Improving Normalizing Flows with the Approximate Mass for Out-of-Distribution Detection

Samy Chali
Université Paris-Saclay, CEA, List, F-91120
Palaiseau, France
samy.chali@cea.fr

Inna Kucher
Université Paris-Saclay, CEA, List, F-91120
Palaiseau, France
inna.kucher@cea.fr

Marc Duranton
Université Paris-Saclay, CEA, List, F-91120
Palaiseau, France
marc.duranton@cea.fr

Jacques-Olivier Klein
Centre de Nanosciences et de Nanotechnologies, CNRS,
Universite Paris-Saclay
91120 Palaiseau, France
jacques-olivier.klein@universite-paris-saclay.fr

Abstract

Normalizing flows are generative models that show poor performance on out-of-distribution (OOD) detection tasks with a likelihood-based test. In this study we focus on the "approximate mass" metric. We show that while it improves OOD detection performance, it has limitations under a maximum likelihood training. To solve this limitation we modify the training objective by incorporating the approximate mass. It smooths the learnt distribution in the vicinity of training in-distribution data. We measure an average of 97.6% AUROC in our experiments on different benchmarks, showing an improvement of 16% with respect to the best baseline we tested against.

1. Introduction

Out-of-distribution (OOD) detection [1] is a binary classification problem where a model assesses whether a data point belongs to a given data distribution (the "in-distribution") or not (if it falls in an "out-distribution"). This problem occurs when deploying machine learning models as they would face unexpected data and should handle OOD inputs properly. Therefore, such systems would greatly benefit from assessing the quality and relevance of the input data beforehand. In this case, generative models are especially interesting because some of them can estimate the likelihood of input data. Normalizing flows (or invertible neural networks) [2] in particular show unique characteristics: not only they can generate samples following the target data distribution but they can also evaluate the exact log-likelihood of data unlike GANs [3] or VAEs [4].

It was shown by previous studies [5–8] that normalizing flows perform poorly on out-of-distribution detection with a likelihood metric: they assign higher likelihood on certain out-distributions than to their training data [5]. Although current explanations of this phenomenon either involve mismatch in entropy between the ID and OOD distributions [8] or overfitting [7], the "typical set hypothesis" (TSH) [5] interpretation was one of the first proposed mechanism to explain the OOD detection problem. The authors of [9] suggested that input data are most likely coming from the typical set which is disjoint from the set of most likely data, thereby making the likelihood irrelevant to OOD detection. Instead, they replace it with a test statistics based on entropy to assess whether a point belongs to the typical set or not. This mechanism was later interpreted in [10] as the model not spreading out the mass of the learnt probability distribution efficiently across the input space, creating abrupt variations around OOD samples. This study gave rise to the approximate mass [10], the metric we focus on in this study. This metric is argued to be more efficient in OOD detection than likelihood, the OOD values having higher approximate mass than ID values in normal conditions. However, the link with the typical set is not clear.

This paper presents several contributions:

- an empirical test of the hypothesis that OOD data are assigned higher approximate mass than ID data on image data as well as an observation that the maximum likelihood objective overfits and reverses the behavior of the approximate mass,
- a modification of the training objective by adding a penalization term proportional to the approximate mass,
- improvements with respect to the state-of-the-art on OOD detection and an exploration of the limits of this metric by assessing the nature of the distribution shifts the approximate mass detects.
- a methodological contribution by correcting how the ROC curve-based metrics are usually reported in the previous works, by balancing the ID and OOD classes. This treatment, as far as we are aware of, is rarely applied in most papers of the OOD detection field.

2. Related works

Normalizing flows learn an invertible transformation T that maps input data x to and from a latent space parameterized by a known distribution. This transformation yields a latent representation that can easily be manipulated [4]. Normalizing flows compute the target distribution by applying the change of variable formula:

$$\log(p_X(x)) = \log(p_U(T(x))) + \log(|det(\frac{\partial T}{\partial x})|)$$

The latent space is parameterized by a fixed tractable distribution, such as a normal distribution, chosen to be easily evaluated and to sample data easily. The transformation T is designed in a way that the determinant of its Jacobian is tractable which is achieved with coupling layers and other invertible transformations as in the RealNVP [11], Glow [12] and NICE [13] models.

The problem of OOD detection can be formulated in different ways, as stated in [1] in a general OOD detection framework. It is defined as a binary detection problem where the aim is to detect shifts in underlying data distribution: covariate shift (shift on the feature distribution $p(x)$) or semantic shift (shift on the label distribution $p(y)$) or a combination of both. In this paper we address both problems: OOD detection (covariate and semantic shifts) and anomaly detection (semantic shift). For OOD detection the training set is drawn from the in-distribution while the OOD distribution is any distribution with different features and underlying semantic. For anomaly detection the model learns one class of a dataset (ID distribution) while the other classes of the dataset make up the OOD distribution. It is more common in the literature to focus on the anomaly detection and OOD detection tasks [1] while the pure covariate shift detection is mostly studied within the adversarial attacks context [14].

If the likelihood is used as a metric to detect OOD samples, it performs poorly when applied to normalizing flows [5–8]. The ”typical set hypothesis” (TSH) [9] attempts to give an explanation to this phenomenon. It assumes that ID data does not originate from sets of high likelihood in the input space but rather comes from the typical set, which is the set of sequences, which probability distribution is on average close to the entropy of the data source. The authors of [9] therefore propose to build a new OOD detection test based on typicality instead of likelihood. This idea was further developed in [10] where the typicality test is replaced by a metric called ”approximate mass” which corresponds to the norm of the gradient of the log-likelihood with respect to input data: $\parallel \frac{\partial \mathcal{L}(x;\theta)}{\partial x} \parallel$, where x is the input, $\mathcal{L}$ is the log-likelihood evaluated by the model with parameters θ and $||.||$ is the euclidean norm. The authors interpret the TSH as meaning that normalizing flows map ID data to regions of high mass while OOD data is mapped to lower mass regions. According to the authors, this metric shows better results than the log-likelihood in OOD detection. It is expected to have smaller values for ID data than to OOD data.

3. Model improvement

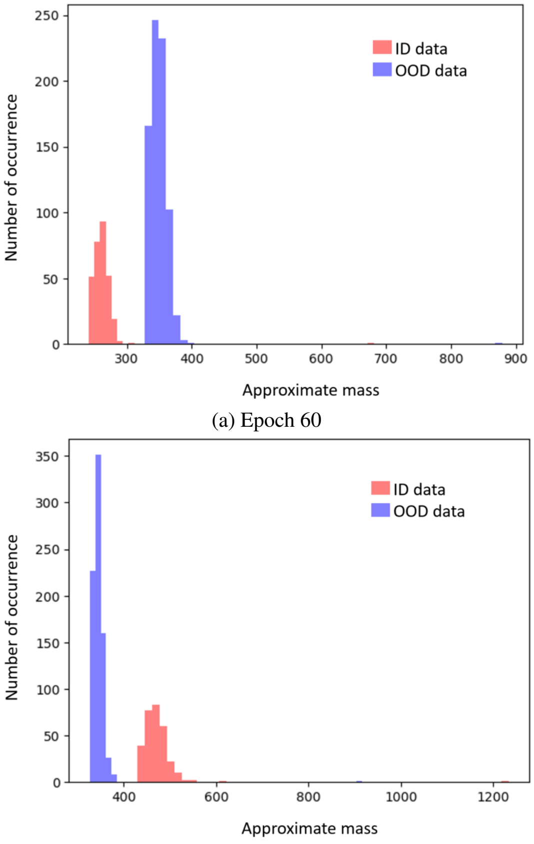
3.1. Observations about normalizing flow training

For most of the models (as well as for our model), no OOD data is available during the training, thus there is no way of controlling how the approximate mass would behave on ID data relatively to OOD inputs. The model might overfit in a way that would not be visible on the likelihood but variations around input points may increase. This sensitivity to input variations was already discussed in [15] where authors first suggested the idea of double backpropagation.

In order to check the evolution of the approximate mass during training, we train the RealNVP model [11] following the same architecture and training settings as in [6] with CIFAR-10 [16] as ID and with SVHN [17] as OOD. At the end of each epoch, we check the approximate mass for both datasets.

We observe in Figure 1 that at first the approximate mass behaves as expected as the model assigns a higher score to OOD data than to ID data. However, after several epochs, the trend reverses and the model ends up assigning higher scores to ID data and lower scores to OOD data, thus making the approximate mass an unreliable metric.

To understand this phenomenon, we break down the approximate mass in the following way:



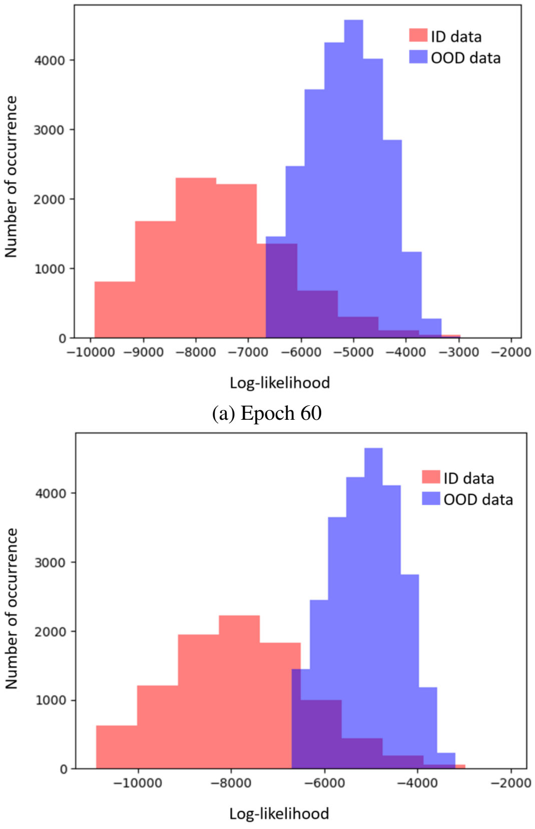
(a) Epoch 60

(b) Epoch 100

Figure 1. Evolution of the approximate mass with the vanilla RealNVP model trained on CIFAR-10 (red) and tested on SVHN (blue).

$$\| \frac{\partial \mathcal{L}(x;\theta)}{\partial x} \| = \frac{1}{p(x;\theta)} \| \frac{\partial p(x;\theta)}{\partial x} \| \tag{1}$$

knowing that $p(x;\theta)$ is a density function therefore $|p(x;\theta)| = p(x;\theta)$. This expression shows that the approximate mass translates the relative changes in the values of the probability density around input x. The maximum likelihood training objective of normalizing flows naturally minimizes the term $\frac{1}{p(x;\theta)}$. However, one may wonder how the approximate mass of in-distribution data evolves during training. Our observations in Figure 1 suggest that this mass of density (the term $\| \frac{\partial p(x;\theta)}{\partial x} \|$) might increase quicker than the term $p(x;\theta)$ during training. Intuitively, this means the model is overfitting: during training, the gap between likelihood values assigned by the model to the training inputs and neighboring points increases too much, resulting in a higher-magnitude derivative. This means that the model doesn't generalize well on the ID distribution.



(a) Epoch 60

(b) Epoch 100

Figure 2. Evolution of the log-likelihood distribution with the vanilla RealNVP model trained on CIFAR-10 (red) and tested on SVHN (blue).

3.2. Penalizing the gradient of the log-likelihood

To improve the metric and fix this silent overfitting issue, we add the approximate mass as a term in the loss function.

Denoting $\mathcal{L}(x;\theta) = \log(p(x;\theta))$ the average log-likelihood the model parameterized by θ on a batch of input data x, we write our new training objective as follows:

$$\min_{\theta} \; -\mathcal{L}(x;\theta) + \alpha \| \frac{\partial \mathcal{L}(x;\theta)}{\partial x} \| \tag{2}$$

where $\alpha > 0$ is a hyperparameter which represents the trade-off between locally increasing the likelihood and decreasing the gradient.

We apply our approach to the image data, as is usually done in OOD detection. We proceed as follows:

1. compute the gradient of the model log-likelihood with respect to the input data,
2. flatten the gradient then compute its norm for each batch of data,
3. average the norms with respect to the batches of data.

Adding this term to the training objective can be compared to other OOD detection methods such as ODIN [18] and Generalized ODIN [19] which smooth the softmax output distribution of a neural network with temperature scaling and add noise to the input. This additional noise is chosen in an adversarial direction [14] to increase the softmax score for any input. The authors argue this perturbation has a stronger effect on in-distribution than out-of-distribution samples, thus making them more separable.

In our approach, the addition of the approximate mass allows the smoothing of the output distribution to be isotropic instead of favoring one specific direction (the direction of optimal uncertainty given by adversarial attacks). This interpretation can be put in parallel with the VAT method [20] where, similarly, the KL divergence between the learned perturbed distribution and the true distribution in a semi-supervised training framework. Another way of interpreting the approximate mass penalization would be by seeing it as the variation of the likelihood around any given data input, similarly to the score function in the Fisher information metric. This score function with respect to the input gives information about how much the log-likelihood changes around the input data. Ideally, as stated above, this variation shouldn't be too high as that would be a sign of overfitting. It would also mean that the model gives too much importance to this specific data sample. Penalizing the model with the approximate mass is expected to make it covariate-shift invariant.

Some concerns can be raised about the complexity of the method as it requires two backpropagation computations per batch during training instead of only one. Further analysis were conducted in [21] regarding complexity and optimization of the double backpropagation family of penalization methods (up to a third of improvement in computation). We chose to focus our work on the results of the application of our newly introduced training objective by relying on the automatic gradient computation methods given by common deep learning libraries.

4. Experiments

4.1. Experimental set-up

We conduct several experiments using RealNVP architecture following the setting in [6]. The models are trained with an Adam optimizer for 200 epochs. When describing models, we refer to the number of blocks and scales following the nomenclature in [6]. A scale, in a multi-scale RealNVP architecture, refers to a block composed of three coupling layers with checkerboard masking, a squeeze operation then three coupling layers with channel-wise masking. A block is a residual block composing the ResNet in the underlying st-network in each coupling layer. Finally, in our experiments we studied the results dependence on the value of the hyperparameter α as introduced in equation 2. We found that $\alpha = 2$ yields the best results. We noticed that there is a range of α where the model behaves reasonably. This range becomes smaller as the model's size decreases for a fixed number of input dimensions.

4.2. Ablation study: removal of the penalization

In order to test out the effect of our penalization on the training of normalizing flows, we perform an ablation study where we compare two modes of training for the RealNVP model, the vanilla training (maximum likelihood) and the penalized training (our loss function as defined in 2). We then compare the results in OOD detection with the approximate mass of both models to assess the impact of approximate mass penalization on OOD detection performance.

Metrics: We report the AUROC on an OOD detection task for all our models.

Datasets: We use classic datasets widely used in the OOD detection setting, namely CIFAR-10 [16] and FashionMNIST [22] as ID datasets and SVHN [17] and MNIST [23] as their respective OOD datasets.

Models: We train RealNVP models on the ID datasets. For grayscale images (FashionMNIST and MNIST), we use a RealNVP model with 4 blocks and 2 scales (≈10 million parameters) and for color pictures (CIFAR-10 and SVHN) we train a RealNVP model with 6 blocks and 3 scales (≈60 million parameters).

Discussion: In table 1 we report the AUROC values for both training modes, on both ID/OOD dataset pairs introduced above. The results show that the observed overfitting in section 3 impacts the OOD detection results. Indeed, the penalized models systematically performs better than their vanilla counterpart. Moreover, we observe that the values of approximate mass of the vanilla models are out of order between ID and OOD data (higher approximate mass for ID than OOD data in both cases). However, the consequences are far more visible for the model trained on CIFAR-10 as the model is almost always perfectly wrong in its classification and systematically ranks ID data higher than OOD data with the approximate mass . This inconsistent behavior observed between the vanilla models trained FashionMNIST and CIFAR-10 shows that the approximate mass is unreliable when not trained with our penalization. Furthermore, it shows that the addition of our penalization effectively changes the behavior of our model as it makes the approximate mass more consistent and systematically perform better than the maximum likelihood objective alone.

4.3. Out-of-distribution detection

Approximate mass illustration: To begin the study of OOD detection, we show both metrics for penalized Real-

Model	Training dataset (ID)	OOD dataset	AUROC
Vanilla	FashionMNIST	MNIST	0.977
Approximate mass	FashionMNIST	MNIST	0.994
Vanilla	CIFAR-10	SVHN	0.0008
Approximate mass	CIFAR-10	SVHN	0.969

Table 1. Results on OOD classification for a vanilla model and a penalized model.

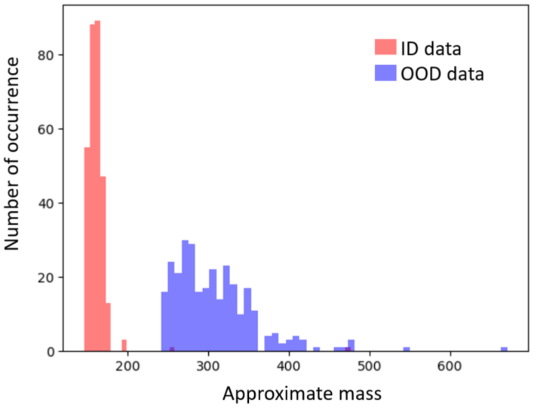


Figure 3. Approximate mass distribution of a penalized RealNVP model trained on FashionMNIST (in-distribution, red) and tested with MNIST (out-distribution, blue). Approximate mass values on the x-axis, number of occurrences on the y-axis.

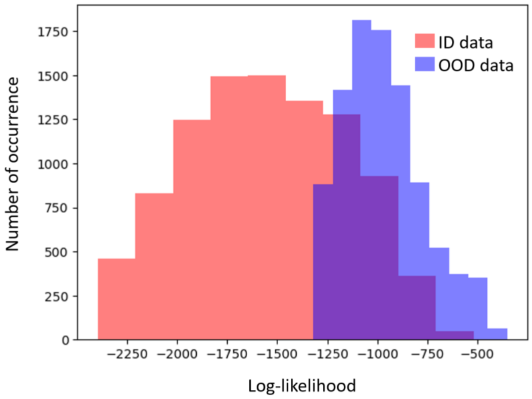


Figure 4. Log-likelihood distribution of a penalized RealNVP model trained on FashionMNIST (in-distribution, red) and tested with MNIST (out-distribution, blue). Log-likelihood values on the x-axis, number of occurrences on the y-axis.

NVP model: approximate mass and log-likelihood in Figure 3 and 4 respectively. The approximate mass shows more clear separation, even visually, than log-likelihood. At the same time it reflects the expected ordering of OOD and ID data, with OOD data having greater values than ID one.

Metrics: To assess the performance of our model on OOD detection, we report common metrics: AUROC (area under the ROC curve), AUPR (area under the precision-recall curve) and TNR at 95% TPR (True Negative Rate at a fixed level of 95% True Positive Rate).

Baselines: To compare the results we chose state-of-the-art methods that are commonly used in OOD detection benchmarks: Generalized ODIN [19], an extension of the original ODIN [18], Mahalanobis [24] and Energy-based OOD detection [25]. The ODIN family of methods rely on adversarial perturbations and smoothing the output distribution, which we will put in parallel to our approach in the following section. The Mahalanobis and energy-based methods use the feature-space to assess whether data samples are ID or OOD. The former uses Gaussian discriminant analysis while the latter uses an energy-based interpretation (different from Joint Energy-based Model [10]). The energy-based model also requires the use of OOD data to train its upper energy bound. We therefore train the energy-based model with both CIFAR-10/FashionMNIST as ID data and SVHN/MNIST as OOD data.

Datasets: We test our approach in a classic OOD detection setting with the CIFAR-10 [16] and FashionMNIST [22] datasets as in-distribution sets. For the models trained on FashionMNIST, we use the MNIST [23] KMINST [26] and EMNIST [27] datasets as OOD distributions while the models trained on CIFAR-10 are tested against the SVHN [17], DTD [28], GTSRB [29], Places365 [30] and iNaturalist [31] (split into ”Animalia” and ”Plantae” parts) datasets as out-of-distribution sets. We chose these datasets as they are semantically distat enough from the ID and the other OOD datasets.

Methodology: The methods are tested with a similar number of parameters, which we change with the dimension of data. We keep the same models as the ones trained in section 4.2 and make the other models in our benchmark match this number of parameters. Finally, the number of OOD and ID data are equalized in order to have meaningful AUROC and TNR measures as the ROC curve is sensitive to class imbalance [32]. This attention we brought on the class balance and the number of parameters is rarely done in the field of OOD detection. If this balance is not present, the models are not strictly comparable with each other, making the interpretation of results impossible.

Results: The results reported in Table 2 show state-of-the-art performance for the penalized model. We report an average AUROC of 97.0% on the CIFAR-10 benchmarks, 98.7% on the FashionMNIST benchmarks and a global average AUROC of 97.6% on all datasets. On the other hand, we measure an average AUROC of 62.4% for Generalized ODIN, 83.8% for Mahalanobis and 84.0% for Energy (or 79.44% when not taking into account MNIST and SVHN as these datasets are used during training to tune the energy bounds).

Discussion: Our state-of-the-art results on OOD detection show that the approximate mass regularization performs well when there is both a covariate shift and a semantic shift between the in-distribution and the out-distributions. Our results surpass by 16% on average the AUROC of the best baseline, Energy with OOD datasets included. Furthermore, we note a more consistent behavior of our model than the baseline as our penalized training yields good performance on all datasets in the benchmark while other models are more irregular in their classification performance depending on the OOD dataset. Finally, a gap can be observed between TNR@95%TPR values and the AUROC values for the other models in the benchmark. We investigated this discrepancy by plotting their respective ROC curve where we observe that lowering the value of the TPR increases the TNR (for example, TNR@95%TPR is greater than TNR@99%TPR but lower than TNR@90%TPR).

In the following section, we assess its applicability in semantic shift detection as the features used to detect OOD data might not transfer similarly to every class of distributional shifts.

4.4. Anomalous class detection

Anomaly detection or anomalous class detection is a task where the model is trained on a single class in a dataset (corresponding to its in-distribution) while the remaining classes of the dataset are treated as the out-distribution. The goal of this study is to assess the applicability of the approximate mass in a semantic shift scenario. We evaluate our model in a similar setting to [33]:

Metrics: The AUROC is usually reported for this task.

Baselines: We compare our approach to two state-of-the-art models in the domain: OCGAN [34] and GradCon [33]. The former is based on the features extracted in the latent space by a GAN [3] while the latter is based on a gradient metric with respect to the model's features (instead of the input) of the reconstruction loss of a VAE [4].

Datasets: Simiarly to [33], we use the MNIST [23] and CIFAR-10 [16] datasets to test performance in an anomalous class setting.

Methodology: We split a dataset in an ID set made of only one class of the dataset while the other classes are OOD. In order to evaluate the model, we train it on a single class of the MNIST (or CIFAR-10) training data then test it on a split of the test set between the ID class and anomalous classes (the remaining classes).

Results: Tables 3 and 4 respectively show the AUROC obtained on a RealNVP model on MNIST and CIFAR-10 in an anomalous class detection setting. We report an average AUROC of 75.2% on MNIST and 76.1% on CIFAR-10 with a penalized RealNVP. The results reported in [33] are an average AUROC of 97.3% on MNIST and 66.4% CIFAR-10 for GradCon, and 97.5% on MNIST and 65.7% CIFAR-10 for OCGAN. This experiment shows that the approximate mass penalized model is sensitive to the semantic of data to some extent as it is able to detect label shifts. The gap in performance between the MNIST and CIFAR-10 datasets is much lower than the one measured for OCGAN or GradCon, consistently with results in section 4.3. However, the AUROC is distributed unevenly across classes in the datasets, some classes being seemingly less distinguishable by the model than others (e.g.: class 1 in the MNIST dataset in Table 3).

Discussion: We explain this result by looking at the information extracted by the model: the approximate mass takes the gradient with respect to the input instead of the features, as is performed by GradCon. Thus the model extracts information much closer to the input image, making it more sensitive to feature level information than semantic information. Furthermore, taking the norm rather than the angle of the gradient with respect to some reference vector, by opposition to GradCon, may also explain the disparity in performance as cosine similarity is most often used when measuring similarity between different semantics.

5. Conclusion

We show that normalizing flows trained with maximum likelihood assign higher approximate mass to ID data than to OOD data. This change of behavior of the approximate mass during training may be due to a displacement of the probability mass around input data. We propose a solution based on simultaneous optimization of the approximate mass and likelihood, yielding better results than the state-of-the-art OOD detection models, improving the baseline by 16% relatively to the best model. This method isotropically smoothes the log-likelihood around in-distribution input data. We show the approximate mass metric is more sensitive to covariate shifts as it allows for OOD detection (covariate and semantic shifts) while not being the best metric for semantic shift detection. Our method generalizes better for different OOD datasets compared to state-of-the-art models. In a follow up work, we aim to integrate this training objective in an adversarial context to test out covariate shift sensitivity.

Model	Training dataset	OOD dataset	AUROC	AUPR	TNR @95%TPR
G-ODIN	CIFAR-10	SVHN	0.810	0.804	0.355
Mahalanobis	CIFAR-10	SVHN	0.936	0.926	0.758
Energy	CIFAR-10	SVHN	**0.999**	**1.0**	**1.0**
Approximate mass	CIFAR-10	SVHN	0.969	0.969	**1.0**
G-ODIN	CIFAR-10	iNaturalist	0.581	0.552	0.124
Mahalanobis	CIFAR-10	iNaturalist	0.745	0.715	0.290
Energy	CIFAR-10	iNaturalist	0.647	0.902	0.872
Approximate mass	CIFAR-10	iNaturalist	**0.968**	**0.906**	**0.994**
G-ODIN	CIFAR-10	iNaturalist (plants)	0.581	0.552	0.124
Mahalanobis	CIFAR-10	iNaturalist (plants)	0.731	0.706	0.166
Energy	CIFAR-10	iNaturalist (plants)	0.587	**0.881**	0.077
Approximate mass	CIFAR-10	iNaturalist (plants)	**0.968**	0.799	**0.991**
G-ODIN	CIFAR-10	DTD	0.882	0.887	0.53
Mahalanobis	CIFAR-10	DTD	0.857	0.788	0.576
Energy	CIFAR-10	DTD	0.734	0.912	0.341
Approximate mass	CIFAR-10	DTD	**0.978**	**0.979**	**1.0**
G-ODIN	CIFAR-10	Places365	0.616	0.577	0.157
Mahalanobis	CIFAR-10	Places365	0.522	0.518	0.08
Energy	CIFAR-10	Places365	0.753	**0.937**	0.177
Approximate mass	CIFAR-10	Places365	**0.968**	0.936	**0.978**
G-ODIN	CIFAR-10	GTSRB	0.411	0.450	0.072
Mahalanobis	CIFAR-10	GTSRB	0.764	0.681	0.356
Energy	CIFAR-10	GTSRB	0.890	**0.975**	0.582
Approximate mass	CIFAR-10	GTSRB	**0.970**	0.970	**1.0**
G-ODIN	FashionMNIST	MNIST	0.535	0. 611	0.008
Mahalanobis	FashionMNIST	MNIST	0.995	0.995	0.994
Energy	FashionMNIST	MNIST	**1.0**	**1.0**	**1.0**
Approximate mass	FashionMNIST	MNIST	0.994	0.971	**1.0**
G-ODIN	FashionMNIST	EMNIST	0.870	0.885	0.425
Mahalanobis	FashionMNIST	EMNIST	**0.995**	**0.995**	0.982
Energy	FashionMNIST	EMNIST	0.990	0.998	0.955
Approximate mass	FashionMNIST	EMNIST	0.969	0.969	**1.0**
G-ODIN	FashionMNIST	KMNIST	0.328	0.391	0.019
Mahalanobis	FashionMNIST	KMNIST	0.990	0.990	0.948
Energy	FashionMNIST	KMNIST	0.981	0.995	0.917
Approximate mass	FashionMNIST	KMNIST	**1.0**	**0.996**	**1.0**

Table 2. Results on OOD classification for different state-of-the-art models. The energy model is trained with the CIFAR-10 (resp. FashionMNIST) as ID data and SVHN (resp. MNIST) datasets as OOD data.

Model	0	1	2	3	4	5	6	7	8	9
GradCon	0.995	**0.999**	**0.952**	**0.973**	0.969	0.977	**0.994**	0.979	0.919	**0.973**
OCGAN	**0.998**	**0.999**	0.942	0.963	**0.975**	**0.980**	0.991	**0.981**	0.939	**0.981**
Approximate mass	0.969	0.064	0.629	0.924	0.796	0.946	0.977	0.682	**0.969**	0.572

Table 3. AUROC results on anomalous class detection on MNIST for different models.

Model	Airplane	Automobile	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck
GradCon	0.760	0.598	0.648	0.586	**0.733**	0.603	0.684	0.567	0.784	**0.678**
OCGAN	0.757	0.531	0.640	0.620	0.723	0.620	0.723	0.575	0.820	0.554
Approximate mass	**0.855**	**0.605**	**0.756**	**0.835**	0.574	**0.801**	**0.797**	**0.862**	**0.871**	0.656

Table 4. AUROC results on anomalous class detection on CIFAR-10 for different models.

References

[1] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized Out-of-Distribution Detection: A Survey. *arXiv:2110.11334 [cs]*, Oct. 2021. 1, 2

[2] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing Flows for Probabilistic Modeling and Inference. *arXiv:1912.02762 [cs, stat]*, Apr. 2021. 1

[3] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. 1, 6

[4] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes, May 2014. 1, 2, 6

[5] Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do Deep Generative Models Know What They Don't Know? *arXiv:1810.09136 [cs, stat]*, Feb. 2019. 1, 2

[6] Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Why Normalizing Flows Fail to Detect Out-of-Distribution Data. *arXiv:2006.08545 [cs, stat]*, June 2020. 1, 2, 4

[7] Lily H Zhang, Mark Goldstein, and Rajesh Ranganath. Understanding Failures in Out-of-Distribution Detection with Deep Generative Models. page 10. 1, 2

[8] Anthony L. Caterini and Gabriel Loaiza-Ganem. Entropic Issues in Likelihood-Based OOD Detection. In *I (Still) Can't Believe It's Not Better! Workshop at NeurIPS 2021*, pages 21–26. PMLR, Feb. 2022. 1, 2

[9] Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, and Balaji Lakshminarayanan. Detecting Out-of-Distribution Inputs to Deep Generative Models Using Typicality. page 15. 1, 2

[10] Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your Classifier is Secretly an Energy Based Model and You Should Treat it Like One. *arXiv:1912.03263 [cs, stat]*, Sept. 2020. 1, 2, 5

[11] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using Real NVP. *arXiv:1605.08803 [cs, stat]*, Feb. 2017. 2

[12] Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative Flow with Invertible 1x1 Convolutions. *arXiv:1807.03039 [cs, stat]*, July 2018. 2

[13] Laurent Dinh, David Krueger, and Yoshua Bengio. NICE: Non-linear Independent Components Estimation, Apr. 2015. 2

[14] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples, Mar. 2015. 2, 4

[15] H. Drucker and Y. Le Cun. Improving generalization performance using double backpropagation. *IEEE Transactions on Neural Networks*, 3(6):991–997, Nov. 1992. 2

[16] Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. page 60. 2, 4, 5, 6

[17] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Ng. Reading Digits in Natural Images with Unsupervised Feature Learning. *NIPS*, Jan. 2011. 2, 4, 5

[18] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. *arXiv:1706.02690 [cs, stat]*, Aug. 2020. 4, 5

[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 *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10948–10957, Seattle, WA, USA, June 2020. IEEE. 4, 5

[20] Takeru Miyato, Shin-Ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1979–1993, Aug. 2019. 4

[21] Christian Etmann. A Closer Look at Double Backpropagation, June 2019. 4

[22] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms. *arXiv:1708.07747 [cs, stat]*, Sept. 2017. 4, 5

[23] Li Deng. The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]. *IEEE Signal Processing Magazine*, 29(6):141–142, Nov. 2012. 4, 5, 6

[24] 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*, volume 31. Curran Associates, Inc., 2018. 5

[25] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based Out-of-distribution Detection. In *Advances in Neural Information Processing Systems*, volume 33, pages 21464–21475. Curran Associates, Inc., 2020. 5

[26] Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep Learning for Classical Japanese Literature, 9999. 5

[27] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: An extension of MNIST to handwritten letters, Mar. 2017. 5

[28] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing Textures in the Wild. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613, June 2014. 5

[29] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks*, pages 1453–1460, July 2011. 5

[30] 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, June 2018. 5

[31] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The iNaturalist Species Classification and Detection Dataset. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8769–8778, Salt Lake City, UT, June 2018. IEEE. 5

[32] Takaya Saito and Marc Rehmsmeier. The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PLoS ONE*, 10(3):e0118432, Mar. 2015. 5

[33] Gukyeong Kwon, Mohit Prabhushankar, Dogancan Temel, and Ghassan AlRegib. Backpropagated Gradient Representations for Anomaly Detection. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, volume 12366, pages 206–226. Springer International Publishing, Cham, 2020. 6

[34] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. OC-GAN: One-Class Novelty Detection Using GANs With Constrained Latent Representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2898–2906, 2019. 6