

# Memorize Early, Then Query: Inlier-Memorization-Guided Active Outlier Detection

Minseo Kang<sup>1</sup>, Seunghwan Park<sup>2</sup>, Dongha Kim<sup>1,3\*</sup>

<sup>1</sup>Department of Statistics, Sungshin Women’s University

<sup>2</sup>Information Statistics, Kangwon National University

<sup>3</sup>Center for Data Science, Sungshin Women’s University

krystalms0306@gmail.com, stat.shpark@kangwon.ac.kr, dongha0718@sungshin.ac.kr

## Abstract

Outlier detection (OD) aims to identify abnormal instances, known as outliers or anomalies, by learning typical patterns of normal data, or inliers. Performing OD under an unsupervised regime—without any information about anomalous instances in the training data—is challenging. A recently observed phenomenon, known as the *inlier-memorization (IM) effect*, where deep generative models (DGMs) tend to memorize inlier patterns during early training, provides a promising signal for distinguishing outliers. However, existing unsupervised approaches that rely solely on the IM effect still struggle when inliers and outliers are not well-separated or when outliers form dense clusters. To address these limitations, we incorporate *active learning* to selectively acquire informative labels, and propose *IMBoost*, a novel framework that explicitly reinforces the IM effect to improve outlier detection. Our method consists of two stages: 1) a *warm-up* phase that induces and promotes the IM effect, and 2) a *polarization* phase in which actively queried samples are used to maximize the discrepancy between inlier and outlier scores. In particular, we propose a novel query strategy and tailored loss function in the polarization phase to effectively identify informative samples and fully leverage the limited labeling budget. We provide a theoretical analysis showing that the IMBoost consistently decreases inlier risk while increasing outlier risk throughout training, thereby amplifying their separation. Extensive experiments on diverse benchmark datasets demonstrate that IMBoost not only significantly outperforms state-of-the-art active OD methods but also requires substantially less computational cost.

## Introduction

**Outlier Detection** An outlier, or anomaly, refers to a data point that deviates significantly from the majority of typical observations (Pang et al. 2021). Such instances can contaminate data integrity and negatively impact downstream analyses, such as regression and classification tasks. Therefore, identifying and removing outliers is essential for ensuring reliable data-driven modeling. Outlier detection (OD) is the task of detecting these abnormal instances. In addition to serving as a pre-processing step for downstream supervised learning applications, OD itself plays a critical role across

various domains, including fraud detection, network intrusion detection, medical diagnosis, and sensor data monitoring (Nassif et al. 2021).

OD tasks can be broadly categorized into three settings based on the availability of anomaly-related information in the training data. Supervised OD (SOD) assumes access to labeled data, where each instance is explicitly annotated as either an inlier or an outlier. Semi-supervised OD (SSOD), also referred to as out-of-distribution (OOD) detection, assumes that all training data are inliers and builds models using only these normal samples. And unsupervised OD (UOD) operates without any label information, relying on training data that may include both inliers and outliers without explicit annotation. In general, many real-world anomaly detection tasks belong to UOD because outliers in large datasets are usually unknown beforehand.

To address UOD tasks, a couple of recent studies have leveraged the so-called *inlier-memorization (IM) effect*, a phenomenon where DGMs tend to learn inlier patterns *earlier* than outlier patterns during training. This arises because inliers are typically more *prevalent* and *densely* distributed, making it more effective to reduce their loss values first when minimizing the overall loss. This early-phase memorization provides a useful signal for distinguishing inliers from outliers without supervision. ODIM (Kim et al. 2024) is the first to exploit this effect by identifying an optimal training point and using per-sample loss as an outlier score. ALTBI (Cho et al. 2025) further enhances the IM effect through adaptive mini-batch-size scheduling and trimmed loss optimization with a dynamic threshold.

Despite their effectiveness, these methods operate in a fully unsupervised manner, which leads to several limitations. First, when inliers and outliers are closely distributed, it becomes inherently difficult to distinguish them without any label supervision. Second, if outliers are more densely clustered than inliers, DGMs may incorrectly memorize outliers before inliers during early training, reversing the intended IM effect and resulting in performance degradation. This issue is especially pronounced when outliers originate from a single point due to a measurement error, causing the model to focus on them prematurely. These limitations underscore the need for a new paradigm that can address such failure modes of unsupervised learning.

\*Corresponding author.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

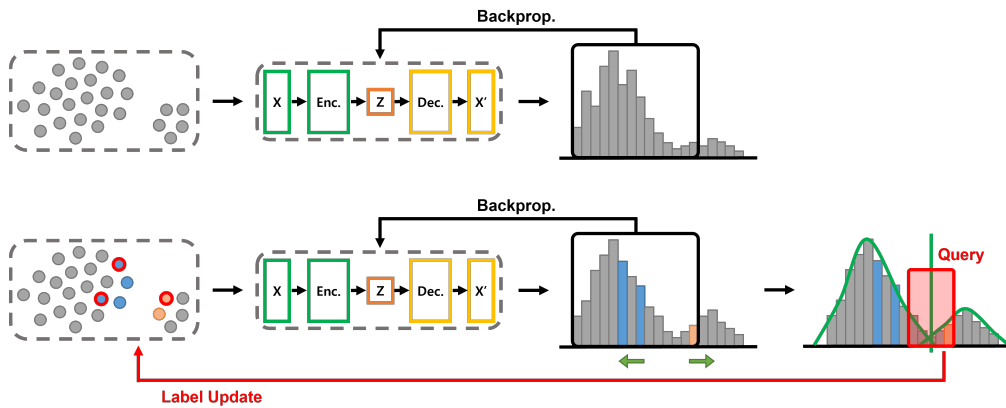


Figure 1: An illustration of IMBoost: (Top) *warm-up* phase and (Down) *polarization* phase.

**Overview of Our Method** As mentioned above, access to anomaly-related information is essential for accurately identifying outliers; however, obtaining such labels is often expensive and time-consuming. To address this challenge, we adopt the idea of *active learning* and propose *IMBoost*, a novel active outlier detection framework that reinforces the IM effect to enhance detection performance. IMBoost operates in two phases: 1) an *warm-up* phase that induces and promotes the IM effect, and 2) a *polarization* phase in which actively queried samples are used to maximize the discrepancy between inlier and outlier scores. In particular, during the polarization phase, we introduce a new querying strategy and a loss function to effectively select informative labeled samples and fully exploit their benefits. An illustration of the IMBoost is shown in Figure 1.

The IMBoost offers a couple of appealing features. It consistently outperforms existing methods by a significant margin in identifying outliers. Through extensive experiments on nearly 60 benchmark datasets, we demonstrate that the IMBoost achieves the best performance among recent approaches with large margins. We also provide a rigorous theoretical analysis, showing that our method progressively increases the separation between inlier and outlier scores, particularly during the polarization phase. Furthermore, by leveraging the IM effect, IMBoost achieves this performance with considerably lower computational cost, as it does not require full model training. These findings highlight the IMBoost as a promising approach for practical OD tasks.

The remainder of this paper is organized as follows. First, we review related work on outlier detection, with a particular focus on approaches that incorporate active learning. We then present a detailed description of the proposed IMBoost framework and discuss its theoretical properties. Next, we report extensive experimental results, including performance comparisons and ablation studies. Finally, we conclude with a summary of our findings and outline potential directions for future work.

## Related Works

We review algorithms for both SSOD and active OD, as the former are often used as base frameworks in active OD tasks.

A representative SSOD approach is support vector data description (SVDD, Tax and Duin (2004)), which models normal data by enclosing it within a hypersphere in feature space. Deep SVDD (DSVDD, Ruff et al. (2018)) extends SVDD by mapping data into a latent space and learning a hypersphere that encloses normal samples. Anomalies are identified as samples that fall outside this sphere. Deep SAD (DSAD, Ruff et al. (2020)) extends Deep SVDD by incorporating a few labeled anomalies, pushing them away from the center while keeping normal samples close.

In real-world scenarios, obtaining a clean dataset composed only of normal samples is challenging due to the high cost of human annotation. Active learning strategies have been introduced to selectively query anomaly labels for informative samples and improve outlier detection performance (Trittenbach, Englhardt, and Böhm 2021). Many existing strategies adopt a greedy query rule, selecting the top-1 or top-K samples with the highest anomaly scores (Lamba and Akoglu 2019; Das et al. 2016). However, such approaches may be suboptimal in active OD tasks, as some lower-ranked instances may provide more valuable information (Zha et al. 2020).

Instead, adaptive query strategies that prioritizes highly uncertain samples have been proposed to provide more informative guidance. Kim et al. (2023) introduced an active OD framework based to DSVDD with an uncertainty-based querying mechanism. The method adaptively updates the latent boundary using feedback from previously queried samples, reducing sensitivity to hyperparameters. They also employed noise contrastive estimation (NCE, Gutmann and Hyvärinen (2010)) to iteratively refine the decision boundary using both labeled normal and abnormal samples.

## Proposed Method

### Preliminaries

**Mathematical Notations** We begin by introducing the notations and definitions used throughout the paper. Let  $X_1, \dots, X_n \in \mathcal{X} \sim \mathbb{P}$  be  $n$  independent input samples drawn from the true data distribution  $\mathbb{P}$ , and let  $\mathcal{D}^{\text{tr}} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$  denote their observed realization, i.e., train-

ing data. We assume that inlier-outlier labels are unavailable unless they are explicitly acquired through oracle queries.

The data distribution  $\mathbb{P}$  is modeled as a mixture of inlier and outlier distributions, i.e.,  $\mathbb{P} = (1 - p_o) \cdot \mathbb{P}_i + p_o \cdot \mathbb{P}_o$ , where  $\mathbb{P}_i$  and  $\mathbb{P}_o$  denote the inlier and outlier distributions, respectively, and  $p_o \in (0, 1)$  is the outlier ratio. We define the supports of  $\mathbb{P}_i$  and  $\mathbb{P}_o$  as  $\mathcal{X}_i$  and  $\mathcal{X}_o$ , respectively, so that the total input space is  $\mathcal{X} = \mathcal{X}_i \cup \mathcal{X}_o$ . We assume disjoint supports for inliers and outliers, i.e.,  $\mathcal{X}_i \cap \mathcal{X}_o = \emptyset$ , which is a natural assumption in OD problems.

For a given sample  $\mathbf{x}$ , we define the per-sample loss with respect to a DGM as  $l(\theta; \mathbf{x})$ , such as the negative log-likelihood, where  $\theta \in \Theta$  represents the model parameters of the DGM. We assume that  $l(\theta; \mathbf{x})$  is differentiable and bounded within the interval  $[0, 1]$  for all  $\mathbf{x} \in \mathcal{X}$  and  $\theta \in \Theta$ .

We denote by  $\mathbb{E}_i$  and  $\mathbb{E}_o$  the expectation operators with respect to  $\mathbb{P}_i$  and  $\mathbb{P}_o$ , respectively. The corresponding risk functions are defined as  $R_i(\theta) = \mathbb{E}_i[l(\theta; X)]$  and  $R_o(\theta) = \mathbb{E}_o[l(\theta; X)]$ , which represent the expected loss over inlier and outlier distributions, respectively. Finally, we denote the minimizer of the inlier risk as  $\theta_*$ , i.e.,  $\theta_* = \operatorname{argmin}_{\theta} R_i(\theta)$ . We assume  $R_i(\theta_*) = 0$ .

**Inlier-Memorization Effect** The *inlier-memorization (IM) effect* is a phenomenon observed in likelihood-based DGMs, where the model tends to learn inlier patterns *earlier* than outlier patterns during training (Kim et al. 2024). As training progresses, outliers are eventually learned as well, but the initial gap in memorization offers a useful signal for distinguishing them without supervision.

Kim et al. (2024) is a pioneering work that first observed the IM effect and applied it to develop an UOD method called ODIM. ODIM trains a DGM for a limited number of updates and uses the per-sample loss as the outlier score, selecting the optimal update step where the IM effect is most prominent. To find this point, ODIM monitors the bimodality of the loss distribution over time by fitting a two-clustered Gaussian mixture model (GMM) and measuring the separation between clusters.

As a follow-up study, ALTBI (Cho et al. 2025) adopts a more dynamic approach by continuously enhancing the IM effect throughout training. It gradually increases the batch size and employs trimmed loss optimization to filter out high-loss (potentially outlier) samples. This strategy helps enlarge the memorization gap between inliers and outliers during training and improves detection performance.

## Loss Function Description of IMBoost

Our procedure consists of two phases: 1) *warm-up* and 2) *polarization*. In the *warm-up* phase, we pre-train a DGM to induce the IM effect and further refine it to promote this effect, in an unsupervised manner. In the *polarization* phase, we employ an active learning scheme to amplify the discrepancy between the inlier and outlier loss distributions.

**Phase 1 (Warm-up)** We begin by training a DGM using a conventional loss function to induce the IM effect. At each iteration, we draw a mini-batch  $\mathcal{D}_0$  of size  $n_0$  and compute

the loss function as follows:

$$\widehat{R}_0(\theta) = \frac{1}{|\mathcal{D}_0|} \sum_{\mathbf{x} \in \mathcal{D}_0} l(\theta; \mathbf{x}). \quad (1)$$

The update process using (1) is iterated for  $T_0$  times.

Next, we train the DGM to further promote the IM effect. We adopt using the strategy proposed in ALTBI (Cho et al. 2025), which gradually increases the mini-batch size and employs a trimmed mean loss with an adaptive threshold. At each iteration  $t$ , we sample a mini-batch  $\mathcal{D}_t \subset \mathcal{D}^u$  whose size increases exponentially with  $t$ , i.e.,  $|\mathcal{D}_t| = n_0 \gamma^{t-1}$  for some constant  $\gamma > 1$ . Instead of computing the loss over all samples in  $\mathcal{D}_t$ , we use a *trimmed loss function* defined as:

$$\widehat{R}_t(\theta) = \frac{\sum_{\mathbf{x} \in \mathcal{D}_t} l(\theta; \mathbf{x}) \cdot I(l(\theta; \mathbf{x}) \leq \tau_t)}{\sum_{\mathbf{x}' \in \mathcal{D}_t} I(l(\theta; \mathbf{x}') \leq \tau_t)}, \quad (2)$$

where  $\tau_t > 0$  is an adaptive threshold. We train the DGM with this trimmed loss (2) for  $T_1$  updates.

**Phase 2 (Polarization)** After the warm-up phase ends, we adopt an active learning scheme. Specifically, at each iteration, we query a small subset of samples to the oracle, resulting in two disjoint labeled subsets:  $\mathcal{I}_t$  for inliers and  $\mathcal{O}_t$  for outliers. In practice, we do not query new samples at every iteration; instead, we query them occasionally and reuse the labeled ones, gradually expanding both the inlier and outlier sets. Along with a randomly drawn unlabeled mini-batch  $\mathcal{D}_t \subset \mathcal{D}^u$  of size  $|\mathcal{D}_t| = n_0 \gamma^{t-1}$  (with the same  $n_0$  and  $\gamma$  used in the phase 1), we use the following loss function:

$$\widehat{R}_t^a(\theta) = \widehat{R}_t(\theta) + \lambda_{1,t} \cdot \widehat{R}_t^i(\theta) - \lambda_{2,t} \cdot \widehat{R}_t^o(\theta), \quad (3)$$

where

$$\widehat{R}_t^i(\theta) = \frac{1}{|\mathcal{I}_t|} \sum_{\mathbf{x} \in \mathcal{I}_t} l(\theta; \mathbf{x}), \quad \widehat{R}_t^o(\theta) = \frac{1}{|\mathcal{O}_t|} \sum_{\mathbf{x} \in \mathcal{O}_t} l(\theta; \mathbf{x}),$$

and  $\lambda_{1,t}, \lambda_{2,t} > 0$  are hyperparameters controlling the relative emphasis on inliers and outliers. We perform  $T_2$  update steps using the loss function (3). After training, we compute the outlier score for a given input  $\mathbf{x}$  as its per-sample loss, i.e.,  $l(\theta; \mathbf{x})$ . A sample is regarded as an outlier if it has a high score, and as an inlier otherwise.

The loss function in (3) can be viewed as an extended version of (2), augmented with additional two terms computed from labeled inliers and outliers, respectively. Minimizing the inlier loss  $\widehat{R}_t^i(\theta)$  is expected to promote a decrease in the inlier risk, while maximizing the outlier loss  $\widehat{R}_t^o(\theta)$  is intended to increase the outlier risk. In the theoretical analysis, we rigorously validate that when the hyperparameters  $\lambda_{1,t}, \lambda_{2,t}$  are set proportional to  $\gamma^{-t}$ , the resulting optimization process indeed enlarges the gap between inliers and outliers, thereby enhancing the discriminative power of the per-sample loss as an outlier score.

## Practical Training Techniques for IMBoost

In this section, we discuss several practical considerations that improve the performance and stability of our method during implementation. The effectiveness of each technique is evaluated through detailed experiments, as presented in the ablation studies.

**Implementation of  $l(\theta; \mathbf{x})$**  Motivated by the empirical success of IM-effect-based approaches such as ODIM (Kim et al. 2024) and ALTBI (Cho et al. 2025), we adopt the negative log-likelihood as our loss function  $l(\theta; \mathbf{x})$ . Various likelihood-based DGMs have been proposed, including VAE-based methods (Kingma and Welling 2013), normalizing flow (NF)-based models (Kingma and Dhariwal 2018), and diffusion-based models (Song et al. 2021). Among them, we employ the importance-weighted autoencoder (IWAE) framework (Burda, Grosse, and Salakhutdinov 2016), a VAE-based that provides a tight and tractable *lower bound* on the log-likelihood and has demonstrated strong empirical performance in prior work.

Unlike the theoretical assumption that  $l(\theta; \mathbf{x})$  lies within the range  $[0, 1]$ , the upper bound of the IWAE loss often remains unsaturated in practice. As a result, directly using  $-l(\theta; \mathbf{x})$  as the outlier loss may lead to instability during training. To address this issue, we adopt the CUBO framework (Dieng et al. 2017), which provides a stable *upper bound* on the log-likelihood. As a result, we use the IWAE loss for the first two terms in (3), and apply the CUBO loss for the last term, which corresponds to the outlier loss. The detailed formulation is provided in Appendix A.

**Threshold Modification** Theoretically, the adaptive threshold  $\tau_t$  is set to be the inlier risk, i.e.,  $\tau_t = R_i(\theta_{t-1})$ . However, when no labeled data are available, this quantity cannot be computed directly. In such cases, using a quantile of the per-sample loss values has been shown to yield good empirical performance (Cho et al. 2025). Therefore, we adopt this quantile-based strategy during the warm-up phase. In contrast, during the polarization phase, labeled data are partially available. We incorporate this information into the computation of the adaptive threshold as follows:

$$\tau_t = (1 - \xi) \cdot q_\rho(\theta_{t-1}) + \xi \cdot \widehat{R}_t^i(\theta_{t-1}), \quad (4)$$

where  $q_\rho$  denotes the  $\rho$ -quantile of the per-sample loss values  $l(\theta_{t-1}; \mathbf{x})$ , and  $\xi \in (0, 1)$  is a weighting parameter. We note that both hyperparameters,  $\rho$  and  $\xi$ , are predetermined.

**Querying Strategies** In the polarization phase, selecting which samples to query based on per-loss values is crucial. We consider three strategies:

- **Random (RD):** randomly selecting samples for labeling.
- **Confidence Poles (CP):** selecting samples with the smallest and largest loss values, which are likely to be inliers and outliers, respectively. Half of the queried samples are chosen from the lowest-loss instances, and the other half from the highest-loss ones.
- **Decision Boundary Using Mixture Model (MM):** targeting ambiguous samples. To this end, we fit a two-component GMM to the current loss values and compute each sample’s posterior probability of belonging to the inlier cluster (the one with the smaller mean). We then select samples whose posterior probability is closest to a predefined threshold  $\alpha \in (0, 1)$ .

Among the three, we adopt the MM approach with  $\alpha = 0.4$  as our final query strategy.

---

Algorithm 1: IMBoost

In practice, we set

$$(n_0, \gamma, \rho, \xi, \alpha, \lambda_{1,t}, \lambda_{2,t}) = (128, 1.03, 0.92, 0.4, 0.4, 2, 1).$$

**Input:** Training data:  $\mathcal{D}^{\text{tr}}$ , parameters of a DGM :  $\theta$ , initial mini-batch size:  $n_0$ , mini-batch increment:  $\gamma$ , quantile value:  $\rho$ , optimizer:  $\mathcal{L}$ , four hyperparameters:  $(\xi, \alpha, \lambda_{1,t}, \lambda_{2,t})$ , four time steps:  $(T_0, T_1, T_2, T_a)$ .

- 1: Initialize  $\theta_0$ .
- 2: Phase 1: Warm-up
- 3: **for**  $(t = 1$  **to**  $T_0)$  **do**
- 4: Draw a mini-batch with the fixed size of  $n_0$ ,  $\mathcal{D}_0 = \{\mathbf{x}_i^{\text{mb}}\}_{i=1}^{n_0}$ , from  $\mathcal{D}^{\text{tr}}$ .  
Calculate the loss function  $\widehat{R}_0(\theta_0)$  in (1).  
 $\theta_0 \leftarrow \mathcal{L}(\widehat{R}_0(\theta_0), \theta_0)$
- 5: **end for**
- 6: **for**  $(t = 1$  **to**  $T_1)$  **do**
- 7: Draw a mini-batch with a size of  $n_t = n_0\gamma^{t-1}$ ,  $\mathcal{D}_t = \{\mathbf{x}_i^{\text{mb}}\}_{i=1}^{n_t}$  from  $\mathcal{D}^{\text{tr}}$ .  
Calculate the loss function  $\widehat{R}_t(\theta_{t-1})$  in (2).  
 $\theta_t \leftarrow \mathcal{L}(\widehat{R}_t(\theta_{t-1}), \theta_{t-1})$
- 8: **end for**
- 9: Phase 2: Polarization
- 10:  $\mathcal{I}_t, \mathcal{O}_t \leftarrow \emptyset, \emptyset$
- 11: **for**  $(t = (T_1 + 1)$  **to**  $(T_1 + T_2))$  **do**
- 12: Draw a mini-batch with a size of  $n_t = n_0\gamma^{t-1}$ ,  $\mathcal{D}_t = \{\mathbf{x}_i^{\text{mb}}\}_{i=1}^{n_t}$  from  $\mathcal{D}^{\text{tr}}$ .
- 13: **if**  $((t - T_1) \bmod (T_2/T_a) \text{ is } 0)$  **then**
- 14: // Query labels and update inlier and outlier sets.  
Update  $\mathcal{I}_t, \mathcal{O}_t$  using the MM strategy based on ensembled losses in (5).
- 15: **end if**  
Calculate the loss function  $\widehat{R}_t^a(\theta_{t-1})$  in (3).  
 $\theta_t \leftarrow \mathcal{L}(\widehat{R}_t^a(\theta_{t-1}), \theta_{t-1})$
- 16: **end for**

**Output (IMBoost score):** Per-sample loss  $l(\theta_{T_1+T_2}; \mathbf{x})$  computed on the training (or test) data

---

For the CP and MM strategies, per-sample loss values over the training data are required, which can be unstable during training. To mitigate this instability, we use loss scores averaged over several recent iterations before each query round. Specifically, at the beginning of query round  $t_q$ , we compute the ensembled loss for each input  $\mathbf{x}$  as:

$$l^{\text{ens}}(\mathbf{x}) = \frac{1}{t_e} \sum_{t=t_q-t_e}^{t_q-1} l(\theta_t; \mathbf{x}), \quad (5)$$

and use these scores to apply the CP or MM strategies. In practice, we set  $t_e = T_2/T_a$ , where  $T_a$  denotes the total number of query rounds, which is set to 5 in the experiments.

The complete pseudo-code of our proposed method, IMBoost, incorporating all the aforementioned techniques, is presented in Algorithm 1.

## Theoretical Analysis

In this section, we provide a theoretical analysis of how IM-Boost progressively enhances the separation between inlier and outlier risks during training. For simplicity in the theoretical analysis, we assume that  $\mathcal{I}_t$  and  $\mathcal{O}_t$  are independently drawn from  $\mathbb{P}_i$  and  $\mathbb{P}_o$ , respectively, and that their sizes are fixed (possibly dependent on  $t$ ). We also set the adaptive threshold to the inlier risk, i.e.,  $\tau_t = R_i(\theta_{t-1})$ .

We begin by presenting three mathematical assumptions, following the setup introduced in Cho et al. (2025).

**Assumption 1 (IM Effect)** *There exist  $0 < a_1 < a_2 < 1$  and  $a_3 \in (0, 1 - a_2)$  such that for any parameter  $\theta$  satisfying  $R_i(\theta) \in [a_1, a_2]$ ,  $R_o(\theta) - R_i(\theta) \geq a_3$ .*

**Assumption 2 (Bounded and Smooth Gradient)** *Denote the gradients of  $l(\theta; \mathbf{x})$  and  $R_i(\theta)$  as  $\nabla_{\theta}l(\theta; \mathbf{x})$  and  $\nabla_{\theta}R_i(\theta)$ , respectively. Then the followings conditions are satisfied:*

1) *For any  $\mathbf{x} \in \mathcal{X}$  and  $\theta \in \Theta$ , there exists a constant  $G > 0$ , such that*

$$\|\nabla_{\theta}l(\theta; \mathbf{x})\| \leq G.$$

2)  *$R_i(\theta)$  and  $R_o(\theta)$  are smooth with a  $L$ -Lipschitz continuous gradient, i.e., there exists a constant  $L > 0$  such that*

$$\begin{aligned} \|\nabla_{\theta}R_i(\theta) - \nabla_{\theta}R_i(\theta')\| &\leq L\|\theta - \theta'\|, \quad \text{and} \\ \|\nabla_{\theta}R_o(\theta) - \nabla_{\theta}R_o(\theta')\| &\leq L\|\theta - \theta'\|, \quad \forall \theta, \theta' \in \Theta. \end{aligned}$$

3) *There exists  $\mu > 0$  such that for any  $\theta \in \Theta$ ,*

$$2\mu(R_i(\theta) - R_i(\theta_*)) = 2\mu R_i(\theta) \leq \|\nabla_{\theta}R_i(\theta)\|^2.$$

**Assumption 3 (Uniform Margin)** *There is a constant  $0 < c < 1$  such that, for any  $\theta$ , the following inequality holds:*

$$\mathbb{E}_i \left[ \sqrt{l(\theta; X)} \right] \leq (1 - c)\sqrt{R_i(\theta)}.$$

Assumption 1 describes a state of the DGM in which a gap has emerged between the risk values of inliers and outliers. Assumption 2 refers to the smoothness properties of the loss function and the associated risk functions. And Assumption 3 ensures that Jensen's inequality holds with a uniform margin for any parameter  $\theta$ .

Additionally, we introduce an assumption stating that, under the presence of the IM effect, a gradient discrepancy exists between inliers and outliers.

**Assumption 4 (Gradient Discrepancy)** *There are two constant  $c_1, c_2 > 0$  such that, for any  $\theta$  satisfying the IM effect (i.e., Assumption 1), the following inequalities hold:*

$$\|\nabla_{\theta}R_o(\theta)\|^2 \geq c_1 \quad \text{and} \quad \nabla_{\theta}R_i(\theta)^T \nabla_{\theta}R_o(\theta) \leq -c_2.$$

Assumption 4 implies that, during the IM effect, the gradients of inlier and outlier risks not only point in different directions, but also cannot be minimized simultaneously. Based on the above assumptions, we establish the following two propositions. The proof of the second proposition is provided in the Appendix B. We note that Proposition 1 is identical to Proposition 2 in Cho et al. (2025), and therefore we omit its proof in this paper.

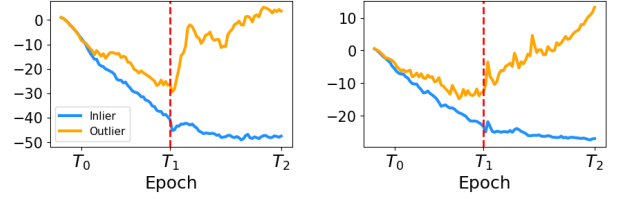


Figure 2: **(1st and 2nd)** Trace plots of inlier and outlier risks during the warm-up and polarization phases on the PageBlocks and Thyroid datasets, respectively.

### Proposition 1 (Warm-up Property (Cho et al. (2025)))

*Suppose that Assumptions 1 to 3 hold. During the warm-up phase (i.e., when  $t \leq T_1$ ), assume that the current parameter  $\theta_{t-1}$  satisfies  $a_1 \leq R_i(\theta_{t-1}) \leq a_2\gamma^{-(t-1)}$ . Then for any given  $\delta > 0$ , there exists a learning rate  $\eta_t > 0$  such that the inlier risk is further reduced, i.e.,  $R_i(\theta_t) \leq \gamma^{-1}R_i(\theta_{t-1})$ , with probability at least  $1 - \delta$ .*

**Proposition 2 (Polarization Property)** *Suppose that Assumptions 1 to 4 are satisfied. During the polarization phase (i.e., when  $t > T_1$ ), suppose that the current parameter  $\theta_{t-1}$  satisfies  $a_1 \leq R_i(\theta_{t-1}) \leq a_2\gamma^{-(t-1)}$ . Then, for any given  $\delta > 0$ , there exists a learning rate  $\eta_t > 0$  and  $\lambda_{1,t}$  and  $\lambda_{2,t}$ , both proportional to  $\gamma^{-(t-T_1-1)}$ , such that the following holds with probability at least  $1 - \delta$ :*

$$R_i(\theta_t) \leq \gamma^{-1}R_i(\theta_{t-1}) \quad \text{and} \quad R_o(\theta_t) > R_o(\theta_{t-1}).$$

Proposition 2 indicates that, during the polarization phase, our proposed loss function (3) guarantees a continued reduction in inlier risk and simultaneously induces an increase in outlier risk, thereby enlarging the discrepancy between inlier and outlier risks.

**Theoretical Implications** These theoretical results provide key insights into our proposed method, with further discussion available in Appendix B. First, a natural question arises regarding *the necessity of employing the ALTBI loss function during the warm-up phase*, as initiating active learning immediately after the emergence of the IM effect may appear more intuitive and efficient by reducing the number of iterations. And this concern is addressed by our theoretical results, particularly Lemma 4 in Appendix B. Lemma 4 shows that repeated use of the trimmed loss function gradually reduces the proportion of outliers included in it. In the early stages following the emergence of the IM effect, the trimmed loss function (2) still includes a relatively high proportion of outliers. As a result, minimizing (2) at this point may unintentionally lead to a reduction in outlier risk, as illustrated in Figure 2. Therefore, applying active learning at this stage does not guarantee a meaningful increase in outlier risk.

Using a large value of  $\lambda_{2,t}$  to amplify the effect of outlier loss term seems to be a plausible alternative. However, this approach may also adversely impact the inlier loss, potentially increasing the inlier risk, which is an undesirable outcome. Therefore, it is crucial to first reduce the influence of outliers by minimizing (2) for several iterations—until it no

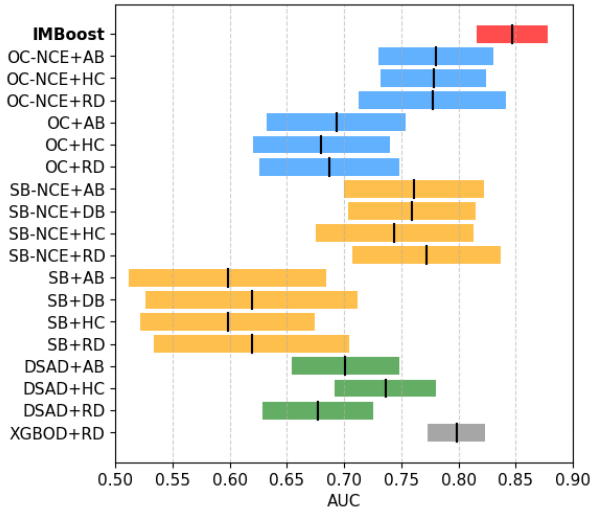


Figure 3: Averaged test AUC results at the final (5th) round across 57 datasets from ADBench, with standard deviations over three independent runs. All implementations were done by us. Color scheme: red (IMBoost), blue (OC-based), orange (SB-based), and green (DSAD-based).

longer decreases the outlier risk—and then introduce active learning to simultaneously enhance the reduction of inlier risk and the increase of outlier risk.

The second question is about *the optimal ratio of labeled inlier and outlier samples*. The proof of Proposition 2 reveals that the increase in outlier risk is maximized when the sizes of the inlier and outlier sets are proportional to their corresponding hyperparameters, i.e.,  $|\mathcal{I}_t| \propto \lambda_{1,t}$  and  $|\mathcal{O}_t| \propto \lambda_{2,t}$ . Assuming that the two hyperparameters are set to similar values, and the number of queried samples is fixed (i.e.,  $|\mathcal{I}_t| + |\mathcal{O}_t| = n_a$  with a constant  $n_a$ ), the outlier risk increases the most when the sets are approximately balanced (i.e.,  $|\mathcal{I}_t| \approx |\mathcal{O}_t| \approx n_a/2$ ). This implies that we should query as *many outliers* as inliers when acquiring labeled data. This finding provides a theoretical justification for why using posterior probability thresholds close to 0.5 in the MM strategy generally yield the favorable results, as shown in the ablation study in the experimental section.

## Experiments

We validate the effectiveness of IMBoost through comprehensive experiments on 57 benchmark datasets across image, text, and tabular domains. The results show that IMBoost achieves state-of-the-art outlier detection performance with notably higher accuracy and lower computational cost. For each experiment, we report the average results over three independent runs with different random initializations. All experiments are conducted using the PyTorch framework on two NVIDIA RTX 3090 GPUs.

**Dataset Description** We evaluate our method on all 57 benchmark datasets from ADBench (Han et al. 2022), following the same preprocessing and dataset settings used in

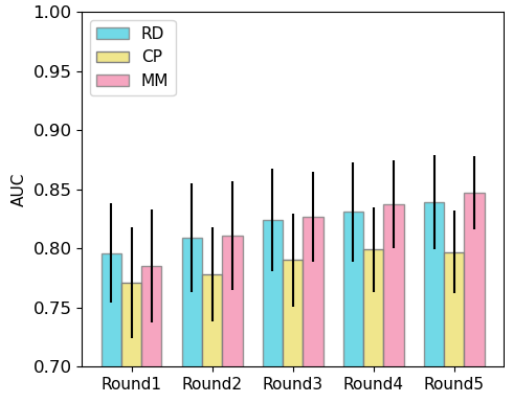


Figure 4: Averaged test AUC results (with standard deviations) of IMBoost using different querying strategies: 1) Random (RD), 2) Confidence Poles (CP), and 3) Mixture Model-based decision boundary (MM). Results are reported at the end of each active learning round.

ALTBI (Cho et al. 2025). We first consider 46 widely used tabular datasets covering diverse domains such as healthcare, finance, and astronautics. Additionally, we include five text datasets using BERT (Devlin et al. 2019) or RoBERTa (Liu et al. 2019) embeddings provided by ADBench. Finally, we evaluate on six image datasets using ViT-based embeddings (Dosovitskiy et al. 2021), also from ADBench. Full dataset details are provided in Appendix C.

All datasets are min-max normalized before training. Each dataset is randomly split into training and test sets with a 7:3 ratio. Following Kim et al. (2023), we perform 5 rounds of active learning, that is,  $T_a = 5$ , setting the query budget 1% of each dataset. For small datasets (fewer than 500 samples), such as *ionosphere*, *arrhythmia*, and *glass*, we instead query 6 samples per round.

**Competing Methods** To evaluate the effectiveness of the proposed IMBoost framework, we compare it against recent state-of-the-art deep learning-based active OD methods. In particular, we mainly follow the experimental settings and baseline configurations analyzed in Kim et al. (2023). As a baseline approach, we first consider DSVDD (Ruff et al. 2018), originally developed for dealing with SSOD tasks. Following Kim et al. (2023), we adopt two variants of DSVDD: one-class DSVDD (OC) and soft-boundary DSVDD (SB). To enhance the discrepancy between inlier and outlier scores, we additionally incorporate noise contrastive estimation (NCE) as a regularization technique. As another baseline, we include DSAD (Ruff et al. 2020), an advanced extension of DSVDD leveraging labeled samples. We also consider a non-deep learning-based approach, XGBOD (Zhao and Hryniewicki 2018), known as one of the strongest OD solvers.

Based on these models, we explore various query strategies for active learning: 1) random sampling (RD), 2) high-confidence sampling (HC) that prioritizes highly abnormal instances, and 3) adaptive boundary (AB), which dynamically searches for a querying region where the ratio of inliers

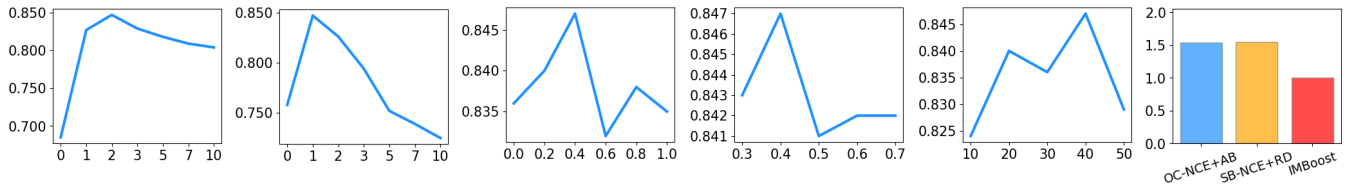


Figure 5: (1st–5th) AUC results with varying values of hyperparameters: 1)  $\lambda_{1,t}$ , 2)  $\lambda_{2,t}$ , 3)  $\xi$ , 4)  $\alpha$ , and 5)  $T_1$ . (6th) Comparison of running time between the IMBoost and other approaches. Each runtime is rescaled relative to that of IMBoost.

and outliers near the boundary is balanced. For XGBOD, we only adapt the RD strategy. For SB, we further consider an additional query strategy, called 4) decision boundary (DB), which selects samples near the estimated decision boundary. We follow all implementation configurations described in Kim et al. (2023).

**Implementation Details** As the DGM framework, we use IWAE, an ELBO-based model that has been shown to be both effective and efficient for IM-based methods by Cho et al. (2025). We adopt the same encoder and decoder architectures as in Cho et al. (2025), with detailed specifications provided in Appendix A.

We use the Adam optimizer (Kingma and Ba 2014) with a learning rate of  $1 \times 10^{-3}$ . Our method involves several hyperparameters. For the warm-up phase, we use the same settings as in Cho et al. (2025), namely  $(n_0, \gamma, \rho) = (128, 1.03, 0.92)$ , and set  $(T_0, T_1) = (10, 40)$ . In the polarization phase, we retain the same values for  $(n_0, \gamma, \rho)$  and set  $(\lambda_{1,t}, \lambda_{2,t}, \xi, \alpha) = (2, 1, 0.4, 0.4)$ . Each active learning round is trained for 10 epochs, resulting in a total of 50 epochs for the polarization phase, i.e.,  $T_2 = 50$ . Note that, given the relatively small number of rounds (five), we keep the hyperparameters  $(\lambda_{1,t}, \lambda_{2,t})$  constant across all rounds. Among the three querying strategies, we adopt the mixture model-based strategy (MM) as the default querying method.

## Performance Results

We compare the anomaly detection performance of our method with other baselines. For each dataset, we compute the mean and standard deviation of the area under the ROC curve (AUC) and average precision (AP) scores over three independent runs. Figure 3 presents the average AUC scores and their standard deviations on the test data, aggregated across all datasets. Detailed results for each dataset, including AP metrics and additional evaluations on the training data, are provided in Appendix C.

We observe that our method achieves the best performance with an average margin of nearly 5% over the second-best baseline (XGBOD+RD). Moreover, it exhibits small standard deviations, indicating that IMBoost is not only superior in accuracy but also consistent in detecting outliers. Given that our method requires only a short warm-up phase, whereas other baselines involve pre-training over 100 epochs, IMBoost offers a highly practical and efficient solution for a wide range of OD tasks.

We also compare the performance of the three querying strategies—RD, CP, and MM—within the IMBoost frame-

work. As shown in Figure 4, the MM strategy usually outperforms both RD and CP throughout all rounds, while CP yields the worst performance among the three. This result aligns with the intuition that samples near the decision boundary—targeted by MM—are the most informative for improving the model. In contrast, CP selects samples with the highest and lowest losses, which are often already well-classified and thus provide limited information gain—even less than randomly selected ones.

Inspired by RD’s strong performance in the first round, we apply RD for the first round and MM for the remaining rounds, but this results in slightly degraded performance (0.840 vs. 0.847 with MM only). From the perspective of viewing RD and MM as *exploration* and *exploitation*, respectively, designing an advanced querying schedule that optimally combines the two would be an intriguing direction for future work.

## Ablation Studies

We conduct additional experiments to examine the sensitivity of IMBoost to various hyperparameter choices, with the results illustrated in Figure 5. To summarize: 1) Setting either  $\lambda_{1,t}$  or  $\lambda_{2,t}$  to zero leads to a significant performance drop. 2) Incorporating inlier risk from  $\mathcal{I}_t$  when determining the adaptive threshold  $\tau_t$  improves performance. 3) Performance remains stable as long as an appropriate posterior probability  $\alpha$  is chosen. 4) Setting  $T_1 = 40$  for the warm-up phase yields optimal results. 5) Our method has the highest computational efficiency among the competitive baselines (excluding XGBOD, which is highly inefficient for large datasets). The detailed results are provided in Appendix C.

## Concluding Remarks

In this paper, we proposed IMBoost, a novel active outlier detection framework that leverages and enhances the IM effect in DGMs. Our method operates in two phases: a warm-up phase that induces and promotes the IM effect, and a polarization phase that leverages actively queried samples to simultaneously reduce inlier risk and increase outlier risk. We theoretically showed that our approach indeed increases the separation between inlier and outlier risks, and empirically demonstrated that IMBoost significantly outperforms existing methods while remaining efficient. A promising future work is to explore optimal querying strategies, for instance by incorporating advanced active learning techniques developed in other domains (Sener and Savarese 2018; Kirsch, van Amersfoort, and Gal 2019; Yoo and Kweon 2019).

## Acknowledgements

DK was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (RS2023-00218231 and RS-2025-24683613). SP was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(RS-2025-25415913).

## References

- Burda, Y.; Grosse, R. B.; and Salakhutdinov, R. 2016. Importance Weighted Autoencoders. In *4th International Conference on Learning Representations, ICLR 2016, Conference Track Proceedings*.
- Cho, S.; Hwang, J.; Bak, K.; and Kim, D. 2025. ALTBI: Constructing Improved Outlier Detection Models via Optimization of Inlier-Memorization Effect. In Walsh, T.; Shah, J.; and Kolter, Z., eds., *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, 11544–11552. AAAI Press.
- Das, S.; Wong, W.-K.; Dietterich, T.; Fern, A.; and Emmott, A. 2016. Incorporating expert feedback into active anomaly discovery. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, 853–858. IEEE.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. 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, Volume 1 (Long and Short Papers)*, 4171–4186. Association for Computational Linguistics.
- Dieng, A. B.; Tran, D.; Ranganath, R.; Paisley, J.; and Blei, D. 2017. Variational Inference via  $\chi$ -Upper Bound Minimization. *Advances in Neural Information Processing Systems*, 30.
- Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR 2021*. OpenReview.net.
- Gutmann, M.; and Hyvärinen, A. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 297–304. JMLR Workshop and Conference Proceedings.
- Han, S.; Hu, X.; Huang, H.; Jiang, M.; and Zhao, Y. 2022. ADBench: Anomaly Detection Benchmark. In *Neural Information Processing Systems (NeurIPS)*.
- Kim, D.; Hwang, J.; Lee, J.; Kim, K.; and Kim, Y. 2024. ODIM: Outlier Detection via Likelihood of Under-Fitted Generative Models. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.
- Kim, M.; Kim, J.; Yu, J.; and Choi, J. K. 2023. Active anomaly detection based on deep one-class classification. *Pattern Recognition Letters*, 167: 18–24.
- Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Kingma, D. P.; and Dhariwal, P. 2018. Glow: Generative Flow with Invertible 1x1 Convolutions. In Bengio, S.; Wallach, H. M.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, 10236–10245.
- Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.
- Kirsch, A.; van Amersfoort, J.; and Gal, Y. 2019. Batch-BALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 7024–7035.
- Lamba, H.; and Akoglu, L. 2019. Learning on-the-job to re-rank anomalies from top-1 feedback. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, 612–620. SIAM.
- Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692.
- Nassif, A. B.; Talib, M. A.; Nasir, Q.; and Dakalbab, F. M. 2021. Machine learning for anomaly detection: A systematic review. *Ieee Access*, 9: 78658–78700.
- Pang, G.; Shen, C.; Cao, L.; and Hengel, A. V. D. 2021. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2): 1–38.
- Ruff, L.; Vandermeulen, R.; Goernitz, N.; Deecke, L.; Siddiqui, S. A.; Binder, A.; Müller, E.; and Kloft, M. 2018. Deep one-class classification. In *International conference on machine learning*, 4393–4402. PMLR.
- Ruff, L.; Vandermeulen, R. A.; Görnitz, N.; Binder, A.; Müller, E.; Müller, K.-R.; and Kloft, M. 2020. Deep Semi-Supervised Anomaly Detection. In *International Conference on Learning Representations*.
- Sener, O.; and Savarese, S. 2018. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
- Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2021. Score-Based Generative Modeling through Stochastic Differential Equations. In *9th International Conference on Learning Representations, ICLR 2021*. OpenReview.net.
- Tax, D. M.; and Duijn, R. P. 2004. Support vector data description. *Machine learning*, 54(1): 45–66.
- Trittenbach, H.; Englhardt, A.; and Böhm, K. 2021. An overview and a benchmark of active learning for outlier detection with one-class classifiers. *Expert Systems with Applications*, 168: 114372.

- Yoo, D.; and Kweon, I. S. 2019. Learning Loss for Active Learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 93–102. Computer Vision Foundation / IEEE.
- Zha, D.; Lai, K.-H.; Wan, M.; and Hu, X. 2020. Meta-AAD: Active anomaly detection with deep reinforcement learning. In *2020 IEEE International Conference on Data Mining (ICDM)*, 771–780. IEEE.
- Zhao, Y.; and Hryniewicki, M. K. 2018. XGBOD: Improving Supervised Outlier Detection with Unsupervised Representation Learning. In *2018 International Joint Conference on Neural Networks, IJCNN 2018, Rio de Janeiro, Brazil, July 8-13, 2018*, 1–8. IEEE.