Contamination-Resilient Anomaly Detection via Adversarial Learning on Partially-Observed Normal and Anomalous Data

Wenxi Lv[1] Qinliang Su[1 2] Hai Wan[1] Hongteng Xu[3] Wenchao Xu[4]

Abstract

Many existing anomaly detection methods assume the availability of a large-scale normal dataset. But for many applications, limited by resources, removing all anomalous samples from a large unlabeled dataset is unrealistic, resulting in contaminated datasets. To detect anomalies accurately under such scenarios, from the probabilistic perspective, the key question becomes how to learn the normal-data distribution from a contaminated dataset. To this end, we propose to collect two additional small datasets that are comprised of partially-observed normal and anomaly samples, and then use them to help learn the distribution under an adversarial learning scheme. We prove that under some mild conditions, the proposed method is able to learn the correct normal-data distribution. Then, we consider the overfitting issue caused by the small size of the two additional datasets, and a correctness-guaranteed flipping mechanism is further developed to alleviate it. Theoretical results under incomplete observed anomaly types are also presented. Extensive experimental results demonstrate that our method outperforms representative baselines when detecting anomalies under contaminated datasets.

1. Introduction

Anomalies are deemed as the instances that look considerably different from the majority ones. Anomalies in real-world applications are generally very diverse and could even have countless types (Ruff et al., 2021). The task of anomaly detection is to identify the diverse exceptional samples from normal ones. With the capability of discovering various kinds of anomalous samples, anomaly detection has been successfully applied to many different fields like disease diagnosis, industrial defect detection, cyber-crimes defense, financial fraud detection, etc (Chandola et al., 2009). Due to the difficulties of collecting anomalies for their diversity and rarity in practical applications, many existing works are established on the availability of only a clean normal dataset, without needing to access any anomalous samples. Under this assumption, these methods first leverage the normal samples to learn the normality of data and then leverage it to determine whether a testing sample is anomalous. Typical examples include the well-known one-class classifier method (Ruff et al., 2018), reconstruction methods (Akcay et al., 2018; Gong et al., 2019) and the self-supervised methods (Qiu et al., 2021; Shenkar & Wolf, 2022) *etc*.

However, for many application scenarios, assuming the availability of a large normal dataset is also unrealistic. What we often have is a contaminated dataset that is composed of both normal and anomalous samples. Contaminated datasets could arise for many reasons, *e.g.*, no labeling or a coarse screening which only removes some obvious anomalies due to limited labor or expertise resources. To alleviate the detrimental impacts caused by the contamination samples (*i.e.*, anomalies), it is proposed in (Zhou & Paffenroth, 2017; Lai et al., 2020) to combine robust PCA or robust projection with the normal-sample-based methods to improve their contamination robustness. But due to the intricate patterns of anomalies, it is often observed that the improvement brought by these methods is limited. Later, some works proposed to first pseudo-label some easy anomalies in the contaminated dataset and then remove them from the dataset to boost the detection accuracy (Qiu et al., 2022; Kim et al., 2023). However, this type of pseudo-labeling methods heavily rely on the accuracy of pseudo-labels, and the labeling error could be accumulated into the subsequent stages, thereby affecting their overall performance.

Instead of only leveraging the contaminated dataset, some recent methods have proposed to further collect a *small* number of anomalous samples to help recognize the anomalies. But due to the diversity nature of anomalies, the collected

[1] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China. [2] Guangdong Key Laboratory of Big Data Analysis and Processing, Guangzhou, China. [3] Gaoling School of Artifical Intelligence, Renmin University of China, Beijing, China. [4] Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR. Correspondence to: Qinliang Su <suqliang@mail.sysu.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

anomalies may only cover a subset of all possible types, which distinguishes the anomaly detection problem from the classical classification tasks. To make use of the collected anomalies, Deep SAD (Ruff et al., 2020) proposed to learn a hyper-sphere that explicitly forces the collected anomalies to locate outside of it, while FeaWAD (Zhou et al., 2021) proposed to directly increase their anomaly scores. In (Tian et al., 2022), an anomaly-aware generative adversarial network (GAN) is developed, which can explicitly avoid to assign probabilities to the collected anomalies and thus can be used to detect anomalies under contaminated scenarios. Recently, (Pang et al., 2023) proposed to construct three different types of pairs with the contaminated dataset and collected anomalies, and then use the pairs to train an ordinal regression model to output different scores for different types of pairs, with the scores later used for detection of anomalies. Despite remarkable gains have been observed, only one extra dataset (*i.e.*, the anomaly dataset) is leveraged in these methods. But for many applications, in addition to anomalous samples, it is also possible to obtain a small amount of normal samples, *e.g.*, hiring domain experts to identify some normal samples from a large unlabeled dataset. Obviously, it is beneficial to further exploit the additional normal dataset for better anomaly detection. In addition, due to the high cost of collecting normal and anomalous samples, the collected normal and anomaly datasets are often small, probably comprised of tens or hundreds of samples at most. However, existing works rarely take into account the overfitting issue caused by the small size of these datasets.

In this paper, we propose to learn the distribution of normal samples based on the contaminated dataset and two small clean datasets, and then leverage the distribution model to detect anomalies. Specifically, we first develop a GAN that can make use of the three available datasets and prove that it can converge to the distribution of normal samples. Then, a mechanism, which randomly flips the samples between normal and anomaly datasets by some probability, is further developed to prevent the discriminator of GAN from easily overfitting to the small number of normal and anomalous samples. It is proved that even with the introducing of the flipping mechanism, the convergence of the proposed GAN is still guaranteed. In addition, we further show that when the anomaly dataset does not cover all possible anomaly types, even though the proposed GAN cannot recover the distribution of normal samples anymore, it can still mitigate the negative influence from contamination samples of observed types, while strengthening the contributions from normal samples in the contaminated dataset. To detect anomalies efficiently, we extend the proposed GANs to the bidirectional paradigm and then use the combination of the reconstruction error and the norm of latent representations to serve as the final detection criteria. Extensive experimental results on both toy and real-world datasets demonstrate the proposed method can effectively exploit the collected normal and anomalous samples even if their number is small and achieve better performance than comparable baselines under the contamination scenarios.

2. Related Work

Due to the rarity of anomalous data, many existing anomaly detection methods assume that the available dataset only contains normal data. One typical paradigm is to train a deep generative model to reconstruct the normal data and the reconstruction error is used to identify anomalies (Gong et al., 2019; Akcay et al., 2018). Another paradigm is to learn a one-class classification model to describe the normality of the training dataset (Schölkopf et al., 1999; Tax & Duin, 2004; Ruff et al., 2018). Self-supervised and contrastive learning methods are also explored to detect anomalies by predicting the type of data transformation or maximizing the relevance between different segments of normal samples (Qiu et al., 2021; Shenkar & Wolf, 2022; Xu et al., 2023c). However, the assumption of the availability of a large clean normal dataset may not hold in practice and the performance of these methods shows degradation when trained on contaminated datasets.

To mitigate the negative influence of contamination in the training dataset, (Zhou & Paffenroth, 2017; Lai et al., 2020) proposed normality-based methods with robust PCA or robust projection which mitigate the negative impact of contamination, while (Qiu et al., 2022; Kim et al., 2023) try to remove the contamination from training dataset through pseudo-labeling on it first. But all of these methods do not make use of the collected anomalous data. To leverage the collected anomalies, one strategy is to leverage collected anomalies to learn a better description of normality. Deep SAD (Ruff et al., 2020) proposed to learn a hyper-sphere that explicitly forces the collected anomalies driven far away from it. Differently, several recent works have proposed to develop new GANs to better characterize the distribution of normal data by equipping it with the ability of perceiving and leveraging the collected anomalies in (Sinha et al., 2021; Tian et al., 2022; Su et al., 2024). In line with these works, we also notice that beyond the anomaly detection task, there are some efforts that simply attempt to learn a clean distribution from contaminated data by making use of auxiliary datasets (Katz-Samuels et al., 2019; Vandermeulen et al., 2020; Tian et al., 2023). Another strategy to make use of the collected anomalies is to train a scoring network to distinguish normal and anomalous data. FeaWAD (Zhou et al., 2021) proposed to train an anomaly scoring network to directly assign high scores to collected anomalies. PReNet (Pang et al., 2023) trains an ordinal regression model to output different scores for three types of pairs constructed with the contaminated dataset and collected anomalies. Recently, to improve the accuracy of pseudo-labeling, (Li et al., 2023) leverages collected anomalies to guide the model pseudo-labeling on the contaminated training dataset. Despite remarkable gains have been observed, in all of the current methods, only one extra dataset (*i.e.*, the anomaly dataset) is leveraged. For many applications, in addition to anomalous samples, it is also possible to obtain a small amount of normal samples. Obviously, it is beneficial to further exploit the additional normal dataset for better anomaly detection. Moreover, since the size of additionally collected normal and anomaly datasets are very small due to the constraints from limited resources, existing works rarely take the overfitting issue caused by them into account.

3. Method

3.1. Problem Formulation

To describe the problem, we first assume the availability of a contaminated dataset

$$\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots \mathbf{x}_N\},$$

which contains both normal and anomalous samples. We can consider the elements in $\mathcal{X}$ come from the data distribution $p_{data}(\mathbf{x})$, which is a mixture distribution of the form

$$p_{\text{data}}(\mathbf{x}) = \pi p^+(\mathbf{x}) + (1-\pi)p^-(\mathbf{x}), \tag{1}$$

where $p^+(\mathbf{x})$ and $p^-(\mathbf{x})$ denote the distributions of normal and anomalous samples, respectively; and π stands for the proportion of normal samples in the contaminated dataset $\mathcal{X}$. Moreover, we also assume the availability of a pure normal dataset $\mathcal{X}^+$ and a pure anomaly dataset $\mathcal{X}^-$, that is,

$$\mathcal{X}^+ = \{\mathbf{x}_1^+, \mathbf{x}_2^+, \dots \mathbf{x}_M^+\}, \quad \mathcal{X}^- = \{\mathbf{x}_1^-, \mathbf{x}_2^-, \dots \mathbf{x}_K^-\},$$

where the normal samples $\mathbf{x}_m^+$ and anomalous samples $\mathbf{x}_k^-$ are drawn from $p^+(\mathbf{x})$ and $p^-(\mathbf{x})$, respectively. Considering the difficulties of collecting normal and anomalous samples in practice, we generally think that the number of collected normal and anomalous samples M and K is much smaller than N. In this paper, our goal is to first learn the distribution of normal data $p^+(\mathbf{x})$ based on the contaminated dataset $\mathcal{X}$ and the two small clean datasets $\mathcal{X}^+$ and $\mathcal{X}^-$ under the GAN framework, and then leverage the GAN that only models normal samples to detect anomalies.

3.2. Learning Normal-data Distribution from the Contaminated Dataset

It is known that training different GANs amounts to minimizing different types of distribution divergence, such as vanilla GAN (Goodfellow et al., 2014) corresponding to the Jensen-Shannon (JS) divergence, least-squared GAN (LSGAN) corresponding to the Pearson χ^2 divergence. It is also known that many widely used divergences like the JS, Pearson χ^2, KL, Reverse KL divergences, can be seen as a special case of the f-divergence. Thus, before delving into how to leverage the contaminated dataset to learn the distribution of normal samples, we first introduce the general form of f-divergence

$$D_f(P(\mathbf{x})||Q(\mathbf{x})) = \int_X Q(\mathbf{x}) f\left(\frac{P(\mathbf{x})}{Q(\mathbf{x})}\right) d\mathbf{x}, \tag{2}$$

where $P(\mathbf{x})$ and $Q(\mathbf{x})$ are two probability distributions; and $f : \mathbb{R}^+ \rightarrow \mathbb{R}$ could be any convex and semi-continuous function with $f(1) = 0$. By setting the function $f(\cdot)$ to different forms, specific divergence can be induced. For examples, the KL, reverse KL and Pearson χ^2 divergences correspond to setting $f(u) = u \log u$, $f(u) = -\log u$ and $f(u) = (u-1)^2$, respectively (Nowozin et al., 2016).

To learn the normal-data distribution with the available datasets $\mathcal{X}$, $\mathcal{X}^+$ and $\mathcal{X}^-$, we propose to define the $P(\mathbf{x})$ and $Q(\mathbf{x})$ as follows

$$P(\mathbf{x}) = (1-\lambda)p_{\text{data}}(\mathbf{x}) + \lambda p^+(\mathbf{x}), \tag{3}$$

$$Q(\mathbf{x}) = (1-\beta)p_g(\mathbf{x}) + \beta p^-(\mathbf{x}), \tag{4}$$

where $p_g(\mathbf{x})$ stands for the generation distribution induced by a neural network generator; and $\lambda, \beta \in [0, 1]$ denote the weights of the distributions of normal and anomalous samples. When λ and β are set closer to 1, it means more emphasis is put on the collected clean datasets $\mathcal{X}^+$ and $\mathcal{X}^-$. It should be noted that in $P(\mathbf{x})$ and $Q(\mathbf{x})$, only the distribution $p_g(\mathbf{x})$ is trainable, while all other distributions are fixed. Thus, when minimizing the f-divergence between $P(\mathbf{x})$ and $Q(\mathbf{x})$, only $p_g(\mathbf{x})$ is changeable. Actually, it can be proved that when minimizing the f-divergence between $P(\mathbf{x})$ and $Q(\mathbf{x})$ in (3) and (4), under some mild conditions, the generation distribution $p_g(\mathbf{x})$ will converge to the normal-data distribution $p^+(\mathbf{x})$.

Theorem 3.1. *If $Supp(p^+(\mathbf{x})) \cap Supp(p^-(\mathbf{x})) = \emptyset$ and $\beta > (1-\lambda)(1-\pi)$ hold, as the function $f(\cdot)$ is set as $f(\cdot) = (u-1)^2$ or $-\log u$, we have*

$$\arg\min_{p_g(\mathbf{x})} D_f(P||Q) = p^+(\mathbf{x}), \tag{5}$$

where $Supp(\cdot)$ denotes the support of a distribution.

To understand the theorem, we substitute $p_{data}(\mathbf{x}) = \pi p^+(\mathbf{x}) + (1-\pi)p^-(\mathbf{x})$ into $P(\mathbf{x})$ in (3), yielding

$$P(\mathbf{x}) = ((1-\lambda)\pi + \lambda)p^+(\mathbf{x}) + (1-\lambda)(1-\pi)p^-(\mathbf{x}). \tag{6}$$

Comparing the $P(\mathbf{x})$ with $Q(\mathbf{x}) = (1-\beta)p_g(\mathbf{x}) + \beta p^-(\mathbf{x})$, we can see that the condition $\beta > (1-\lambda)(1-\pi)$ in Theorem 3.1 is equivalent to ensure the ratio of $p^-(\mathbf{x})$ in $Q(\mathbf{x})$ is larger than that in $P(\mathbf{x})$ in (6). Thus, by further taking into account the condition of disjoint support between $p^+(\mathbf{x})$

and $p^-(\mathbf{x})$, we can see that under the condition $\beta > (1-\lambda)(1-\pi)$, the trainable $p_g(\mathbf{x})$ will not attempt to allocate any probabilities on the support of $p^-(\mathbf{x})$, but instead directly equals to $p^+(\mathbf{x})$ so that the divergence between $P(\mathbf{x})$ and $Q(\mathbf{x})$ can be minimized. The rigorous proof can be found in the Appendix A.

Under the setting $f(\cdot) = (u-1)^2$, as we minimize the f-divergence $D_f(P||Q)$, it is known that we are actually minimizing the Pearson χ^2 divergence between $P(\mathbf{x})$ and $Q(\mathbf{x})$. As revealed in (Mao et al., 2017), minimizing the Pearson χ^2 divergence between two distributions can be achieved via the least-squared GAN, which employs the least-squared loss, instead of the commonly-used cross-entropy loss, to train the discriminator and generator. Specifically, to minimize the Pearson χ^2 divergence, we can optimize the following two sub-problems alternatively

$$\min_D V(D) = \mathbb{E}_{P(\mathbf{x})}[(D(\mathbf{x})–1)^2] + \mathbb{E}_{Q(\mathbf{x})}[(D(\mathbf{x})-0)^2], \quad (7)$$

$$\min_G V(G) = \mathbb{E}_{Q(\mathbf{x})}[(D(\mathbf{x})-0.5)^2], \quad (8)$$

where D and G denote the discriminator and generator, respectively. Substituting $P(\mathbf{x})$ in (3) and $Q(\mathbf{x})$ in (4) into the above $V(D)$ and $V(G)$ gives

$$\begin{aligned}\min_D V(D) =& (1-\lambda)\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[(D(\mathbf{x})-1)^2] \\ &+ \lambda\mathbb{E}_{\mathbf{x}\sim p^+(\mathbf{x})}[(D(\mathbf{x})-1)^2] \\ &+ (1-\beta)\mathbb{E}_{\mathbf{x}\sim p_g(\mathbf{x})}[(D(\mathbf{x})-0)^2] \\ &+ \beta\mathbb{E}_{\mathbf{x}\sim p^-(\mathbf{x})}[(D(\mathbf{x})-0)^2],\end{aligned} \quad (9)$$

$$\begin{aligned}\min_G V(G) =& (1-\beta)\mathbb{E}_{\mathbf{x}\sim p_g(\mathbf{x})}[(D(\mathbf{x})-0.5)^2] \\ &+ \beta\mathbb{E}_{\mathbf{x}\sim p^-(\mathbf{x})}[(D(\mathbf{x})-0.5)^2],\end{aligned} \quad (10)$$

in which the expectation $\mathbb{E}[\cdot]$ w.r.t. $p_{data}(\mathbf{x})$, $p^+(\mathbf{x})$, $p^-(\mathbf{x})$ and $p_g(\mathbf{x})$ can be approximately evaluated with samples from $\mathcal{X}$, $\mathcal{X}^+$, $\mathcal{X}^-$ and the generated samples, respectively. Therefore, the Pearson χ^2 divergence between $P(\mathbf{x})$ and $Q(\mathbf{x})$ can be minimized by solving the optimization problems (9) and (10) alternatively. After convergence, it is known from Theorem 3.1 that the generation distribution $p_g(\mathbf{x})$ will equal to the normal-data distribution $p^+(\mathbf{x})$.

3.3. Alleviating the Overfitting Caused by the Small Size of Normal and Anomaly Datasets

As discussed before, the clean normal and anomaly datasets $\mathcal{X}^+$ and $\mathcal{X}^-$ are generally very small. Consequently, as we use (9) to train the discriminator D, it may easily memorize them and directly output 1 for samples from $\mathcal{X}^+$ and 0 for samples from $\mathcal{X}^-$. To alleviate the overfitting issue, our basic idea is to prevent the discriminator from easily memorizing the samples in $\mathcal{X}^+$ and $\mathcal{X}^-$. To this end, we propose to modify the $P(\mathbf{x})$ and $Q(\mathbf{x})$ as

$$\tilde{P}(\mathbf{x}) = (1-\gamma)[(1-\lambda)p_{\text{data}}(\mathbf{x}) + \lambda p^+(\mathbf{x})] + \gamma p^-(\mathbf{x}), \quad (11)$$

$$\tilde{Q}(\mathbf{x}) = (1-\gamma)[(1-\beta)p_g(\mathbf{x}) + \beta p^-(\mathbf{x})] + \gamma p^+(\mathbf{x}), \quad (12)$$

where $\gamma \in [0,1]$ indicates the flipping probability. Before explaining the implications of the modification, we first present the theoretical result that when we minimize the f-divergence between the modified $\tilde{P}(\mathbf{x})$ and $\tilde{Q}(\mathbf{x})$, under some mild conditions, the generation distribution $p_g(\mathbf{x})$ can still converge to the normal-data distribution $p^+(\mathbf{x})$.

Theorem 3.2. *If $Supp(p^+(\mathbf{x})) \cap Supp(p^-(\mathbf{x})) = \emptyset$ and $(1-\gamma)\beta > (1-\lambda)(1-\pi) + \gamma$ hold, as the function $f(\cdot)$ is set as $f(\cdot) = (u-1)^2$ or $-\log u$, we have $\arg\min_{p_g(\mathbf{x})} D_f(\tilde{P}||\tilde{Q}) = p^+(\mathbf{x})$.*

The theorem can be understood in a similar way as Theorem 3.1. That is, by substituting $p_{data}(\mathbf{x}) = \pi p^+(\mathbf{x}) + (1-\pi)p^-(\mathbf{x})$ into $\tilde{P}(\mathbf{x})$ in (11), we obtain

$$\begin{aligned}\tilde{P}(\mathbf{x}) =& (1-\gamma)((1-\lambda)\pi + \lambda)p^+(\mathbf{x}) \\ &+ ((1-\lambda)(1-\pi) + \gamma)\, p^-(\mathbf{x}).\end{aligned} \quad (13)$$

Comparing the coefficient of $p^-(\mathbf{x})$ in (13) with that in $\tilde{Q}(\mathbf{x})$ in (12), we can see that the condition $(1-\gamma)\beta > (1-\lambda)(1-\pi) + \gamma$ is equivalent to ensure the ratio of $p^-(\mathbf{x})$ in $\tilde{Q}(\mathbf{x})$ is larger than that in $\tilde{P}(\mathbf{x})$. Thus, given the disjoint support condition $Supp(p^+(\mathbf{x})) \cap Supp(p^-(\mathbf{x})) = \emptyset$, the trainable distribution $p_g(\mathbf{x})$ will not assign any probabilities on the support of $p^-(\mathbf{x})$, but directly equals to $p^+(\mathbf{x})$. The rigorous proof is provided in the Appendix B.

As the function $f(\cdot)$ is set as $(u-1)^2$, the f-divergence is reduced to the Pearson divergence. Thus, we can resort to the LSGAN to minimize the f-divergence by solving the two sub-problems (7) and (8), except replacing $P(\mathbf{x})$ and $Q(\mathbf{x})$ with the modified distributions $\tilde{P}(\mathbf{x})$ and $\tilde{Q}(\mathbf{x})$. To see why the modified $\tilde{P}(\mathbf{x})$ and $\tilde{Q}(\mathbf{x})$ can alleviate the overfitting issue, we substitute the $\tilde{P}(\mathbf{x})$ in (11) and $\tilde{Q}(\mathbf{x})$ in (12) into the $V(D)$ in (7), yielding

$$\begin{aligned}\min_D V(D) =& (1-\gamma)(1-\lambda)\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[(D(\mathbf{x})-1)^2] \\ &+ (1-\gamma)\lambda\mathbb{E}_{\mathbf{x}\sim p^+(\mathbf{x})}[(D(\mathbf{x})-1)^2] \\ &+ \gamma\mathbb{E}_{\mathbf{x}\sim p^-(\mathbf{x})}[(D(\mathbf{x})-1)^2] \\ &+ (1-\gamma)(1-\beta)\mathbb{E}_{\mathbf{x}\sim p_g(\mathbf{x})}[(D(\mathbf{x})-0)^2] \\ &+ (1-\gamma)\beta\mathbb{E}_{\mathbf{x}\sim p^-(\mathbf{x})}[(D(\mathbf{x})-0)^2] \\ &+ \gamma\mathbb{E}_{\mathbf{x}\sim p^+(\mathbf{x})}[(D(\mathbf{x})-0)^2],\end{aligned} \quad (14)$$

where the expectation $\mathbb{E}[\cdot]$ w.r.t. $p_{data}(\mathbf{x})$, $p^+(\mathbf{x})$, $p^-(\mathbf{x})$ and $p_g(\mathbf{x})$ are estimated with samples from $\mathcal{X}$, $\mathcal{X}^+$, $\mathcal{X}^-$ and the generated samples, respectively. Comparing the objective function (14) to the original objective (9), we can see that instead of simply forcing the discriminator to output

1 for samples from $\mathcal{X}^+$ and 0 for samples from $\mathcal{X}^-$, we will randomly select some samples from the anomaly dataset $\mathcal{X}^-$ with a probability γ and then require the discriminator to output 1 for them. The same also applies to the output value of 0. Obviously, the randomly selected samples can somehow confuse the discriminator, making it harder to memorize what to output for samples in $\mathcal{X}^+$ and $\mathcal{X}^-$, thereby alleviating the overfitting issue. More importantly, despite the flipping mechanism is introduced, $p_g(\mathbf{x})$ can still converge to the normal-data distribution $p^+(\mathbf{x})$.

To ensure the satisfaction of condition $(1-\gamma)\beta > (1-\lambda)(1-\pi)+\gamma$ in Theorem 3.2, the flipping probability γ cannot be set too large. For example, we can set it to a value like 0.05 or 0.1. But in practice, we find that the model can work better if we can increase γ slightly if the discriminator is deemed overfitting too much or decrease it otherwise. Therefore, we propose to adaptively adjust the value of γ according to the estimated overfitting degree of discriminator as follows

$$\gamma = \begin{cases} 1 & \text{if } \gamma \geq 1 \\ \gamma + \Delta\gamma & \text{if } \gamma < 1 \text{ and } S_o > \tau \\ \gamma - \Delta\gamma & \text{if } \gamma > 0 \text{ and } S_o \leq \tau \\ 0 & \text{if } \gamma \leq 0, \end{cases} \tag{15}$$

where S_0 is the estimated value that measures the degree of overfitting; τ is the threshold to increase or decrease γ, which is set to a value close to 1 in our experiments; and $\Delta\gamma$ is the adjustment stepsize. In our experiments, we estimate the overfitting degree using the difference between the discriminator's averaged output value for samples from $\mathcal{X}^+$ and the averaged output value for samples from $\mathcal{X}^-$, that is, we let $S_0 = \mathbb{E}_{\mathbf{x}\sim\mathcal{X}^+}[D(\mathbf{x})] - \mathbb{E}_{\mathbf{x}\sim\mathcal{X}^-}[D(\mathbf{x})]$. In our experiments, only a proportion of samples are used to estimate S_0 for efficiency.

3.4. Considering $\mathcal{X}^-$ only Covering Incomplete Anomaly Types

So far, we assume the auxiliary anomaly dataset $\mathcal{X}^-$ covers all types of anomalies by default. But due to the diversity nature of anomalous data in real-world applications, this assumption may not hold. In this case, the anomaly dataset $\mathcal{X}^-$ may only represent a subset of all possible anomaly types. To reflect this point, we assume the distribution of the anomalous data is composed of two components

$$p^-(\mathbf{x}) = (1-\alpha)p_u^-(\mathbf{x}) + \alpha p_c^-(\mathbf{x}), \tag{16}$$

where $p_c^-(\mathbf{x})$ denotes the distribution of anomalies that are observed in $\mathcal{X}^-$, while $p_u(\mathbf{x})$ represents the distribution of anomalies unseen in the collected anomaly dataset $\mathcal{X}^-$; and $\alpha \in [0,1]$ indicates the proportion of the distribution $p_c^-(\mathbf{x})$. In this case, since we only the anomalous samples drawn from $p_c^-(\mathbf{x})$ are observed, we propose to modify the distribution $P(\mathbf{x})$ and $Q(\mathbf{x})$ as following form

$$\tilde{P}(\mathbf{x}) = (1-\gamma)[(1-\lambda)p_{\text{data}}(\mathbf{x}) + \lambda p^+(\mathbf{x})] + \gamma p_c^-(\mathbf{x}), \tag{17}$$

$$\tilde{Q}(\mathbf{x}) = (1-\gamma)[(1-\beta)p_g(\mathbf{x}) + \beta p_c^-(\mathbf{x})] + \gamma p^+(\mathbf{x}), \tag{18}$$

in which only the distribution of anomalies from observed types $p_c^-(\mathbf{x})$ is used. When minimizing the f-divergence between $\tilde{P}(\mathbf{x})$ and $\tilde{Q}(\mathbf{x})$, we have the following theorem.

Theorem 3.3. *Denote $\kappa_1 = (1-\gamma)(1-\lambda)\pi + \lambda$, $\kappa_2 = (1-\gamma)(1-\lambda)(1-\pi)(1-\alpha)$ and $\kappa_3 = (1-\gamma)(1-\beta)+\gamma$. If $Supp(p^+(\mathbf{x})) \cap Supp(p^-(\mathbf{x})) = \emptyset$, $Supp(p_u^-(\mathbf{x})) \cap Supp(p_c^-(\mathbf{x})) = \emptyset$, $(1-\gamma)(1-\alpha+\beta+\alpha\pi+\alpha\lambda-\alpha\lambda\pi) > 1$ and $\kappa_1\kappa_3 > \gamma(\kappa_1+\kappa_2)$ hold, as the function $f(\cdot)$ is set as $f(\cdot) = (u-1)^2$ or $-\log u$, we have*

$$\begin{aligned} arg\min_{p_g(\mathbf{x})} D_f(\tilde{P}||\tilde{Q}) = & \frac{\kappa_1\kappa_3 - \gamma(\kappa_1+\kappa_2)}{(1-\gamma)(1-\beta)(\kappa_1+\kappa_2)} p^+(\mathbf{x}) \\ & + \frac{\kappa_2\kappa_3}{(1-\gamma)(1-\beta)(\kappa_1+\kappa_2)} p_u^-(\mathbf{x}). \end{aligned}$$

Please refer to Appendix C for the proof. From Theorem 3.3, we can see that under the scenario with incomplete collected anomalies, the generation distribution $p_g(\mathbf{x})$ cannot converge to normal-data distribution $p^+(\mathbf{x})$ anymore, but instead to a mixture of $p^+(\mathbf{x})$ and $p_u^-(\mathbf{x})$, which represents the anomaly distribution of unobserved types. Therefore, with the help of incomplete anomaly dataset $\mathcal{X}^-$, the model can still mitigate the influence from anomalies of observed types. Furthermore, if we increase the weight of normal samples λ, due to $\kappa_2 = (1-\gamma)(1-\lambda)(1-\pi)(1-\alpha)$, the coefficient $\frac{\kappa_2\kappa_3}{(1-\gamma)(1-\beta)(\kappa_1+\kappa_2)}$ controlling the weighting of $p_u^-(\mathbf{x})$ will become small. Thus, although we cannot use $\mathcal{X}^-$ to mitigate the influence of anomalies from unobserved types, we can still use the normal dataset $\mathcal{X}^+$ to dampen their influence in the final converged distribution $p_g(\mathbf{x})$. But due to the small size of $\mathcal{X}^+$, the value of λ cannot be set as large as we want. That is because this could easily result in overfitting to these normal samples.

3.5. Detection Method

Despite our model is able to learn the distribution of normal data (without considering the incompleteness of collected anomalies), which implies that the anomalies will locate at the low-density region, due to the high cost of directly evaluating the density value, a surrogate metric is developed to detect anomalies efficiently. To this end, we train our proposed GAN under the paradigm of bidirectional GAN (BiGAN) (Donahue et al., 2017; Li et al., 2017), which, in addition to the generator, also includes an encoder $E(\cdot)$ to encode samples into the latent space. Since the model can capture the distribution of normal data, theoretically, the reconstruction errors $||\mathbf{x} - G(E(\mathbf{x}))||_2^2$ on normal samples should be small. Moreover, the latent representation $E(\mathbf{x})$

of normal samples should follow a standard normal distribution, thus the norm $||E(\mathbf{x})||_2^2$ for normal samples should be small, too. Thus, we can employ either of the two metrics to detect anomalies. In this paper, we propose to compute the anomaly score of a sample $\mathbf{x}$ by combining them in an intuitive way as

$$S(\mathbf{x}) = \frac{||\mathbf{x} - G(E(\mathbf{x}))||_2^2 - a_{min}}{a_{max} - a_{min}} + \rho \frac{||E(\mathbf{x})||_2^2 - b_{min}}{b_{max} - b_{min}},$$

where ρ is a weight coefficient; a_{max} and a_{min} can be set as the maximum and minimum value of $||\mathbf{x} - G(E(\mathbf{x}))||_2^2$ on the validation set, respectively; the same applies to b_{max} and b_{min}, which, however, is computed using $||E(\mathbf{x})||_2^2$. Using this anomaly score, samples with higher scores, which imply poor reconstruction and deviation from the prior distribution, are more likely to be anomalies.

4. Experiments

4.1. Datasets and Settings

Datasets To control the number of anomaly types observed in $\mathcal{X}^-$, we first experiment with four classification datasets, as done in the previous semi/weakly-supervised anomaly detection works (Ruff et al., 2020; Chen et al., 2021; Tian et al., 2022). Specifically, we use three image datasets (MNIST, F-MNIST and CIFAR10) and one textual dataset (20newsgroups), in which the textual features are extracted from a pre-trained BERT (Devlin et al., 2019) as proposed in ADBench (Han et al., 2022). For more details about the toy datasets, please refer to Appendix D.1. In addition, we also experiment with two real-world large anomaly detection datasets (UNSW-NB15 and HAR), as well as other nine classical anomaly detection datasets. For the UNSW-NB15 dataset, we follow (Pang et al., 2023) to select the Backdoor, DoS, Fuzzers and Reconnaissance anomaly types to constitute the detection dataset. HAR is tabular data on human activities. Following (Pang et al., 2021), we treat going downstairs and upstairs activities as anomalies, while viewing the other four activities as normal. For more details about these datasets, please refer to Appendix D.2.

Training For the toy datasets, we treat samples from a subset of categories as normal data, while viewing samples from the left categories as anomalies. To obtain the contaminated dataset $\mathcal{X}$, we randomly select a proportion of ϵ_p samples from the anomaly categories and mix them with the normal data, where ϵ_p denotes the ratio between the number of selected anomalies and the total number in $\mathcal{X}$. The clean normal dataset $\mathcal{X}^+$ is comprised of samples randomly selected from all normal categories. But to mimic the situation that we are unable to collect all types of anomalies due to the diversity nature of anomalies, the anomaly dataset $\mathcal{X}^-$ is constructed by only selecting some samples from one of the anomaly categories randomly. The size of $\mathcal{X}^+$ and $\mathcal{X}^-$ are controlled by the parameters ϵ_n and ϵ_a, which are defined in a similar way as ϵ_p. Unless specified otherwise, in our experiments on toy datasets, the three parameters ϵ_p, ϵ_n and ϵ_a are set as 20%, 5% and 1%, respectively. More normal samples are used here because they are generally considered to be easier to collect than anomalies in practice. As for the datasets UNSW-NB15 and HAR, the three datasets $\mathcal{X}$, $\mathcal{X}^+$ and $\mathcal{X}^-$ are constructed in the same way as toy datasets. But since these datasets contain normal and anomalous samples by themselves, we do not need to manually specify which categories are normal and which are abnormal. For the two datasets, due to the scarcity of available anomalies, when constructing the contaminated dataset $\mathcal{X}$, the contamination ratio ϵ_p is only set to 5% and 10%, respectively, while ϵ_n and ϵ_a are set the same as toy datasets. For each of these datasets, a validation dataset is built to help find the appropriate values for hyper-parameters[1].

Evaluation We train the model for every anomaly type that is observed in $\mathcal{X}^-$, and the averaged performance on the testing dataset is then reported. In our paper, following the previous works (Ruff et al., 2020; Tian et al., 2022), the area under the receiver operating characteristic curve (AUROC) is employed as the performance criterion. Please refer to the Appendix E for more training details.

4.2. Baseline

For comparison, two unsupervised methods and two self-supervised methods are used: **Deep SVDD** (Ruff et al., 2018), **Deep Isolation Forest** (Xu et al., 2023a), Scale Learning for Anomaly Detection (**SLAD**) (Xu et al., 2023c), Internal Contrastive Learning (**ICL**) (Shenkar & Wolf, 2022). In addition, since auxiliary datasets are used in our methods, we also compare with five semi-supervised methods that all make use of an extra clean anomaly dataset: **Deep SAD** (Ruff et al., 2020), **FeaWAD** (Zhou et al., 2021), **RoSAS** (Xu et al., 2023b), **PReNet** (Pang et al., 2023), **AA-BiGAN** (Tian et al., 2022), **SOEL**(Li et al., 2023).

4.3. Performance and Analyses

Performance on Datasets with Diverse Normal Data In the experiments of previous weakly/semi-supervised detection methods on toy datasets, the normal data is often assumed to be comprised of only one type of samples (Ruff et al., 2020; Tian et al., 2022; Jiang et al., 2023). However, we think this setting may not be very reasonable since normal data could be more diverse than this. Thus, we first explore the performance of our method and the baselines when the normal data is comprised of different number of

[1] The code is available at `https://github.com/Vanssssry/CR-GAN`.

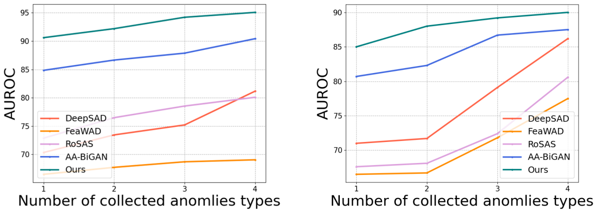
Table 1. AUROC performance under different numbers of normal types contained in the training dataset $\mathcal{X}$. ϵ_p is set as 20% here.

Dataset	K	Deep SVDD	DIF	SLAD	ICL	Deep SAD	Fea WAD	Ro SAS	AA-Bi GAN	Ours
MNIST	1	69.7	68.6	88.8	88.5	86.6	70.7	75.2	90.8	**93.7**
	3	65.7	56.2	74.8	71.3	70.3	66.5	72.8	82.4	**90.6**
	5	58.7	52.9	55.6	58.7	63.6	56.1	59.4	73.7	**83.5**
FMNIST	1	69.3	70.8	82.1	84.3	80.8	76.4	73.3	90.2	**91.1**
	3	62.1	69.8	79.2	80.3	71.0	66.5	67.6	80.7	**85.2**
	5	61.3	65.2	78.9	83.6	69.8	61.7	62.0	88.7	**90.0**
CIFAR10	1	59.0	56.7	62.3	62.1	68.5	64.3	67.1	77.0	**78.4**
	3	51.3	45.0	55.6	53.3	64.5	61.4	64.6	63.7	**67.5**
	5	48.3	42.7	51.2	50.4	57.8	51.6	53.4	56.5	**58.2**
20NEWS	1	63.4	63.9	75.1	73.3	66.8	56.1	68.6	70.8	**75.2**
	2	52.8	56.6	65.0	60.8	63.0	53.2	63.5	64.4	**72.5**
	3	52.1	56.3	60.3	52.3	63.5	52.4	64.6	70.7	**74.0**

Table 2. AUROC under different sizes of the auxiliary dataset.

Dataset	Size	Deep SAD	Fea WAD	Ro SAS	AA-Bi GAN	Ours
MNIST	10	65.1	64.9	55.8	75.1	**87.4**
	20	65.9	65.2	58.1	76.9	**87.9**
	30	66.5	64.8	58.2	78.0	**90.1**
	50	67.8	65.3	59.1	80.8	**91.4**
FMNIST	10	73.1	51.2	54.5	80.6	**84.2**
	20	73.2	52.1	55.6	81.3	**85.4**
	30	72.6	53.7	56.1	82.7	**86.1**
	50	72.8	54.0	56.8	83.0	**86.5**
CIFAR10	10	57.5	51.7	54.3	63.2	**64.7**
	20	60.1	53.1	55.7	63.9	**65.8**
	30	60.9	54.6	55.9	64.2	**66.8**
	50	61.7	56.9	57.1	64.8	**67.9**
20NEWS	10	54.6	50.8	54.8	62.8	**69.1**
	20	60.0	51.7	58.8	65.1	**71.8**
	30	61.0	52.1	60.6	67.0	**73.3**
	50	59.8	53.8	63.7	72.9	**77.1**
HAR	10	67.4	64.2	62.3	86.0	**89.4**
	20	73.8	67.6	72.9	88.9	**91.8**
	30	78.6	77.1	81.4	90.5	**92.1**
	50	81.7	82.1	85.3	92.4	**93.3**
UNSW-NB15	10	64.5	82.3	65.8	79.0	**83.1**
	20	69.1	82.4	67.5	81.8	**84.8**
	30	73.0	83.1	69.1	82.6	**85.0**
	50	74.2	83.6	71.3	82.7	**85.3**

categories. When the number of normal categories is set to 1, we follow exactly the same setting as previous work (Ruff et al., 2020; Tian et al., 2022) to evaluate the performance. When the number of normal categories is set to a value K more than one, we view the first K categories as normal, while viewing the remaining as anomalous. Table 1 shows the performance of our method and baselines on four toy datasets under different numbers of normal categories. It can be seen from Table 1 that the performance of all methods decreases as the normal data becomes more diverse. But thanks to the exploitation of a normal dataset, the results show that the performance of our method deteriorates more slowly than the baselines, which makes our method more suitable for practical scenarios. In practice, normal data would not be as simple as comprised of just one category. Thus, in our subsequent experiments on toy datasets, the number of normal categories is always set to three.



(a) MNIST (b) Fashion-MNIST

Figure 1. AUROC performance under different numbers of collected types in the clean anomalous dataset.

Performance under Different Contamination Ratios To study the performance of the proposed method under the scenario with different contamination ratios, we have conducted experiments with different ϵ_p. Table 3 demonstrates the performance of different anomaly detection methods on different datasets under different values of ϵ_p. It is worth mentioning that due to the lack of anomalous samples in HAR and UNSW-NB15 datasets, the contamination range considered for these two datasets are smaller than other toy datasets. From the table, it can be seen that the performance of all anomaly detection methods decreases as the level of contamination increases. But our approach drops slower than other methods in all six datasets across. This suggests that the introduction of two auxiliary datasets brings more information into the modeling of normal data, thus effectively resisting contamination in the training dataset.

Impact of the Number of Types of Collected Anomalies The aforementioned experimental results all assume

Table 3. AUROC under different ratios of contamination ϵ_p.

Dataset	ϵ_p	Deep SVDD	DIF	SLAD	ICL	Deep SAD	Fea WAD	Ro SAS	AA-Bi GAN	Ours
MNIST	0.05	71.3	57.1	84.8	85.1	75.6	73.1	81.8	89.5	**94.9**
	0.1	69.7	58.7	81.2	82.8	73.7	68.9	76.7	86.8	**93.0**
	0.2	65.8	56.2	74.8	71.4	70.4	66.5	72.9	82.5	**90.6**
	0.3	63.8	54.9	67.7	63.8	67.5	63.1	69.1	79.9	**87.8**
FMNIST	0.05	63.5	71.7	83.4	85.8	77.9	69.0	71.0	83.5	**88.0**
	0.1	62.5	70.3	80.4	85.0	74.7	67.2	68.4	83.0	**86.1**
	0.2	61.5	69.8	79.2	84.3	71.0	66.5	67.6	80.7	**85.2**
	0.3	59.8	68.6	78.1	83.5	68.3	63.2	64.3	80.1	**84.6**
CIFAR10	0.05	58.4	50.9	60.3	58.7	65.8	64.3	67.0	65.3	**68.4**
	0.1	54.4	49.1	59.6	56.2	64.9	63.7	65.8	64.5	**67.9**
	0.2	51.3	45.0	55.6	53.3	64.5	61.4	64.6	63.7	**67.5**
	0.3	51.1	47.3	53.8	52.0	63.0	57.8	62.7	62.1	**66.2**
20NEWS	0.05	54.7	57.9	64.1	59.9	69.0	55.8	72.0	74.7	**77.6**
	0.1	53.4	57.1	63.9	56.7	65.3	53.2	66.4	73.0	**76.1**
	0.2	52.1	56.3	60.3	52.3	63.5	52.4	64.6	70.7	**74.0**
	0.3	50.9	56.2	59.2	51.1	62.7	51.6	61.1	69.7	**72.3**
HAR	0.01	85.4	85.9	89.9	90.8	90.4	88.2	92.7	95.2	**97.1**
	0.05	81.3	84.1	86.6	87.5	85.7	84.8	86.2	91.4	**93.2**
	0.1	77.6	83.1	85.1	81.5	79.5	82.4	85.9	89.9	**91.6**
	0.2	62.4	75.5	82.3	80.9	70.0	76.6	72.9	88.3	**89.3**
UNSW-NB15	0.025	65.5	75.8	75.4	72.9	87.5	83.7	85.6	88.8	**93.8**
	0.05	63.2	74.8	71.3	66.5	87.3	82.4	84.0	88.0	**92.7**
	0.075	60.6	73.4	67.3	58.2	84.7	81.5	83.2	84.0	**92.0**
	0.1	58.4	71.8	65.5	54.8	82.3	80.6	79.3	83.6	**91.7**

Table 4. AUROC of our method under different γ.

Dataset	$\gamma = 0$	$\gamma = 0.05$	$\gamma = 0.1$	Adaptive
MNIST	88.2	89.6	89.9	**90.6**
FMNIST	83.7	84.9	84.3	**85.2**
20NEWS	72.3	72.4	70.8	**74.0**
HAR	91.8	92.4	92.2	**92.6**
UNSW-NB15	90.0	92.1	92.4	**92.7**

the dataset $\mathcal{X}^-$ is built from only one type of anomalous data. Figure 1 shows that how the performance varies as the number of types of collected anomalies varies from 1 to 4 on MNIST and F-MNIST datasets. Obviously, as the number of collected anomalous types increases, the performance of all the methods improves, but our method remains the best over all number of types considered.

Impact of the Number of Collected Samples To validate the effectiveness of the proposed flipping mechanism, experiments have been conducted with varying sizes of auxiliary datasets. Table 2 shows that given a limited amount of collected data, varying from 10 to 50, our method with the proposed flipping mechanism outperforms other weakly/semi-supervised methods. Specifically, the averaged performance gains of our method over the best baseline in MNIST, FMNIST and 20NEW are 11.5%, 3.7% and 5.9% respectively. This confirms that slightly flipping the labels to confuse the discriminator can alleviate the overfitting issue of discriminator and thereby help the model exploit the small amount of collected samples more effectively.

Impact of Flipping Probability γ We conduct experiments under different fixed values of γ as well as adaptive values adjusted according to (15). When γ is fixed at 0, labels of two clean datasets will not flip at all. From Table 4, our method with the fixed $\gamma = 0.05$ achieves better performance than the non-flipping one. We can also see that by using the proposed adaptive probability adjusting scheme, the best performance can be achieved. This indicates that slightly flipping the labels of auxiliary datasets is helpful when collected clean datasets are small. However, the fixed γ should be set carefully to avoid the negative impacts.

Performance on Classical Anomaly Detection Datasets Our method is also evaluated on nine other classic anomaly detection datasets. Due to the scarcity of anomalous data in these datasets, ϵ_a and ϵ_n both are fixed at 1%, and ϵ_p is also fixed at 1% if anomalous data are adequacy. From Table 5, we can see that our method achieves the best performance on seven of the nine datasets, and the second-best performance on the Shuttle dataset. This result indicates that our method still works well without critical contamination, demonstrating the competitiveness of our method.

Table 5. AUROC on nine classic anomaly detection datasets.

Dataset	Deep SAD	Fea WAD	Ro SAS	PRe Net	AA-Bi GAN	SO EL	Ours
Arrhythmia	78.5	73.8	61.2	64.4	79.9	83.6	**84.8**
Cardio	96.1	73.9	91.7	90.7	97.6	95.7	**99.4**
Satellite	83.8	89.6	86.3	78.5	85.4	85.9	**91.8**
Satimage-2	96.0	**99.9**	99.2	99.1	99.7	99.7	**99.9**
Shuttle	99.4	97.9	98.6	99.1	99.0	**99.7**	99.5
Thyroid	**99.6**	65.6	**99.6**	96.2	99.1	99.1	**99.6**
Bank	75.5	61.1	89.8	62.6	87.5	89.9	**90.7**
Amazon	89.9	58.2	85.5	76.2	85.7	**90.5**	89.7
Yelp	90.5	55.9	92.2	79.0	89.3	91.3	**92.6**

Table 6. Ablation Study.

Dataset	Additional Dataset Leveraging	Flipping Mechanism	AUROC
MNIST	×	×	84.0
	✓	×	88.2
	✓	✓	**90.6**
FMNIST	×	×	80.6
	✓	×	83.7
	✓	✓	**85.2**

Ablation Study We have further conducted an ablation study to investigate the effectiveness of different modules in the model, including the module using the additionally collected normal and anomaly datasets and the module flipping mechanism used to alleviate overfitting. From the Table 6, we can clearly observe that by using the additionally collected normal and anomaly datasets, the performance can be improved substantially. Furthermore, by using the theoretically supported flipping mechanism, a further improvement can be observed due to the mechanism's ability to alleviate the overfitting problem caused by the small size of the additional collected datasets.

Sensitivity Analysis of Parameter λ and β : From the Theorem 3.2, the condition $(1-\gamma)\beta > (1-\lambda)(1-\pi)+\gamma$ should be fulfilled. To study the sensitivity of our model to the parameters λ and β, we conduct experiments with various values of λ and β. Table 7 shows that λ and β could be roughly set in the range $[0.6, 0.8]$ to better resist the contamination.

Sensitivity Analysis of Parameter ρ To study the sensitivity of our model to the parameter ρ used as a weight coefficient in Eq.19, we conduct experiments with various values of ρ. From the Table 8, we can see that the performance of our method is quite robust to the choice of the value of ρ. As long as it is chosen within the range $[1, 8]$, the performance does not have too much difference on the considered datasets. In our experiments, we simply set it to 4 for all experiments.

Table 7. Sensitivity analysis w.r.t λ and β, where $\lambda = \beta$.

Dataset	0.2	0.4	0.6	0.8	0.9
MNIST	87.6	88.7	87.1	**90.2**	89.5
FMNIST	84.6	84.3	84.9	**85.8**	84.7
20NEWS	72.4	72.6	**73.4**	71.6	71.7
HAR	91.5	91.9	92.1	**92.6**	91.5
UNSW-NB15	92.3	92.0	**92.7**	92.6	92.6

Table 8. Sensitivity Analysis of ρ.

Dataset	$\rho = 1$	$\rho = 4$	$\rho = 8$
MNIST	90.5	**90.8**	90.4
FMNIST	84.5	**85.9**	85.6
20news	70.3	**74.0**	73.7
HAR	91.5	**91.7**	91.0
UNSW-NB15	91.8	92.8	**93.0**

5. Conclusion

In this paper, we studied the problem of anomaly detection when the training dataset is contaminated and the sizes of available clean datasets are small. To alleviate the negative impact of contamination in training datasets, a novel GANs, which can exploit partially observed normal and anomalous data, is developed to learn the normal-data distribution. We theoretically prove that the proposed GANs can recover the distribution of normal data from the distribution of contaminated dataset. Furthermore, we introduced a flipping mechanism to alleviate the overfitting issue caused by the small size of two clean datasets. The case that the observed anomalous data only covers incomplete anomaly types is considered and the proposed method can still mitigate the influence of observed anomalies. Extensive experimental results demonstrate that our method can effectively leverage the observed normal and anomalous data and outperform the comparable baselines on vast datasets.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 62276280, U1811264), Guangzhou Science and Technology Planning Project (No. 2024A04J9967), the Fundamental Research Funds of the Central Universities, Sun Yat-Sen University (No. 23ptpy78).

Impact Statement

This paper proposes an anomaly detection method that utilizes collected samples for adversarial learning, which can effectively alleviate the negative impact of contaminated training datasets. Anomaly detection plays an important role in identifying fraud, diseases, and industrial defects. However, the training dataset with contamination results in a biased detection model that fails to detect anomalies. Our method advances anomaly detection in real-world application scenarios and makes it more reliable.

References

Akcay, S., Atapour-Abarghouei, A., and Breckon, T. P. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Computer Vision–ACCV*, pp. 622–637, 2018.

Chandola, V., Banerjee, A., and Kumar, V. Anomaly detection: A survey. *ACM computing surveys*, 41(3):1–58, 2009.

Chen, C., Liu, J., Xie, Y., Ban, Y. X., Wu, C., Tao, Y., and Song, H. Latent regularized generative dual adversarial network for abnormal detection. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pp. 760–766, 2021.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186, 2019.

Donahue, J., Krähenbühl, P., and Darrell, T. Adversarial feature learning. In *5th International Conference on Learning Representations*, 2017.

Gong, D., Liu, L., Le, V., Saha, B., Mansour, M. R., Venkatesh, S., and Hengel, A. v. d. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1705–1714, 2019.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Han, S., Hu, X., Huang, H., Jiang, M., and Zhao, Y. Adbench: Anomaly detection benchmark. *Advances in Neural Information Processing Systems*, 35:32142–32159, 2022.

Jiang, M., Hou, C., Zheng, A., Hu, X., Han, S., Huang, H., He, X., Yu, P. S., and Zhao, Y. Weakly supervised anomaly detection: A survey. *arXiv preprint arXiv:2302.04549*, 2023.

Katz-Samuels, J., Blanchard, G., and Scott, C. Decontamination of mutual contamination models. *Journal of machine learning research*, 20(41):1–57, 2019.

Kim, M., Yu, J., Kim, J., Oh, T.-H., and Choi, J. K. An iterative method for unsupervised robust anomaly detection under data contamination. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2023.

Lai, C.-H., Zou, D., and Lerman, G. Robust subspace recovery layer for unsupervised anomaly detection. In *International Conference on Learning Representations*, 2020.

Li, A., Qiu, C., Kloft, M., Smyth, P., Mandt, S., and Rudolph, M. Deep anomaly detection under labeling budget constraints. In *International Conference on Machine Learning*, pp. 19882–19910, 2023.

Li, C., Liu, H., Chen, C., Pu, Y., Chen, L., Henao, R., and Carin, L. Alice: Towards understanding adversarial learning for joint distribution matching. *Advances in neural information processing systems*, 30, 2017.

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., and Paul Smolley, S. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2794–2802, 2017.

Nowozin, S., Cseke, B., and Tomioka, R. f-gan: Training generative neural samplers using variational divergence minimization. *Advances in neural information processing systems*, 29, 2016.

Pang, G., van den Hengel, A., Shen, C., and Cao, L. Toward deep supervised anomaly detection: Reinforcement learning from partially labeled anomaly data. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 1298–1308, 2021.

Pang, G., Shen, C., Jin, H., and van den Hengel, A. Deep weakly-supervised anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1795–1807, 2023.

Qiu, C., Pfrommer, T., Kloft, M., Mandt, S., and Rudolph, M. Neural transformation learning for deep anomaly detection beyond images. In *International Conference on Machine Learning*, pp. 8703–8714, 2021.

Qiu, C., Li, A., Kloft, M., Rudolph, M., and Mandt, S. Latent outlier exposure for anomaly detection with contaminated data. In *International Conference on Machine Learning*, pp. 18153–18167, 2022.

Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S. A., Binder, A., Müller, E., and Kloft, M. Deep one-class classification. In *International conference on machine learning*, pp. 4393–4402. PMLR, 2018.

Ruff, L., Vandermeulen, R. A., Görnitz, N., Binder, A., Müller, E., Müller, K., and Kloft, M. Deep semi-supervised anomaly detection. In *8th International Conference on Learning Representations*, 2020.

Ruff, L., Kauffmann, J. R., Vandermeulen, R. A., Montavon, G., Samek, W., Kloft, M., Dietterich, T. G., and Müller, K.-R. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5):756–795, 2021.

Schölkopf, B., Williamson, R. C., Smola, A., Shawe-Taylor, J., and Platt, J. Support vector method for novelty detection. *Advances in neural information processing systems*, 12, 1999.

Shenkar, T. and Wolf, L. Anomaly detection for tabular data with internal contrastive learning. In *International Conference on Learning Representations*, 2022.

Sinha, A., Ayush, K., Song, J., Uzkent, B., Jin, H., and Ermon, S. Negative data augmentation. In *9th International Conference on Learning Representations*, 2021.

Su, Q., Tian, B., Wan, H., and Yin, J. Anomaly detection under contaminated data with contamination-immune bidirectional gans. *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–16, 2024. doi: 10.1109/TKDE.2024.3404027.

Tax, D. M. and Duin, R. P. Support vector data description. *Machine learning*, 54:45–66, 2004.

Tian, B., Su, Q., and Yin, J. Anomaly detection by leveraging incomplete anomalous knowledge with anomaly-aware bidirectional gans. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pp. 2255–2261, 2022.

Tian, B., Su, Q., and Yu, J. Leveraging contaminated datasets to learn clean-data distribution with purified generative adversarial networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 9989–9996, 2023.

Vandermeulen, R. A., Saitenmacher, R., and Ritchie, A. A proposal for supervised density estimation. In *Proc. NeurIPS Pre-Registration Workshop*, 2020.

Xu, H., Pang, G., Wang, Y., and Wang, Y. Deep isolation forest for anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 2023a.

Xu, H., Wang, Y., Pang, G., Jian, S., Liu, N., and Wang, Y. Rosas: Deep semi-supervised anomaly detection with contamination-resilient continuous supervision. *Information Processing & Management*, 60(5):103459, 2023b.

Xu, H., Wang, Y., Wei, J., Jian, S., Li, Y., and Liu, N. Fascinating supervisory signals and where to find them: Deep anomaly detection with scale learning. In *International Conference on Machine Learning*, volume 202, pp. 38655–38673, 2023c.

Zhou, C. and Paffenroth, R. C. Anomaly detection with robust deep autoencoders. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 665–674, 2017.

Zhou, Y., Song, X., Zhang, Y., Liu, F., Zhu, C., and Liu, L. Feature encoding with autoencoders for weakly supervised anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 33(6):2454–2465, 2021.

A. Proof of Theorem 3.1

Theorem A.1. *If $Supp(p^+(\mathbf{x})) \cap Supp(p^-(\mathbf{x})) = \emptyset$ and $\beta > (1-\lambda)(1-\pi)$, as the function $f(\cdot)$ is set as $f(\cdot) = (u-1)^2$ or $-\log u$, we have*

$$\arg\min_{p_g(\mathbf{x})} D_f(P||Q) = p^+(\mathbf{x}), \tag{19}$$

where $Supp(\cdot)$ denotes the support of a distribution.

Proof. For simplicity, we use $p_{\text{data}}, p^+, p^-, p_g$ to denote the probability density function $p_{\text{data}}(\mathbf{x}), p^+(\mathbf{x}), p^-(\mathbf{x}), p_g(\mathbf{x})$, respectively. Replacing $P(\mathbf{x})$ and $Q(\mathbf{x})$ with Eq. (3) and (4). Then we have the following:

$$\begin{aligned}
&D_f((1-\lambda)p_{\text{data}} + \lambda p^+||(1-\beta)p_g + \beta p^-) \\
&= \int_X ((1-\beta)p_g + \beta p^-) f(\frac{(1-\lambda)p_{\text{data}} + \lambda p^+}{(1-\beta)p_g + \beta p^-}) d\mathbf{x} \\
&= \int_X (1-\beta)p_g f(\frac{(1-\lambda)[\pi p^+ + (1-\pi)p^-] + \lambda p^+}{(1-\beta)p_g + \beta p^-}) d\mathbf{x} \\
&+ \int_X \beta p^- f(\frac{(1-\lambda)[\pi p^+ + (1-\pi)p^-] + \lambda p^+}{(1-\beta)p_g + \beta p^-}) d\mathbf{x}
\end{aligned} \tag{20}$$

Because of $Supp(p^+) \cap Supp(p^-) = \emptyset$, it implies that both p^+ and p^- cannot be greater than 0 at the same time. We denote $\int_{X^-} p_g d\mathbf{x} = t, \int_{X^+} p_g d\mathbf{x} = 1-t$. With the face that f is convex and Jensen's inequality, Eq.(20) can be transformed as follows:

$$\begin{aligned}
&= \int_{X^+} (1-\beta)p_g f(\frac{(1-\lambda)\pi p^+ + \lambda p^+}{(1-\beta)p_g}) d\mathbf{x} + [(1-\beta)t + \beta] \int_{X^-} \frac{1}{(1-\beta)t+\beta} [(1-\beta)p_g + \beta p^-] f(\frac{(1-\lambda)(1-\pi)p^-}{(1-\beta)p_g + \beta p^-}) d\mathbf{x} \\
&\geq \int_{X^+} (1-\beta)p_g f(\frac{(1-\lambda)\pi p^+ + \lambda p^+}{(1-\beta)p_g}) d\mathbf{x} + [(1-\beta)t + \beta] f(\frac{1}{(1-\beta)t+\beta} \int_{X^-} [(1-\beta)p_g + \beta p^-] \frac{(1-\lambda)(1-\pi)p^-}{(1-\beta)p_g + \beta p^-} d\mathbf{x}) \\
&= (1-\beta)(1-t) \int_{X^+} \frac{p_g}{1-t} f(\frac{(1-\lambda)\pi p^+ + \lambda p^+}{(1-\beta)p_g}) d\mathbf{x} + [(1-\beta)t + \beta] f(\frac{(1-\lambda)(1-\pi)}{(1-\beta)t+\beta}) \\
&\geq (1-\beta)(1-t) f(\int_{X^+} \frac{p_g}{1-t} \frac{(1-\lambda)\pi p^+ + \lambda p^+}{(1-\beta)p_g} d\mathbf{x}) + [(1-\beta)t + \beta] f(\frac{(1-\lambda)(1-\pi)}{(1-\beta)t+\beta}) \\
&= (1-\beta)(1-t) f(\frac{(1-\lambda)\pi + \lambda}{(1-\beta)(1-t)}) + [(1-\beta)t + \beta] f(\frac{(1-\lambda)(1-\pi)}{(1-\beta)t+\beta})
\end{aligned} \tag{21}$$

In order to facilitate the study of the properties of Eq. (21), is there a set of values for λ and β such that the corresponding function f is monotonically increasing with respect to t? f is set as $(u-1)^2$ which is the f-divergence corresponding to LS-GAN (Mao et al., 2017). Then we have the derivative of Eq. (21):

$$\frac{(Q_1^2 - (1-\beta)^2(1-t)^2)(1-\beta)}{(1-\beta)^2(1-t)^2} - \frac{(Q_2^2 - ((1-\beta)t+\beta)^2)(1-\beta)}{((1-\beta)t+\beta)^2}, \tag{22}$$

where the Q_1 denotes $(1-\lambda)\pi + \lambda$ while the Q_2 denotes $(1-\lambda)(1-\pi)$. In order to keep Eq. (22) larger than zero, it is equivalent to:

$$(Q_1^2 - (1-\beta)^2(1-t)^2)((1-\beta)t+\beta)^2 > (Q_2^2 - ((1-\beta)t+\beta)^2)(1-\beta)^2(1-t)^2 \tag{23}$$

Then we have:

$$(Q_1 + Q_2)(1-\beta)t > Q_2(1-\beta) - Q_1\beta \tag{24}$$

Due to domain of Q_1, Q_2, β, t, to make the inequality 24 hold equal to:

$$Q_2(1-\beta) - Q_1\beta < 0 \tag{25}$$

Then we have:

$$(1-\lambda)\pi + \beta + \lambda > 1 \tag{26}$$

We also study the case when f is set as $-logu$ which is one of f-divergence, So we have another form for the derivative of Eq. (21):

$$(1-\beta)log(\frac{(1-\beta)t+\beta}{(1-\lambda)(1-\pi)}) - (1-\beta)log(\frac{(1-\beta)(1-t)}{(1-\lambda)\pi+\lambda}) \tag{27}$$

It is easy to obtain that Eq. (27) is a monotonically increasing function with respect to $t \in [0,1]$ Its minimum value can be obtained as $t=0$, and Eq. (27) equal to:

$$(1-\beta)log(\frac{\beta}{(1-\lambda)(1-\pi)}) - (1-\beta)log(\frac{(1-\beta)}{(1-\lambda)\pi+\lambda}) = log(\frac{\beta[(1-\lambda)\pi+\lambda]}{(1-\beta)(1-\lambda)(1-\pi)}) \tag{28}$$

To make the minimum value of Eq. (27) greater than 0. It is equivalent to:

$$\frac{\beta[(1-\lambda)\pi+\lambda]}{(1-\beta)(1-\lambda)(1-\pi)} > 1 \tag{29}$$

Then we have:

$$(1-\lambda)\pi + \beta + \lambda > 1 \tag{30}$$

As long as $(1-\lambda)\pi + \beta + \lambda > 1$, Eq. (27) and Eq. (22) greater than 0 for all $t \in [0,1]$. It indicates that there is a set of values for λ and β such that Eq. (21) is a monotonically increasing function with respect to $t \in [0,1]$. Its minimum value can be obtained as $t=0$, and Eq. (21) equal to:

$$(1-\beta)f(\frac{(1-\lambda)\pi+\lambda}{1-\beta}) + \beta f(\frac{(1-\lambda)(1-\pi)}{\beta}) \tag{31}$$

Substituting p_g for p^+ in the Eq .(20), we have:

$$\begin{aligned}
&D((1-\lambda)p_{\text{data}} + \lambda p^+ || (1-\beta)p^+ + \beta p^-) \\
=&\int_X ((1-\beta)p^+ + \beta p^-) f(\frac{(1-\lambda)p_{\text{data}} + \lambda p^+}{(1-\beta)p^+ + \beta p^-}) d\mathbf{x} \\
=&\int_X ((1-\beta)p^+ + \beta p^-) f(\frac{(1-\lambda)[\pi p^+ + (1-\pi)p^-] + \lambda p^+}{(1-\beta)p^+ + \beta p^-}) d\mathbf{x} \\
=&\int_{X^+} (1-\beta)p^+ f(\frac{(1-\lambda)\pi p^+ + \lambda p^+}{(1-\beta)p^+}) d\mathbf{x} + \int_{X^-} \beta p^- f(\frac{(1-\lambda)(1-\pi)p^-}{\beta p^-}) d\mathbf{x} \\
=&(1-\beta)f(\frac{(1-\lambda)\pi+\lambda}{1-\beta}) + \beta f(\frac{(1-\lambda)(1-\pi)}{\beta})
\end{aligned} \tag{32}$$

Thus, the inequality in Eq .(32) will be equality only when $p_g = p^+$, indicating that the generator distribution will recover the normal-data distribution upon convergence.

□

B. Proof of Theorem 3.2

Theorem B.1. *If $Supp(p^+(\mathbf{x})) \cap Supp(p^-(\mathbf{x})) = \emptyset$ and $(1-\gamma)\beta > (1-\lambda)(1-\pi)+\gamma$ hold, as the function $f(\cdot)$ is set as $f(\cdot) = (u-1)^2$ or $-\log u$, we have $\arg\min\limits_{p_g(\mathbf{x})} D_f(\tilde{P}||\tilde{Q}) = p^+(\mathbf{x})$.*

Proof. For simplicity, we use $p_{\text{data}}, p^+, p^-, p_g$ to denote the probability density function $p_{\text{data}}(\mathbf{x}), p^+(\mathbf{x}), p^-(\mathbf{x}), p_g(\mathbf{x})$, respectively. To optimize the f-divergence of $\tilde{P}(\mathbf{x})$ and $\tilde{Q}(\mathbf{x})$ of Eq. (11) and Eq. (12), we have:

$$\arg\min_{p_g} D_f(\tilde{P}||\tilde{Q}) = \arg\min_{p_g} D_f((1-\gamma)[(1-\lambda)p_{\text{data}} + \lambda p^+] + \gamma p^- ||(1-\gamma)[(1-\beta)p_g + \beta p^-] + \gamma p^+) \tag{33}$$

where the D_f refers to the f-divergence. Then we have the following:

$$\begin{aligned}
&D_f((1-\gamma)[(1-\lambda)p_{\text{data}} + \lambda p^+] + \gamma p^- ||(1-\gamma)[(1-\beta)p_g + \beta p^-] + \gamma p^+) \\
&= \int_X ((1-\gamma)[(1-\beta)p_g + \beta p^-] + \gamma p^+) f(\frac{(1-\gamma)[(1-\lambda)p_{\text{data}} + \lambda p^+] + \gamma p^-}{(1-\gamma)[(1-\beta)p_g + \beta p^-] + \gamma p^+}) d\mathbf{x} \\
&= \int_X (1-\gamma)(1-\beta) f(\frac{(1-\gamma)[(1-\lambda)p_{\text{data}} + \lambda p^+] + \gamma p^-}{(1-\gamma)[(1-\beta)p_g + \beta p^-] + \gamma p^+}) d\mathbf{x} \\
&+ \int_X (1-\gamma)\beta p^- f(\frac{(1-\gamma)[(1-\lambda)p_{\text{data}} + \lambda p^+] + \gamma p^-}{(1-\gamma)[(1-\beta)p_g + \beta p^-] + \gamma p^+}) d\mathbf{x} \\
&+ \int_X \gamma p^+ f(\frac{(1-\gamma)[(1-\lambda)p_{\text{data}} + \lambda p^+] + \gamma p^-}{(1-\gamma)[(1-\beta)p_g + \beta p^-] + \gamma p^+}) d\mathbf{x}
\end{aligned} \tag{34}$$

Because of $Supp(p^+) \cap Supp(p^-) = \emptyset$, it implies that $p^+p^- = 0$. We have:

$$\begin{aligned}
&= \int_{X^+} [(1-\gamma)(1-\beta)p_g + \gamma p^+] f(\frac{(1-\gamma)[(1-\lambda)\pi p^+ + \lambda p^+]}{(1-\gamma)(1-\beta)p_g + \gamma p^+}) d\mathbf{x} \\
&+ \int_{X^-} [(1-\gamma)(1-\beta)p_g + (1-\gamma)\beta p^-] f(\frac{(1-\gamma)(1-\lambda)(1-\pi)p^- + \gamma p^-}{(1-\gamma)[(1-\beta)p_g + \beta p^-]}) d\mathbf{x}
\end{aligned} \tag{35}$$

We denote $\int_{X^-} p_g d\mathbf{x} = t, \int_{X^+} p_g d\mathbf{x} = 1-t$. With the fact that f is convex and Jensen's inequality, Eq .(35) can be transformed as follows:

$$\begin{aligned}
&= [(1-\gamma)(1-\beta)(1-t) + \gamma] \int_{X^+} \frac{(1-\gamma)(1-\beta)p_g + \gamma p^+}{(1-\gamma)(1-\beta)(1-t) + \gamma} f(\frac{(1-\gamma)[(1-\lambda)\pi p^+ + \lambda p^+]}{(1-\gamma)(1-\beta)p_g + \gamma p^+}) d\mathbf{x} \\
&+ [(1-\gamma)(1-\beta)t + (1-\gamma)\beta] \int_{X^-} \frac{(1-\gamma)[(1-\beta)p_g + \beta p^-]}{(1-\gamma)(1-\beta)t + (1-\gamma)\beta} f(\frac{(1-\gamma)(1-\lambda)(1-\pi)p^- + \gamma p^-}{(1-\gamma)[(1-\beta)p_g + \beta p^-]}) d\mathbf{x} \\
&\geq [(1-\gamma)(1-\beta)(1-t) + \gamma] f(\int_{X^+} \frac{(1-\gamma)(1-\beta)p_g + \gamma p^+}{(1-\gamma)(1-\beta)(1-t) + \gamma} \frac{(1-\gamma)[(1-\lambda)\pi p^+ + \lambda p^+]}{(1-\gamma)(1-\beta)p_g + \gamma p^+} d\mathbf{x}) \\
&+ [(1-\gamma)(1-\beta)t + (1-\gamma)\beta] f(\int_{X^-} \frac{(1-\gamma)[(1-\beta)p_g + \beta p^-]}{(1-\gamma)(1-\beta)t + (1-\gamma)\beta} \frac{(1-\gamma)(1-\lambda)(1-\pi)p^- + \gamma p^-}{(1-\gamma)[(1-\beta)p_g + \beta p^-]} d\mathbf{x}) \\
&= [(1-\gamma)(1-\beta)(1-t) + \gamma] f(\frac{(1-\gamma)[(1-\lambda)\pi + \lambda]}{(1-\gamma)(1-\beta)(1-t) + \gamma}) \\
&+ [(1-\gamma)(1-\beta)t + (1-\gamma)\beta] f(\frac{(1-\gamma)(1-\lambda)(1-\pi) + \gamma}{(1-\gamma)(1-\beta)t + (1-\gamma)\beta})
\end{aligned} \tag{36}$$

In order to facilitate the study of the properties of Eq. (36), is there a set of values for λ and β such that the corresponding function f is monotonically increasing with respect to t? f is set as $(u-1)^2$ corresponding to LS-GAN (Mao et al., 2017). Then we have the derivative of Eq. (36):

$$(1-\beta)(1-\gamma)\left[\left(\frac{Q_1}{(1-\beta)(1-\gamma)(1-t)+\gamma}\right)^2 - 1\right] - (1-\beta)(1-\gamma)\left[\left(\frac{Q_2}{(1-\gamma)[(1-\beta)t+\beta]}\right)^2 - 1\right], \tag{37}$$

where the Q_1 denotes $(1-\gamma)((1-\lambda)\pi+\lambda)$ while the Q_2 denotes $(1-\gamma)(1-\lambda)(1-\pi)+\gamma$. In order to keep Eq. (37) larger than zero, it is equivalent to:

$$Q_1(1-\gamma)[(1-\beta)t+\beta] > Q_2((1-\beta)(1-\gamma)(1-t)+\gamma) \tag{38}$$

Then we have:

$$(Q_1+Q_2)(1-\gamma)(1-\beta)t > Q_2(1-\beta)(1-\gamma) - Q_1\beta + \gamma Q_2 \tag{39}$$

Due to domain of Q_1, Q_2, β, t, to make the Inequality (39) hold equal to:

$$Q_2(1-\beta)(1-\gamma) - Q_1\beta + \gamma Q_2 < 0 \tag{40}$$

Then we have:

$$(1-\gamma)((1-\lambda)\pi+\beta+\lambda) > 1 \tag{41}$$

We also study the case when f is set as $-logu$ which is one of f-divergence. So we have another form for the derivative of Eq. (36):

$$(1-\gamma)(1-\beta)log(\frac{(1-\gamma)[(1-\beta)t+\beta]}{\gamma+(1-\lambda)(1-\pi)(1-\gamma)}) - (1-\beta)(1-\gamma)log(\frac{(1-\beta)(1-\gamma)(1-t)+\gamma}{[(1-\lambda)\pi+\lambda](1-\gamma)}) \tag{42}$$

It is easy to obtain that Eq. (42) is a monotonically increasing function with respect to $t \in [0,1]$ Its minimum value can be obtained as $t=0$, and Eq. (42) equal to:

$$(1-\gamma)(1-\beta)log(\frac{(1-\gamma)\beta}{\gamma+(1-\lambda)(1-\pi)(1-\gamma)}) - (1-\beta)(1-\gamma)log(\frac{(1-\beta)(1-\gamma)+\gamma}{[(1-\lambda)\pi+\lambda](1-\gamma)}) \tag{43}$$

To make the minimum value of Eq. (27) greater than 0. It is equivalent to:

$$\frac{(1-\gamma)^2\beta[(1-\lambda)\pi+\lambda]}{[\gamma+(1-\lambda)(1-\pi)(1-\gamma)][(1-\beta)(1-\gamma)+\gamma]} > 1 \tag{44}$$

Then we have the following:

$$(1-\gamma)((1-\lambda)\pi+\beta+\lambda) > 1 \tag{45}$$

Due to γ indicates the probability of labels flipping under the overall which is a small value and $\pi > 0.5$, inequality (44) is easily satisfied, i.e., when $\pi = 0.8, \gamma = 0.1, \lambda+\beta = 1$. Therefore, Eq. (42) and Eq. (37) is greater than 0 for all $t \in [0,1]$. It indicates that there is a set of values for λ and β such that Eq. (21) is a monotonically increasing function with respect to $t \in [0,1]$. Its minimum value can be obtained as $t=0$, and Eq. (36) equal to:

$$[(1-\gamma)(1-\beta)+\gamma]f(\frac{(1-\gamma)[(1-\lambda)\pi+\lambda]}{(1-\gamma)(1-\beta)+\gamma}) + (1-\gamma)\beta f(\frac{(1-\gamma)(1-\lambda)(1-\pi)+\gamma}{(1-\gamma)\beta}) \tag{46}$$

Substituting p_g for p^+ in the Eq .(34), we have:

$$\begin{aligned}
&D((1-\gamma)[(1-\lambda)p_{\text{data}}+\lambda p^+]+\gamma p^- || (1-\gamma)[(1-\beta)p_g+\beta p^-]+\gamma p^+) \\
&= \int_X ((1-\gamma)[(1-\beta)p^+ + \beta p^-]+\gamma p^+) f(\frac{(1-\gamma)[(1-\lambda)p_{\text{data}}+\lambda p^+]+\gamma p^-}{(1-\gamma)[(1-\beta)p^+ + \beta p^-]+\gamma p^+}) d\mathbf{x} \\
&= \int_{X^+} [(1-\gamma)(1-\beta)p^+ + \gamma p^+] f(\frac{(1-\gamma)[(1-\lambda)\pi p^+ + \lambda p^+]}{(1-\gamma)(1-\beta)p^+ + \gamma p^+}) d\mathbf{x} \\
&+ \int_{X^-} (1-\gamma)\beta p^- f(\frac{(1-\gamma)(1-\lambda)(1-\pi)p^- + \gamma p^-}{(1-\gamma)\beta p^-}) d\mathbf{x} \\
&= [(1-\gamma)(1-\beta)+\gamma]f(\frac{(1-\gamma)[(1-\lambda)\pi+\lambda]}{(1-\gamma)(1-\beta)+\gamma}) + (1-\gamma)\beta f(\frac{(1-\gamma)(1-\lambda)(1-\pi)+\gamma}{(1-\gamma)\beta})
\end{aligned} \tag{47}$$

Thus, the inequality in Eq .(36) will be equality only when $p_g = p^+$, indicating that the generator distribution will recover the normal-data distribution upon convergence. □

C. Proof of Theorem 3.3

Theorem C.1. *Denote* $\kappa_1 = (1-\gamma)(1-\lambda)\pi + \lambda$, $\kappa_2 = (1-\gamma)(1-\lambda)(1-\pi)(1-\alpha)$ *and* $\kappa_3 = (1-\gamma)(1-\beta)+\gamma$*. If* $Supp(p^+(\mathbf{x})) \cap Supp(p^-(\mathbf{x})) = \emptyset, Supp(p_u^-(\mathbf{x})) \cap Supp(p_c^-(\mathbf{x})) = \emptyset, (1-\gamma)(1-\alpha+\beta+\alpha\pi+\alpha\lambda-\alpha\lambda\pi) > 1$ *and* $\kappa_1\kappa_3 > \gamma(\kappa_1+\kappa_2)$ *hold, as the function* $f(\cdot)$ *is set as* $f(\cdot) = (u-1)^2$ *or* $-\log u$*, we have*

$$\begin{aligned} arg\min_{p_g(\mathbf{x})} D_f(\tilde{P}||\tilde{Q}) = & \frac{\kappa_1\kappa_3 - \gamma(\kappa_1+\kappa_2)}{(1-\gamma)(1-\beta)(\kappa_1+\kappa_2)} p^+(\mathbf{x}) \\ & + \frac{\kappa_2\kappa_3}{(1-\gamma)(1-\beta)(\kappa_1+\kappa_2)} p_u^-(\mathbf{x}). \end{aligned}$$

Based on Theorem 3.2, we can easily analyse to obtain Theorem 3.3.

Proof. For simplicity, we use $p_{\text{data}}, p^+, p^-, p_u^-, p_c^-, p_g$ to denote the probability density function $p_{\text{data}}(\mathbf{x}), p^+(\mathbf{x}), p^-(\mathbf{x}), p_u^-(\mathbf{x}), p_c^-(\mathbf{x}), p_g(\mathbf{x})$, respectively. To optimize the f-divergence of $\tilde{P}$ and $\tilde{Q}$ of Eq. (17) and Eq. (18), we have:

$$arg\min_{p_g} D_f(\tilde{P}||\tilde{Q}) = arg\min_{p_g} D_f((1-\gamma)[(1-\lambda)p_{\text{data}} + \lambda p^+] + \gamma p_c^- || (1-\gamma)[(1-\beta)p_g + \beta p_c^-] + \gamma p^+) \quad (48)$$

The target of Eq. (48) is to find an appropriate p_g to minimize the f-divergence between distribution $\tilde{P}(\mathbf{x}) : (1-\gamma)[(1-\lambda)p_{\text{data}} + \lambda p^+] + \gamma p_c^-$ and distribution $\tilde{Q}(\mathbf{x}) : (1-\gamma)[(1-\beta)p_g + \beta p_c^-] + \gamma p^+$.

Distribution $\tilde{P}(\mathbf{x})$ and $\tilde{Q}(\mathbf{x})$ can be rewritten as

$$\begin{aligned} \tilde{P}(\mathbf{x}) =& ((1-\gamma)(1-\lambda)\pi + \lambda)p^+ + (1-\gamma)(1-\lambda)(1-\pi)p^- + \gamma p_c^- \\ \tilde{Q}(\mathbf{x}) =& (1-\gamma)(1-\beta)p_g + \gamma p^+ + (1-\gamma)\beta p_c^- \end{aligned}$$

Because $p^- = (1-\alpha)p_u^- + \alpha p_c^-$, we have:

$$\begin{aligned} \tilde{P}(\mathbf{x}) =& ((1-\gamma)(1-\lambda)\pi + \lambda)p^+ + (1-\gamma)(1-\lambda)(1-\pi)(1-\alpha)p_u^- + [(1-\gamma)(1-\lambda)(1-\pi)\alpha + \gamma]p_c^- \\ \tilde{Q}(\mathbf{x}) =& (1-\gamma)(1-\beta)p_g + \gamma p^+ + (1-\gamma)\beta p_c^- \end{aligned}$$

Let $\hat{p}_{\text{data}} = ((1-\gamma)(1-\lambda)\pi + \lambda)p^+ + (1-\gamma)(1-\lambda)(1-\pi)(1-\alpha)p_u^-$ and $\hat{p}_g = (1-\gamma)(1-\beta)p_g + \gamma p^+$. Then, distribution $\tilde{P}(\mathbf{x})$ and $\tilde{Q}(\mathbf{x})$ can be rewritten as:

$$\begin{aligned} \tilde{P}(\mathbf{x}) &= \hat{p}_{\text{data}} + [(1-\gamma)(1-\lambda)(1-\pi)\alpha + \gamma]p_c^- \\ \tilde{Q}(\mathbf{x}) &= \hat{p}_g + (1-\gamma)\beta p_c^- \end{aligned}$$

Based on Theorem 3.1, when $(1-\gamma)(1-\alpha+\beta+\alpha\pi+\alpha\lambda-\alpha\lambda\pi) > 1$ holds, we have $(1-\gamma)\beta > (1-\gamma)(1-\lambda)(1-\pi)\alpha+\gamma$. It indicates that no more density from p_g will be assigned to p_c^-. Therefore, the seen contamination is erased from the generator distribution. With κ_1 denotes $(1-\gamma)(1-\lambda)\pi + \lambda$, κ_2 denotes $(1-\gamma)(1-\lambda)(1-\pi)(1-\alpha)$, and κ_3 denotes $(1-\gamma)(1-\beta)+\gamma$, we have:

$$\hat{p}_g = (1-\gamma)(1-\beta)p_g + \gamma p^+ = \frac{\int_X (1-\gamma)(1-\beta)p_g + \gamma p^+ dx}{\int_X \kappa_1 p^+ + \kappa_2 p_u^- dx}(\kappa_1 p^+ + \kappa_2 p_u^-) = \frac{\kappa_3}{\kappa_1+\kappa_2}(\kappa_1 p^+ + \kappa_2 p_u^-) \quad (49)$$

Then we can obtain the generator distribution as follow:

$$p_g = \frac{\kappa_1\kappa_3 - \gamma(\kappa_1+\kappa_2)}{(1-\gamma)(1-\beta)(\kappa_1+\kappa_2)} p^+(\mathbf{x}) + \frac{\kappa_2\kappa_3}{(1-\gamma)(1-\beta)(\kappa_1+\kappa_2)} p_u^-(\mathbf{x}) \quad (50)$$

With $\kappa_1\kappa_3 > \gamma(\kappa_1+\kappa_2)$, we can preserve the normal-data distribution in the generator. Therefore, Theorem 3.3 is proved. □

D. Details of Dataset

D.1. Details of Toy Data

MNIST A dataset of 60000 training and 10000 testing 28×28 handwritten digits from 10 classes. The first three classes are chosen as normal data while the rest seven classes are viewed as anomalies.

Fashion-MNIST A dataset of 60000 training and 10000 testing 28×28 clothing images from 10 classes. The first three classes are chosen as normal data while the rest seven classes are viewed as anomalies.

CIFAR10 A dataset of 50000 training and 10000 testing 32×32×3 RGB images from 10 classes. The first three classes are chosen as normal data while the rest seven classes are viewed as anomalies.

20NEWS From ADBench (Han et al., 2022), we use the anomaly detection consist of 20news features extracted by BERT(Devlin et al., 2019). The details of this dataset are shown in Table 9

Table 9. The details of 20NEWS anomaly detection dataset

Label	Category	#Features	#Samples
Normal	Computer	768	3090
	Recreation	768	2514
	Science	768	2497
Anomaly	Miscellaneous	768	615
	Politics	768	1657
	Religion	768	1532

D.2. Details of Anomaly Datasets

UNSW-NB15 This dataset is about cyber defense containing nine types of network attacks. We select four of them to form this anomaly detection dataset as same as (Pang et al., 2023). The details of this dataset are shown in Table 10.

Table 10. The details of UNSW-NB15

Category	#Features	#Samples	#Anomaly	%Anomaly
Backdoor	196	95329	2329	2.44%
DoS	196	109353	16353	14.95%
Fuzzers	196	96000	3000	3.13%
Reconnaissance	196	106987	13987	13.07%

HAR This dataset is from human activity recognition. The details of this dataset are shown in Table 11.

Table 11. The details of 20NEWS anomaly detection dataset

Label	Category	#Features	#Samples
Normal	Walking	561	1226
	Sitting	561	1286
	Laying	561	1407
	Standing	561	1374
Anomaly	Downstairs	561	986
	Upstairs	561	1073

Classic Anomaly Datasets The details of classical anomaly datasets are shown in Table 12. Yelp and Amazon datasets are from ADBench (Han et al., 2022), whose features are extracted by BERT (Devlin et al., 2019).

Table 12. The details of nine classical datasets

Dataset	#Features	#Samples	#Anomaly	%Anomaly
Arrhythmia	274	452	66	14.60%
Cardio	21	1831	176	9.61%
Satellite	36	6435	2036	31.6%
Satimage-2	36	5803	71	1.22%
Shuttle	9	49097	3511	7.15%
Thyroid	6	3772	93	2.47%
Bank	62	41188	4640	11.27%
Amazon	768	10000	500	5.00%
Yelp	768	10000	500	5.00%

E. Training Details

For the image benchmarks: MNIST and FMNIST, we employ DCGAN architecture, which is implemented on Pytorch. Adam optimizer is selected for optimization during training. The learning rate of generator and encoder is set as 0.001, while the learning rate of discriminator is set as 0.0001. For the tabular anomaly detection datasets, 3-layer MLP architecture is employed. The test dataset is splitted into validation and test datasets with a proportion of 20% and 80%.

For the baselines, except for the results of AA-BiGAN, which are reported by running its source code, the above methods are all implemented in DeepOD.

F. Additional Experimental Results

F.1. Complete Tables of Experimental Results

Table 13 demonstrates the complete experimental results under different numbers of collected anomalous types on four datasets. Due to the HAR dataset only having two anomalous types, we have not conducted this experiment on the HAR dataset. The setting of ϵ_p is fixed at 20% for toy datasets and 5% for UNSW-NB15, while ϵ_a and ϵ_n are fixed at 5% and 1% for all of them. From Table 13, our method outperforms other baselines in all levels.

Table 13. AUROC performance under different numbers of collected types in the auxiliary anomalous dataset.

Dataset	k	DeepSAD	FeaWAD	RoSAS	PReNet	AA-BiGAN	Ours
MNIST	1	70.3	66.5	72.8	56.1	84.8	**90.6**
	2	73.4	67.7	76.4	57.2	86.6	**92.1**
	3	75.2	68.7	78.5	57.9	87.8	**94.2**
	4	81.1	69.0	80.0	59.4	90.4	**95.0**
FMNIST	1	71.0	66.5	67.6	55.1	80.7	**85.2**
	2	71.7	66.7	68.1	55.7	82.3	**88.0**
	3	79.1	71.8	72.4	57.2	86.7	**89.2**
	4	86.2	77.5	80.6	62.8	87.5	**90.0**
20NEWS	1	63.5	52.4	64.6	52.9	70.7	**74.0**
	2	67.2	53.6	67.8	54.0	71.2	**75.2**
	3	64.2	54.4	72.1	53.1	72.7	**76.7**
UNSW	1	87.3	82.4	84.0	60.7	88.0	**92.0**
	2	91.7	87.3	94.8	66.1	88.7	**95.3**
	3	92.6	91.2	95.1	71.9	92.7	**95.6**
	4	93.0	92.8	96.2	74.2	94.2	**96.8**

Table 14 demonstrates the complete experimental results on nine tabular anomaly detection datasets, including the unsupervised methods and self-supervised method. From Table 14, our method still outperforms other baselines. This result indicates the superiority of our method.

Table 14. AUROC on nine classic anomaly detection datasets.

Dataset	Deep SVDD	DIF	SLAD	ICL	Deep SAD	Fea WAD	Ro SAS	PRe Net	AA-Bi GAN	SOEL	Ours
Arrhythmia	73.5	79.4	78.1	77.7	78.5	73.8	61.2	64.4	79.9	83.6	**84.8**
Cardio	87.7	96.1	96.8	89.5	96.1	73.9	91.7	90.7	97.6	95.7	**99.4**
Satellite	71.9	76.4	80.5	81.5	83.8	89.6	86.3	78.5	85.4	85.9	**91.8**
Satimage-2	65.4	99.6	97.7	75.3	96.0	**99.9**	99.2	99.1	99.7	99.7	**99.9**
Shuttle	73.0	99.1	99.3	94.5	99.4	97.9	98.6	99.1	99.0	**99.7**	99.5
Thyroid	89.0	94.9	98.7	59.8	**99.6**	65.6	**99.6**	96.2	99.1	99.1	**99.6**
Bank	55.6	67.0	73.0	69.4	75.5	61.1	89.8	62.6	87.5	89.9	**90.7**
Amazon	53.9	56.5	62.6	59.4	89.9	58.2	85.5	76.2	85.7	**90.5**	89.7
Yelp	59.3	60.9	65.8	63.4	90.5	55.9	92.2	79.0	89.3	91.3	**92.6**