RETHINKING THRESHOLDING AS A NEGLECTED FRONTIER IN ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Anomaly detection aims to identify data samples that deviate from expected patterns and is widely applied across domains. In practice, it consists of two stages: (i) computing continuous anomaly scores; and (ii) converting these scores into binary decisions via thresholding. While extensive research has focused on scoring mechanisms, thresholding is often treated as a minor detail, despite being essential for evaluation, comparison, and deployment. Over half of studies and libraries select thresholds using supervision, introducing oracle knowledge that conflicts with the unsupervised setting. Through a diagnostic study of 38 thresholding methods and 8 scaling strategies across three data modalities–time-series from TSB-AD, and tabular and image data from ADBench– we show that violations of common parametric assumptions cause complete thresholding failures on one-third of datasets. Score normality, for example, holds in fewer than 2% of empirical distributions. Thresholding choices alone can also reverse detector rankings. To address these limitations, we propose the **Energy-Based Adaptive Boundary (EAB)**, a nonparametric, parameter-free method that partitions anomaly scores using a Gini-mean-difference-calibrated two-sample energy distance. Centroid aggregation over the energy landscape provides robustness to autocorrelation, while a prefix-sum reformulation reduces naive cubic computation to $O(n \log n)$ time and $O(n)$ space. Compared with the nine strongest unsupervised baselines, EAB achieves statistically significant F1 improvements against every baseline, with gains of up to 17.7% on time-series data and 16.2% on tabular and image data, confirmed by Wilcoxon signed-rank tests. These results establish EAB as an efficient, modality-agnostic framework for unsupervised anomaly thresholding.

1 INTRODUCTION

Anomaly detection (AD) is a fundamental task with applications across diverse domains, encompassing challenges from anomaly score generation to the conversion of scores into binary decisions through thresholding (Siffer et al., 2017; Perini et al., 2023). In many practical settings, labeled anomalies are scarce or unavailable, necessitating unsupervised approaches. Nevertheless, deployment and evaluation using binary predictions require a threshold, making threshold selection an essential component of the detection pipeline. Although extensive research has advanced anomaly scoring (Pang et al., 2021; Boniol et al., 2024) and benchmarking (Han et al., 2022; Liu & Paparrizos, 2024), thresholding remains comparatively underexplored, despite its direct influence on reported performance, model comparison, and practical deployment.

We systematically examine thresholding practices in over 30 recent experimental studies and 11 widely used AD libraries, summarized in Figure 1(a). More than half employ supervised threshold selection, including maximizing the F1-score using ground-truth labels (*BestF*) or setting the predicted anomaly proportion to the ground-truth anomaly ratio (*Ratio*). These practices introduce oracle information unavailable in unsupervised deployment, limiting the practical interpretation of benchmark results. Beyond label dependence, threshold selection can substantially alter conclusions about detector quality. Figure 1(b) illustrates this effect using 10 representative detectors from ADBench (Han et al., 2022): rankings change markedly between supervised and unsupervised thresholding, with XGB moving from a middle-ranked position to the top. Thus, apparent differences in detector performance can arise from threshold selection alone, even when the underlying anomaly scores remain unchanged.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

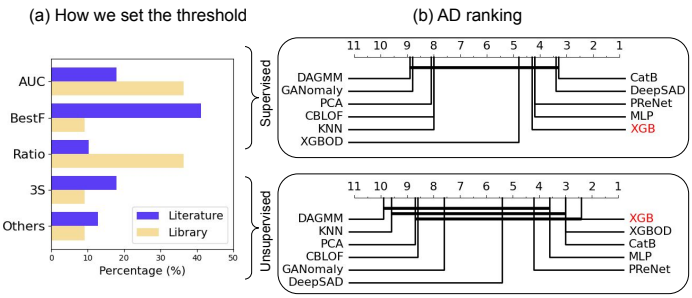


Figure 1: Thresholding practices and their impact on anomaly detector rankings. (a) Prevalence of thresholding methods in the literature and AD libraries. (b) Detector rankings under supervised and unsupervised thresholding, illustrating the sensitivity of model comparisons to threshold selection.

Even without supervision, widely adopted thresholding methods often depend on restrictive distributional assumptions. A prominent example is the KS deviation rule, which places the threshold K standard deviations above the mean, with $K = 3$ ($3S$) commonly motivated by normally distributed scores (Barnett, 1994). However, as shown in Section 3.2, normality holds in fewer than 2% of the empirical score distributions we examine, and $3S$ fails completely on approximately one-third of datasets. These findings highlight the limitations of applying fixed parametric rules to heterogeneous anomaly score distributions and motivate adaptive alternatives.

To characterize these limitations systematically, we conduct a large-scale empirical study of 38 thresholding methods and 8 score-scaling strategies across 242 time-series from TSB-AD (Liu & Paparrizos, 2024), spanning 17 domains, and 10 tabular and image datasets from ADBench (Han et al., 2022). On each time-series/dataset we apply the top 10 detectors (see Section 3.1) resulting in total to 2420 and 100 anomaly-score sets respectively from the two benchmarks. This evaluation isolates the effects of scaling and thresholding from anomaly scoring, allowing us to examine their influence on detection performance, ranking stability, and computational cost. We provide implementations and standardized evaluation protocols to support reproducible comparisons. We share our code and results in our anonymous GitHub repo https://anonymous.4open.science/r/Last-Mile-AD.

Guided by this analysis, we propose the **Energy-Based Adaptive Boundary (EAB)**, a nonparametric, parameter-free algorithm that formulates threshold selection as a two-sample distributional separation problem. For each candidate threshold, EAB partitions the scores into putative inlier and outlier subsets and measures their separation using the two-sample energy distance. The criterion is calibrated for the no-anomaly case using the Gini mean difference, and the final threshold is obtained through an energy-weighted centroid over high-separation candidates. A prefix-sum reformulation reduces naive cubic computation to $O(n \log n)$ time and $O(n)$ space. EAB achieves the best average rank among the evaluated unsupervised methods on both benchmarks, with statistically significant F1 improvements of up to 17.7 percentage points on TSB-AD and 16.2 percentage points on ADBench over the strongest competitor (Wilcoxon signed-rank tests, $p < 0.05$).

Together, our diagnostic findings and methodological results establish thresholding as a critical component of both AD evaluation and algorithm design. We present contributions as follows:

- **Distributional assumptions.** We test common parametric assumptions on anomaly scores and show that they rarely hold, causing complete thresholding failures (Section 3.2).
- **Score scaling and smoothing.** We examine how scaling and, for time series, smoothing affect thresholding robustness across data modalities (Sections 3.3 and 3.4).
- **Ranking sensitivity.** We demonstrate how thresholding choices alter detector rankings and benchmark conclusions (Figure 1 and Appendix D).
- **Design requirements.** We distill these findings into four requirements for unsupervised thresholding (Section 3.6).
- **EAB algorithm.** We introduce a parameter-free, unsupervised method combining energy-based separation, Gini mean difference calibration, and centroid aggregation (Section 4).
- **Comparative evaluation and efficiency.** We benchmark EAB against 38 thresholding methods and evaluate its robustness and performance–cost trade-offs (Section 5).

Finally, we discuss the implications of our findings (Section 6).

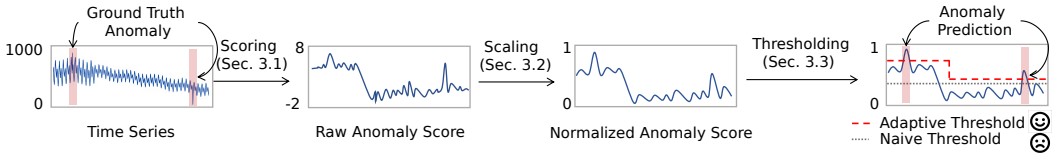


Figure 2: Illustration of the processing steps from obtaining the anomaly score to generating the anomaly prediction, with time-series data shown as an illustrative example. Detailed discussions of scaling and thresholding are provided in Sections 2.2 and 2.3, respectively.

2 Background: The Anomaly Scoring Pipeline

Anomaly Detection (AD) presents a set of fundamental challenges that distinguish it from standard classification or regression tasks due to the complex and heterogeneous nature of real-world data. Such data often exhibit diverse anomaly manifestations, including isolated outliers, context-dependent anomalies, and collective abnormal patterns (Paparrizos et al., 2022; Han et al., 2022). In addition, noise is an inherent characteristic of many real-world datasets, arising from random fluctuations or measurement errors that can obscure true anomalies and affect the reliability of detection methods. Labeled anomalies are typically scarce, limiting the applicability of supervised learning and necessitating unsupervised or semi-supervised approaches, while anomaly characteristics may vary substantially across datasets and application domains, requiring flexible and adaptive solutions. These challenges directly influence the anomaly scores produced by AD models, rendering threshold selection a non-trivial component of the pipeline.

Problem statement. Formally, in anomaly detection, a dataset consists of a collection of n samples $X = \{x_1, \ldots, x_n\} \in \mathbb{R}^{n \times d}$, where each sample $x_i \in \mathbb{R}^d$ has d features. AD methods aim to identify abnormal patterns by assigning an anomaly score to each sample in X, yielding an anomaly score vector $S = \{s_1, \ldots, s_n\}$ with $s_i \in \mathbb{R}$, where higher values indicate a greater likelihood of anomalous behavior. While these scores provide a continuous measure of abnormality, real-world applications ultimately require discrete decisions. This leads to the central problem considered in this work: **given an anomaly score vector S, how can one accurately derive corresponding binary anomaly predictions P, where each decision $p_i \in \{0, 1\}$ indicates whether sample x_i is anomalous?** This process concerns the final stages of the AD pipeline illustrated in Figure 2. In practice, anomaly scores are often first transformed via scaling functions Sc, producing scaled scores $Sc(S) = S'$, and subsequently converted into binary predictions $P = \{p_1, \ldots, p_n\}$ through a **thresholding method**. Importantly, neither the scaling procedure nor the thresholding rule is uniquely determined by the anomaly scores themselves, but instead encodes assumptions whose consequences are rarely examined. This dependence motivates a closer investigation of thresholding practices and their implications, which is the focus of this paper.

This section presents the anomaly prediction pipeline as a sequence of transformations that map raw data to binary anomaly decisions. We discuss representative state-of-the-art anomaly detection models in Section 2.1, introduce anomaly score scaling procedures in Section 2.2, and describe a taxonomy of thresholding methods in Section 2.3. Importantly, while these steps are often presented independently, our central argument is that their interaction critically determines evaluation outcomes.

2.1 Anomaly Scoring

Anomaly scoring constitutes the first stage of the pipeline, where raw data are processed by an AD model to produce real-valued scores indicating the degree of abnormality of each sample. Figure 2 illustrates this process for the time-series modality. A broad spectrum of AD techniques has been proposed in the literature (Aggarwal, 2017; Boniol et al., 2024), commonly grouped into three categories. *Distance-based* methods identify anomalies based on their dissimilarity to neighboring points in the input space. *Density-based* methods estimate data density using representations such as graphs, trees, or histograms, flagging low-density regions as anomalous. *Prediction-based* methods model normal behavior and use prediction errors as anomaly scores; these approaches often perform well in semi-supervised settings. We focus on widely used and well-performing detectors drawn from established benchmarks. We adopt the top-ranked anomaly detection methods from the TSB-AD benchmark for time-series data (Liu & Paparrizos, 2024) and from ADBench for tabular and image data (Han et al., 2022); they are listed in Appendix A (Tables 2 and 3). These models provide diverse anomaly score distributions that serve as inputs to subsequent pipeline stages, allowing us to isolate the impact of scaling and thresholding independently of scoring model design.

2.2 Anomaly Score Scaling

Anomaly detectors produce scores that vary widely in scale, range, and distribution, which complicates direct comparison and downstream decision-making. As a result, raw anomaly scores $s_t \in S$ are commonly transformed prior to thresholding. The goal of score scaling is to improve interpretability and comparability, ideally mapping scores to a common scale where higher values correspond more clearly to higher anomaly likelihood (Kriegel et al., 2011). We refer to this process as **scaling**, encompassing any method or combination of methods that adjusts anomaly scores before threshold selection. Scaling may consist of one or both of the following steps: (i) transforming scores defined on $[0, \infty)$ so that small values indicate inliers and large values indicate outliers; and (ii) normalizing scores into a bounded range, typically $[0, 1]$. A critical requirement of scaling procedures is that they preserve the ranking of anomaly scores, ensuring that relative ordering remains unchanged. **Transformation.** We consider three transformations. The *Base* transformation, $\text{Base}(s_t) = s_t - \min(S)$, shifts scores to a non-negative domain. The *Logarithmic* transformation, $\text{Log}(s_t) = \log(s_t - \min(S) + \epsilon)$, emphasizes relative differences and can attenuate heavy-tailed behavior. Finally, *EWMA* smoothing, $\text{Ewma}(s_t) = \alpha s_t + (1 - \alpha)\text{Ewma}(s_{t-1})$ with $\text{Ewma}(s_0) = s_0$, is applied in the time-series setting to reduce noise while preserving temporal trends. **Normalization.** We study normalizers that map scores to a bounded range while preserving their rank ordering: (i) *MinMax* normalization, $\text{MinMax}(s_t) = \frac{s_t - \min(S)}{\max(S) - \min(S)}$; (ii) *Gaussian* scaling, which assumes approximate normality and applies the error function, $\text{Gaussian}(s_t) = \max(0, \text{erf}((s_t - \mu_S)/(\sigma_S\sqrt{2})))$; and (iii) *Gamma* scaling, which accommodates skewed distributions via a Gamma fit, $\text{Gamma}(s_t) = \max\left(0, \frac{\text{cdf}_S^{\gamma}(t) - \mu_{\text{cdf}}}{1 - \mu_{\text{cdf}}}\right)$. Other monotonic normalizations, such as z-norm, unit scaling, or no normalization, preserve score ranking and therefore do not affect thresholding outcomes under rank-based methods. We empirically evaluate the impact of different scaling choices in Section 5.

2.3 Thresholding methods

The final stage of the anomaly prediction pipeline converts scaled anomaly scores into binary decisions. Thresholding methods may produce static thresholds, which remain fixed over the dataset, or dynamic thresholds, which adapt over time to address non-stationarity and concept drift. In addition, thresholds can be determined in offline settings, where the entire dataset is available in advance, or in online settings, where samples are processed incrementally. In this work, we analyze 38 thresholding methods drawn from multiple methodological families (Perini et al., 2023; Kulik et al., 2025). We organize them into nine families based on their underlying principles (the full list of methods per family is given in Table 4, Appendix B):

Statistical-based: Methods that rely on assumed distributional properties (Siffer et al., 2017).
Statistical test-based: Approaches that apply iterative hypothesis testing (Alrawashdeh, 2021).
Kernel-based: Techniques that estimate score densities or decision boundaries (Ren et al., 2018; Joneidi, 2013).
Transformation-based: Methods that reshape score distributions or identify structural shifts (Raymaekers & Rousseeuw, 2024; Fearnhead & Rigaill, 2019).
Filter-based: Methods that apply smoothing or filtering operations to anomaly scores (Hashemi et al., 2019).
Mixture model-based: Approaches that model scores as mixtures of inlier and outlier distributions, often using Gaussian mixtures or clustering (Perini et al., 2023; Klawonn & Rehm, 2009).
Quantile-based: Methods that define anomalies according to empirical quantiles of the score.
Regression-based: Techniques that employ regression or classifier models, typically trained on synthetic data, to identify deviations (Aggarwal, 2017).
Curve-based: Geometric approaches using elliptical boundaries or winding numbers (Friendly et al., 2013; Jacobson et al., 2013).

Each method is described in detail in Appendix B. While diverse in formulation, all thresholding methods encode assumptions about score distributions and anomaly prevalence; Section 3 examines whether these assumptions hold in practice.

3 Why Does Thresholding Fail? A Diagnostic Study

Section 2 showed that every thresholding method encodes assumptions about the anomaly scores it receives. Before designing a new method, we test these assumptions empirically. We ask four

questions: whether anomaly scores follow the distributions that parametric rules assume (Section 3.2), how much scaling changes the outcome (Section 3.3), how noise and temporal dependence in time-series scores affect thresholds (Section 3.4), and whether simple adaptiveness can repair failed thresholds (Section 3.5). Together with the practical need for efficiency, the answers yield four requirements that guide the design of EAB (Section 3.6).

3.1 Experimental Setup

Both the diagnostic study in this section and the evaluation of EAB in Section 5 share the following setup. We conduct a comprehensive empirical evaluation across three data modalities: time-series, tabular, and image data. For time-series data, we consider 10 anomaly detection methods that achieve state-of-the-art performance on carefully curated benchmark datasets from TSB-AD (Liu & Paparrizos, 2024), as summarized in Table 2 (Appendix A). Given the anomaly scores produced by these detectors, we apply 38 thresholding methods described in Section 2.3. In addition, we evaluate three normalization strategies (MinMax, Gamma, Gaussian) and three transformation techniques (Base, Logarithmic, EWMA), as detailed in Section 2.2, allowing us to isolate the effect of scaling and thresholding independently of the scoring model. To examine whether these effects extend beyond time-series data, we further conduct experiments on multimodal datasets from ADBench (Han et al., 2022), covering 10 datasets, including tabular and image data, and 10 representative detectors (resulting in 100 sets of anomaly scores). The same scaling and normalization strategies are applied across modalities, with the exception of EWMA, which is specific to time-series data and therefore omitted. This setup allows us to evaluate the generality of thresholding-related phenomena across different data types and anomaly score distributions. Throughout our analysis, we adopt the *F1* score as the evaluation metric, reflecting the fact that anomaly detection systems ultimately require binary decisions. While *threshold-independent* metrics such as AUROC or VUS are useful for ranking scoring functions, they are insufficient for studying thresholding behavior, as they abstract away the decision boundary itself. In contrast, *F1* directly captures the precision–recall trade-off induced by threshold selection and is therefore appropriate for evaluating the practical consequences of thresholding choices. Detailed implementation and experimental documentation are available at https://anonymous.4open.science/r/Last-Mile-AD.

3.2 Do Anomaly Scores Follow a Normal Distribution?

Many statistical thresholding methods (Schmidl et al., 2024; Paparrizos et al., 2022; Lai et al., 2021; Ahmad et al., 2017; Mejri et al., 2024; Choudhary et al., 2017; Alnegheimish et al., 2024) implicitly assume that anomaly scores follow a normal distribution, enabling unsupervised threshold selection. However, this assumption is rarely verified in practice. Methods such as the commonly used $3S$ rule rely directly on score normality to compute thresholds. To answer about normality, we applied the D'Agostino–Pearson normality test (d'Agostino, 1971; D'agostino & Pearson, 1973) to 3,610 anomaly score sets obtained from: (i) all data modalities, (ii) 10 time-series AD methods evaluated on 350 time series, and (iii) 11 AD methods evaluated on 10 tabular and image datasets. Across all settings, fewer than 2% of anomaly score distributions satisfy the normality assumption ($p = 0.05$). To mitigate skewness, prior work often applies logarithmic transformations; however, such transformations are only effective for log-normal data and may increase skewness otherwise (Feng et al., 2014). Reapplying the normality test after log transformation yields similar results, with fewer than 2% of scores exhibiting normality. As a consequence, normality-based methods such as $3S$ fail to produce a valid threshold in **34%** of the evaluated cases, yielding thresholds that exceed all anomaly scores. Therefore, normality assumptions underlying many thresholding methods are often violated in practice. *Takeaway: thresholds should be derived without assuming a parametric shape for the score distribution.*

3.3 What Is the Impact of Scaling?

Although prior work has proposed scaling anomaly scores before thresholding (Hundman et al., 2018), its impact is rarely examined systematically and is often overlooked in empirical studies. Scaling is commonly viewed as a minor preprocessing step rather than a factor that can substantially influence anomaly prediction.

We analyze the effect of MinMax, Gaussian, and Gamma normalization in combination with the transformations described in Section 2.2. Figure 3 shows that scaling choices significantly affect performance distributions. In particular, Gamma normalization combined with logarithmic transformation yields consistent performance improvements across detectors. These results suggest that

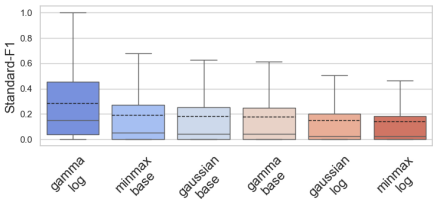


Figure 3: Scaling combinations ranking.

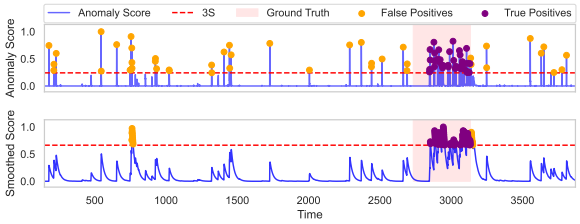


Figure 4: TimesFM scores before and after smoothing.

scaling alone can materially alter anomaly prediction outcomes, underscoring its importance prior to threshold selection. *Takeaway: a thresholding method should be robust to the choice of scaling, since no single scaling is universally best; we test this for EAB in Section 5.2.*

3.4 Noise and Smoothing in Time-Series Scores

Time-series data add a further complication: samples are ordered and temporally dependent, so anomaly scores are autocorrelated and often noisy (Boniol et al., 2024). In time-series anomaly detection (TSAD), an additional post-processing step-*smoothing*-leverages temporal neighborhoods to refine anomaly signals. Prior work (Hundman et al., 2018; Shipmon et al., 2017) shows that noise and transient spikes can degrade thresholding performance by inflating score variance, increasing false positives, or producing overly conservative thresholds. This effect directly impacts statistical thresholding methods such as $3S$ and is particularly pronounced in deep learning models, where noisy anomaly scores are a known limitation of unsupervised approaches (Chalapathy & Chawla, 2019). Figure 4 illustrates how noise and spurious spikes adversely affect threshold selection. Applying smoothing via EWMA substantially improves anomaly prediction by reducing false positives and stabilizing score distributions. Across all normalizations, detectors, thresholding methods, and datasets, smoothing improves *F1* performance by an average of **28%**. For deep learning–based models (TimesFM, MOMENT, LSTMAD, CNN, USAD), the effect is even stronger, with average improvements reaching **51%** (see Appendix D). *Takeaway: thresholds that hinge on a single extreme statistic are fragile under noisy, autocorrelated scores; a robust method should aggregate evidence over many candidate boundaries rather than select one extremum.*

3.5 Can Simple Adaptiveness Rescue Failed Thresholds?

In unsupervised AD, thresholding methods inevitably rely on assumptions about the behavior of anomaly scores. When these assumptions are violated, or when anomaly scores exhibit unexpected or degenerate behavior, thresholding can fail altogether, leading to poor performance (see Appendix D). Given that widely used methods such as $3S$ fail to detect a single anomaly in **34%** of the evaluated cases (Section 3.2), we ask whether a minimal form of adaptiveness can rescue them. We conduct an experiment in which thresholds are incrementally adjusted when a method produces no anomaly predictions. Concretely, if no anomaly score exceeds ($>$) the threshold returned by a method, the threshold is iteratively reduced to a percentage of its original value until at least one anomaly is detected. This adaptive adjustment substantially improves performance (for $3S$ 56% improvement on ADBench and 65.22% on time-series data). These results indicate that even simple forms of adaptiveness can mitigate failure modes caused by violated assumptions and rigid thresholding rules. However, this fix is reactive: the relaxation schedule is arbitrary, the threshold remains anchored to the original parametric rule, and the resulting low thresholds increase false positives. *Takeaway: adaptiveness helps, but it should be intrinsic—the boundary should be derived from the score distribution itself, together with a test of whether an apparent separation is meaningful at all.*

3.6 Requirements for Unsupervised Thresholding

The above findings translate into four requirements for an unsupervised thresholding method:

R1 Distribution-free. No parametric form may be assumed for the scores (Section 3.2).
R2 Calibrated adaptiveness. The boundary must come from the data and must not be triggered by homogeneous scores that contain no real separation (Section 3.5).
R3 Robustness. The threshold must be stable under noisy, autocorrelated scores and under different scalings (Sections 3.3 and 3.4).
R4 Efficiency and no tuning. The method must scale to long score sequences and have no hyperparameters to tune without labels.

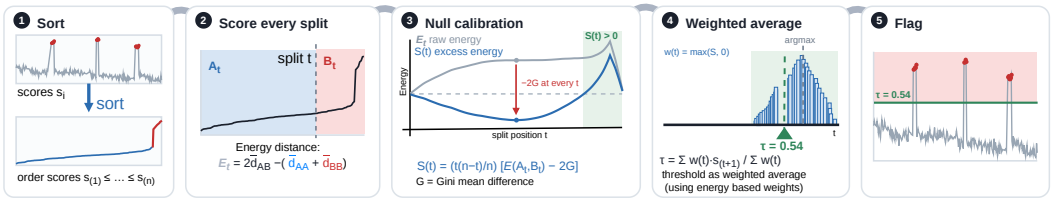


Figure 5: Energy-Based Adaptive Boundary for thresholding anomaly scores.

4 Energy-Based Adaptive Boundary

The diagnostic study shows that reactive fixes, such as incrementally relaxing a failed static threshold, mitigate catastrophic failures but treat the symptom rather than the cause: they remain tied to parametric assumptions that rarely hold (Sections 3.2 and 3.5). We therefore introduce the **Energy-based Adaptive Boundary (EAB)**, a non-parametric, parameter-free thresholding method that measures the distributional separation between normal and anomalous scores. Each component of EAB addresses one requirement from Section 3.6: a distribution-free separation criterion (R1), a null calibration that suppresses spurious boundaries (R2), centroid aggregation over candidate thresholds (R3), and a prefix-sum implementation with no hyperparameters (R4). Figure 5 illustrates the procedure. **Non-parametric Separation.** Given a set of anomaly scores $s_1, \ldots, s_n$ and their sorted order $s_{(1)} \le \cdots \le s_{(n)}$, any candidate threshold corresponds to a split index t partitioning the scores into a normal subset $A_t = \{s_{(1)}, \ldots, s_{(t)}\}$ and an anomalous subset $B_t = \{s_{(t+1)}, \ldots, s_{(n)}\}$. EAB evaluates candidate splits using the two-sample energy distance (Székely & Rizzo, 2013): $\mathcal{E}(A_t, B_t) = 2\bar{d}_{AB} - \bar{d}_{AA} - \bar{d}_{BB}$, where $\bar{d}_{XY}$ represents the mean absolute difference between elements of sets X and Y (Figure 5 panels 1 and 2). Because energy distance is zero if and only if the two samples originate from the same distribution, it responds to any difference in distributional shape without requiring density estimation, or binning parametric assumptions.

Null Calibration. Naively maximizing the energy distance inherently biases the split toward the median, even for completely homogeneous data lacking anomalies. Inspired by the gap statistic (Tibshirani et al., 2001), to ensure EAB only flags genuine separation, EAB subtracts a baseline score derived from a completely flat distribution. Specifically, it subtracts the score that would be expected from a uniform set of data, as this represents the worst-case scenario most prone to generating false boundaries. The expected baseline score energy evaluates exactly to $2G$, where $G = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}|s_i - s_j|$ is the Gini mean difference of the sample (Figure 5 panel 3). The calibrated excess energy for a split is thus weighted by the partition sizes to reflect statistical evidence:

$$S(t) = \frac{t(n-t)}{n}\left[\mathcal{E}(A_t, B_t) - 2G\right] \tag{1}$$

Centroid Thresholding. Anomaly scores, especially in autocorrelated time-series data, often produce flat regions of high excess energy, making a simple maximum highly susceptible to sampling noise. EAB mitigates this (Figure 5 panel 4), by calculating the final threshold τ as a centroid weighted by the positive excess energy:

$$\tau = \frac{\sum w(t)s_{(t+1)}}{\sum w(t)}, \quad \text{where} \quad w(t) = \max(S(t), 0) \tag{2}$$

Computational Efficiency. By leveraging prefix sums, EAB collapses the naively cubic cost of evaluating all candidate splits to a single linear pass. The entire procedure, detailed in Algorithm 1, executes in $\mathcal{O}(n \log n)$ time, dominated solely by the initial sort, and requires strictly $\mathcal{O}(n)$ space.

5 Evaluation of EAB

We now evaluate EAB against the 38 thresholding methods and the supervised oracles *BestF* and *Ratio*, using the setup of Section 3.1. We report overall accuracy (Section 5.1), robustness to scaling (Section 5.2, requirement R3), and runtime (Section 5.3, requirement R4).

5.1 Overall Performance

Figure 6 summarizes the overall performance of all evaluated thresholding methods across two benchmark settings: mixed tabular and image datasets (ADBench) and time-series datasets (TSB-

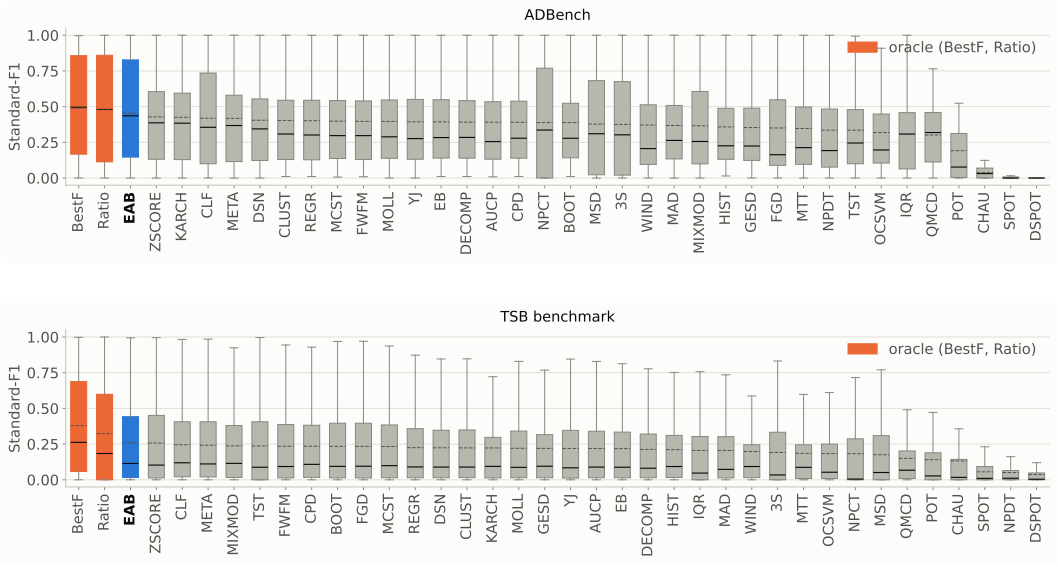


Figure 6: Box-plot results of thresholding methods on the ADBench and TSB benchmarks, sorted by mean performance.

(a) TSB-AD

Competitor	W	L	T	$\Delta\bar{F}_1$ (%)	Sig.
FGD	1397	806	218	+10.7%	✓
MCST	1403	837	181	+11.1%	✓
FWFM	1401	813	207	+10.3%	✓
GESD	1392	832	197	+17.9%	✓
CPD	1318	920	183	+10.3%	✓
META	1340	907	174	+7.0%	✓
MIXMOD	1288	980	153	+9.1%	✓
ZSCORE	1215	944	262	+1%	✓
CLF	1196	999	226	+5.5%	✓

(b) ADBench

Competitor	W	L	T	$\Delta\bar{F}_1$ (%)	Sig.
DECOMP	54	32	9	+16.2%	✓
EB	56	30	9	+15.6%	✓
DSN	55	31	9	+12.4%	✓
MCST	56	30	9	+14.3%	✓
REGR	56	30	9	+13.3%	✓
MOLL	53	33	9	+14.7%	✓
CLUST	53	33	9	+12.8%	✓
KARCH	51	34	10	+6.9%	✓
ZSCORE	48	36	11	+6.2%	✓

Table 1: Pairwise comparison of EAB with the nine best thresholders at their best normalisation. W/L/T denote the cases on which EAB wins, loses, and ties. $\Delta\bar{F}_1$ denotes the relative improvement of EAB in mean F_1. Significance is assessed by the two-sided Wilcoxon signed-rank test over paired per-task scores. EAB is significantly better than all nine competitors on both benchmarks.

AD). **EAB ranks third overall and first among all unsupervised methods on both benchmarks.** The two methods ranked above EAB-*BestF* and *Ratio*-are supervised oracles that select the threshold maximizing F1 given ground-truth labels, representing a ceiling unattainable in deployment without labeled data. Among genuinely unsupervised methods, EAB achieves the highest median F1 across datasets on both benchmarks, outperforming parametric mixture models, change-point detectors, z-score heuristics, and regression-based approaches. Tables 1a and 1b report pairwise win/loss statistics against nine representative unsupervised baselines, with statistical significance assessed via the Wilcoxon signed-rank test ($p < 0.05$). In contrast, common methods ($3S$, POT, IQR) often underperform, with POT being particularly sensitive to parameter choices (Liu et al., 2023). Finally, we observe that using top thresholding methods is better than native decision boundaries of AD methods in 84% of cases (Appendix D) and we report a complementary critical difference diagram analysis in Appendix D.

5.2 Robustness Across Score Normalizations

We examine EAB performance broken down by score normalisation strategy. Across the three normalisations evaluated on ADBench, EAB ranks first under all three; the margin is widest under min-max (mean F1 0.512 vs. 0.470 for *ZSCORE*, with 9/9 pairwise comparisons reaching significance).

Under Gaussian normalisation, EAB leads by average rank but is level with *CLUST* on mean F1 (0.448 vs. 0.447). The gamma case is weaker: EAB retains the top rank but *REGR* is marginally ahead on mean F1 (0.426 vs. 0.424) and only 4/9 comparisons reach significance, partly because this panel rests on 81 tasks rather than 143–148, as most competitors fail under gamma on the harder datasets. On TSB-AD, EAB ranks first under 8 of 10 normalisation strategies. Its strongest absolute results occur under the smoothing family (*max*, *mean*, *EWMA*, and *median*) where mean F1 reaches 0.256–0.271 and all 9 pairwise comparisons are significant. The two strategies where EAB does not lead are min-max (4th) and Gaussian (3rd); z-score yields a nominal first that carries little weight, as *FWFM* achieves a higher mean F1 there and only 1/9 comparisons is significant. **Mechanistic interpretation.** The pattern follows directly from EAB's design. Min-max normalisation is an affine map that preserves the heavy tail of the raw score distribution; the excess-energy surface is then dominated by a small number of extreme points, making the centroid estimator less stable. Smoothing transforms (EWMA, median, mean) and CDF-style transforms compress that tail into the regime for which the centroid estimator is designed, producing the observed performance gap. This analysis identifies a natural deployment heuristic: pairing EAB with a smoothing or rank-compressing normalisation maximises its advantage over parametric competitors.

5.3 Runtime Performance

Thresholding methods efficiency Figure 7 compares the runtime of the top-performing thresholding methods from both domains, along with the widely used *3S* baseline, against the fastest and slowest anomaly detection models in our study (POLY and TimesFM, both from the TSB-AD domain). **EAB, implemented via prefix-sum decomposition with O(n log n) complexity, runs at roughly 10 milliseconds per dataset-fastest than the fastest anomaly detection models and orders of magnitude faster than parametric alternatives.** For context, *MIXMOD* is slower than the slowest anomaly detector evaluated; *CLF* suffers from quadratic time complexity, leading to poor scalability; 3S (and ZSCORE, which performs the same iterations) with the lowest possible complexity O(n), are faster than EAB (as expected) but with trade-off in performance. Overall, EAB combines top-ranked accuracy with runtime consistently faster than the fastest anomaly detector, resolving the efficiency-accuracy tension that afflicts parametric competitors.

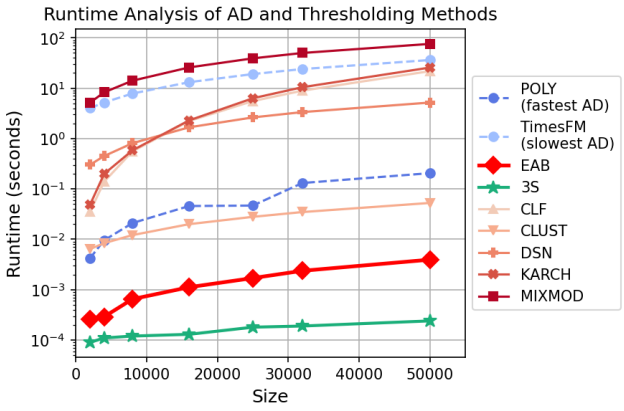


Figure 7: Runtime of top thresholding methods and $3S$ vs. fastest/slowest AD methods.

6 Conclusion

We introduced the Energy-Based Adaptive Boundary (EAB), a parameter-free, distribution-free algorithm for unsupervised anomaly score thresholding. EAB grounds threshold selection in the two-sample energy distance, calibrates against the Gini mean difference to avoid false triggers on homogeneous data, and identifies the final threshold as the energy-weighted centroid of high-separation candidate points-all in O(n log n) time via prefix-sum decomposition. Evaluated on benchmark datasets spanning time-series (TSB-AD, 17 domains) and tabular/image data (ADBench), EAB achieves the highest average rank among all unsupervised thresholding methods on both benchmarks, with statistically significant F1 gains of up to +17.7 percentage points on TSB-AD and +16.2 percentage points on ADBench over the strongest competitor (Wilcoxon signed-rank, $p < 0.05$), while running at speeds competitive with simple heuristics. Our study also finds that more than half of the published thresholding methods examined require labeled data, while the parametric distributional assumptions assessed are violated in nearly all evaluated cases. These findings underscore the need for adaptive, distribution-free thresholding methods. We hope EAB and the accompanying open benchmark suite encourage broader recognition of thresholding as a first-class component of anomaly detection and advance reproducible evaluation and principled threshold selection.

Reproducibility Statement

The EAB algorithm is fully specified by the pseudocode in Algorithm 1 and the mathematical derivations in Section 4. It has no hyperparameters requiring tuning. All 38 thresholding baselines are implemented using publicly available code (see https://anonymous.4open.science/r/Last-Mile-AD); implementation references and version numbers are provided in Appendix B. Evaluation uses the publicly available TSB-AD (Liu & Paparrizos, 2024) and ADBench (Han et al., 2022) benchmarks. Upon publication, we will release code reproducing all experiments, including baseline wrappers, evaluation scripts, and dataset loaders, at an anonymized repository.

Ethics Statement

This work presents an algorithm for anomaly score thresholding and a benchmark evaluation of existing methods. We foresee no direct ethical concerns arising from this research. Anomaly detection systems informed by this work could be applied in sensitive domains such as medical diagnostics or network security; practitioners should pair any automated threshold selection with domain-appropriate validation, especially where false positives or false negatives carry significant consequences. This work complies with the ICLR Code of Ethics.

AI Usage Disclosure

The authors used AI-assisted writing tools for grammar and phrasing suggestions during manuscript preparation. All scientific content, experimental design, analysis, and conclusions are solely the work of the authors.

References

Charu C. Aggarwal. *Outlier Analysis*. Springer International Publishing, 2017. ISBN 9783319475783. doi: 10.1007/978-3-319-47578-3. URL http://dx.doi.org/10.1007/978-3-319-47578-3.

Subutai Ahmad, Alexander Lavin, Scott Purdy, and Zuha Agha. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing*, 262:134–147, 2017.

Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian conference on computer vision*, pp. 622–637. Springer, 2018.

Sarah Alnegheimish, Laure Berti-Equille, and Kalyan Veeramachaneni. Orionbench: Benchmarking time series generative models in the service of the end-user. In *2024 IEEE International Conference on Big Data (BigData)*, pp. 1215–1222. IEEE, 2024.

Mufda Jameel Alrawashdeh. An adjusted grubbs' and generalized extreme studentized deviation. *Demonstratio Mathematica*, 54(1):548–557, 2021.

Daichi Amagata, Makoto Onizuka, and Takahiro Hara. Fast and exact outlier detection in metric spaces: a proximity graph-based approach. In *Proceedings of the 2021 International Conference on Management of Data*, pp. 36–48, 2021.

Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A Zuluaga. Usad: Unsupervised anomaly detection on multivariate time series. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 3395–3404, 2020.

Vilijandas Bagdonavičius and Linas Petkevičius. Multiple outlier detection tests for parametric models. *Mathematics*, 8(12), 2020. ISSN 2227-7390. doi: 10.3390/math8122156. URL https://www.mdpi.com/2227-7390/8/12/2156.

Alberto Barbado, Óscar Corcho, and Richard Benjamins. Rule extraction in unsupervised anomaly detection for model explainability: Application to oneclass svm. *Expert Systems with Applications*, 189:116100, 2022.

Jean-Marc Bardet and Solohaja-Faniaha Dimby. A new non-parametric detector of univariate outliers for distributions with unbounded support. *Extremes*, 20(4):751–775, 2017.

V Barnett. Outliers in statistical data. *John Wiley & Sons google schola*, 2:705–708, 1994.

Graciela Boente, Ana M Pires, and Isabel M Rodrigues. Influence functions and outlier detection under the common principal components model: A robust approach. *Biometrika*, 89(4):861–875, 2002.

LN Bol'shev and M Ubaidullaeva. Chauvenet's test in the classical theory of errors. *Theory of Probability & Its Applications*, 19(4):683–692, 1975.

Paul Boniol, John Paparrizos, Themis Palpanas, and Michael J. Franklin. Sand: streaming subsequence anomaly detection. *Proc. VLDB Endow.*, 14(10):1717–1729, June 2021. ISSN 2150-8097. doi: 10.14778/3467861.3467863. URL https://doi.org/10.14778/3467861.3467863.

Paul Boniol, Qinghua Liu, Mingyi Huang, Themis Palpanas, and John Paparrizos. Dive into time-series anomaly detection: A decade review. *arXiv preprint arXiv:2412.20512*, 2024.

Raghavendra Chalapathy and Sanjay Chawla. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019.

Tianqi Chen. Xgboost: A scalable tree boosting system. *Cornell University*, 2016.

Dhruv Choudhary, Arun Kejariwal, and Francois Orsini. On the runtime-efficacy trade-off of anomaly detection techniques for real-time streaming data. *arXiv preprint arXiv:1710.04735*, 2017.

Daniele Coin. Testing normality in the presence of outliers. *Statistical Methods and Applications*, 17: 3–12, 2008.

Ralph D'agostino and Egon S Pearson. Tests for departure from normality. empirical results for the distributions of b 2 and b. *Biometrika*, 60(3):613–622, 1973.

Ralph B d'Agostino. An omnibus test of normality for moderate and large size samples. *Biometrika*, 58(2):341–348, 1971.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10148–10167. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/das24c.html.

Paul Fearnhead and Guillem Rigaill. Changepoint detection in the presence of outliers. *Journal of the American Statistical Association*, 114(525):169–183, 2019.

Changyong Feng, Hongyue Wang, Naiji Lu, Tian Chen, Hua He, Ying Lu, and Xin M Tu. Log-transformation and its implications for data analysis. *Shanghai Archives of Psychiatry*, 26(2): 105–109, April 2014. ISSN 1002-0829. doi: 10.3969/j.issn.1002-0829.2014.02.009.

Michael Friendly, Georges Monette, and John Fox. Elliptical Insights: Understanding Statistical Methods through Elliptical Geometry. *Statistical Science*, 28(1):1 – 39, 2013. doi: 10.1214/12-STS402. URL https://doi.org/10.1214/12-STS402.

Apostolos Giannoulidis, Anastasios Gounaris, Nikodimos Nikolaidis, Athanasios Naskos, and Daniel Caljouw. Investigating thresholding techniques in a real predictive maintenance scenario. *SIGKDD Explor. Newsl.*, 24(2):86–95, December 2022. ISSN 1931-0145. doi: 10.1145/3575637.3575651. URL https://doi.org/10.1145/3575637.3575651.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: a family of open time-series foundation models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Songqiao Han, Xiyang Hu, Hailiang Huang, Minqi Jiang, and Yue Zhao. Adbench: Anomaly detection benchmark. *Advances in neural information processing systems*, 35:32142–32159, 2022.

Navid Hashemi, Eduardo Verdugo German, Jonatan Pena Ramirez, and Justin Ruths. Filtering approaches for dealing with noise in anomaly detection. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 5356–5361. IEEE, 2019.

Zengyou He, Xiaofei Xu, and Shengchun Deng. Discovering cluster-based local outliers. *Pattern recognition letters*, 24(9-10):1641–1650, 2003.

Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 387–395, 2018.

Dmitri Iouchtchenko, Neil Raymond, Pierre-Nicholas Roy, and Marcel Nooijen. Deterministic and quasi-random sampling of optimized gaussian mixture distributions for vibronic monte carlo. *arXiv preprint arXiv:1912.11594*, 2019.

Alec Jacobson, Ladislav Kavan, and Olga Sorkine-Hornung. Robust inside-outside segmentation using generalized winding numbers. *ACM Transactions on Graphics (TOG)*, 32(4):1–12, 2013.

Mohsen Joneidi. Sparse auto-regressive: Robust estimation of ar parameters. *arXiv preprint arXiv:1306.3317*, 2013.

Hermann Karcher. Riemannian center of mass and mollifier smoothing. *Communications on pure and applied mathematics*, 30(5):509–541, 1977.

Michiel A Keyzer, BGJS Sonneveld, et al. Using the mollifier method to characterize datasets and models: the case of the universal soil loss equation. *ITC Journal*, 3(4):263–272, 1997.

Frank Klawonn and Frank Rehm. *Cluster Analysis for Outlier Detection*, pp. 214–218. IGI Global, 2009. doi: 10.4018/978-1-60566-010-3.ch035. URL http://dx.doi.org/10.4018/978-1-60566-010-3.ch035.

Krzysztof Kotowski, Christoph Haskamp, Jacek Andrzejewski, Bogdan Ruszczak, Jakub Nalepa, Daniel Lakey, Peter Collins, Aybike Kolmas, Mauro Bartesaghi, Jose Martinez-Heras, et al. European space agency benchmark for anomaly detection in satellite telemetry. *arXiv preprint arXiv:2406.17826*, 2024.

Hans-Peter Kriegel, Peer Kroger, Erich Schubert, and Arthur Zimek. Interpreting and unifying outlier scores. In *Proceedings of the 2011 SIAM International Conference on Data Mining*, pp. 13–24. SIAM, 2011.

Daniel Kulik, Sebastian Schmidl, and Lorenzo Perini. Kulikdm/pythresh: v1.0.0 (v1.0.0), 2025.

Kwei-Herng Lai, Daochen Zha, Junjie Xu, Yue Zhao, Guanchu Wang, and Xia Hu. Revisiting time series outlier detection: Definitions and benchmarks. In *Thirty-fifth conference on neural information processing systems datasets and benchmarks track (round 1)*, 2021.

Jia Li, Shimin Di, Yanyan Shen, and Lei Chen. Fluxev: a fast and effective unsupervised framework for time-series anomaly detection. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pp. 824–832, 2021.

Zhi Li, Hong Ma, and Yongbing Mei. A unifying method for outlier and change detection from data streams based on local polynomial fitting. In Zhi-Hua Zhou, Hang Li, and Qiang Yang (eds.), *Advances in Knowledge Discovery and Data Mining*, pp. 150–161, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. ISBN 978-3-540-71701-0.

Liang Liu, Ling Tian, Zhao Kang, and Tianqi Wan. Spacecraft anomaly detection with attention temporal convolution networks. *Neural Computing and Applications*, 35(13):9753–9761, May 2023. ISSN 1433-3058. doi: 10.1007/s00521-023-08213-9. URL https://doi.org/10.1007/s00521-023-08213-9.

Qinghua Liu and John Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=R6kJtWsTGy.

Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, and Puneet Agarwal. Long short term memory networks for anomaly detection in time series. In *23rd European Symposium on Artificial Neural Networks, ESANN 2015, Bruges, Belgium, April 22-24, 2015*, 2015. URL https://www.esann.org/sites/default/files/proceedings/legacy/es2015-56.pdf.

Michael A Martin and Steven Roberts. An evaluation of bootstrap methods for outlier detection in least squares regression. *Journal of Applied Statistics*, 33(7):703–720, 2006.

Nesryne Mejri, Laura Lopez-Fuentes, Kankana Roy, Pavel Chernakov, Enjie Ghorbel, and Djamila Aouada. Unsupervised anomaly detection in time-series: An extensive evaluation and analysis of state-of-the-art methods. *Expert Systems with Applications*, pp. 124922, 2024.

Mohsin Munir, Shoaib Ahmed Siddiqui, Andreas Dengel, and Sheraz Ahmed. Deepant: A deep learning approach for unsupervised anomaly detection in time series. *IEEE Access*, 7:1991–2005, 2019. doi: 10.1109/ACCESS.2018.2886457.

Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2):1–38, 2021.

Guansong Pang, Chunhua Shen, Huidong Jin, and Anton Van Den Hengel. Deep weakly-supervised anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1795–1807, 2023.

John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S Tsay, Themis Palpanas, and Michael J Franklin. Tsb-uad: an end-to-end benchmark suite for univariate time-series anomaly detection. *Proceedings of the VLDB Endowment*, 15(8):1697–1711, 2022.

SS Pawar et al. Periodicity detection of outlier sequences using constraint based pattern tree with mad. *arXiv preprint arXiv:1507.01685*, 2015.

Lorenzo Perini, Paul-Christian Bürkner, and Arto Klami. Estimating the contamination factor's distribution in unsupervised anomaly detection. In *International Conference on Machine Learning*, pp. 27668–27679. PMLR, 2023.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31, 2018.

Zhuang Qi, Dazhi Jiang, and Xiaming Chen. Iterative gradient descent for outlier detection. *International Journal of Wavelets, Multiresolution and Information Processing*, 19(04):2150004, 2021.

Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. Efficient algorithms for mining outliers from large data sets. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 427–438, 2000.

Jakob Raymaekers and Peter J Rousseeuw. Transforming variables to central normality. *Machine Learning*, 113(8):4953–4975, 2024.

Ke Ren, Haichuan Yang, Yu Zhao, Wu Chen, Mingshan Xue, Hongyu Miao, Shuai Huang, and Ji Liu. A robust auc maximization framework with simultaneous outlier detection and feature selection for positive-unlabeled classification. *IEEE transactions on neural networks and learning systems*, 30(10):3072–3083, 2018.

Divish Rengasamy, Benjamin C Rothwell, and Grazziela P Figueredo. Towards a more reliable interpretation of machine learning outputs for safety-critical systems using feature importance fusion. *Applied Sciences*, 11(24):11854, 2021.

Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. *arXiv preprint arXiv:1906.02694*, 2019.

Sebastian Schmidl, Felix Naumann, and Thorsten Papenbrock. Autotsad: Unsupervised holistic anomaly detection for time series data. *Proceedings of the VLDB Endowment*, 17(11):2987–3002, 2024.

Dominique Shipmon, Jason Gurevitch, Paolo M Piselli, and Steve Edwards. Time series anomaly detection: Detection of anomalous drops with limited features and sparse examples in noisy periodic data. Technical report, Google Inc., 2017. URL `https://arxiv.org/abs/1708.03665`.

Mei-Ling Shyu, Shu-Ching Chen, Kanoksri Sarinnapakorn, and LiWu Chang. A novel anomaly detection scheme based on principal component classifier. 2003.

Alban Siffer, Pierre-Alain Fouque, Alexandre Termier, and Christine Largouet. Anomaly detection in streams with extreme value theory. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1067–1075, 2017.

Junho Song, Keonwoo Kim, Jeonglyul Oh, and Sungzoon Cho. Memto: Memory-guided transformer for multivariate time series anomaly detection. *Advances in Neural Information Processing Systems*, 36:57947–57963, 2023.

Gábor J. Székely and Maria L. Rizzo. Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143(8):1249–1272, 2013.

KK Thanammal, JS Jayasudha, RR Vijayalakshmi, and S Arumugaperumal. Effective histogram thresholding techniques for natural images using segmentation. *Journal of Image and Graphics*, 2 (2):113–116, 2014.

Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B*, 63(2):411–423, 2001.

Willem van Veluw. Application of mixture models to threshold anomaly scores. Master's thesis, Utrecht University, 2023.

Frank Wilcoxon. Individual comparisons by ranking methods. In *Breakthroughs in statistics: Methodology and distribution*, pp. 196–202. Springer, 1992.

Jiehui Xu. Anomaly transformer: Time series anomaly detection with association discrepancy. *arXiv preprint arXiv:2110.02642*, 2021.

Takehisa Yairi, Yoshikiyo Kato, and Koichi Hori. Fault detection by mining association rules from house-keeping data. In *Proc. of International Symposium on Artificial Intelligence, Robotics and Automation in Space*, volume 3. Citeseer, 2001.

Jiawei Yang, Susanto Rahardja, and Pasi Fränti. Outlier detection: how to threshold outlier scores? In *Proceedings of the international conference on artificial intelligence, information processing and cloud computing*, pp. 1–6, 2019.

Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. Matrix profile i: All pairs similarity joins for time series: A unifying view that includes motifs, discords and shapelets. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pp. 1317–1322, 2016. doi: 10.1109/ICDM.2016.0179.

Aoqian Zhang, Shuqing Deng, Dongping Cui, Ye Yuan, and Guoren Wang. An experimental evaluation of anomaly detection in time series. *Proceedings of the VLDB Endowment*, 17(3): 483–496, 2023.

Yue Zhao and Maciej K Hryniewicki. Xgbod: improving supervised outlier detection with unsupervised representation learning. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2018.

Yue Zhao, Ryan A Rossi, and Leman Akoglu. Automating outlier detection via meta-learning. *arXiv preprint arXiv:2009.10606*, 2020.

Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International conference on learning representations*, 2018.

Detector	Description
KMeansAD	(Yairi et al., 2001) Measures distance to the centroid of the assigned cluster using k-means.
KShapeAD	(Boniol et al., 2021) Uses k-Shape clustering to identify normal patterns; scores are distances between subsequences and patterns.
Matr. Pr. (MP)	(Yeh et al., 2016) Detects abnormal subsequences using nearest neighbor distance.
POLY	(Li et al., 2007) Fits a polynomial model to forecast values; prediction errors indicate anomalies.
Sub-PCA	(Aggarwal, 2017) Projects data to a lower-dimensional plane; distant points are anomalies.
USAD	(Audibert et al., 2020) Combines reconstruction and discriminator losses from adversarially trained autoencoders.
CNN	(Munir et al., 2019) Predicts the next timestamp using CNN and compares it to actual values.
LSTMAD	(Malhotra et al., 2015) Uses LSTM to predict future values; errors are used as anomaly scores.
TimesFM	(Das et al., 2024) Pretrained decoder-attention model using time-series patching.
MOMENT	(Goswami et al., 2024) Pretrained TS encoder using masked time-series modeling.

Table 2: SOTA time-series AD algorithms.

Detector	Description
CBLOF	(He et al., 2003) Uses the size of its cluster and its distance to the nearest large cluster to assign anomaly scores.
KNN	(Ramaswamy et al., 2000) Uses k nearest neighbors' distances to determine outliers.
PCA	(Shyu et al., 2003) Projects data into PCA space; distant points are flagged as outliers.
DAGMM	(Zong et al., 2018) Autoencoder + Gaussian mixture; high-energy points are anomalies.
DeepSAD	(Ruff et al., 2019) Learns a space where normal data clusters tightly and anomalies are pushed away.
GANomaly	(Akcay et al., 2018) Encoder-decoder-encoder with adversarial training for anomaly detection.
PReNet	(Pang et al., 2023) Learns pairwise relations between anomalies and unlabeled data.
XGBOD	(Zhao & Hryniewicki, 2018) Combines outlier scores with features using XGBoost.
MLP	(Rosenblatt, 1958) Learns nonlinear patterns to distinguish anomalies.
XGB	(Chen, 2016) Boosted trees to identify deviations.
CatB	(Prokhorenkova et al., 2018) Ordered boosted trees for anomaly detection.

Table 3: SOTA AD algorithms in ADBench.

A Anomaly Detection Models

Tables 2 and 3 list the anomaly detectors whose scores we threshold, adopted from the top-ranked methods of TSB-AD (Liu & Paparrizos, 2024) and ADBench (Han et al., 2022), respectively.

B Description of Thresholding Techniques

Thresholding methods can be categorized based on how they set thresholds: static thresholds, which stay fixed over the entire time series, and dynamic thresholds, which adjust over time to handle concept drift. These methods can also function in two different settings: an offline setting, where the entire time series is available beforehand, and an online setting, where data points are processed incrementally as they arrive. In this work, we investigate 38 diverse thresholding methods spanning multiple families (Perini et al., 2023; Kulik et al., 2025). Table 4 lists them by family, and each is described below.

Statistical-based: POT (Siffer et al., 2017), which utilizes the Extreme Value Theory to calculate a threshold, and its extensions $SPOT$, $DSPOT$, $FluxEV$ (Li et al., 2021). The KS deviation method is a widely used unsupervised thresholding method that sets the threshold at K times the standard deviation S from the mean. Its dynamic counterpart, MSD (Moving Standard Deviation), adapts this approach for streaming data. TST (Yang et al., 2019) applies the $2S$ rule twice (excluding anomalies from first threshold). Another extension of the KS rationale is $NPDT$ (Hundman et al., 2018), which continuously perform an optimization function to derive on the correct K for anomaly score streams, where $NPCT$ is used for the batch application of the same logic. MAD (Pawar et al., 2015) instead of S uses the scaled Median Absolute Deviation for calculating threshold. $BOOT$ (Martin & Roberts, 2006) uses bootstrapping to calculate S, and leverages the confidence interval of

Family	Thresholding method
Statistical-based	POT (Siffer et al., 2017), $SPOT$ (Li et al., 2021), $DSPOT$ (Li et al., 2021), $FluxEV$ (Li et al., 2021), KS ($\mu + \mathrm{K} \cdot \sigma$), $ZSCORE$ (Bagdonavičius & Petkevičius, 2020), MSD (Giannoulidis et al., 2022), TST (Yang et al., 2019), $NPDT$ (Hundman et al., 2018), $NPCT$ (Hundman et al., 2018), MAD (Pawar et al., 2015), $BOOT$ (Martin & Roberts, 2006), $KARCH$ (Karcher, 1977)
Statistical test-based	$GESD$ (Alrawashdeh, 2021), $MCST$ (Coin, 2008), MTT (Rengasamy et al., 2021)
Kernel-based	$AUCP$ (Ren et al., 2018), FGD (Qi et al., 2021), $FWFM$ (Joneidi, 2013), $OCSVM$ (Barbado et al., 2022)
Transformation-based	$MOLL$ (Keyzer et al., 1997), YJ (Raymaekers & Rousseeuw, 2024), CPD (Fearnhead & Rigaill, 2019), $DECOMP$ (Boente et al., 2002)
Filter-based	$FILTER$ (Hashemi et al., 2019), $HIST$ (Thanammal et al., 2014), $META$ (Zhao et al., 2020)
Mixture Model-based	γGMM (Perini et al., 2023), $MIXMOD$ (Veluw, 2023), $CHAU$ (Bol'shev & Ubaidullaeva, 1975), DSN (Amagata et al., 2021), $CLUST$ (Klawonn & Rehm, 2009)
Quantile-based	IQR (Bardet & Dimby, 2017), $QMCD$ (Iouchtchenko et al., 2019)
Regression-based	CLF (Aggarwal, 2017), $REGR$ (Aggarwal, 2017)
Curve-based	EB (Friendly et al., 2013), $WIND$ (Jacobson et al., 2013)

Table 4: Anomaly score thresholding methods.

the standard deviation of the anomaly scores to derive a threshold. Finally, $KARCH$ employs the Riemannian Center of Mass (Karcher, 1977) to compute a threshold, identifying outliers as scores exceeding the (Karcher mean) by one standard deviation.

Statistical test-based: $GESD$ (Alrawashdeh, 2021) iteratively applies the Generalized Extreme Studentized test to identify outliers. $MCST$ (Coin, 2008) iteratively applies the Shapiro-Wilk test to anomaly scores, while the Modified Thompson Tau test (MTT) (Rengasamy et al., 2021) can be used to iteratively identify anomalies.

Kernel-based: $AUCP$ (Ren et al., 2018) and FGD (Qi et al., 2021) leverage a kernel density estimator (KDE) to estimate outliers. Moreover, $FWFM$ (Joneidi, 2013) estimates the KDE of anomaly scores, identifies the main peak, computes its full width at minimum base, and sets the threshold as the normalized base width over the KDE points. Finally, $OCSVM$ (Barbado et al., 2022) uses a one-class SVM to find anomalies.

Transformation-based: $MOLL$ performs smoothing of anomaly scores using Friedrichs' mollifier (Keyzer et al., 1997), whereas YJ (Raymaekers & Rousseeuw, 2024) applies the Yeo-Johnson transformation to convert a non-normal distribution into a normal distribution. A Change Point Detection method, CPD (Fearnhead & Rigaill, 2019), finds a threshold by detecting a change point in the probability density function (PDF) of anomaly scores. $DECOMP$ (Boente et al., 2002) isolates regions where anomalies are likely to occur based on the cumulative distribution function (CDF) of the anomaly scores.

Filter-based: $FILTER$ (Hashemi et al., 2019) filters the scores based on a preselected filter (e.g. Gaussian). $HIST$ (Thanammal et al., 2014) applies different filters upon histograms of the anomaly scores. $META$ (Zhao et al., 2020) uses pre-trained meta-models to classify outliers based on a majority vote across meta-model predictions.

Mixture Model-based: γGMM (Perini et al., 2023) estimates the contamination factor γ by considering scores from multiple detectors, fitting a Dirichlet Process Gaussian Mixture Model. $MIXMOD$ (Veluw, 2023) models anomaly scores as a mixture of two distributions, fits them via Expectation-Maximization, and selects the best model using a scoring criterion. The threshold is set where the posterior probabilities of both components are equal.

Normality-based: $CHAU$ (Bol'shev & Ubaidullaeva, 1975) computes z-scores and applies Chauvenet's Criterion to set a probability-based threshold. DSN (Distance Shift from Normal) (Amagata et al., 2021) identifies outliers by measuring the shift in distance between the PDF or CDF of the initial anomaly score distribution and a normal distribution. $CLUST$ (Klawonn & Rehm, 2009)

Algorithm 1 Energy-based Adaptive Boundary (EAB)

Require: Anomaly scores $s_1, \ldots, s_n$
Sort scores to obtain $s_{(1)} \leq \cdots \leq s_{(n)}$ ▷ $\mathcal{O}(n \log n)$
Form prefix sums $C_k = \sum_{i \leq k} s_{(i)}$ and $D_k = \sum_{i \leq k} i\, s_{(i)}$ ▷ $\mathcal{O}(n)$
Compute Gini mean diff: $G \leftarrow \frac{2}{n^2}\left[2D_n - (n+1)C_n\right]$
for $t = 1, \ldots, n-1$ **do**
    Compute $\bar{d}_{AB}, \bar{d}_{AA}, \bar{d}_{BB}$ using C_t, D_t, C_n, D_n ▷ $\mathcal{O}(1)$ each
    $S(t) \leftarrow \frac{t\,(n-t)}{n}\left[2\bar{d}_{AB} - \bar{d}_{AA} - \bar{d}_{BB} - 2G\right]$
    $w(t) \leftarrow \max(S(t), 0)$
end for
if $\sum w(t) > 0$ **then**
    $\tau \leftarrow \frac{\sum_t w(t)\, s_{(t+1)}}{\sum_t w(t)}$
else
    $\tau \leftarrow s_{(\arg\max_t S(t)+1)}$ ▷ Fallback to extreme tail
end if
return $\hat{y}_i = \mathbf{1}\{s_i \geq \tau\}$

employs clustering with two or more clusters and identifies outliers as data points that are not assigned to the "main" cluster.

Quantile-based: IQR Inter-Quartile Region (Bardet & Dimby, 2017) sets as outlier values beyond the third quartile plus 1.5 times the inter-quartile region. Quasi-Monte Carlo Discrepancy ($QMCD$) (Iouchtchenko et al., 2019) defines threshold as the quintile of one minus the Quasi-Monte Carlo discrepancy.

Regression-based: The coefficients of a pre-trained linear classifier on synthetic datasets with known labels are used in CLF method (Aggarwal, 2017). $REGR$ (Aggarwal, 2017) also identifies outliers by performing linear regression between anomaly scores and normalized values of a normal distribution.

Curve-based: EB (Friendly et al., 2013) uses elliptical boundaries with random eccentricities to compute thresholds, selecting the boundary that best aligns with the median inlier count. Finally, $WIND$ (Jacobson et al., 2013) detects outliers using the winding number to identify distribution shifts, setting the threshold at the mean interaction point with a reference normal distribution.

C EAB Algorithm

Algorithm 1 presents the pseudocode of EAB thresholding method.

D Supplementary Analysis of Thresholding Methods Performance

Smoothing of Anomaly Scores

As mentioned in Section 3.4, the beneficial impact of smoothing anomaly scores has been observed across various methods. Here, we provide further detail supporting this observation. For instance, noise can influence the standard deviation of scores, thereby affecting methods such as $3S$. This issue is particularly prevalent in deep learning models, where noise is a major limitation of unsupervised deep anomaly detection methods (Chalapathy & Chawla, 2019). Such observation is supported by our findings, demonstrated in Figure 8, where all except one AD method benefit from the use of smoothing on the anomaly scores. The figure shows the percentage of cases each method achieved better, worse, and equal results, while ✓and symbols indicate whether results are statistically better (or worse) when using smoothing. We perform all pairwise comparisons with the Wilcoxon test (Wilcoxon, 1992), testing whether the performance of smoothed scores result in significant better or worse results.

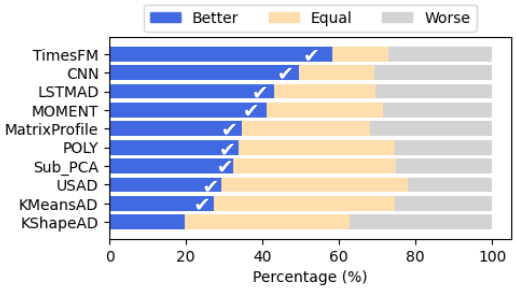


Figure 8: AD performance with and without score smoothing. 51 statistical difference.

Further Theoretical Analysis of Thresholding Performance Under Varying Data Conditions

One key reason certain thresholding approaches perform poorly is their reliance on strong assumptions about anomaly score distributions, which, as shown in Section 4.2 and RQ1, often do not hold. Additionally, methods like POT, SPOT, and DSPOT (Siffer et al., 2017) are designed for point anomalies, focusing on extreme values, as reflected in the results of Figure 7a. Importantly, current methods do not account for whether the anomaly is a point or a subsequence, and most thresholding approaches are not adaptive to this distinction. There are no existing tools to automatically infer the nature (point vs. subsequence), structure (e.g., periodicity, seasonality), or distributional form of the anomaly scores prior to thresholding. These issues become even more pronounced in streaming settings, where the data distribution may shift over time.

Extra Discussion on Runtime Performance

Here, we expand on the runtime performance discussion presented in Section 5.3. Figure 7 shows that $KARCH$, CLF, and DSN can process 10,000 timestamps in under a second. While DSN is slower on small datasets, it scales better on larger ones due to its linear complexity compared to CLF's quadratic complexity . In contrast, $MIXMOD$ takes over 10 seconds, as it models anomaly scores using Expectation-Maximization, which can slow execution if convergence is not achieved. In resource-constrained environments, applying threshold techniques in batches of anomaly scores, rather than processing the entire anomaly score set at once, can help reduce computational overhead, especially for methods with quadratic complexity. For instance, when evaluating CLF (one of the top-performing techniques), we tested its effectiveness by segmenting a time series into batches of 5,000 and 10,000 data points. This approach significantly reduced CLF's runtime but also impacted the *F1* score by decreasing it by 0.07%, as shown in Figures 9 and 10.

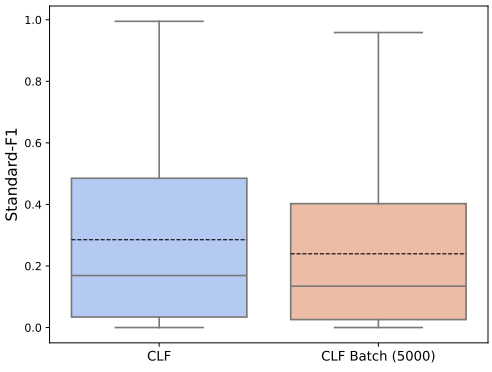


Figure 9: CLF Performance (original vs running on batches). We include only time series larger than 5000 length.

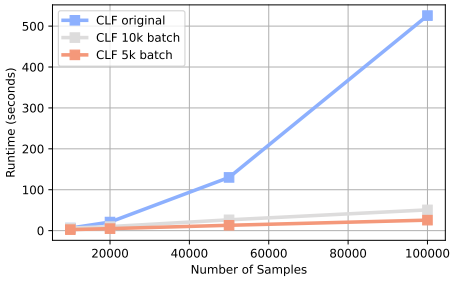


Figure 10: Execution time of CLF for different size of batches.

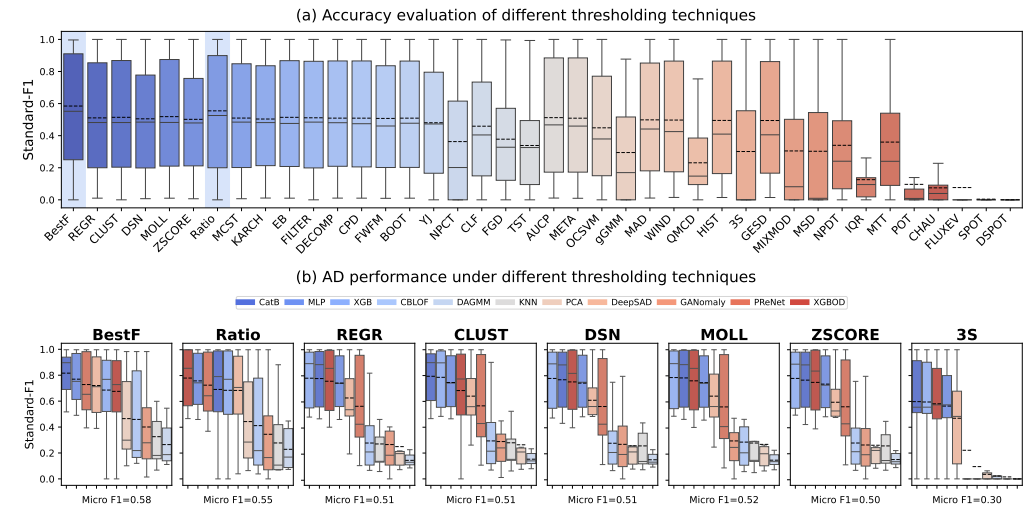


Figure 11: Accuracy evaluation (a) and performance discrepancy of AD methods (b) using different threshold methods on tabular datasets.

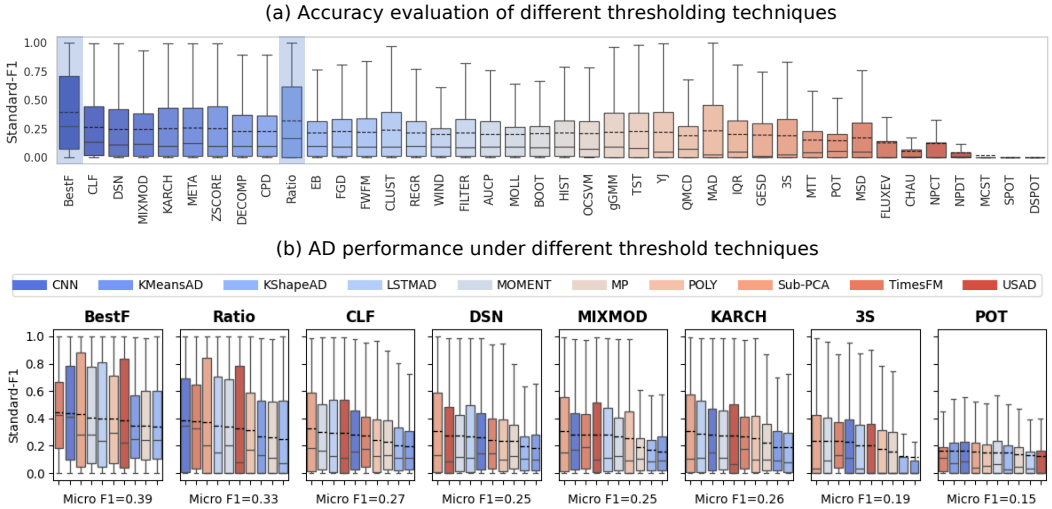


Figure 12: Accuracy evaluation (a) and performance discrepancy of AD methods (b) using different threshold methods on time series datasets.

TOP-PERFORMING THRESHOLDING METHODS

Figures 11 and 12 report extended results on the average *F1* score across thresholding methods, and demonstrate how different thresholding choices affect the relative ranking of anomaly detection algorithms. We observe that the supervised methods $BestF$ and $Ratio$, marked with a darker background, perform better than most of the unsupervised thresholding methods. One crucial observation is that the ranking of AD methods (Figure 11), especially in the time-series domain, differs between thresholding strategies. More importantly, although the ranking of methods between supervised thresholds is very similar, it changes significantly when using the top 7 thresholding methods. For instance, Sub-PCA ranks first under most thresholding methods, yet drops to third place when supervised thresholds are used. In Figure 12, we observe that, in the tabular domain, the ranking of the top anomaly detection algorithms also varies, although with less fluctuation than in the time-series domain (which presents inherent difficulties, as discussed earlier). $CatB$, MLP, and XGB generally achieve strong performance. However, an important observation is that the ranking of AD algorithms changes markedly when the $Ratio$ thresholding method is used. This sensitivity to the chosen contamination factor raises concerns about the robustness and reliability of $Ratio$-based thresholding, even though it is widely adopted in the literature (Perini et al., 2023; Zhang et al., 2023; Song et al., 2023; Kotowski et al., 2024; Xu, 2021).

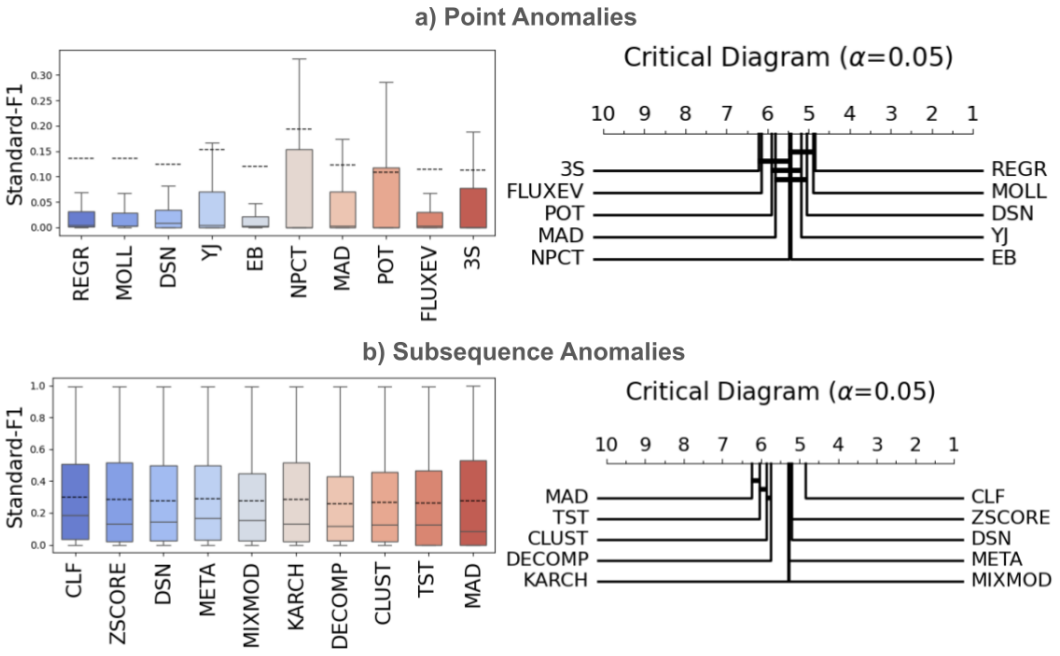


Figure 13: Evaluation of 10 SOTA thresholding methods (by average *F1*) for a) point and b) collective anomalies.

Point and Collective anomalies in TSAD

Figure 13 presents the results for 10 SOTA methods, ranked based on the average *F1* score.

Methods with Native Decision Boundaries

We additionally analyze the One-Class SVM anomaly detector, a canonical unsupervised anomaly detection method that natively performs both anomaly scoring and decision boundary selection. Specifically, we compare OCSVM's native binary predictions against post-hoc thresholding applied to its anomaly scores. For each dataset, using both Gamma scaling and raw anomaly scores without scaling, we apply the eight top-performing thresholding methods identified in Figure 4(a) and compare their *F1* scores against OCSVM's built-in decision boundary. A "win" is defined as a case where post-hoc thresholding achieves a higher *F1* score than the native OCSVM predictions. Across datasets, post-hoc thresholding achieves a win-rate of 0.84, outperforming the native decision boundary in 84% of cases, with an average relative performance improvement of 31%. These results further support the claim that threshold selection remains a critical and non-trivial component of the anomaly detection pipeline, even for methods equipped with native decision rules.

Comparison of performance usin Critical Diagrams

We reporte CD analysis, based on the Friedman test with the Nemenyi post-hoc test at $\alpha = 0.05$, on the top 10 unsupervised thresholding methods in Figure 14 and Figure 15 for the ADBench and TSB benchmarks respectively. EAB have a better ranking than competitors in both cases, with clear statistical significant difference in the TSB.

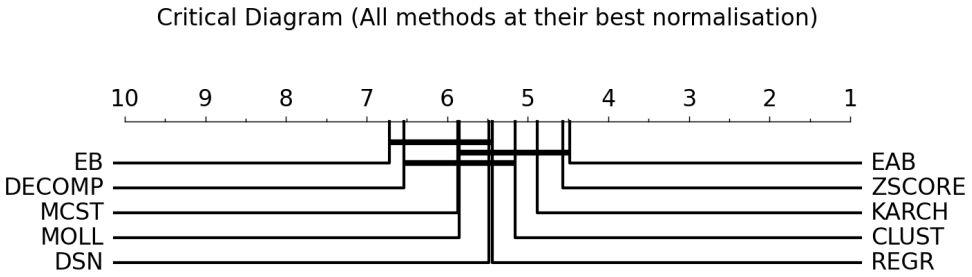


Figure 14: CD of thresholding performance on the ADBench datasets.

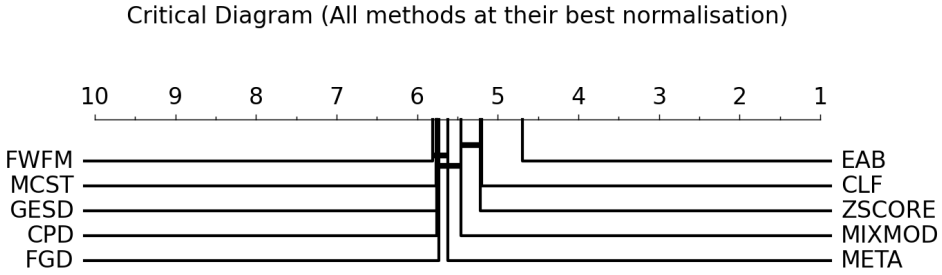


Figure 15: CD of thresholding performance on the TSB datasets.