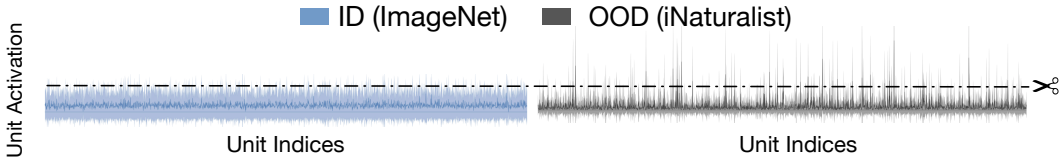

Figure 4: The distribution of per-unit activations in the penultimate layer for ID (ImageNet [4]) and OOD (iNaturalist [18]) on model trained with weight normalization [44].

	Methods	OOD Datasets								Average	
		iNaturalist		SUN		Places		Textures			
		FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
GroupNorm	w.o. ReAct	65.38	88.45	65.11	85.52	65.46	84.34	69.17	83.22	66.28	85.38
	w/ ReAct	**39.45**	**92.95**	**51.57**	**87.90**	**52.78**	**87.32**	**62.50**	**81.76**	**51.58**	**87.48**
WeightNorm	w.o. ReAct	40.71	92.52	48.07	89.39	50.92	87.87	61.65	80.71	50.34	87.62
	w/ ReAct	**19.73**	**95.91**	**31.39**	**93.21**	**42.34**	**88.94**	**13.74**	**96.98**	**26.80**	**93.76**

Table 6: **Effectiveness of ReAct for different normalization methods.** ReAct consistently improves OOD detection performance for model trained with GroupNorm and WeightNorm. In-distribution is ImageNet-1k dataset. **Bold** numbers are superior results.

We apply ReAct on models trained with WeightNorm and GroupNorm, and report results in Table 6. Our results suggest that ReAct is consistently effective under various normalization schemes. For example, ReAct reduces the average FPR95 by **23.54**% and **14.7**% respectively. Overall, ReAct has shown broad efficacy and compatibility with different OOD scoring functions (Section 4.2) as well as training methods.

7 Related Work

Out-of-distribution detection with discriminative models The softmax confidence score has become a common baseline for OOD detection [15]. Several works attempt to improve the OOD uncertainty estimation using deep ensembles [28], ODIN score [32], Mahalanobis distance-based confidence score [31], generalized ODIN score [19], and the energy score [33]. However, previous methods primarily derive OOD scores using original activations. In contrast, our work is motivated by novel analysis on internal activations, and shows that rectifying activations is a surprisingly effective approach for OOD detection. A separate line of methods uses an auxiliary OOD dataset for model regularization during training [10, 14, 16, 30, 34, 41, 50]. However, in many applications, it can be prohibitive to construct an auxiliary outlier dataset since the in-distribution data has a much larger coverage of concepts (e.g., ImageNet). Therefore, our method does not assume the availability of any auxiliary data and can be broadly used for any pre-trained model using standard cross-entropy loss.

Neural network activation analysis Neural networks have been studied at the granularity of the activation of individual layers [21, 35, 56, 60]. In particular, [37] showed that neural networks with ReLU activation can lead to arbitrary high activation for inputs far away from the training data. We show that using ReAct could efficiently alleviate this undesirable phenomenon. The idea of rectifying unit activation [26], which is known as Relu6, was used to facilitate the learning of sparse features. In this paper, we first show that rectifying activation can drastically reduce the overconfidence for OOD data, and as a result, improve OOD detection performance.

OOD detection with generative models Alternative approaches resort to generative models that directly estimate in-distribution density [5, 23, 43, 48, 49]. An input is considered as OOD if it lies in the low-likelihood regions. However, Nalisnick et al. [38] showed that deep generative models can assign a high likelihood to OOD data. Several methods were proposed to improve OOD detection using improved metrics [2], likelihood ratios [42, 46] and likelihood regret [54]. However, generative models can be prohibitively challenging to train and optimize [17], and the performance can often lag behind the discriminative counterpart [25, 51]. In contrast, our method relies on a discriminative classifier, which is much easier to optimize and achieves stronger performance.

8 Conclusion

This paper provides a simple activation rectification strategy termed ReAct, which truncates the high activations during test time for OOD detection. We provide both empirical and theoretical insights characterizing and explaining the mechanism by which ReAct improves OOD uncertainty estimation. By rectifying the activations, the outsized contribution of hidden units on OOD output can be attenuated, resulting in a stronger separability from ID data. Extensive experiments show ReAct can significantly improve the performance of OOD detection on both common benchmarks and large-scale image classification models. We hope that our insights inspire future research to further examine the internal mechanisms of neural networks for OOD detection.

References

[1] Tianshi Cao, Chinwei Huang, David Yu-Tung Hui, and Joseph Paul Cohen. A benchmark of medical out of distribution detection. *arXiv preprint arXiv:2007.04250*, 2020.

[2] Hyunsun Choi and Eric Jang. Generative ensembles for robust anomaly detection. 2018.

[3] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613, 2014.

[4] 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*, pages 248–255. Ieee, 2009.

[5] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[6] Yilun Du and Igor Mordatch. Implicit generation and generalization in energy-based models. *arXiv preprint arXiv:1903.08689*, 2019.

[7] Angelos Filos, Panagiotis Tigkas, Rowan McAllister, Nicholas Rhinehart, Sergey Levine, and Yarin Gal. Can autonomous vehicles identify, recover from, and adapt to distribution shifts? In *International Conference on Machine Learning*, pages 3145–3153. PMLR, 2020.

[8] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[9] Will Grathwohl, Kuan-Chieh Wang, Joern-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. In *International Conference on Learning Representations*, 2019.

[10] T. Guenais, D. Vamvourellis, Y. Yacoby, F. Doshi-Velez, and W. Pan. Bacoun: Bayesian classifers with out-of-distribution uncertainty. *ICML Workshop on Uncertainty in Deep Learning*, 1:1–24, 2020.

[11] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016.

[14] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

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

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

[17] Tobias Hinz, Stefan Heinrich, and Stefan Wermter. Generating multiple objects at spatially distinct locations. *arXiv preprint arXiv:1901.00686*, 2019.

[18] Grant Horn, Oisin Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. *The iNaturalist Species Classification and Detection Dataset*. 06 2018.

[19] Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10951–10960, 2020.

[20] Rui Huang and Yixuan Li. Towards scaling out-of-distribution detection for large semantic space. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[21] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 448–456, Lille, France, 07–09 Jul 2015. PMLR.

[22] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[23] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[24] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems*, pages 10215–10224, 2018.

[25] Polina Kirichenko, Pavel Izmailov, and Andrew G Wilson. Why normalizing flows fail to detect out-of-distribution data. *Advances in Neural Information Processing Systems*, 33, 2020.

[26] Alex Krizhevsky. Convolutional deep belief networks on cifar-10, 2010.

[27] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[28] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in neural information processing systems*, pages 6402–6413, 2017.

[29] Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

[30] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=ryiAv2xAZ.

[31] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, pages 7167–7177, 2018.

[32] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *6th International Conference on Learning Representations, ICLR 2018*, 2018.

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

[34] 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.

[35] Ari S. Morcos, David G.T. Barrett, Neil C. Rabinowitz, and Matthew Botvinick. On the importance of single directions for generalization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=r1iuQjxCZ.

[36] Govind S Mudholkar and Alan D Hutson. The epsilon–skew–normal distribution for analyzing near-normal data. *Journal of statistical planning and inference*, 83(2):291–309, 2000.

[37] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, pages 807–814. Omnipress, 2010.

[38] 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.

[39] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[40] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 427–436, 2015.

[41] Aristotelis-Angelos Papadopoulos, Nazim Shaikh, Jiamian Wang, and Mohammad Reza Rajati. Simultaneous classification and out-of-distribution detection using deep neural networks, 2020. URL https://openreview.net/forum?id=Hyez1CVYvr.

[42] 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 *Advances in Neural Information Processing Systems*, pages 14680–14691, 2019.

[43] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

[44] Tim Salimans and Diederik P Kingma. Weight normalization: a simple reparameterization to accelerate training of deep neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 901–909, 2016.

[45] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[46] Joan Serrà, David Álvarez, Vicenç Gómez, Olga Slizovskaia, José F. Núñez, and Jordi Luque. Input complexity and out-of-distribution detection with likelihood-based generative models. In *International Conference on Learning Representations*, 2020.

[47] Jiaming Song, Yang Song, and Stefano Ermon. Unsupervised out-of-distribution detection with batch normalization. *arXiv preprint arXiv:1910.09115*, 2019.

[48] Esteban G Tabak and Cristina V Turner. A family of nonparametric density estimation algorithms. *Communications on Pure and Applied Mathematics*, 66(2):145–164, 2013.

[49] Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. In *Advances in neural information processing systems*, pages 4790–4798, 2016.

[50] Sachin Vernekar, Ashish Gaurav, Vahdat Abdelzad, Taylor Denouden, Rick Salay, and Krzysztof Czarnecki. Out-of-distribution detection in classifiers via generation. *CoRR*, abs/1910.04241, 2019. URL http://arxiv.org/abs/1910.04241.

[51] Ziyu Wang, Bin Dai, David Wipf, and Jun Zhu. Further analysis of outlier detection with deep generative models. *Advances in Neural Information Processing Systems*, 33, 2020.

[52] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[53] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, pages 3485–3492. IEEE Computer Society, 2010.

[54] Zhisheng Xiao, Qing Yan, and Yali Amit. Likelihood regret: An out-of-distribution detection score for variational auto-encoder. *Advances in Neural Information Processing Systems*, 33, 2020.

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

[56] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/375c71349b295fbe2dcdca9206f20a06-Paper.pdf.

[57] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

[58] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[59] 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*, 40(6):1452–1464, 2017.

[60] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Revisiting the importance of individual units in cnns via ablation. *CoRR*, abs/1806.02891, 2018. URL http://arxiv.org/abs/1806.02891.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [N/A]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 5.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Section A in Supplementary.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A] Code and data will be released publicly to facilitate reproducible research.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A] All methods use the same pre-trained model and test datasets, so the evaluation process doesn't involve randomness.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix D.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4.
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]