On Out-of-distribution Detection with Energy-based Models

Sven Elflein [1] **Bertrand Charpentier** [1] **Daniel Zügner** [1] **Stephan Günnemann** [1]

Abstract

Several density estimation methods have shown to fail to detect out-of-distribution (OOD) samples by assigning higher likelihoods to anomalous data. Energy-based models (EBMs) are flexible, unnormalized density models which seem to be able to improve upon this failure mode. In this work, we provide an extensive study investigating OOD detection with EBMs trained with different approaches on tabular and image data and find that EBMs do not provide consistent advantages. We hypothesize that EBMs do not learn semantic features despite their discriminative structure similar to Normalizing Flows. To verify this hypotheses, we show that supervision and architectural restrictions improve the OOD detection of EBMs independent of the training approach.

1. Introduction

To leverage deep learning in security-critical application areas, such as medical applications and autonomous driving, robustness, and uncertainty have recently received increased attention (Varshney, 2016). An open question is to ensure that the model does account for out-of-distribution (OOD) data where recent work has shown that models tend to make over-confident predictions (Lakshminarayanan et al., 2017; Hein et al., 2019). One approach to solve this problem is to estimate the training data distribution and reject samples if the density at that point is low. However, Normalizing Flows, (Rezende & Mohamed, 2016) which are powerful density estimators based on a sequence of invertible transformations, tend to assign higher likelihoods to the OOD than the in-distribution (ID) data (Nalisnick et al., 2019a). Another promising class of density estimators without restrictions on the architecture are Energy-based models (EBM) (LeCun et al.). Recently, Grathwohl et al. (2020b) improved OOD detection by interpreting discriminative models as EBMs. This encourages that EBMs might be better suited for the task of OOD detection. In this work, we aim to investigate this claim and the main factors facilitating superior OOD detection of EBMs.

We summarize our contributions as follows: **(1)** we find that EBMs do not strictly outperform Normalizing Flows across multiple training methods, **(2)** identify that learning semantic features induced by supervision improves OOD detection in recent discriminative EBMs (Grathwohl et al., 2020b) and, **(3)** show that one can use architectural modifications to improve OOD detection with EBMs similar to Normalizing Flows (Kirichenko et al., 2020).

2. Related Work

Classifier-based OOD detection. Initially, Hendrycks & Gimpel (2018) proposed to use the maximum softmax probability as OOD score. Liang et al. (2020); Hsu et al. (2020) augment this approach by temperature scaling. Other methods add additional loss terms to the objective to encourage maximum entropy predictions for OOD inputs (Hendrycks et al., 2019; Lee et al., 2018a; Sricharan & Srivastava, 2018; Hein et al., 2019). Malinin & Gales (2018; 2019); Charpentier et al. (2020) obtain uncertainty estimates for OOD detection by predicting parameters of a Dirichlet distribution for classification.

Density-based OOD detection. A set of methods estimates the distribution over activations at multiple layers (Lee et al., 2018b; Zisselman & Tamar, 2020). Other methods focus on the data distribution directly: Nalisnick et al. (2019a) discovered that the density learned by generative models cannot distinguish between ID and OOD inputs. Various works study this observation identifying background statistic (Ren et al., 2019), excessive influence of input complexity (Serrà et al., 2020), and mismatch between the typical set and high-density regions (Nalisnick et al., 2019b; Choi et al., 2019; Morningstar et al., 2020) as causes. In comparison to our work, these methods focus on flow-based and autoregressive density methods with tractable likelihood.

Recently, there has also been increasing interest in leveraging EBMs as generative models for OOD detection. Du & Mordatch (2020) investigate the generative capabilities and generalization of EBMs to OOD inputs. Zhai et al. (2016) train EBM architectures with a score matching objective for anomaly detection. Grathwohl et al. (2020b;a) derive optimization procedures for hybrid EBMs and investigate

[1]Technical University Munich, Germany. Correspondence to: Sven Elflein <sven.elflein@in.tum.de>.

Presented at the ICML 2021 Workshop on Uncertainty and Robustness in Deep Learning., Copyright 2021 by the author(s).

their OOD detection performance. However, existing work does not study the factors leading to improved OOD detection with EBMs compared to other generative models. Thus, most relevant to our work are the studies by Kirichenko et al. (2020); Schirrmeister et al. (2020) which found that Normalizing Flows learn low-level features common to image datasets and thus struggle with detecting OOD inputs. We aim to provide similar insight for EBMs.

3. Method

In the following, we specify the structure of the EBMs and provide an overview of the training methods considered in this work.

Energy-based model. EBMs (LeCun et al.) are defined by an energy-function E_θ which defines a density over the data x as

$$p_\theta(x) = \frac{\exp(-E_\theta(x))}{Z(\theta)} \tag{1}$$

where $Z(\theta) = \int \exp(-E_\theta(x))dx$ is the normalizing constant and θ are learnable parameters. In particular, E_θ can be any function $E : \mathbb{R}^D \mapsto \mathbb{R}$ placing no restrictions on the model compared to Normalizing Flows.

Joint Energy model. We additionally consider Joint Energy models (JEM) for discriminative EBMs (Grathwohl et al., 2020b). Given a classifier $f : \mathbb{R}^D \mapsto \mathbb{R}^C$ assigning logits for C classes for a datapoint $x \in \mathbb{R}^D$, the probabilities over the classes are defined as

$$p_\theta(y \mid x) = \frac{\exp(f_\theta(x)[y])}{\sum_{y'} \exp(f_\theta(x)[y']} \tag{2}$$

where $f_\theta(x)[y]$ denotes the y-th logit. The logits $f_\theta(x)[y]$ can be interpret as unnormalized probabilities of the joint distribution $p_\theta(x, y)$ which yields the marginal distribution over x as

$$p_\theta(x) = \sum_y p_\theta(x, y) = \sum_y \frac{\exp(f(x)[y])}{Z(\theta)} \tag{3}$$

We follow (Grathwohl et al., 2020b) and optimize the factorization $\log p_\theta(x, y) = \log p_\theta(x) + \log p_\theta(y \mid x)$ using Equation (2) and Equation (3). In particular, we use a Cross Entropy objective to optimize $p_\theta(y \mid x)$ weighted with hyperparameter γ.

For optimizing $p_\theta(x)$, we consider different approaches which have shown to scale to high-dimensional data. Note that this term should contribute significantly to the OOD detection performance of the model. We introduce the training approaches used in this work in the following.

Sliced score matching. Hyvärinen (2005) propose to learn an unnormalized density by approximating the score of the distribution $s_\theta(x) = \nabla_x p(x)$. Song et al. (2019) introduce an efficient update formula based on random projection $\mathbb{E}_{p_v}\mathbb{E}_{p(x)}\left[v^T \nabla_x s_\theta(x) v + \frac{1}{2}\|s_\theta(x)\|_2^2\right]$ where $v \sim p_v$ is a simple distribution of random vectors.

Contrastive divergence. Hinton (2002) approximates the gradient of the maximum likelihood objective by $\nabla_\theta p_\theta(x) = \mathbb{E}_{p_\theta(x')}\left[\nabla_\theta E_\theta(x')\right] - \nabla_\theta E_\theta(x)$ Following recent literature (Du & Mordatch, 2020), we approximate the expectation with samples obtained through Stochastic Gradient Langevin Dynamics (Welling & Teh, 2011).

VERA. Lastly, we consider the recently proposed Variational Entropy Regularized Approximate maximum likelihood (VERA) training (Grathwohl et al., 2020a) which learns the parameters ϕ of a auxiliary distribution q_ϕ as the optimum of $\log Z(\theta) = \max_{q_\phi} \mathbb{E}_{q_\phi(x)}\left[f_\theta(x)\right] + H(q_\phi)$ which can be plugged into Equation (1) to obtain an alternative method for training EBMs. Grathwohl et al. (2020a) propose a variational approximation to estimate the entropy term H_{q_ϕ} circumventing the need for sampling (Dieng et al., 2019; Titsias & Ruiz, 2019).

While more approaches for training EBMs exist, they either assume knowledge about a noise distribution close to the ground-truth data distribution (Gutmann & Hyvarinen; Ceylan & Gutmann, 2018) or have shown to require prohibitive amounts of training time in our experiments (Gao et al., 2020).

4. Experiments

We investigate the OOD detection performance of EBMs trained with the approaches discussed in Section 3. In particular, we verify the following hypotheses improving OOD detection with EBMs in recent works (Grathwohl et al., 2020b;a) compared to Normalizing Flows:

Dimensionality reduction. The manifold hypotheses (Fefferman et al., 2013) suggests that high-dimensional data such as images reside on a lower-dimensional manifold. Normalizing Flows require invertible transformations and thus operate in the original data space. We hypothesize that this hinders OOD detection as they need to model off-manifold directions. Contrarily, EBMs do not require invertibility, which allows pruning of redundant dimensions without semantic content.

Supervision. Kirichenko et al. (2020); Schirrmeister et al. (2020) show that Normalizing Flows learn low-level features without semantic meaning (smoothness, etc.) common to all natural images (Serrà et al., 2020). We hypothesize that label information encourages semantic, high-level features instead, improving OOD detection.

Setup. We perform OOD detection by comparing the density of ID and OOD inputs under the learned $p_\theta(x)$. For evaluation, we consider OOD detection as a binary classification problem with labels 1 for ID and 0 for OOD and report average precision (AUC-PR) as commonly done in

Table 1. AUC-PR for OOD detection on the respective in-distribution dataset.

ID dataset	CIFAR-10					FMNIST			Segment	Sensorless
OOD dataset	CIFAR-100	CelebA	LSUN	SVHN	Textures	KMNIST	MNIST	NotMNIST	Segment OOD	Sensorless OOD
CE	**62.76 ± 1.46**	64.47 ± 2.44	65.18 ± 5.79	47.51 ± 4.58	39.17 ± 2.28	69.07 ± 6.73	**82.5 ± 12.27**	50.9 ± 6.73	33.35 ± 1.82	33.02 ± 1.32
NF	58.34	**74.68**	62.99	31.58	50.23	62.22	49.03	**93.68**	**99.45 ± 0.18**	**94.35**
CD	50.51 ± 2.13	43.86 ± 5.85	54.43 ± 11.37	**60.72 ± 24.59**	**76.21 ± 17.44**	50.52 ± 9.39	31.69 ± 0.9	76.85 ± 2.66	98.18 ± 2.18	72.83 ± 16.19
SSM	53.82 ± 3.12	57.72 ± 7.0	52.79 ± 3.16	45.75 ± 7.24	48.82 ± 4.34	58.98 ± 5.48	67.86 ± 11.4	57.27 ± 13.73	79.43 ± 24.29	67.14 ± 20.31
VERA	55.95 ± 2.68	73.97 ± 2.63	**67.39 ± 2.57**	37.27 ± 4.66	46.29 ± 8.1	**78.11 ± 21.05**	67.53 ± 21.63	76.22 ± 22.11	94.63 ± 7.22	45.66 ± 10.55

ID dataset	CIFAR-10			FMNIST			Segment		Sensorless	
OOD dataset	Constant	Noise	OODomain	Constant	Noise	OODomain	Constant	Noise	Constant	Noise
CE	45.26 ± 8.8	61.13 ± 21.02	30.69 ± 0.0	35.5 ± 3.08	55.84 ± 22.32	30.74 ± 0.11	41.74 ± 18.57	33.66 ± 2.77	32.38 ± 1.19	31.97 ± 1.26
NF	30.87	83.65	**100.0**	**71.07**	98.04	**100.0**	99.95	**100.0**	**100.0**	**100.0**
CD	**58.75 ± 28.17**	**100.0 ± 0.0**	58.41 ± 37.96	70.59 ± 12.84	**100.0 ± 0.0**	**100.0 ± 0.0**	95.47 ± 2.34	95.14 ± 3.71	**100.0 ± 0.0**	**100.0 ± 0.0**
SSM	47.24 ± 15.56	70.28 ± 31.39	68.57 ± 25.2	47.57 ± 15.18	49.45 ± 21.19	76.76 ± 21.1	73.91 ± 25.44	81.1 ± 17.87	69.79 ± 6.62	64.61 ± 17.35
VERA	31.51 ± 0.66	**100.0 ± 0.0**	63.48 ± 34.37	53.24 ± 22.65	79.34 ± 27.34	72.42 ± 37.61	**100.0 ± 0.0**	**100.0 ± 0.0**	**100.0 ± 0.0**	99.85 ± 0.31

the literature (Hendrycks & Gimpel, 2018). We train EBMs with Sliced Score Matching (*SSM*), Contrastive Divergence (*CD*), and *VERA* as described in Section 3. For baselines, we compare with Normalizing Flow (*NF*) and the energy score of a classifier (Liu et al., 2020) (*CE*) [1].

Datasets. Following (Charpentier et al., 2020), we consider the tabular datasets *Sensorless drive* and *Segment*, with dimensionality 18 and 49 and 4 and 11 classes, respectively. To obtain a representative OOD dataset, we remove one class (*sky*) from Segment and two classes (*10*, *11*) from Sensorless drive. Further, we evaluate on image datasets. We use FMNIST (Xiao et al., 2017) as ID dataset and MNIST (Lecun et al., 1998), NotMNIST (Bulatov, 2011), KMNIST (Clanuwat et al., 2018) as OOD datasets. Additionally, we train on CIFAR-10 (Krizhevsky, 2009) and use LSUN (Yu et al., 2016), Textures (Huang et al., 2020), CIFAR-100 (Krizhevsky, 2009), SVHN (Netzer et al., 2011) and CelebA (Liu et al., 2015) as out-of-distribution. In the following, we refer to these OOD datasets as *natural* OOD datasets. Finally, we generate *non-natural* OOD datasets with *noise* and *constant* input. As proposed by Charpentier et al. (2020), we also consider an *OODomain* dataset where the input data is not normalized into the range $[0, 1]$.

Natural vs. non-natural datasets. Note that the differentiation of *natural* and *non-natural* datasets allows evaluating distinct properties of the learned density: A model able to distinguish *natural* inputs can recognize semantic features of the high-level content of images, e.g., corresponding to classes, while *non-natural* inputs are easily detected semantically but lie farther away from the data manifold, thus, require the model to decrease the density when moving away from the data distribution. **Architectures.** We use MLPs on the tabular datasets and WideResNet-10-2 (Zagoruyko & Komodakis, 2017) for the image datasets. For the Normalizing Flow baseline, radial flows (Rezende & Mohamed, 2016) are used on the tabular and Glow (Kingma & Dhariwal, 2018) on the image datasets. We provide more details in Appendix C.

[1] We provide code at https://github.com/selflein/EBM-OOD-Detection

4.1. Are EBMs better than baselines in general?

Experiment 1. We establish baseline results by training EBMs and baseline models. In Table 1, we find that EBMs consistenly outperform the *CE* baseline by 62.9%, 55.0%, and 36.4% for *CD*, *VERA*, and *SSM* respectively. The improvements are moderate in comparison to the Normalizing Flow baselines with 11.9%, 4.3%, and -4.3%. Notably, improvements are mostly on *natural* datasets. As EBMs perform dimensionality reduction since they map to the scalar energy and do not consistently outperform Normalizing Flows in this experiment across all training methods, we conclude that dimensionality reduction plays a minor role in the OOD detection performance of recent EBMs (Grathwohl et al., 2020b). We attribute slight improvements on *natural* data to the ability to discard non-semantic dimensions in EBMs.

4.2. Does supervision improve OOD detection?

Next, we consider two ways of incorporating labels to investigate the influence of supervision. Firstly, by applying an additional loss term as in (Grathwohl et al., 2020b) which affects the optimization directly, and secondly, performing density estimation on embeddings of a classification model which incorporates supervision indirectly through class-related features.

Experiment 2. We consider JEMs as introduced in Section 3 and apply a cross entropy objective with weighting hyperparameter γ optimizing $p_\theta(y \mid x)$. In Table 2, we find substantial improvements in OOD detection on most datasets compared to the baseline models. Using label information within the model encourages discriminative features relevant for classification, improving detection of *natural*, OOD inputs by 29.61%. These results indicate that EBM training tends to assign high-likelihood to all natural, structured images, an issue also observed in other generative models (Ren et al., 2019). Note however that supervision decreases performance on some *natural* datasets and consistently worsens results at differentiating *non-natural* inputs (-11.74%). Investigating the difference in results between *natural* and *non-natural* datasets, we observe in Figure 1

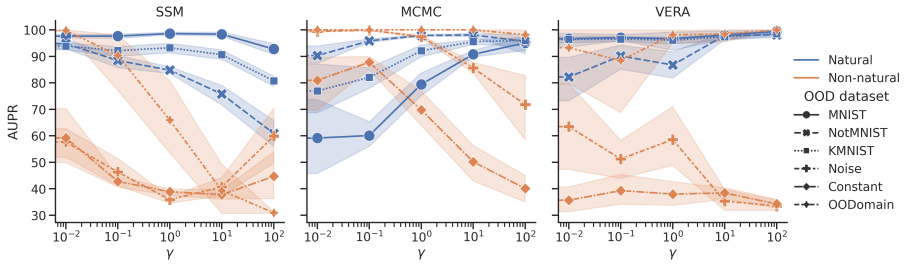


Figure 1. AUC-PR for OOD detection for different settings of the weighting hyperparameter γ of the cross entropy objective. FMNIST is used as the in-distribution dataset.

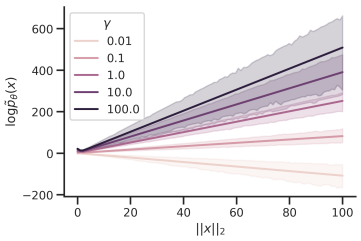


Figure 2. Unnormalized density $\tilde{p}_\theta(x)$ of inputs with increasing L_2-norm.

Table 2. % improvement in AUC-PR for OOD detection when using additional supervision during training.

Model	ID dataset	Natural	Non-natural
CD	CIFAR-10	-10.82	-9.11
	FMNIST	47.17	3.24
	Segment	1.85	0.89
	Sensorless	29.72	-0.02
SSM	CIFAR-10	7.33	-27.94
	FMNIST	50.61	-20.26
	Segment	25.89	-21.94
	Sensorless	22.13	-40.73
VERA	CIFAR-10	-1.16	-3.00
	FMNIST	33.66	-15.53
	Segment	4.98	-0.57
	Sensorless	97.93	0.07

Table 3. % improvement in AUC-PR for OOD detection when training on embeddings.

Model	ID dataset	Natural	Non-natural
CD	CIFAR-10	48.60	3.37
	FMNIST	95.79	-13.52
SSM	CIFAR-10	53.84	-2.31
	FMNIST	58.40	59.59
VERA	CIFAR-10	50.16	16.97
	FMNIST	15.12	1.80

Table 4. % improvement in AUC-PR for OOD detection after introducing bottlenecks.

Model	ID dataset	Natural	Non-natural
CD	CIFAR-10	20.18	20.38
	FMNIST	67.95	10.88
SSM	CIFAR-10	14.76	33.34
	FMNIST	1.75	-5.92
VERA	CIFAR-10	19.66	33.22
	FMNIST	26.84	32.94

that the OOD detection on *non-natural* images is negatively impacted by increasing weighting of the cross-entropy objective. In Figure 2, we observe that the EBM assigns exponentially increasing density to datapoints distant from the training data distribution for higher settings of γ similar to what has been proven for the confidence in ReLU networks (Hein et al., 2019). As a result *non-natural* inputs which are further away from the training data than *natural* images become increasingly harder to detect. We conclude that training with this factorization requires tuning of γ to achieve high OOD detection performance on both *natural* and *non-natural* inputs.

Experiment 3. Sidestepping the issue of tuning γ, we follow Kirichenko et al. (2020) noticing that training Normalizing Flows on high-level features improves OOD detection. To investigate this behavior for EBMs, we store the features from a classifier trained with cross-entropy objective after convolutional layers. Subsequently, we train EBMs on these embeddings. In Table 3, we observe that density estimation on embeddings significantly improves results on *natural* datasets compared to the baseline trained on images directly ($+53.65\%$). Further, performance on *non-natural* datasets does not deteriorate with this approach and increases performance by 10.98% on average. As training on discriminative features directly improves OOD detection, this supports our hypotheses that EBMs trained on high-dimensional data such as images struggle to learn semantic features.

4.3. Can we encourage semantic features?

While EBMs inherently perform dimensionality reduction, the previous experiments suggest this being insufficient to capture semantic features within the data. As shown by Kirichenko et al. (2020), introducing a bottleneck in the coupling transforms of Normalizing Flows enforces the network to learn semantic features improving OOD detection. This can also be interpret in the frame of compression (Serrà et al., 2020) where redundant information is removed.

Experiment 4. We introduce bottlenecks after every block of the WRN through a set of 1×1 convolutions mapping to $0.2\times$ the original dimensionality. In Appendix A, we provide results for other settings of the bottleneck. In Table 4, we observe that this simple adjustment yields improvements in OOD detection on *natural* images for all training methods. The bottlenecks force the network to compress the features removing redundant information and enable improved OOD detection supporting the hypotheses that generic EBMs retain non-semantic features. We provide further investigation on low-level features in Appendix A.1.

5. Conclusion

Overall, we find that **(1)** EBMs struggle with OOD detection on high-dimensional data but to a lower degree than Normalizing Flows, **(2)** incorporating task-specific priors such as supervision significantly improves OOD detection on *natural* OOD data, and **(3)** architectural modifications can be used to improve the OOD detection performance.

References

Bulatov, Y. Machine Learning, etc: notMNIST dataset, September 2011.

Ceylan, C. and Gutmann, M. U. Conditional Noise-Contrastive Estimation of Unnormalised Models. *arXiv:1806.03664 [cs, stat]*, June 2018.

Charpentier, B., Zügner, D., and Günnemann, S. Posterior Network: Uncertainty Estimation without OOD Samples via Density-Based Pseudo-Counts. *arXiv:2006.09239 [cs, stat]*, June 2020.

Choi, H., Jang, E., and Alemi, A. A. WAIC, but Why? Generative Ensembles for Robust Anomaly Detection. *arXiv:1810.01392 [cs, stat]*, May 2019.

Clanuwat, T., Bober-Irizar, M., Kitamoto, A., Lamb, A., Yamamoto, K., and Ha, D. Deep Learning for Classical Japanese Literature. *arXiv:1812.01718 [cs, stat]*, 2018. doi: 10.20676/00000341.

Dieng, A. B., Ruiz, F. J. R., Blei, D. M., and Titsias, M. K. Prescribed Generative Adversarial Networks. *arXiv:1910.04302 [cs, stat]*, October 2019.

Du, Y. and Mordatch, I. Implicit Generation and Generalization in Energy-Based Models. *arXiv:1903.08689 [cs, stat]*, June 2020.

Fefferman, C., Mitter, S., and Narayanan, H. Testing the Manifold Hypothesis. *arXiv:1310.0425 [math, stat]*, December 2013.

Gao, R., Nijkamp, E., Kingma, D. P., Xu, Z., Dai, A. M., and Wu, Y. N. Flow Contrastive Estimation of Energy-Based Models. *arXiv:1912.00589 [cs, stat]*, April 2020.

Grathwohl, W., Kelly, J., Hashemi, M., Norouzi, M., Swersky, K., and Duvenaud, D. No MCMC for me: Amortized sampling for fast and stable training of energy-based models. *arXiv:2010.04230 [cs]*, October 2020a.

Grathwohl, W., Wang, K.-C., Jacobsen, J.-H., Duvenaud, D., Norouzi, M., and Swersky, K. Your Classifier is Secretly an Energy Based Model and You Should Treat it Like One. *arXiv:1912.03263 [cs, stat]*, September 2020b.

Gutmann, M. and Hyvarinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. pp. 8.

Hein, M., Andriushchenko, M., and Bitterwolf, J. Why ReLU networks yield high-confidence predictions far away from the training data and how to mitigate the problem. *arXiv:1812.05720 [cs, stat]*, May 2019.

Hendrycks, D. and Gimpel, K. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. *arXiv:1610.02136 [cs]*, October 2018.

Hendrycks, D., Mazeika, M., and Dietterich, T. Deep Anomaly Detection with Outlier Exposure. *arXiv:1812.04606 [cs, stat]*, January 2019.

Hinton, G. E. Training Products of Experts by Minimizing Contrastive Divergence. *Neural Computation*, 14 (8):1771–1800, August 2002. ISSN 0899-7667. doi: 10.1162/089976602760128018.

Hsu, Y.-C., Shen, Y., Jin, H., and Kira, Z. Generalized ODIN: Detecting Out-of-distribution Image without Learning from Out-of-distribution Data. *arXiv:2002.11297 [cs, eess]*, March 2020.

Huang, Y., Qiu, C., Wang, X., Wang, S., and Yuan, K. A Compact Convolutional Neural Network for Surface Defect Inspection. *Sensors*, 2020. doi: 10.3390/s20071974.

Hyvärinen, A. Estimation of Non-Normalized Statistical Models by Score Matching. *The Journal of Machine Learning Research*, 6:695–709, December 2005. ISSN 1532-4435.

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

Kirichenko, P., Izmailov, P., and Wilson, A. G. Why Normalizing Flows Fail to Detect Out-of-Distribution Data. *arXiv:2006.08545 [cs, stat]*, June 2020.

Krizhevsky, A. Learning Multiple Layers of Features from Tiny Images. /paper/Learning-Multiple-Layers-of-Features-from-Tiny-Krizhevsky/5d90f06bb70a0a3dced62413346235c02b1aa086, 2009.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. *arXiv:1612.01474 [cs, stat]*, November 2017.

LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., and Huang, F. J. A Tutorial on Energy-Based Learning. pp. 59.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998. ISSN 1558-2256. doi: 10.1109/5.726791.

Lee, K., Lee, H., Lee, K., and Shin, J. Training Confidence-calibrated Classifiers for Detecting Out-of-Distribution Samples. *arXiv:1711.09325 [cs, stat]*, February 2018a.

Lee, K., Lee, K., Lee, H., and Shin, J. A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. *arXiv:1807.03888 [cs, stat]*, October 2018b.

Liang, S., Li, Y., and Srikant, R. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. *arXiv:1706.02690 [cs, stat]*, August 2020.

Liu, W., Wang, X., Owens, J. D., and Li, Y. Energy-based Out-of-distribution Detection. *arXiv:2010.03759 [cs]*, October 2020.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Malinin, A. and Gales, M. Predictive Uncertainty Estimation via Prior Networks. *arXiv:1802.10501 [cs, stat]*, November 2018.

Malinin, A. and Gales, M. Reverse KL-Divergence Training of Prior Networks: Improved Uncertainty and Adversarial Robustness. *arXiv:1905.13472 [cs, stat]*, December 2019.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral Normalization for Generative Adversarial Networks. *arXiv:1802.05957 [cs, stat]*, February 2018.

Morningstar, W. R., Ham, C., Gallagher, A. G., Lakshminarayanan, B., Alemi, A. A., and Dillon, J. V. Density of States Estimation for Out-of-Distribution Detection. *arXiv:2006.09273 [cs, stat]*, June 2020.

Nalisnick, E., Matsukawa, A., Teh, Y. W., Gorur, D., and Lakshminarayanan, B. Do Deep Generative Models Know What They Don't Know? *arXiv:1810.09136 [cs, stat]*, February 2019a.

Nalisnick, E., Matsukawa, A., Teh, Y. W., and Lakshminarayanan, B. Detecting Out-of-Distribution Inputs to Deep Generative Models Using Typicality. *arXiv:1906.02994 [cs, stat]*, October 2019b.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Reading Digits in Natural Images with Unsupervised Feature Learning. *undefined*, 2011.

Nijkamp, E., Hill, M., Zhu, S.-C., and Wu, Y. N. Learning Non-Convergent Non-Persistent Short-Run MCMC Toward Energy-Based Model. *arXiv:1904.09770 [cs, stat]*, November 2019.

Ren, J., Liu, P. J., Fertig, E., Snoek, J., Poplin, R., DePristo, M. A., Dillon, J. V., and Lakshminarayanan, B. Likelihood Ratios for Out-of-Distribution Detection. *arXiv:1906.02845 [cs, stat]*, December 2019.

Rezende, D. J. and Mohamed, S. Variational Inference with Normalizing Flows. *arXiv:1505.05770 [cs, stat]*, June 2016.

Schirrmeister, R. T., Zhou, Y., Ball, T., and Zhang, D. Understanding Anomaly Detection with Deep Invertible Networks through Hierarchies of Distributions and Features. *arXiv:2006.10848 [cs, stat]*, November 2020.

Serrà, J., Álvarez, D., Gómez, V., Slizovskaia, O., Núñez, J. F., and Luque, J. Input complexity and out-of-distribution detection with likelihood-based generative models. *arXiv:1909.11480 [cs, stat]*, January 2020.

Song, Y., Garg, S., Shi, J., and Ermon, S. Sliced Score Matching: A Scalable Approach to Density and Score Estimation. *arXiv:1905.07088 [cs, stat]*, June 2019.

Sricharan, K. and Srivastava, A. Building robust classifiers through generation of confident out of distribution examples. *arXiv:1812.00239 [cs, stat]*, December 2018.

Tieleman, T. Training restricted Boltzmann machines using approximations to the likelihood gradient. In *ICML '08*, 2008. doi: 10.1145/1390156.1390290.

Titsias, M. K. and Ruiz, F. J. R. Unbiased Implicit Variational Inference. *arXiv:1808.02078 [cs, stat]*, February 2019.

Varshney, K. R. Engineering safety in machine learning. In *2016 Information Theory and Applications Workshop (ITA)*, pp. 1–5, January 2016. doi: 10.1109/ITA.2016.7888195.

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pp. 681–688, Madison, WI, USA, June 2011. Omnipress. ISBN 978-1-4503-0619-5.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms. *arXiv:1708.07747 [cs, stat]*, September 2017.

Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., and Xiao, J. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. *arXiv:1506.03365 [cs]*, June 2016.

Zagoruyko, S. and Komodakis, N. Wide Residual Networks. *arXiv:1605.07146 [cs]*, June 2017.

Zhai, S., Cheng, Y., Lu, W., and Zhang, Z. Deep Structured Energy Based Models for Anomaly Detection. *arXiv:1605.07717 [cs, stat]*, June 2016.

Zisselman, E. and Tamar, A. Deep Residual Flow for Out of Distribution Detection. *arXiv:2001.05419 [cs, stat]*, July 2020.

Appendix

A. Additional results

For completeness, we report the full results for OOD detection on *natural* datasets in Table 5 and on *non-natural* datasets in Table 6. Model with *-E* suffix correspond to models trained on embeddings of the classifier, while models with the *-S* suffix correspond to model trained with additional supervision in the form of cross-entropy objective weighted with parameter $\gamma = 1$.

We also present the results for different choices of the bottleneck dimensionality in Table 7.

In addition to the results on the effect of the weighting parameter γ of the cross-entropy loss on OOD detection in EBMs on FMNIST in the main paper, we add results for the Segment dataset in Figure 6, the Sensorless dataset in Figure 7 and CIFAR-10 in Figure 8. Our findings hold that the choice of γ heavily affects the OOD detection performance in particular on high-dimensional datasets.

A.1. Low-level features in EBMs

In the main paper, we argue that supervision encourages semantic features while unsupervised EBMs learn generic local pixel correlations (low-level features) common to all *natural* images as shown by Schirrmeister et al. (2020) which results in worse OOD detection performance on these datasets.

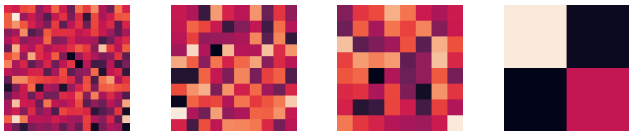


Figure 3. Example images generated with pooling sizes 2, 3, 4, and 16. Note that images become smoother the higher the pooling size.

Experiment 5 To provide further evidence for our observation that low-level features affect the likelihood of unsupervised EBMs, we include density histograms for datasets with varying low-level features. We take inspiration from (Serrà et al., 2020) and generate images with varying smoothness properties which has shown to affect the likelihood of samples in other generative models. In order to obtain images with different smoothness, we sample uniform noise at each pixel independently, apply average pooling with different pooling sizes, and resize to the original image dimensions with nearest neighbour upsampling. Images after this pre-processing procedure are shown in Figure 3. Subsequently, we estimate the density of 1000 images generated at each pooling fidelity under our models. Note that we use average pooling in comparison to max pooling in Serrà et al. (2020) since max pooling leads to images with different statistics (higher mean) for higher pooling sizes. Average pooling allows use to isolate the contribution of the change of features independent of image statistics.

In Figure 4a, we observe that unsupervised EBMs assign higher likelihood to smoother versions of the dataset (corresponding to higher pooling sizes), while the supervised EBM is not affected by the change of low-level features in Figure 4b. This demonstrates that unsupervised EBMs are susceptible to low-level features affecting the likelihood of samples, while supervised EBMs rely on higher-level, semantic features to assign likelihoods.

In Figure 4c and Figure 4d, we investigate the effect of applying the bottleneck to the architecture of the unsupervised EBM. We observe that the EBM with bottleneck assigns higher relative likelihood to the FashionMNIST test set vs. the artificial noise datasets containing low-level features only. This supports our observation in the main paper that including bottlenecks within the EBM helps the model to learn semantic features rather than local, low-level feature correlations.

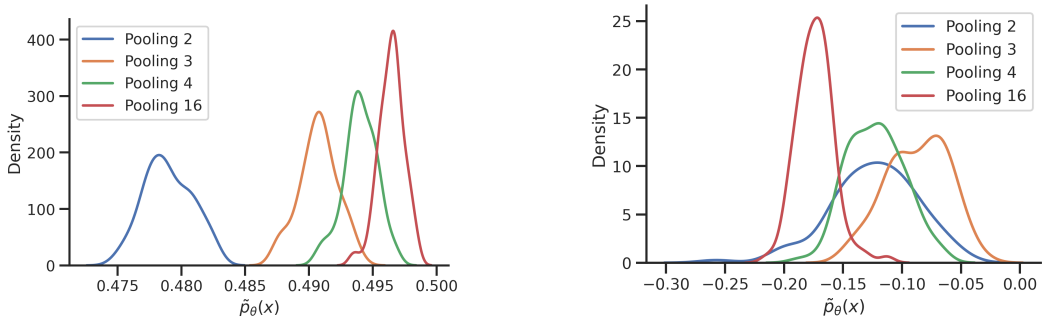
Experiment 6 Finally, we investigate images under the learned EBMs. Samples from the FashionMNIST dataset can be found in Figure 5a. We optimize the likelihood of these samples under the model and visualize samples for unsupervised EBM in Figure 5b and for supervised EBM in Figure 5c.

We observe that while the semantic content of samples under the unsupervised model becomes almost indistinguishable, the samples under the supervised model largely preserve their class semantics.

This result once more highlights that low-level features are the driving factor for high likelihood in unsupervised EBMs, while a notion of semantics is learned in supervised EBMs.

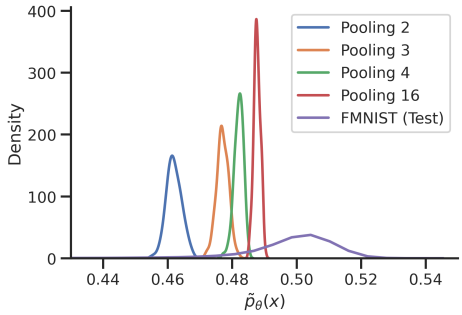
B. Training details

In this section, we provide further details on the training procedures and hyperparameters used for individual methods. Unless otherwise specified we use the Adam optimizer with default parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Further, we use learning rate warm-up with 2500 steps across all models. We train the models on the tabular datasets for $10,000$ steps and the image datasets for 50 epochs. We perform model selection based on the AUC-PR on an OOD validation dataset. For CIFAR-10, we use the validation sets of CelebA and

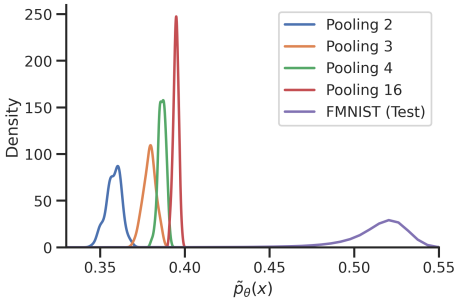


(a) Unsupervised EBM.

(b) Supervised EBM.



(c) EBM (unsupervised) without bottleneck.



(d) EBM (unsupervised) with bottleneck.

Figure 4. (a, b) Density histograms of generated dataset of noise images with different smoothness under different EBMs. Higher pooling corresponds to higher smoothness of the input images as visualized in Figure 3. (c, d) Comparison of density histogram of ID test set of FMNIST vs. low-level feature datasets for an EBM with and without bottleneck.

CIFAR-100, while for FMNIST we use the validation sets of MNIST and KMNIST. On the tabular dataset, we use 10% of the data of the removed classes for model selection and the remaining data as the test set.

Contrastive Divergence Following (Grathwohl et al., 2020b; Du & Mordatch, 2020), we use persistent contrastive divergence (Tieleman, 2008) which significantly reduces compute compared to seeding new chains at every iteration as in (Nijkamp et al., 2019). For the parameters of the Stochastic Gradient Langevin Dynamics sampler, we use the settings of Grathwohl et al. (2020b) and set the step size α to 1 and reinitialize samples from the replay buffer with probability 0.05. The size of the buffer is set to 10000. In contrast to (Grathwohl et al., 2020b), we found that training with 20 SGLD steps consistently diverged, thus, we set the number of SGLD steps to 100 which lead to stable convergence. Further, we set the initial learning rate to 0.001. We add additive Gaussian noise with variance 0.1 to the inputs in order to stabilize training (Du & Mordatch, 2020; Nijkamp et al., 2019).

VERA We use the default hyperparameters proposed in (Grathwohl et al., 2020a) and initialize the variance of the variational approximation η with 0.1 and clamp it in the range $[0.01, 0.3]$. We perform a grid search for the entropy regularizer in $[1e-4, 1]$ and found $1e^{-4}$ to yield to the best results in terms for training stability. Further, we set the learning rate of the EBM to $3e^{-4}$ and the learning rate of the generator to $6e^{-4}$. The Adam optimizer is used to optimize the generator with parameters $\beta_1 = 0.0$ and $\beta_2 = 0.9$.

For the generator architecture, we use a 5-layer MLP for the tabular datasets with hidden dimension 100 and leaky ReLU activations with slope 0.2. The latent distribution is a 16-dim. isotropic Normal distribution. For the image datasets, we follow (Grathwohl et al., 2020a) and use the generator from (Miyato et al., 2018) based on ResNet blocks with latent dimension 128.

Sliced Score Matching We set the distribution p_v to a multivariate Rademacher distribution which enables to use the variance reduced objective (SSM-VR) where the expectation $\mathbb{E}_{p_v}\left[v^T s_m(x;\theta))^2\right] = \|s_m(x;\theta)\|_2^2$ can be integrated analytically (Song et al., 2019). s_m denotes the score function of the EBM. During training, we use a single projection vector v from p_v to compute the objective.

Normalizing Flow We train Normalizing Flow models with maximum likelihood and learning rate $1e^{-3}$. We perform early stopping based on the log-likelihood with patience 10.

Cross-entropy classifier We train our cross-entropy baseline with learning rate $1e^{-3}$ on the tabular and $1e^{-4}$ on the image datasets. Further, we use weight decay with weight $5e^{-4}$ and perform early stopping based on the accuracy of the model.

C. Architecture details

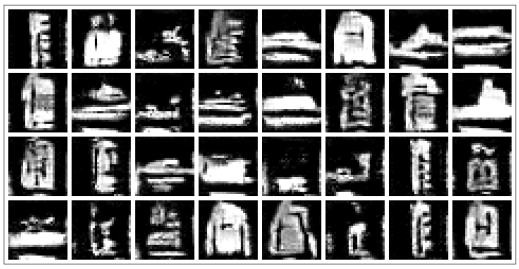
For the Normalizing Flows on the image datasets, we use a Glow (Kingma & Dhariwal, 2018) implementation with $L = 3$ layers, $K = 32$ steps, and $C = 512$ channels[2]. On the tabular datasets, we use 20 stacked radial transforms (Rezende & Mohamed, 2016). For all other models, we use a 5-layer MLP with ReLU activations on the tabular datasets and WideResNet-10-2 (Zagoruyko & Komodakis, 2017) on the image datasets.

D. Dataset details

In this section, we provide additional details on how we generate *non-natural* OOD datasets used in the paper. For the *Noise* dataset, we use an equal amount of samples from a Gaussian distribution $N(0, 1)$ and a uniform distribution $\mathcal{U}(-1, 1)$. The *Constant* dataset is sampled by drawing a scalar from $\mathcal{U}(-1, 1)$ and then filling a tensor with the same shape as the input data with the sampled value. Finally, *OODomain* inputs are the SVHN dataset and KMNIST dataset, where we do not apply normalization, for the in-distribution datasets of CIFAR-10 and FMNIST, respectively. As a result, the data is in the range $[0, 255]$.

[2]https://github.com/chrischute/glow

(a) Dataset samples.



(b) Optimized samples under **unsupervised** EBM.

(c) Optimized samples under **supervised** EBM.

Figure 5. Samples from the FMNIST dataset.

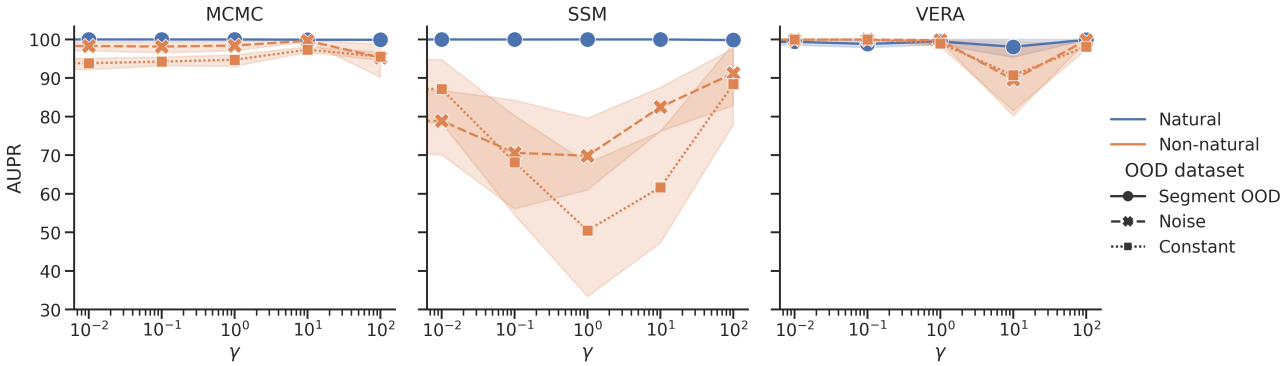


Figure 6. AUC-PR for OOD detection for different settings of the weighting hyperparameter γ of the cross entropy objective. Segment is used as the in-distribution dataset.

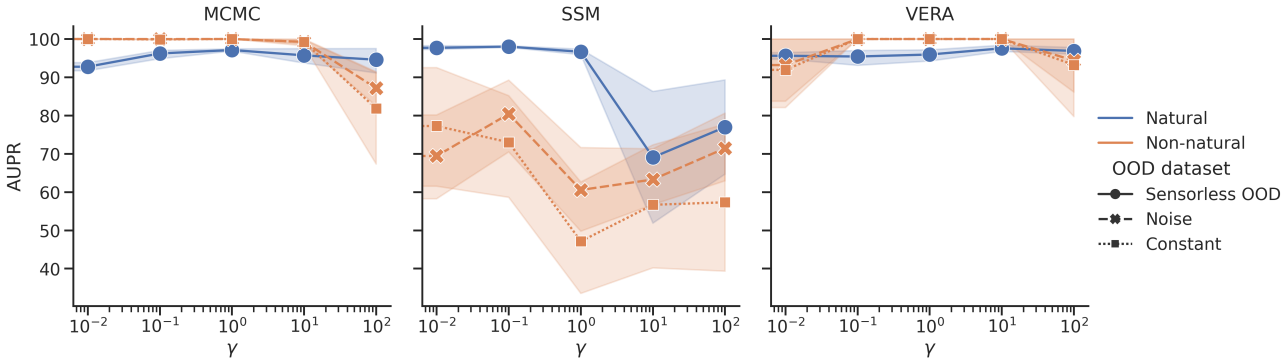


Figure 7. AUC-PR for OOD detection for different settings of the weighting hyperparameter γ of the cross entropy objective. Sensorless is used as the in-distribution dataset.

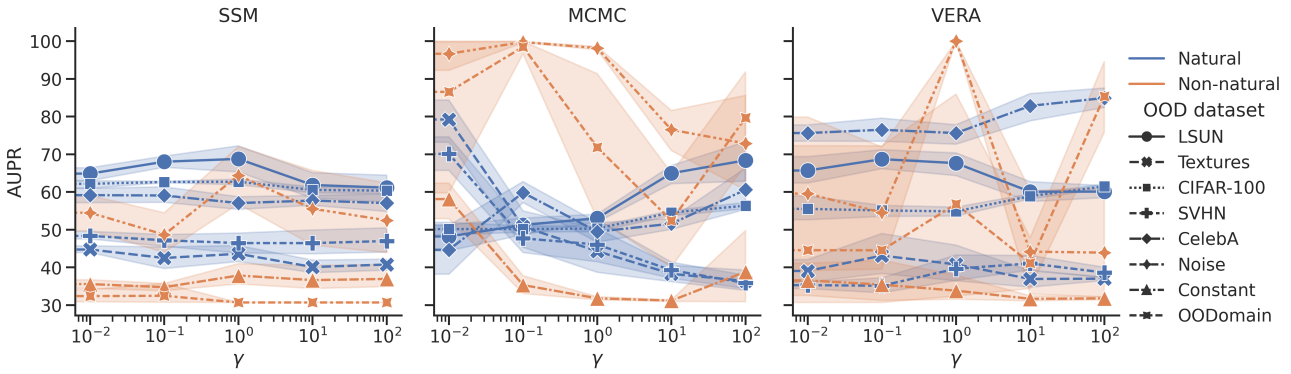


Figure 8. AUC-PR for OOD detection for different settings of the weighting hyperparameter γ of the cross entropy objective. CIFAR10 is used as the in-distribution dataset.

Table 5. AUC-PR for OOD detection on the natural datasets when trained on the respective in-distribution dataset.

ID dataset	CIFAR-10					FMNIST			Segment	Sensorless
OOD dataset / Model	CIFAR-100	CelebA	LSUN	SVHN	Textures	KMNIST	MNIST	NotMNIST	Segment OOD	Sensorless OOD
CE	62.76 ± 1.46	64.47 ± 2.44	65.18 ± 5.79	47.51 ± 4.58	39.17 ± 2.28	69.07 ± 6.73	82.5 ± 12.27	50.9 ± 6.73	33.35 ± 1.82	33.02 ± 1.32
NF	58.34	74.68	62.99	31.58	50.23	62.22	49.03	93.68	99.12	94.35
CD	50.51 ± 2.13	43.86 ± 5.85	54.43 ± 11.37	60.72 ± 24.59	76.21 ± 17.44	50.52 ± 9.39	31.69 ± 0.9	76.85 ± 2.66	98.18 ± 2.18	72.83 ± 16.19
CD-E	77.88 ± 1.61	66.21 ± 1.94	80.61 ± 4.26	**97.38 ± 1.15**	**98.6 ± 0.56**	90.53 ± 3.38	93.05 ± 2.88	88.05 ± 4.5	-	-
CD-S	49.81 ± 1.15	50.15 ± 4.24	53.09 ± 1.04	45.14 ± 7.55	46.55 ± 4.37	93.88 ± 1.44	83.47 ± 2.93	**98.03 ± 0.53**	**100.0 ± 0.0**	**94.48 ± 2.11**
SSM	53.82 ± 3.12	57.72 ± 7.0	52.79 ± 3.16	45.75 ± 7.24	48.82 ± 4.34	58.98 ± 5.48	67.86 ± 11.4	57.27 ± 13.73	79.43 ± 24.29	67.13 ± 20.31
SSM-E	84.73 ± 0.67	78.62 ± 2.23	**89.4 ± 1.18**	74.69 ± 2.76	69.8 ± 2.72	97.88 ± 0.66	95.57 ± 0.97	96.44 ± 0.83	-	-
SSM-S	62.71 ± 0.98	57.15 ± 2.18	68.77 ± 4.5	46.54 ± 3.54	43.51 ± 2.74	93.77 ± 1.08	99.22 ± 0.19	84.93 ± 2.07	**100.0 ± 0.0**	81.99 ± 21.79
VERA	55.95 ± 2.68	73.97 ± 2.63	67.39 ± 2.57	37.27 ± 4.66	46.29 ± 8.1	78.11 ± 21.05	67.53 ± 21.63	76.22 ± 22.11	94.63 ± 7.22	45.66 ± 10.55
VERA-E	76.66 ± 3.23	73.41 ± 6.68	81.31 ± 3.92	83.6 ± 7.45	78.52 ± 7.98	85.8 ± 15.18	88.52 ± 13.91	79.58 ± 15.61	-	-
VERA-S	61.37 ± 0.74	**85.02 ± 2.38**	58.91 ± 3.66	38.35 ± 1.08	36.68 ± 0.52	**98.89 ± 0.61**	**99.64 ± 0.53**	97.75 ± 1.72	99.35 ± 1.08	90.38 ± 3.78

Table 6. AUC-PR for OOD detection on the non-natural datasets when trained on respective in-distribution dataset.

ID dataset	CIFAR-10			FMNIST			Segment		Sensorless	
OOD dataset / Model	Constant	Noise	OODomain	Constant	Noise	OODomain	Constant	Noise	Constant	Noise
CE	45.26 ± 8.8	61.13 ± 21.02	30.69 ± 0.0	35.5 ± 3.08	55.84 ± 22.32	30.74 ± 0.11	42.57 ± 18.3	33.82 ± 3.14	32.42 ± 1.04	31.96 ± 1.28
NF	30.87	83.65	**100.0**	71.07	98.04	**100.0**	99.97	**100.0**	**100.0**	**100.0**
CD	58.75 ± 28.17	**100.0 ± 0.0**	58.41 ± 37.96	70.59 ± 12.84	**100.0 ± 0.0**	**100.0 ± 0.0**	96.13 ± 2.55	95.43 ± 3.58	**100.0 ± 0.0**	**100.0 ± 0.0**
CD-E	**99.92 ± 0.07**	87.5 ± 24.31	30.69 ± 0.0	**96.63 ± 7.53**	86.7 ± 26.75	35.83 ± 7.8	-	-	-	-
CD-S	31.32 ± 0.29	98.01 ± 0.93	70.86 ± 30.57	71.17 ± 10.16	97.79 ± 1.02	**100.0 ± 0.0**	94.57 ± 2.11	98.68 ± 1.9	99.97 ± 0.06	99.98 ± 0.03
SSM	47.24 ± 15.56	70.28 ± 31.39	68.57 ± 25.2	47.57 ± 15.18	49.45 ± 21.19	76.76 ± 21.1	74.86 ± 23.37	80.35 ± 17.38	69.66 ± 6.12	64.81 ± 17.03
SSM-E	65.64 ± 3.66	64.5 ± 4.05	42.74 ± 6.73	82.51 ± 4.95	87.54 ± 2.17	98.49 ± 1.94	-	-	-	-
SSM-S	37.8 ± 3.49	64.24 ± 12.88	30.69 ± 0.0	37.61 ± 1.83	33.71 ± 1.07	70.03 ± 17.45	51.57 ± 25.04	70.09 ± 17.57	37.5 ± 13.23	41.6 ± 12.05
VERA	31.51 ± 0.66	**100.0 ± 0.0**	63.48 ± 34.37	53.24 ± 22.65	79.34 ± 27.34	72.42 ± 37.61	**100.0 ± 0.0**	**100.0 ± 0.0**	**100.0 ± 0.0**	99.88 ± 0.25
VERA-E	83.95 ± 8.71	36.19 ± 4.43	30.69 ± 0.0	77.82 ± 21.24	60.28 ± 10.11	60.29 ± 28.5	-	-	-	-
VERA-S	31.7 ± 0.55	45.32 ± 25.14	92.1 ± 8.59	36.01 ± 4.24	33.73 ± 3.21	**100.0 ± 0.0**	99.01 ± 1.45	99.9 ± 0.3	**100.0 ± 0.0**	**100.0 ± 0.01**

Table 7. AUC-PR for OOD detection of EBMs with different choices of the dimension of the bottleneck introduced into the WideResNet-10-2.

	ID dataset	CIFAR-10								FashionMNIST					
Model	OOD dataset / Bottleneck	SVHN	LSUN	CelebA	CIFAR-100	Textures	Noise	OODomain	Constant	KMNIST	MNIST	NotMNIST	Noise	OODomain	Constant
MCMC	0.05	81.29	42.16	42.82	53.09	59.13	100.0	76.77	78.4	65.27	43.26	80.72	100.0	100.0	73.18
	0.10	67.93	48.3	52.71	53.47	61.99	100.0	62.19	70.85	73.8	53.23	83.25	100.0	100.0	78.27
	0.20	77.97	43.8	42.19	51.85	63.53	100.0	85.99	79.71	73.46	51.8	81.2	99.99	100.0	64.49
	1.00	60.72	54.43	43.86	50.51	76.21	100.0	58.41	58.75	50.52	31.69	76.85	100.0	100.0	70.59
SSM	0.05	53.6	49.44	54.06	53.47	43.3	51.57	89.8	67.7	56.79	62.09	49.05	42.48	77.2	46.42
	0.10	52.51	49.4	57.4	51.54	40.74	40.81	87.05	62.91	58.32	69.04	48.75	46.52	65.72	42.82
	0.20	52.69	49.29	59.96	52.37	49.31	49.26	71.39	57.82	62.31	53.76	68.8	67.07	65.84	56.88
	1.00	45.75	52.79	57.72	53.82	48.82	70.28	68.57	47.24	58.98	67.86	57.27	49.45	76.76	47.57
VERA	0.05	33.34	89.65	82.69	55.22	58.13	72.28	86.14	30.92	87.52	80.64	75.22	74.43	100.0	33.09
	0.10	34.52	80.64	79.4	54.44	54.84	45.02	84.56	33.65	86.85	85.65	66.3	75.58	96.28	31.54
	0.20	33.95	73.29	76.1	54.03	45.01	48.29	73.57	31.31	88.7	89.06	60.7	59.56	100.0	36.01
	1.00	37.27	67.39	73.97	55.95	46.29	100.0	63.48	31.51	78.11	67.53	76.22	79.34	72.42	53.24