BOUNDARY OF DISTRIBUTION SUPPORT GENERATOR (BDSG): SAMPLE GENERATION ON THE BOUNDARY

Nikolaos Dionelis, Mehrdad Yaghoobi, Sotirios A. Tsaftaris

The University of Edinburgh, Edinburgh, UK

ABSTRACT

Generative models, such as Generative Adversarial Networks (GANs), have been used for unsupervised anomaly detection. While performance keeps improving, several limitations exist particularly attributed to difficulties at capturing multimodal supports and to the ability to approximate the underlying distribution closer to the tails, i.e. the boundary of the distribution's support. This paper proposes an approach that attempts to alleviate such shortcomings. We propose an invertible-residual-network-based model, the Boundary of Distribution Support Generator (BDSG). GANs generally do not guarantee the existence of a probability distribution and here, we use the recently developed Invertible Residual Network (IResNet) and Residual Flow (ResFlow), for density estimation. These models have not yet been used for anomaly detection. We leverage IResNet and ResFlow for Out-of-Distribution (OoD) sample detection and for sample generation on the boundary using a compound loss function that forces the samples to lie on the boundary. The BDSG addresses non-convex support, disjoint components, and multimodal distributions. Results on synthetic data and data from multimodal distributions, such as MNIST and CIFAR-10, demonstrate competitive performance compared to methods from the literature.

Index Terms— Anomaly detection, invertible models

1. INTRODUCTION

Anomaly detection is the identification of samples different from typical data [1, 2]. When anomalies are not known in advance, unsupervised learning with generative models is used. The aim is to learn a model of "normality" with anomalies being detected as deviations from this model [3, 4]. Important goals are reducing misdetections and false alarms, estimating the support of the "normal" data distribution, detecting anomalies close to the support boundary, generating within-distribution and Out-of-Distribution (OoD) data, and providing decision boundaries for inference of within and OoD.

Existing approaches to anomaly detection use probability, reconstruction [5, 6], and domain based models. GANs are trained to generate samples and fit the "normal" data distribution [7, 8]. During inference, an anomaly score of a queried test sample, $\mathbf{x}^*$, is computed by evaluating the prob-

ability of obtaining $\mathbf{x}^*$ with the generator [9]. Such models belong to the probability-based methods (e.g. AnoGAN) [10, 11]. However, these models do not directly address the major problems of multimodal support and the ability to generate on the tails/boundaries. Recent approaches have tried to improve performance and alleviate shortcomings (e.g. MinLGAN and FenceGAN) [12, 13]. At present, generative models based on invertible residual networks, such as [14, 15], are lacking for unsupervised anomaly detection [16, 17]. Anomaly detection techniques show discernible limitations for detecting anomalies near the support of multimodal distributions [18, 19].

This work aims at addressing these limitations. Our aim is to detect abnormalities and generate samples on the boundary of the underlying multimodal distribution of the "normal data". We train invertible models [14] to estimate the density of typical samples and propose a loss function for the boundary generator. We pay particular attention to anomalies close to the boundary of the data distribution and to anomalies near high-probability normal samples. We focus on the ability to model multimodal distributions with non-convex support and disjoint components. Our model is denoted by Boundary of Distribution Support Generator (BDSG). It achieves competitive performance on synthetic and typically used benchmark data. In summary, our contributions are: (a) Training invertible generative models and evaluating the use of inference for anomaly detection, and (b) Sample generation on the tails.

2. RELATED WORK: BOUNDARY GENERATION

The GAN discriminator estimates the distance between the target and model distributions, while the generator learns the mapping from the latent space, $\mathbf{z}$, to the data space, $\mathbf{x}$. The GAN optimization is $\text{argmin}_{\boldsymbol{\theta}_g} \text{dist}(p_{\mathbf{x}}(\mathbf{x}), p_g(\mathbf{x}))$, where the distance metric is given by $\text{argmax}_{\boldsymbol{\theta}_d} f(D, p_{\mathbf{x}}(\mathbf{x}), p_g(\mathbf{x}))$, e.g. $\text{dist}(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_{\infty} = \max_i |x_i - y_i|$. The GAN loss is

$$\text{argmin}_{\boldsymbol{\theta}_g} \text{argmax}_{\boldsymbol{\theta}_d} \mathbb{E}_{\mathbf{x}}[\log(D(\mathbf{x}; \boldsymbol{\theta}_d))] \\ + \mathbb{E}_{\mathbf{z}}[\log(1 - D(G(\mathbf{z}; \boldsymbol{\theta}_g); \boldsymbol{\theta}_d))], \tag{1}$$

where $\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})$, $\mathbf{x} \sim p_{\mathbf{x}}(\mathbf{x})$, and $G(\mathbf{z}) \sim p_g(\mathbf{x})$. To perform anomaly detection, we need to change (1) and create a discriminator that can distinguish normal from abnormal. Yet, this implies the ability to have learned all underlying
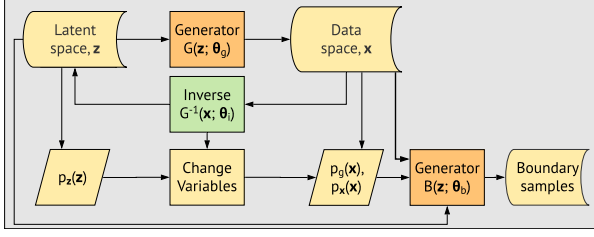
Fig. 1. Flowchart of BDSG to learn the mapping $B : \mathbf{z} \to \mathbf{x}$.

modes and have covered the full support of the distribution from limited data. Unfortunately, GANs tend to learn the mass of the underlying multimodal distribution well, focusing less towards the low probability regions, i.e. the tails, and have discernible problems with mode collapse [20, 19].

MinLGAN uses minimum likelihood regularization to generate data on the tail of the normal data distribution [12]. FenceGAN performs both sample generation on the boundary and anomaly detection using the generator and discriminator, respectively [13]. The generator loss is reinforced with bespoke losses to help model the boundary and the output of the discriminator is used as an anomaly threshold. However, FenceGAN does not succeed to form multimodal supports and to detect anomalies near discontinuous boundaries.

3. THE PROPOSED BDSG MODEL

We propose the BDSG to detect strong anomalies which are near the boundary of the normal data distribution. The BDSG flowchart is shown in Fig. 1. The premise of our approach is to use two generators: G models data of the distribution and B models data that lie close to the support boundary of the distribution. Specifically, we first train an invertible generator, $G(\mathbf{z})$, in the form of IResNet [14] and ResFlow [22], $G(\mathbf{z}) \sim p_g(\mathbf{x})$. $\mathbf{z}$ follows a standard Gaussian distribution, $\mathbf{z} \sim N(\mathbf{0}, \mathbf{I})$, and the mapping from the latent space, $\mathbf{z} \in \mathbb{R}^d$, to the data space, $\mathbf{x} \in \mathbb{R}^d$, is given by $G(\mathbf{z})$. The inverse is given by $G^{-1}(\mathbf{x})$. The second step is to train a generator, $B(\mathbf{z})$, to perform sample generation on the support boundary of the data distribution, learning the mapping $B : \mathbf{z} \to \mathbf{x}$.

We now formulate the BDSG loss function. The first term, $L_0(\boldsymbol{\theta}_b, \mathbf{z})$, guides to find the boundary, while the second term, $L_1(\boldsymbol{\theta}_b, \mathbf{z})$, penalizes deviations from the "normal class" using the distance from a point to a set. The third term, $L_2(\boldsymbol{\theta}_b, \mathbf{z})$, is for the scattering of the $B(\mathbf{z})$ samples in the $\mathbf{x}$ space. $L_2(\boldsymbol{\theta}_b, \mathbf{z})$ is for dispersion and diversity and is the ratio of distances in the $\mathbf{z}$ and $\mathbf{x}$ spaces. With $L_2(\boldsymbol{\theta}_b, \mathbf{z})$, BDSG addresses the mode collapse problem. The loss function for $B(\mathbf{z}; \boldsymbol{\theta}_b)$ is

$$\operatorname{argmin}_{\boldsymbol{\theta}_b} L(\boldsymbol{\theta}_b, \mathbf{z}, \mathbf{x}, G, \lambda_1, \lambda_2), \tag{2}$$

where the loss, $L(\boldsymbol{\theta}_b, \mathbf{z}, \mathbf{x}, G, \lambda_1, \lambda_2)$, is given by

$$L_0(\boldsymbol{\theta}_b, \mathbf{z}, G) + \lambda_1 L_1(\boldsymbol{\theta}_b, \mathbf{z}, \mathbf{x}) + \lambda_2 L_2(\boldsymbol{\theta}_b, \mathbf{z}) \tag{3}$$

$$= \frac{1}{N} \sum_{i=1}^{N} \left[p_g(B(\mathbf{z}_i; \boldsymbol{\theta}_b)) + \lambda_1 \min_{j=1}^{M} ||B(\mathbf{z}_i; \boldsymbol{\theta}_b) - \mathbf{x}_j||_2 \right.$$
$$\left. + \lambda_2 \frac{1}{N-1} \sum_{j=1, j \neq i}^{N} \frac{||\mathbf{z}_i - \mathbf{z}_j||_2}{||B(\mathbf{z}_i; \boldsymbol{\theta}_b) - B(\mathbf{z}_j; \boldsymbol{\theta}_b)||_2} \right], \tag{4}$$

where λ_1 and λ_2 are hyper-parameters of the BDSG. In (3) and (4), the first term, $L_0(\boldsymbol{\theta}_b, \mathbf{z}, G)$, is given by

$$\frac{1}{N} \sum_{i=1}^{N} \left[p_{\mathbf{z}}(G^{-1}(B(\mathbf{z}_i; \boldsymbol{\theta}_b))) |\det \mathbf{J}_G(B(\mathbf{z}_i; \boldsymbol{\theta}_b))|^{-1} \right] \tag{5}$$

$$= \frac{1}{N} \sum_{i=1}^{N} \left[\exp(\log(p_{\mathbf{z}}(G^{-1}(B(\mathbf{z}_i; \boldsymbol{\theta}_b))))) \right.$$
$$\left. - \log(|\det \mathbf{J}_G(B(\mathbf{z}_i; \boldsymbol{\theta}_b))|)) \right], \tag{6}$$

where $\log(p_{\mathbf{z}}(G^{-1}(B(\mathbf{z}_i))))$ and $\log(|\det \mathbf{J}_G(B(\mathbf{z}_i))|)$ are estimated by an invertible model. The $\boldsymbol{\theta}_b$ parameters are obtained by running Gradient Descent on $L(\boldsymbol{\theta}_b, \mathbf{z}, \mathbf{x})$, which can decrease to zero and is written in terms of the sample size, M, and the batch size, $N \leq M$. In the loss in (4), the effective dimensionality of $\mathbf{z} \sim \mathrm{N}(\mathbf{0}, \mathbf{I})$ is lower than that of $\mathbf{x}$.

### 3.1. BDSG Benefits in Sampling Complexity, Anomaly Detection, and Generation of Strong Anomalies

The Sampling Complexity Problem: To perform anomaly detection, FenceGAN estimates $p(\mathbf{x}|D(\mathbf{x}) < \delta)$. This is difficult due to the rarity problem since at least δ^{-2} points are needed on the tail of the distribution. Sampling from a distribution could fail to have even a single point in low probability regions [23, 24]. However, the FenceGAN loss does not succeed to generate a discrete boundary around multimodal distributions separately because it is based on the parallel simultaneous estimation of the density and of sample generation on the boundary. In contrast, the proposed BDSG obviates the rarity problem achieving better sampling complexity.

Anomaly Detection: During inference, a test sample, $\mathbf{x}^*$, is anomalous if $p_g(\mathbf{x}^*) = 0$ and normal otherwise. In practice, a threshold, ϵ, is used instead of 0. The first term of the loss in (4) discriminates between normal and abnormal data.

Generating Strong Anomalies: The BDSG can generate samples lying on the tail of the data distribution, i.e. strong anomalies. First, the boundary generator, $B(\mathbf{z})$, generates $Q \gg N$ samples. Then, the probability of each of these boundary samples is computed and $L_0(\boldsymbol{\theta}_b, \mathbf{z}, G)$ in (4), and if $p_g(B(\mathbf{z})) < \epsilon$, then $B(\mathbf{z})$ is a strong anomalous sample.

4. EVALUATION OF THE BDSG

We evaluate BDSG on synthetic and image data considering several criteria that measure its ability to approximate the boundary and detect anomalies. We evaluate the BDSG for
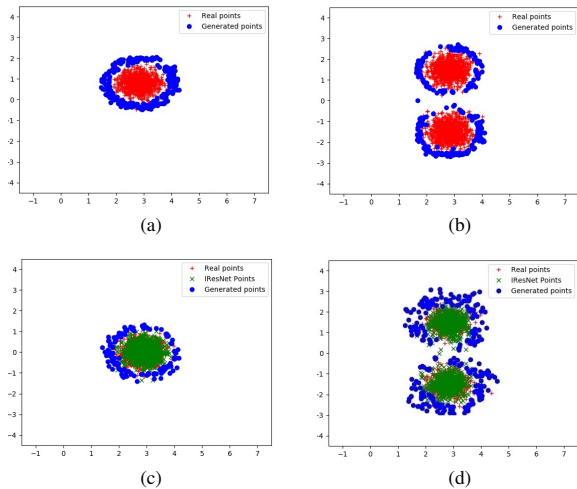
Fig. 2. (a,b) CFS BDSG for uni and multimodal distributions. (c,d) IResNet BDSG. The red points are samples, the green points IResNet samples, and the blue points BDSG samples.

anomaly detection using the Area Under the Receiver Operating Characteristics Curve (AUROC) and the Area Under the Precision-Recall Curve (AUPRC). Using the leave-one-out methodology, we compare the BDSG with the state-of-the-art models of GANomaly, AnoGAN, MinLGAN, and FenceGAN on MNIST, CIFAR-10, and other datasets for OoD.

Setup: *Synthetic data:* We test BDSG using two experimental setups using the multivariate Gaussian distribution, where we know the closed-form of the underlying probability density function. The first setup uses a closed-form solution (CFS) evaluation of $L_0(\boldsymbol{\theta}_b, \mathbf{z}, G)$ model distribution, in lieu of $p_g(\mathbf{x})$. The second setup uses $p_g(\mathbf{x})$ from IResNet [14].

Benchmark data: We also evaluate the BDSG on MNIST by first training an invertible generator, ResFlow, for density estimation. We then train the BDSG using a convolutional neural network (CNN), applying (4). Then, we evaluate the performance of the BDSG on CIFAR-10. Further, we evaluate the performance of the BDSG trained on MNIST and CIFAR-10 and tested on OoD data using the algorithm convergence criteria of the proposed loss and its second term, L_1 [21].

Models: We use a fully-connected $B(\mathbf{z})$ model for synthetic data and CNN and batch normalization for images.

4.1. B(z) Model Architecture for Synthetic Data

CFS BDSG Model: Based on sensitivity analyses, we use dense fully-connected layers for $B(\mathbf{z})$, $M = 1024$, $N = 256$, $\lambda_1 = 0.3$, and $\lambda_2 = 0.025$. The sample size, M, affects the BDSG performance. The batch size, N, affects the convergence speed and can lead to a thinner boundary. Figure 2(a) shows the boundary formed using the CFS BDSG for a unimode distribution. The red points are from the normal data

distribution; the blue $B(\mathbf{z})$ points are on the estimated boundary. The 2-8-8-2 model for $B(\mathbf{z})$ achieves a low loss function value and converges the samples to the boundary. For a bimodal distribution in Fig. 2(b), a 2-8-8-8-2 network leads to low loss values and accurate boundary formation. The average probability of the $B(\mathbf{z})$ points, which are on the boundary, is $L_0(\boldsymbol{\theta}_b, \mathbf{z}, G) = 0.007$ in (3). We obtain descending loss values, successfully converging $B(\mathbf{z})$ to the boundary.

IResNet-Based BDSG: To show that BDSG yields competitive performance on synthetic data from multimodal distributions, we also perform a second experiment. We train our chosen invertible model, IResNet, and use the estimated density to create the boundary. If $p_g(\mathbf{x})$ is estimated correctly, then BDSG estimates the boundary of $p_\mathbf{x}(\mathbf{x})$. In Fig. 2(c), we use a 2-8-8-8-2 network for $B(\mathbf{z})$ for the unimode distribution, $M = 1024$, $N = 128$, $\lambda_1 = 0.3$, and $\lambda_2 = 0.025$.

For the bimodal distribution in Fig. 2(d), we use a deeper architecture for $B(\mathbf{z})$, $N = 256$, and $\lambda_2 = 0.25$. An ablation study found that $L_2(\boldsymbol{\theta}_b, \mathbf{z})$ in (4) is necessary, and otherwise mode collapse is encountered. In Fig. 2(d), for evaluation, we also use the boundary clustering algorithm given by

$$\operatorname{argmin}_{i=1,...,K} \min_{j=1,...,L} ||B(\mathbf{z}) - \mathbf{x}_{i,j}||_2, \qquad (7)$$

where $K = 2$ clusters from the bimodal distribution, $L = 0.5M$ samples from each mode, and $\mathbf{x}_{i,j} \in \mathbb{R}^d$ is the j-th sample of mode i. Here, $\min_{j=1,...,L} ||B(\mathbf{z}) - \mathbf{x}_{i,j}||_2$ is negligible, smaller than the distance from a mode/set to a set.

Figures 2(b) and (d) show that BDSG achieves successful boundary formation and stable convergence without mode collapse. BDSG is compared to FenceGAN and FenceGAN yields incomplete boundary formation between the modes.

4.2. Binary Classification and Boundary Precision

We create a grid of equidistant points in the 2D space and associate each grid point with a probability using the distribution in Fig. 2(d). Using a threshold, ϵ, to detect anomalies, we evaluate the inference performance of L_0 in (4) by computing binary classification metrics. To examine the influence of the choice of ϵ, we compute precision, recall, F1 score, and accuracy, and these scores are higher than 0.99 for $\epsilon \geq 0.5\%$. To examine how accurate we estimate the boundary and to compare with IResNet, we define two Boundary Precision (BP) scores. By analogy with precision, BP1 is the percentage of $B(\mathbf{z})$-points that satisfy $p_\mathbf{x}(B(\mathbf{z})) \in [\gamma, \epsilon]$. BP2 is defined as the intersection of the grid points with IResNet. BP1 is always higher than BP2: BP1 = 80%, BP2 = 68% when $\epsilon = 1\%$.

4.3. Evaluation of the BDSG on Image Data

MNIST: *Setup:* We train ResFlow until convergence on MNIST using the leave-one-out evaluation where the anomaly class is the leave-out digit and the normal class is the remaining digits. We then train the BDSG using a CNN with batch
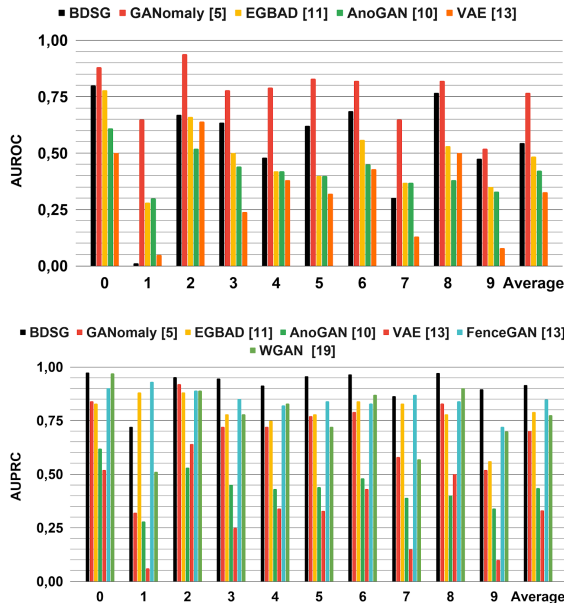
Fig. 3. AUROC and AUPRC evaluation on MNIST data.

MNIST	Loss	L_1	**CIFAR-10**	Loss	L_1
Digits 1-9	0.74	0.93	CIFAR-10	3.16	8.94
Digit 0	20.36	66.32	CIFAR-100	7.50	23.43
Fashion-MNIST	9.92	31.44	SVHN	7.18	22.36
KMNIST	9.28	29.37	STL-10	10.01	31.75

Table 1. Evaluation of BDSG comparing normality (MNIST Digits 1-9 and CIFAR-10) with the abnormal class and the anomaly cases (MNIST Digit 0, Fashion-MNIST, KMNIST and CIFAR-100, SVHN, STL-10), using L and L_1 in (4).
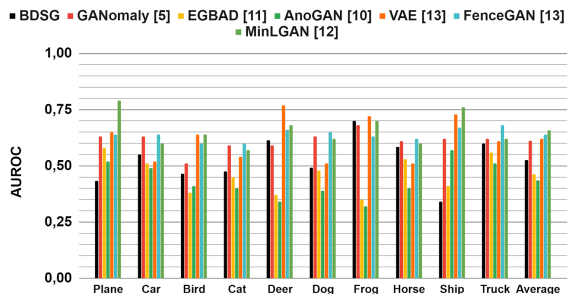

Fig. 4. AUROC evaluation of BDSG on CIFAR-10 data.

normalization, using (4). We also examine different models such as feed-forward and residual. For M, we use the entire training set and we also examine different values for N in (4). After convergence, the loss is 0.7, $L_0(\boldsymbol{\theta}_b, \mathbf{z}, G) = 0$, and $L_1(\boldsymbol{\theta}_b, \mathbf{z}, \mathbf{x}) = 0.8$. This $L_1(\boldsymbol{\theta}_b, \mathbf{z}, \mathbf{x})$, which is the distance from a point to a set, is smaller than the minimum set distance of every pair of MNIST digits which is approximately 10. For evaluation, we compare the proposed BDSG with state-of-the-art models using AUROC and AUPRC as they are commonly used evaluation criteria in the literature [5].

Findings: Figure 3 shows that BDSG achieves competitive performance compared to the alternative techniques and on average and for most digits, BDSG outperforms EGBAD, AnoGAN, and VAE in AUROC and GANomaly, EGBAD, AnoGAN, VAE, FenceGAN, and WGAN in AUPRC.

Going beyond the leave-one-out setting, we assess how BDSG performs when other OoD data are used as anomaly samples considering MNIST as normal and Fashion-MNIST and KMNIST as OoD abnormal [1]. We report results in Table 1 using algorithm convergence criteria, the proposed loss and L_1. The loss and L_1 are lower for the normal class, digits 1 to 9, than for the anomaly class, digit 0, and the abnormal OoD data indicating that the proposed loss and its first term can be used for anomaly detection with a threshold of 1.

CIFAR-10: *Setup:* We train ResFlow and IResNet for density estimation on CIFAR-10 [15]. Next, we train BDSG using a CNN with batch normalization and applying (4).

Findings: Figure 4 presents the AUROC for each CIFAR-10 class. On a leave-one-out evaluation, the BDSG outperforms on average EGBAD and AnoGAN. We demonstrate

the efficacy of the proposed BDSG model which achieves competitive performance in AUROC compared to EGBAD, AnoGAN, and VAE. Table 1 presents the performance evaluation of the BDSG to detect abnormal OoD data from CIFAR-100, SVHN, and STL-10 using the algorithm criteria of the loss and $L_1(\boldsymbol{\theta}_b, \mathbf{z}, \mathbf{x})$. Both L and L_1 in (4) are high for the anomaly cases deviating from normality, indicating that an anomaly detection threshold can be imposed on either the proposed cost or its second term, e.g. 4 on L and 9 on L_1.

5. CONCLUSION

For anomaly detection, the accurate determination of the support boundary is critical and in this paper, we present the BDSG which uses the loss in (4) and leverages reversibility to compute the probability at any point in $\mathbf{x}$. It addresses the rarity problem and the detection of strong anomalies, and maps from $\mathbf{z}$ to $\mathbf{x}$ concentrating the images of $\mathbf{z}$ on the boundary. Using invertible models has advantages in improving the anomaly detection methodology by allowing to devise a generator for creating boundary samples. The BDSG performs sample generation on the boundary, addresses mode collapse, and achieves competitive performance on synthetic data from multimodal distributions and on MNIST and CIFAR-10.

6. ACKNOWLEDGMENT

7. REFERENCES

[1] E. Nalisnick, A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan. Do deep generative models know what they don't know?. In Proc. International Conference on Learning Representations (ICLR), May 2019.

[2] D. Hendrycks, M. Mazeika, and T. Dietterich. Deep Anomaly Detection with Outlier Exposure. In Proc. International Conference on Learning Representations (ICLR), May 2019.

[3] H. Choi, E. Jang, and A. A. Alemi. WAIC, but Why? Generative Ensembles for Robust Anomaly Detection. *arXiv preprint, arXiv:1810.01392v3*, Febr. 2019.

[4] L. Deecke, R. Vandermeulen, L. Ruff, S. Mandt, and M. Kloft. Image Anomaly Detection with Generative Adversarial Networks. In Machine Learning and Knowledge Discovery in Databases. Lecture Notes in Computer Science, vol. 11051. Springer, 2019.

[5] S. Akçay, A. Atapour-Abarghouei, and T. Breckon. GANomaly: Semi-Supervised Anomaly Detection. *arXiv preprint, arXiv:1805.06725v3 [cs.CV]*, Nov. 2018.

[6] Samet Akçay, Amir Atapour-Abarghouei, and Toby P. Breckon. Skip-GANomaly: Skip Connected and Adversarially Trained Encoder-Decoder Anomaly Detection. *arXiv preprint, arXiv:1901.08954 [cs.CV]*, Jan. 2019.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. In Proc. Advances in neural information processing systems (NIPS), 2014.

[8] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs created equal? A study. *arXiv preprint, arXiv:1711.10337*, 2017.

[9] Nazly Rocio, Santos Buitrago, Loek Tonnaer, Vlado Menkovski, and Dimitrios Mavroeidis. Anomaly Detection for imbalanced datasets with Deep Generative Models. *arXiv preprint, arXiv:1811.00986*, 2018.

[10] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs. Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. In Proc. Information Processing in Medical Imaging. Lecture Notes, vol. 10265. Springer, 2017.

[11] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. Ramaseshan Chandrasekhar. Efficient GAN-Based Anomaly Detection. Workshop track, International Conference on Learning Representations (ICLR), 2018.

[12] Chu Wang, Yan-Ming Zhang, and Cheng-Lin Liu. Anomaly Detection via Minimum Likelihood Generative Adversarial Networks. In Proc. 24th International Conference on Pattern Recognition (ICPR), 2018.

[13] C. P. Ngo, A. A. Winarto, C. K. K. Li, S. Park, F. Akram, and H. K. Lee. Fence GAN: Towards Better Anomaly Detection. *arXiv preprint, arXiv:1904.01209*, 2019.

[14] J. Behrmann, W. Grathwohl, R. T. Q. Chen, D. Duvenaud, and J.-H. Jacobsen. Invertible Residual Networks. In Proc. 36th International Conference on Machine Learning (ICML), pp. 573-582, 2019.

[15] R. T. Q. Chen, J. Behrmann, D. Duvenaud, and J.-H. Jacobsen. Residual Flows for Invertible Generative Modeling. In Proc. Advances in Neural Information Processing Systems (NIPS), Dec. 2019.

[16] D. Gongy, L. Liuy, V. Lez, B. Sahaz, M. Reda Mansourx, S. Venkateshz, and A. van den Hengel. Memorizing Normality to Detect Anomaly: Memory-augmented Deep Autoencoder for Unsupervised Anomaly Detection. *arXiv preprint, arXiv:1904.02639v1*, April 2019.

[17] D. T. Nguyen, Z. Lou, M. Klar, and T. Brox. Anomaly Detection With Multiple-Hypotheses Predictions. *arXiv preprint, arXiv:1810.13292v4 [cs.CV]*, Jan. 2019.

[18] Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the Support of a High-Dimensional Distribution. In Proc. Neural Computation, vol. 13, pp. 1443-1471, 2001.

[19] Ilyass Haloui, Jayant Sen Gupta, and Vincent Feuillard. Anomaly detection with Wasserstein GAN. *arXiv preprint, arXiv:1812.02463v2 [stat.ML]*, Dec. 2018.

[20] Hao Ge, Yin Xia, Xu Chen, Randall Berry, and Ying Wu. Fictitious GAN: Training GANs with Historical Models. In Proc. Computer Vision Foundation (CVF) ECCV, Springer, Sept. 2018.

[21] Dan Hendrycks and Kevin Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. *arXiv preprint, arXiv:1610.02136v1 [cs.NE]*, 2016.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770-778, 2016.

[23] Sham Kakade. On the Sample Complexity of Reinforcement Learning. PhD Thesis, University College London: Gatsby Computational Neuroscience Unit, 2003.

[24] R. Devon Hjelm, Athul Paul Jacob, Tong Che, Adam Trischler, Kyunghyun Cho, and Yoshua Bengio. Boundary-Seeking Generative Adversarial Networks. *arXiv preprint, arXiv:1702.08431v4 [stat.ML]*, Febr. 2018.