# Contrastive Generative Adversarial Network for Anomaly Detection

**Laya Rafiee Sevyeri** laya.rafiee@gmail.com
**Thomas Fevens** fevens@cs.concordia.ca
Gina Cody School of Engineering and Computer Science
Concordia University
Montréal, QC, Canada

## Abstract

Anomaly detection (AD) is a fundamental challenge in machine learning that finds samples that do not belong to the distribution of the training data. Recently self-supervised learning approaches and, in particular, contrastive learning show promising results in various machine vision applications mitigating the hunger of traditional supervised deep learning approaches for an enormous amount of labeled data. In this work, we adopt the idea of contrastive learning for reconstruction-based anomaly detection models. Our contrastive learning approach contrasts the sample with local feature maps of itself instead of contrasting a given sample with other instances as in conventional contrastive learning approaches. Our anomaly detection model based on contrastive generative adversarial network, AD-CGAN, is shown to obtain state-of-the-art performance in multiple benchmark datasets. AD-CGAN outperforms the existing reconstruction-based approaches by more than $15\%$ ROC-AUC in several benchmark experiments.

## 1 Introduction

Anomaly detection (AD), also known as outlier/out-of-distribution detection, has a long history in artificial intelligence. Anomaly detection refers to identifying those samples that do not come from the expected distribution. Supervised learning models address anomaly detection using classification approaches such as outlier exposure [4]. On the other hand, unsupervised learning approaches, such as reconstruction-based methods [13, 17], mitigate the problem of limited labeled data and unknown anomalies. In these approaches, the model learns the distribution of the normal data and then a reconstruction loss targets anomalies. AnoGAN [13] proposes using generative adversarial networks (GANs) to find anomalies in the medical domain. They introduce a new anomaly score based on the distance of the reconstructed sample and the sample itself to find anomalies. AnoGAN suffers from its long inference procedure to find the inverse mapping of an image in low-dimensional representations. Besides, the intrinsic problems of GANs such as mode collapse [5], catastrophic forgetting [7, 1], unstable training, and difficulty in convergence of GAN [11] limit the ability of the model to learn a suitable representation for the task of AD. Although recent studies tried to overcome the limitations of AnoGAN [12, 15, 16], they achieved little performance gain. Lee et al. [10] showed that maximizing mutual information on the discriminator side and contrastive learning on the generator side in training GANs increases the quality of generated images by simultaneously mitigating catastrophic forgetting and mode collapse of the discriminator and generator respectively.

In this work, we propose a reconstruction-based Anomaly Detection approach using Contrastive Generative Adversarial Network (AD-CGAN). The proposed model contains three main sub-modules: a contrastive GAN, an autoencoder, and a discriminator (different from the discriminator in GAN) on the latent representations. We train all modules simultaneously on the normal (in-distribution) data

Bayesian Deep Learning workshop, NeurIPS 2021.

to learn a discriminative representation for each image while keeping each image's local and global features as close as possible. The other discriminator trains on the hidden representations of two different reconstruction-based models, i.e., GAN and autoencoder, to provide more discriminative representations. We show that having a contrastive GAN while maximizing the mutual information between local and global features of an image provides more semantic and discriminative features for anomaly detection. The representations obtained by a contrastive GAN in our anomaly detection model can greatly increase the performance of reconstruction-based anomaly detection approaches. Our work is the first to investigate using contrastive generative adversarial networks for anomaly detection to the best of our knowledge.

## 2 Anomaly Detection

Reconstruction-based models are unsupervised approaches that rely on the reconstruction loss of samples, where a higher loss implies an anomalous sample. We consider the idea of mutually informative contrastive GAN [10] in our AD model to detect anomalous samples in images. This model benefits from contrastive learning while maximizing the similarity of a single image's local and global features using mutual information. Our anomaly detection model classifies anomalies with their higher reconstruction loss.

AD-CGAN contains a contrastive GAN, an autoencoder, and a discriminator on the latent representations which train simultaneously to learn the true distribution of normal training data, which will be explained in Section 2.1. We introduce a new normality score using the representations obtained by the trained AD-CGAN to separate normal and anomalous samples in Section 2.2.

### 2.1 AD-CGAN

Our Anomaly Detection model of the Contrastive Generative Adversarial Network, AD-CGAN, uses two discriminators to obtain more discriminative representations of normal (in-distribution) samples–a discriminator in the contrastive GAN ($D_{cgan}$) and a discriminator on latent representations of an input image and the input random noise to the GAN ($D_z$) (Appendix A.1). AD-CGAN trains on the training set $\mathcal{D}_{train} = \{x_1, x_2, ..., x_k \sim P_{ind}\}$ which contains samples drawn from $P_{ind}$, normal distribution (in-distribution). To evaluate our model we use a test set $\mathcal{D}_{test} = \{\bar{x_1}, \bar{x_2}, ..., \bar{x_n} \sim P_{ind} \cup P_{ood}\}$ including both normal and anomalous samples drawn from normal and anomalous distribution ($P_{ood}$) respectively. AD-CGAN includes a contrastive GAN which we refer to as $CGAN$ and an auxiliary autoencoder, $AE$. $AE$ trains with the mean squared error (MSE) reconstruction loss function, $L_{AE} = \|x - G(E(x))\|^2$ where the decoder of $AE$ and the generator of CGAN shares their weights and $G(E(x))$ is the output of $AE$. $CGAN$ contains a generator $G$ and a discriminator $D_{cgan}$. Training the $CGAN$ incorporates two losses: a contrastive loss, $L_{cgan}$, and an adversarial loss, $L_{adv}$. Given an image $x \in X$, we consider the penultimate and ultimate representations of $D_{cgan}$ as local ($C_\psi(x)$) and global ($E_\psi(x)$) features of $x$. Then $\phi_\theta(E_\psi(x))$ and $C_\psi(x)$ go to the contrastive pairing phase to create positive/negative sets for the contrastive learning. In conventional contrastive learning, a given image $x$ is contrasted with other samples, while in AD-CGAN, each image is contrasted with its own local feature maps to create positive/negative sets. For a given image $x$, the set of positive samples is the pair $(C_\psi^{(i)}(x), \phi_\theta(E_\psi(x)))$ for $i \in A = \{0, 1, ..., M^2 - 1\}$ of a $M \times M$ local feature map, whereas the negative samples are defined as the pairs $(C_\psi^{(j)}(x), \phi_\theta(E_\psi(x)))$ $j \in A,\ j \neq i$. The contrastive loss of AD-CGAN follows the loss presented in [10], with a slight modification to fit the architectural design of our model (Eq. 1).

$$
\begin{aligned}
L_{cgan}(X) &= -\mathbb{E}_{(x \in X)} \mathbb{E}_{(i \in A)} [log\ p(C_\psi^{(i)}(x), E_\psi(x) | X)] \\
&= -\mathbb{E}_{(x \in X)} \mathbb{E}_{(i \in A)} [log \frac{exp(g_\theta(C_\psi^{(i)}(x), E_\psi(x)))}{\sum_{(x\prime, i) \in X \times A} exp(g_\theta(C_\psi^{(i)}(x\prime), E_\psi(x)))}]
\end{aligned} \tag{1}
$$

The function $g_\theta(C_\psi^{(i)}(x), E_\psi(x)) = C_\psi^{(i)}(x)^T \phi_\theta(E_\psi(x))$ maps the local/global features with $K$ dimensions to a scalar score. We used relativistic loss [6] as $L_{adv}$. In order to stabilize training, we constrained the discriminator $D_{cgan}$ and the generator ($G$) to learn only from the contrastive loss of real image and fake image features, respectively, as suggested in [10]. The final loss of the generator

and the discriminator of our $CGAN$ is a combination of its adversarial and contrastive losses:

$$L_G = L_{adv_G} + \alpha L_{cgan}(X_f), \quad L_{Dcgan} = L_{adv_D} + \beta L_{cgan}(X_r) \tag{2}$$

We further regularized the encoded latent space of encoder, $E(x)$, and random noise, $z$, with an additional adversarially learned discriminator $D_z$, called $L_{dz}$ (Eq. 3).

$$L_{dz} = \mathbb{E}_{z \sim P_z}[logD_z(z,z)] + \mathbb{E}_{z \sim P_z,\ x \in X}[1 - logD_z(z, E(x))] \tag{3}$$

### 2.2 Normality Score

The trained model evaluates on the $\mathcal{D}_{test}$ to identify anomalous samples. The normality score presented here encompasses two reconstruction losses: the generation reconstruction loss, $L_{Gr}$, which involves the scores obtained from the trained $CGAN$, and the feature reconstruction loss, $L_{Fr}$, which incorporate the scores obtained from latent representations of a given image. The normality score for a given image $x \in \mathcal{D}_{test}$ is $NS(x) = L_{Gr} + L_{Fr}$ where the generation and feature reconstruction losses are defined as

$$L_{Gr} = \lambda L_{GD}(x) + (1-\lambda) L_{GR}(x), \quad L_{Fr} = \rho L_{FE}(x) + (1-\rho) L_{FL}(z) \tag{4}$$

Here, $L_{Gr}$ includes discrimination loss, $L_{GD}(x) = \sum |f_D(x) - f_D(G(E(x)))|$ with intermediate representation of a given image $x$ from $D_{cgan}$ as $f_D(x)$, and residual loss, $L_{GR}(x) = \sum |x - G(E(x))|$, similar to the loss presented in [13]. It should be noted that $f_D(x)$ here refers to the internal representation of image $x$ obtained from the penultimate layer of the $D_{cgan}$. $L_{Fr}$ contains encoded, $L_{FE}$, and latent, $L_{FL}$, feature reconstruction losses (Eq. 5).

$$\begin{aligned} L_{FE}(x) &= \sum |E(x) - E(G(E(x)))| \\ L_{FL}(z) &= \|D_z(E(G(z)), z) - D_z(E(x), E(G(z)))\|_1 \end{aligned} \tag{5}$$

where $E(x)$ is the encoded representation of $x$ from the encoder $E$, $z$ is the input random noise to the generator $G$ of CGAN, and $G(z)$ is the generated output of $G$.

| | all-vs-one | | | | one-vs-all | | | |
|---|---|---|---|---|---|---|---|---|
| Datasets | ALAD | ADGAN | AE-GAN | AD-CGAN | ALAD | ADGAN | AE-GAN | AD-CGAN |
| CIFAR-10 | 51.1± 0.08† | 48.3† | 55.5 | **84.5**± 0.05 | 60.7 | 60.6 | 61.1 | **86.0**± 0.04 |
| fMNIST | 50.8± 0.12† | 54.0† | 52.2± 0.18† | **87.7**± 0.04 | 78.1± 0.12† | 75.4† | 69.0± 0.16† | **93.9**± 0.02 |
| MNIST | 53.9± 0.13† | 62.5† | 55.7 | **85.0**± 0.05 | 62.4± 0.09† | 91.5 | 69.3 | **92.3**± 0.03 |
| CatsVsDogs | - | - | - | - | 53.4† | 49.0† | 51.6† | **89.8**± 0.04 |

Table 1: ROC-AUC (%) comparison of reconstruction-based models on all four datasets with *one-vs-all* and *all-vs-one* schemes. The symbol † represents results reported from our implementations. All of the results from our implementations are averaged over three different runs. We use 0.3 for $\alpha$ and $\beta$ for the losses of the generator and discriminator, respectively, of our CGAN. $\lambda = 0.1$ and $\rho = 0.5$ are used for all the experiments.

## 3 Experiments and Results

We considered four benchmark datasets in our experiments: CIFAR-10 [8], FashionMNIST (fMNIST) [14], MNIST [9], and CatsVsDogs [3]. All of the datasets except CatsVsDogs include 10 classes. In order to evaluate the model on AD tasks, we employ two different schemes. We considered soft (one-vs-all) and hard (all-vs-one) anomaly detection experiments. In one-vs-all, one of the classes is chosen to be normal and the rest forms the anomalous class, and reverse in all-vs-one.

We compare the performance of our model with several state-of-the-art reconstruction-based methods (Table 1). ALAD [16], ADGAN [2], and AE-GAN [12] are the recent AD methods use GANs as their reconstruction models. As illustrated in Table 1, AD-CGAN outperforms all the baselines. The improvement is more notable on CIFAR-10 and FashionMNIST by a minimum of 15% on the soft and 30% on the hard scheme of ROC-AUC. As expected, the performance in the hard scheme is lower compared with the soft scheme since the normal class contains multiple labels, each having a different distribution. This is more notable in FashionMNIST and MNIST datasets with around 7% drop in the performance. We argue that given the similar pattern in several labels of these two

datasets, even AD-CGAN with its discriminative representations obtained by the contrastive loss may have difficulty in the hard scheme. More details on the two proposed soft and hard schemes, the experimental setup, and the results on each individual class are presented in the Appendix A.2, A.3, and A.4.

In order to have a better understanding of the effect of each of the components in the proposed normality score, in various experiments, we measure their effects in different anomaly detection settings (see Appendix A.5).

## References


[1] Ting Chen, Xiaohua Zhai, Marvin Ritter, Mario Lucic, and Neil Houlsby. Self-supervised gans via auxiliary rotation loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12154–12163, 2019.

[2] Lucas Deecke, Robert Vandermeulen, Lukas Ruff, Stephan Mandt, and Marius Kloft. Anomaly detection with generative adversarial networks, 2018. URL `https://openreview.net/forum?id=S1EfylZ0Z`.

[3] Jeremy Elson, John R Douceur, Jon Howell, and Jared Saul. Asirra: a captcha that exploits interest-aligned manual image categorization. *CCS*, 7:366–374, 2007.

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

[5] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[6] Alexia Jolicoeur-Martineau. The relativistic discriminator: a key element missing from standard GAN. In *International Conference on Learning Representations*, 2019.

[7] Ronald Kemker, Marc McClure, Angelina Abitino, Tyler Hayes, and Christopher Kanan. Measuring catastrophic forgetting in neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[8] Alex Krizhevsky. Learning multiple layers of features from tiny images. Master's thesis, Computer Science Department, University of Toronto, 2009.

[9] Yann LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[10] Kwot Sin Lee, Ngoc-Trung Tran, and Ngai-Man Cheung. Infomax-gan: Improved adversarial image generation via information maximization and contrastive learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3942–3952, 2021.

[11] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs created equal? a large-scale study. In *Proc. of NIPS*, pages 700–709, 2018.

[12] Laya Rafiee and Thomas Fevens. Unsupervised anomaly detection with a gan augmented autoencoder. In *International Conference on Artificial Neural Networks*, pages 479–490. Springer, 2020.

[13] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *Proc. of IPMI*, pages 146–157. Springer, 2017.

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

[15] Houssam Zenati, Chuan Sheng Foo, Bruno Lecouat, Gaurav Manek, and Vijay Ramaseshan Chandrasekhar. Efficient gan-based anomaly detection. *arXiv preprint arXiv:1802.06222*, 2018.

[16] Houssam Zenati, Manon Romain, Chuan-Sheng Foo, Bruno Lecouat, and Vijay Chandrasekhar. Adversarially learned anomaly detection. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 727–736. IEEE, 2018.

[17] Chong Zhou and Randy C Paffenroth. Anomaly detection with robust deep autoencoders. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 665–674. ACM, 2017.

# A Appendix

## A.1 AD-CGAN

For the adversarial loss, we used relativistic loss [6] which is shown in Eq. 6.

$$\begin{aligned} L_{adv_D} &= -\mathbb{E}_{(x_r \in X_r,\, x_f \in X_f) \sim (\mathbb{P},\mathbb{Q})}[\log(\sigma(C(x_r) - C(x_f)))] \\ L_{adv_G} &= -\mathbb{E}_{(x_r \in X_r,\, x_f \in X_f) \sim (\mathbb{P},\mathbb{Q})}[\log(\sigma(C(x_f) - C(x_r)))] \end{aligned} \tag{6}$$

where $L_{adv_G}$ and $L_{adv_D}$ are the losses of the generator and the discriminator of the CGAN, $\sigma$ is the sigmoid function, $X_r$ and $X_f$ represent sets of real and fake images respectively, $\mathbb{P}$ is the distribution of the real data, $\mathbb{Q}$ is the distribution of the fake data, and $C$ is the critic.

A schematic view of the proposed AD-CGAN is also presented in Fig. 1.

## A.2 Datasets

In order to evaluate the model on AD tasks, we employ two different schemes. We introduce soft and hard anomaly detection experiments. In the soft experiments, we consider one-vs-all scheme. In this scheme, a dataset with $C$ classes will lead to $C$ different anomaly detection experiments. A given class $1 \leq c_{ind} \leq C$ is considered as the *normal* class, while $c_{ood}$ defines *anomalous* class of the rest of $C-1$ classes. In the hard AD scheme, we introduce all-vs-one scheme. Similar to the soft scheme, each dataset with $C$ classes will lead to $C$ different experiments. However, in contrast with the soft scheme, $1 \leq c_{ood} \leq C$ includes only a single class while $c_{ind}$ contains the remaining $C-1$ classes. The soft and hard schemes were not applicable to the experiments on CatsVsDogs, as it has only two classes (either Cats or Dogs). Hence, each of the two classes was treated as *normal* in a separate experiment.

## A.3 Experimental Setup

The detailed hyperparameters of the AD-CGAN are presented in Table 2. We used the same hyperparameters for MNIST and fashionMNIST with the same image size ($28 \times 28$). All the images of CatsVsDogs are rescaled to $64 \times 64$. For the one-vs-all scheme, we trained the models for 15 epochs with 5 warm-up epochs for the autoencoders for each dataset. The model trained with ADAM optimizer, latent size of 100 for all of the datasets except CatsVsDogs with latent size of 200, batch size of 32, and learning rate of $3 \times 10^{-4}$ and $2 \times 10^{-4}$ for $CGAN$ and $AE$ respectively. For the all-vs-one, we used the same configuration, except that we trained the models on each dataset with 35 epochs plus 15 warm-up epochs for $AE$. All of the experiments were conducted using Python 3.7 with the Pytorch library on a GeForce GTX 1080 Ti GPU.

We use 0.3 for $\alpha$ and $\beta$ for the losses of the generator and discriminator, respectively, of our CGAN. $\lambda = 0.1$ and $\rho = 0.5$ were chosen based on validation sets (10% of the training sets) from each of the datasets.

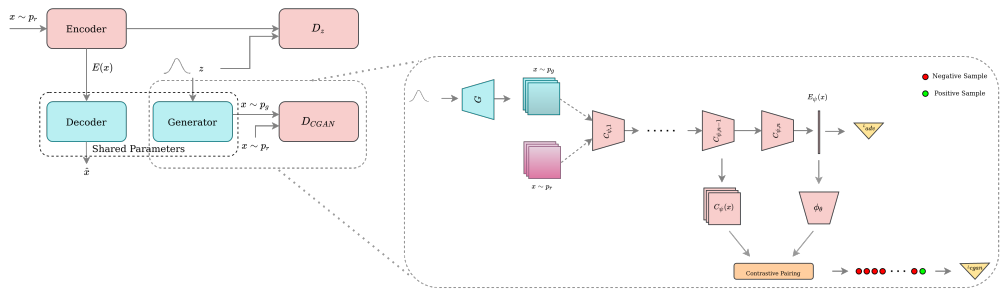


Figure 1: AD-CGAN; different components of our anomaly detection model are depicted on the left. On the right, a zoom-in view of the discriminator of AD-CGAN is shown.

Table 2: Architecture and hyperparameters of AD-CGAN on each dataset; the generator of the GAN and the decoder of autoencoder use weight sharing. *Conv* represents convolutional layers, *Deconv* represents deconvolutional layers, and *Linear* represents dense layers.

| Module | CIFAR10 | | | fMNIST/MNIST | | | CatsVsDogs | | |
|---|---|---|---|---|---|---|---|---|---|
| | Conv | Deconv | Linear | Conv | Deconv | Linear | Conv | Deconv | Linear |
| Encoder ($E$) | 5 | - | - | 5 | - | - | 5 | - | - |
| Generator/Decoder ($G_z/G(E(x))$) | 3 | 4 | - | 2 | 4 | - | 2 | 5 | - |
| Critic ($D_{cgan}$) | 6 | - | 2 | 4 | - | 2 | 6 | - | 2 |
| $D_z$ | - | - | 3 | - | - | 3 | - | - | 3 |

### A.4 Results

The detailed performance of each of the models, in soft and hard schemes, for each of the classes of $c_{ind}$ is presented in Table 3.

### A.5 Ablation Study

AD-CGAN comprises several training components as well as multiple normality score components. We believe that each of the training/normality score components is critical in the models' performance. In order to have a better understanding of the effect of each of the components in the proposed normality score, in various experiments, we measure their effects in different anomaly detection settings (see Table 4). Feature reconstruction loss is added to the normality score to measure how discriminative the latent representations of the two reconstruction models are. Several experiments on MNIST and FashionMNIST on soft and hard AD schemes showed that adding $D_z$ leads to more discriminative latent representation, which affects the normality scores obtained by the feature reconstruction loss. We defined four distinct models of AD-CGAN based on the normality score components they have access: AD-CGAN$_{LG}$ represents AD-CGAN with only generation reconstruction loss; AD-CGAN$_{LFL}$ represents AD-CGAN with only latent feature reconstruction loss, $L_{FL}$; AD-CGAN$_{LFE}$ represents AD-CGAN with only encoded feature reconstruction loss, $L_{FE}$; and AD-CGAN$_{GF}$ contains both $L_{Gr}$ and $L_{Fr}$ in its normality score. It should be mentioned that in each of these models, generation reconstruction loss is considered as part of the normality score.

As the results reveal, removing the feature reconstruction loss (ignoring $D_z$) negatively affects the performance of AD-CGAN. The impact is more severe in the case of the all-vs-one (hard) scheme. This effect can be interpreted by the idea that removing $D_z$ diminishes the power of AD-CGAN to obtain more discriminative representations, which can be more important in the hard scheme. On the other hand, $AD-CGAN_{LFE}$ that trains with $D_z$ with only encoded feature reconstruction loss, improved $AD-CGAN_{LG}$ with a large boost in both datasets. The results on $AD-CGAN_{LFL}$, which substitutes the encoded feature reconstruction loss with the latent feature reconstruction loss, show similar behaviour. However, it is important to note that in all the experiments, ignoring any training and/or normality score component results in lower performance. It also should be considered that the best results are achieved when all the components with the right amount of contributions are considered, as it is shown in $AD-CGAN_{GF}$.

| | MNIST (%) | | fMNIST (%) | |
|---|---|---|---|---|
| model | all-vs-one | one-vs-all | all-vs-one | one-vs-all |
| $AD-CGAN_{LG}$ | 56.8± 0.13 | 72.3± 0.12 | 68.7± 0.11 | 80.8± 0.11 |
| $AD-CGAN_{LFE}$ | 66.6± 0.12 | 84.4± 0.11 | 70.4± 0.13 | 86.1± 0.08 |
| $AD-CGAN_{LFL}$ | 73.2± 0.09 | 84.9± 0.05 | 74.5± 0.08 | 87.0± 0.05 |
| $AD-CGAN_{GF}$ | **85.0**± 0.05 | **92.3**± 0.03 | **87.7**± 0.04 | **93.9**± 0.02 |

Table 4: Ablation studies on MNIST and FashionMNIST given different normality score components of AD-CGAN. We used $\lambda = 0.1$ in cases where the generation reconstruction loss had been used. We set $\rho = 1$ and $\rho = 0$ for $AD-CGAN_{LFE}$ and $AD-CGAN_{LFL}$, respectively. The ROC-AUC (%) results are averaged over three different runs.

| | | all-vs-one | | | | one-vs-all | | | |
|---|---|---|---|---|---|---|---|---|---|
| Datasets | Class | ALAD | ADGAN | AE-GAN | Ours | ALAD | ADGAN | AE-GAN | Ours |
| CIFAR-10 | 0 | 61.2± 0.02 | 44.3 | 63 | **89.8**± 0.12 | 67 | 62.7 | 67 | **83.8**± 0.04 |
| | 1 | 61.1± 0.02 | 39.6 | 63 | **89.5**± 0.13 | 46 | 54.6 | 49 | **87.2**± 0.03 |
| | 2 | 40.7± 0.00 | 58.2 | 60 | **84.7**± 0.06 | 64 | 56.1 | 63 | **80.1**± 0.10 |
| | 3 | 48.8± 0.01 | 44.7 | 54 | **78.6**± 0.09 | 63 | 59.5 | 56 | **86.0**± 0.07 |
| | 4 | 35.5± 0.01 | 66.1 | 35 | **81.7**± 0.14 | 66 | 58.6 | 73 | **85.4**± 0.04 |
| | 5 | 53.5± 0.02 | 44.5 | 52 | **72.8**± 0.01 | 53 | 62.8 | 52 | **81.6**± 0.02 |
| | 6 | 47.8± 0.01 | 61.5 | 60 | **87.6**± 0.06 | 78 | 60.4 | 72 | **94.6**± 0.03 |
| | 7 | 49.7± 0.01 | 47.4 | 51 | **90.7**± 0.03 | 52 | 62.3 | 63 | **81.7**± 0.04 |
| | 8 | 52.9± 0.03 | 45.7 | 54 | **82.6**± 0.02 | 75 | 70.2 | 68 | **89.3**± 0.06 |
| | 9 | 59.4± 0.01 | 31.3 | 63 | **86.9**± 0.02 | 43 | 59.1 | 48 | **90.7**± 0.04 |
| | Average | 51.1± 0.08 | 48.3 | 55.5⋆ | **84.5**± 0.05 | 60.7⋆ | 60.6⋆ | 61.1⋆ | **86.0**± 0.04 |
| fMNIST | 0 | 54.0± 0.02 | 48.4 | 45.3± 0.09 | **89.5**± 0.06 | 79.4± 0.02 | 74.1 | 74.4± 0.03 | **93.3**± 0.04 |
| | 1 | 68.2± 0.04 | 63.7 | 32.8± 0.12 | **85.8**± 0.03 | 94.1± 0.04 | 92.3 | 92.3± 0.01 | **95.9**± 0.04 |
| | 2 | 55.5± 0.03 | 40.4 | 57.9± 0.08 | **90.4**± 0.07 | 60.6± 0.09 | 71.1 | 67.7± 0.03 | **94.0**± 0.06 |
| | 3 | 47.9± 0.04 | 60.5 | 23.0± 0.05 | **81.7**± 0.10 | 79.5± 0.05 | 81.6 | 80.0± 0.03 | **93.7**± 0.03 |
| | 4 | 60.3± 0.13 | 47.8 | 34.9± 0.07 | **81.4**± 0.02 | 76.4± 0.06 | 73.6 | 82.5± 0.01 | **93.1**± 0.02 |
| | 5 | 22.2± 0.01 | 66.9 | 80.4± 0.02 | **93.8**± 0.04 | 85.5± 0.01 | 77.3 | 36.5± 0.06 | **93.5**± 0.02 |
| | 6 | 45.1± 0.01 | 34.5 | 52.1± 0.06 | **92.1**± 0.02 | 61.2± 0.08 | 70.0 | 55.1± 0.04 | **91.7**± 0.03 |
| | 7 | 44.1± 0.05 | 67.1 | 55.5± 0.09 | **87.0**± 0.09 | 94.9± 0.02 | 91.0 | 77.9± 0.07 | **98.5**± 0.01 |
| | 8 | 50.2± 0.09 | 54.1 | 76.0± 0.02 | **83.8**± 0.01 | 62.6± 0.03 | 50.3 | 49.9± 0.06 | **91.6**± 0.08 |
| | 9 | 60.7± 0.07 | 56.6 | 63.6± 0.09 | **91.1**± 0.06 | 86.5± 0.13 | 73.2 | 73.4± 0.13 | **93.7**± 0.07 |
| | Average | 50.8± 0.12 | 54.0 | 52.2± 0.18 | **87.7**± 0.04 | 78.1± 0.12 | 75.4 | 69.0± 0.16 | **93.9**± 0.02 |
| MNIST | 0 | 61.0± 0.05 | 42.7 | 73 | **94.6**± 0.07 | 74.7± 0.12 | 97.2 | 85 | **97.1**± 0.02 |
| | 1 | 87.1± 0.03 | **93.1** | 56 | 85.8± 0.11 | 69.8± 0.16 | **99.7** | 98 | 95.7± 0.02 |
| | 2 | 44.5± 0.04 | 39.7 | 61 | **91.6**± 0.05 | 50.4± 0.09 | 87.4 | 54 | **92.1**± 0.03 |
| | 3 | 47.7± 0.05 | 61.2 | 55 | **86.6**± 0.08 | 65.7± 0.05 | 84.8 | 69 | **91.2**± 0.03 |
| | 4 | 56.7± 0.05 | 70.2 | 49 | **77.4**± 0.04 | 63.6± 0.06 | 91.0 | 72 | **95.5**± 0.01 |
| | 5 | 50.1± 0.06 | 53.1 | 49 | **82.8**± 0.03 | 56.1± 0.04 | **91.6** | 54 | 87.8± 0.02 |
| | 6 | 51.8± 0.11 | 59.8 | 55 | **86.9**± 0.08 | 53.0± 0.08 | **95.7** | 60 | 88.6± 0.06 |
| | 7 | 56.4± 0.09 | 75.2 | 44 | **76.1**± 0.02 | 49.6± 0.01 | **93.7** | 68 | 92.5± 0.01 |
| | 8 | 41.2± 0.08 | 58.5 | 59 | **83.9**± 0.02 | 75.3± 0.07 | 81.6 | 69 | **87.2**± 0.05 |
| | 9 | 42.4± 0.02 | 71.1 | 56 | **84.5**± 0.06 | 65.2± 0.09 | 92.4 | 64 | **95.2**± 0.02 |
| | Average | 53.9± 0.13 | 62.5 | 55.7⋆ | **85.0**± 0.05 | 62.4± 0.09 | 91.5⋆ | 69.3⋆ | **92.3**± 0.03 |
| CatsVsDogs | Cats | - | - | - | - | 52.6 | 53.1 | 51.7 | **92.7**± 0.03 |
| | Dogs | - | - | - | - | 54.1 | 44.9 | 52.1 | **86.9**± 0.05 |
| | Average | - | - | - | - | 53.4 | 49.0 | 51.6 | **89.8**± 0.04 |

Table 3: ROC-AUC (%) comparison of reconstruction-based models on all four datasets with *one-vs-all* and *all-vs-one* schemes. In the *one-vs-all* scheme, the class number defines $c_{ind}$, while in *all-vs-one*, it refers to $c_{ood}$. The results are averaged over three different runs. $\lambda = 0.1$ and $\rho = 0.5$ are used for all the experiments. The symbol $^\star$ represents results reported from the original paper. For simplicity, for each of the classes of CIFAR-10 and FashionMNIST (fMNIST), we use ordinal numbers instead of their label.