PVBOOST: POSTERIOR-CALIBRATED VARIANCE BOOSTING FOR TIME-SERIES ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Time-series anomaly detection (TSAD) is typically performed in an unsupervised manner and follows a two-step pipeline: *scoring*, which quantifies the abnormality of each time step, and *thresholding*, which converts continuous scores into actionable decisions. Prior work largely emphasizes scoring, leaving thresholding to ad hoc heuristics despite their inherent coupling. This raises a key question: *given anomaly scores from a black-box model, how can we refine the scores and select a decision threshold jointly, in a principled, unsupervised manner?* To address this question, we propose PVBOOST, an unsupervised, parameter-light, and model-agnostic framework that jointly refines black-box anomaly scores and selects a decision threshold through a posterior-variance feedback loop. Mixture modeling estimates anomaly posteriors, which gate variance-based boosting to emphasize informative uncertainty while suppressing noise. Refined scores update the posteriors, whose trajectory also guides threshold selection, unifying refinement and decision-making without labels or access to detector internals. Experiments with 15 statistical and neural detectors on an established TSAD benchmark comprising 530 time series across nine domains demonstrate gains in both scoring and detection. In the univariate and multivariate settings, respectively, PVBOOST improves average VUS-PR by 4.3% and 11.4% over the original scores and average F-score by 68.3% and 107.3% over widely used three-sigma thresholding of those scores. These findings support coupled refinement and thresholding as a practical approach to more accurate unsupervised TSAD. Our implementation is available at `https://anonymous.4open.science/r/PVBoost`.

1 INTRODUCTION

Time-series anomaly detection (TSAD) aims to identify time steps that deviate significantly from underlying normal patterns of a sequence and is critical in applications such as system monitoring, fraud detection, and industrial diagnostics (Blázquez-García et al., 2021; Boniol et al., 2024; Zamanzadeh Darban et al., 2024). Due to the scarcity and annotation cost of labeled anomalies, TSAD is typically performed without supervision through two stages: *scoring*, which quantifies the abnormality of each time step, and *thresholding*, which converts continuous scores into binary decisions. Reliable detection therefore depends on both the quality of anomaly scores and the decision threshold used to interpret them. Figure 1 contrasts two strategies for converting anomaly scores into decisions: applying a heuristic threshold to raw scores and jointly refining the scores and selecting a threshold.

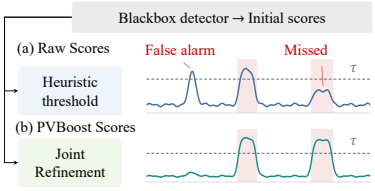

Figure 1: Heuristic thresholding (a) versus joint refinement (b). Shading marks anomalies; dashed lines indicate thresholds.

Despite extensive research on TSAD, obtaining reliable anomaly scores remains challenging. Unsupervised detectors rely on assumptions about anomalous behavior, such as deviation from local neighborhoods (Breunig et al., 2000; Ramaswamy et al., 2000), low likelihood under a learned distribution (Schölkopf et al., 1999; Aggarwal, 2017), or discrepancies between observed and reconstructed signals (Sakurada & Yairi, 2014; Tuli et al., 2022). These assumptions may not hold across time series with heterogeneous temporal dynamics, noise characteristics, and anomaly types. Accordingly, prior studies have found that there is no one-size-fits-all model for TSAD (Schmidl et al.,

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.

2022; Liu & Paparrizos, 2024; Qiu et al., 2025). The resulting scores may overemphasize normal fluctuations while assigning insufficient prominence to genuine anomalies.

Thresholding, the *last mile* from anomaly scores to actionable decisions, has received comparatively less attention. Common approaches use labels to select the threshold that maximizes F-score or to specify the anomaly ratio (Han et al., 2022; Liu & Paparrizos, 2024). Such information is unavailable in unsupervised deployment, where practitioners instead rely on statistical rules or tail-modeling methods whose assumptions and parameters may not transfer across datasets (Barnett et al.; Siffer et al., 2017). More fundamentally, threshold selection alone cannot resolve incorrect score ordering. Figure 1(a) illustrates a normal fluctuation receiving a higher score than a true anomaly: no threshold on these fixed scores can simultaneously reject the former and detect the latter. This limitation motivates addressing score quality and threshold selection together, as illustrated in Figure 1(b).

These observations raise a central question: *given a time series and anomaly scores from a black-box detector, how can we jointly refine the scores and select a decision threshold in a principled, unsupervised manner?* Addressing this question entails three challenges. First, score distributions vary across detectors and datasets, requiring distribution-aware refinement and decision rules. Second, refinement and thresholding are inherently coupled: refinement benefits from identifying likely anomalies, while threshold selection depends on the quality of the refined scores. Third, the absence of labels leaves only indirect and noisy signals for refinement, e.g., variance across successive refinements is larger on anomalies (Tifrea et al., 2022; Ye et al., 2023), which are intrinsically harder to model, but is equally inflated by ordinary noise on normal points.

In this paper, we propose PVBOOST, an unsupervised, parameter-light, and model-agnostic framework built around a *posterior-variance feedback loop*. Given a time series and a detector's scalar scores, PVBOOST refines them without access to detector internals or gradients. Its central insight is that anomaly posteriors and prediction variance are complementary: the posterior indicates *where* uncertainty is informative, and the variance *how much* the score should move. At each iteration, mixture-model posteriors of the current scores gate variance-based boosting to emphasize informative uncertainty over noise, and the refined scores inform the next posteriors. The resulting posterior trajectory also sets the threshold, linking refinement and decision-making in one procedure. Posterior momentum and variance-reduced weak learners stabilize refinement, and blending trajectory and final-score posteriors stabilizes the threshold. PVBOOST needs no labels or anomaly ratio, replaces user-set quantiles and tail parameters with a posterior-derived threshold, and uses one configuration for all detectors and time series without per-dataset or per-detector tuning.

We evaluate PVBOOST with 15 statistical and neural detectors on the TSB-AD benchmark (Liu & Paparrizos, 2024), comprising 530 univariate and multivariate time series across nine domains. On the univariate and multivariate suites, PVBOOST improves average VUS-PR (Boniol et al., 2025) by 4.3% and 11.4%, respectively. Its complete refinement-and-thresholding procedure improves average F-score by 68.3% and 107.3% relative to applying the widely used three-sigma rule to the original scores. These results demonstrate the practical value of coupling score refinement with threshold selection to improve unsupervised detection across heterogeneous base models.

We start with a review of related work (Section 2), then we present our contributions:

- **Last-mile problem framing.** We formulate the joint problem of score refinement and threshold selection on black-box detectors, and identify three fundamental challenges (Section 3.1).
- **Posterior-variance feedback framework.** We introduce PVBOOST, which couples mixture-modeling posteriors with variance-based boosting and uses the resulting posterior trajectory to guide threshold selection (Sections 3.2–3.4).
- **Stabilization mechanisms.** We incorporate posterior momentum, variance-reduced weak learners, and dual-source posterior blending to stabilize unsupervised refinement (Section 3.5).
- **Broad empirical validation.** Experiments across diverse detectors and domains demonstrate improvements in average scoring quality and detection F-score, with ablation and sensitivity analyses examining the contributions of individual components (Section 4).

Finally, we conclude with the implications of our work (Section 5).

2 Related Work

In this section, we first review TSAD methods (Section 2.1), followed by automated solutions (Section 2.2), and conclude with a discussion of thresholding techniques (Section 2.3).

2.1 TIME-SERIES ANOMALY DETECTION

TSAD identifies unusual patterns in temporal data, with challenges arising from diverse anomaly types and manifestations. Time series are categorized as univariate or multivariate, and anomalies as point, contextual, or collective (Paparrizos et al., 2025). Decades of research have produced diverse TSAD methods (Schmidl et al., 2022; Boniol et al., 2024), broadly grouped by their underlying principles. *Distance-based methods* measure subsequence deviations from reference patterns or nearest neighbors (Breunig et al., 2000; Boniol et al., 2021a). *Density-based methods* identify anomalies in low-density regions using distributional sparsity rather than pairwise distances alone (Liu et al., 2008; Aggarwal & Aggarwal, 2017). *Prediction-based methods* model normal temporal dynamics and detect anomalies through reconstruction or forecasting errors (Sakurada & Yairi, 2014; Munir et al., 2018). Despite their success, these methods rely on assumptions about anomaly characteristics that may not generalize across heterogeneous real-world time series, motivating adaptive, model-agnostic approaches.

2.2 AUTOML IN ANOMALY DETECTION

Heterogeneous anomaly patterns and time-series characteristics limit the effectiveness of any single detector across domains. Benchmark studies confirm the absence of a universal TSAD solution (Liu & Paparrizos, 2024; Qiu et al., 2025), motivating growing interest in automated approaches. AutoML for TSAD comprises three paradigms (Liu et al., 2025b;a): *model selection* chooses the best detector from a candidate pool (Sylligardos et al., 2023); *ensembling* combines detectors to improve robustness (Aggarwal & Sathe, 2015; Rayana & Akoglu, 2016; Schmidl et al., 2024); and *model generation* constructs new detectors using information from one or more candidate models, beyond selecting or aggregating them (Cao et al., 2023; Ye et al., 2023). Our framework follows the model-generation paradigm and is closely related to variance-based boosting methods such as UADB (Ye et al., 2023), which exploit prediction discrepancies as refinement signals. Whereas prior approaches primarily rely on individual heuristics, PVBOOST introduces posterior calibration to give these signals a probabilistic interpretation, improving robustness and interpretability.

2.3 ANOMALY SCORE THRESHOLDING

Anomaly detection typically comprises *scoring*, which quantifies abnormality using detector-specific criteria (Section 2.1), and *thresholding*, which classifies instances as anomalous or normal according to whether their scores exceed a threshold. Threshold selection has received considerably less attention than scoring. Many studies select thresholds by maximizing F-score or using ground-truth anomaly ratios (Han et al., 2022; Liu & Paparrizos, 2024). These supervised strategies are incompatible with fully unsupervised deployment, where labels and the true anomaly (or contamination) ratio are typically unavailable. Consequently, practitioners often use the *three-sigma rule* (Barnett et al.), which sets a threshold three standard deviations above the mean but may poorly fit complex score distributions. POT (Siffer et al., 2017) instead models score tails using Extreme Value Theory, while γGMM (Perini et al., 2023) fits a Dirichlet-process GMM to scores from multiple detectors to infer a posterior over the contamination rate. Despite these advances, restrictive assumptions or reliance on supervision make thresholding a critical, underexplored part of unsupervised anomaly detection.

3 METHODOLOGY

In this section, we first formalize the problem and key challenges (Section 3.1), then present an overview of PVBOOST (Section 3.2). Next, we detail its three core components (Section 3.3 – 3.5).

3.1 PROBLEM FORMULATION AND CHALLENGES

Problem setup. Given a time series $\mathcal{X} = \{x_1, \ldots, x_n\}$, $x_i \in \mathbb{R}^d$, a base detector f_S produces scores $\mathbf{s} \in \mathbb{R}^n$, with higher values indicating greater anomaly likelihood. We address the fully unsupervised *last mile*: converting these scores into binary predictions $\mathbf{p} \in \{0, 1\}^n$ through two coupled tasks: (i) *refinement*, which learns a booster f_B atop f_S to improve score quality, and (ii) *thresholding*, which converts refined scores into decisions. We treat f_S as a black box, requiring only its scalar outputs without access to internals or gradients, ensuring compatibility with classical and deep detectors.

Challenges. Despite its practical importance, the last mile is dominated by ad-hoc heuristics (e.g., fixed quantiles, 3σ rules, manual tuning), and we identify three fundamental challenges that make principled solutions elusive. **(C1) Score-distribution mismatch:** score distributions vary drastically

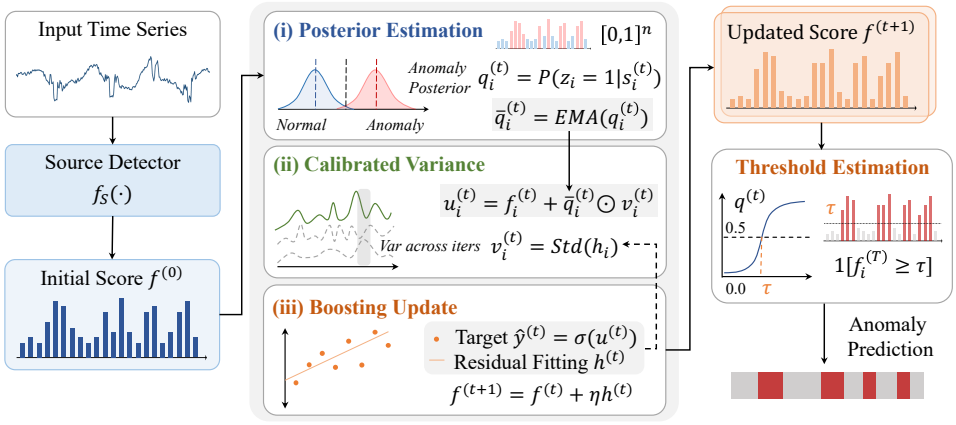

Figure 2: Overview of PVBOOST. The base detector is treated as a black box; its scores enter an iterative loop with three components: (i) posterior estimation, (ii) calibrated variance, and (iii) boosting update. The threshold is read off the same posterior trajectory that drove refinement.

across detectors in scale, shape, and tail behavior, so any thresholding rule or refinement signal defined directly on raw scores is neither comparable across detectors nor transferable across datasets. **(C2) Chicken-and-egg coupling:** refinement and thresholding are mutually dependent (i.e., refining scores ideally requires knowing which points are anomalous, while principled thresholding benefits from well-calibrated scores), yet prior work breaks this loop sequentially and propagates upstream errors downstream. **(C3) Double-edged unsupervised uncertainty:** prediction variance is a natural refinement signal in the absence of labels, since anomalous patterns are intrinsically harder to model and tend to exhibit higher variance under perturbation (Ye et al., 2023); but it is also amplified by noise in normal regions without supervision to distinguish the two. Existing methods address these challenges in isolation; to our knowledge, we are the first to tackle all three jointly.

3.2 Overview of PVBoost

As illustrated in Figure 2, PVBOOST is a Posterior-calibrated Variance Boosting framework that addresses C1–C3 through a *self-reinforcing posterior–variance feedback loop*. The loop combines two complementary unsupervised signals: the posterior $\mathbf{q}$ identifies *where* uncertainty is informative but provides no direction for score refinement, whereas prediction variance $\mathbf{v}$ across boosting iterations indicates *how much* scores should change but contains spurious fluctuations on normal points. Posterior-gated variance exploits this complementarity to refine scores, yielding sharper posteriors in subsequent iterations. PVBOOST implements this loop through three components: (i) *posterior estimation via mixture modeling* (Section 3.3) fits a two-component Gaussian mixture to normalized scores to obtain a scale-invariant posterior, addressing **C1**; (ii) *posterior-calibrated variance* (Section 3.4) gates raw variance with a smoothed posterior, preserving informative uncertainty while suppressing noise, addressing **C3**; (iii) *iterative boosting and thresholding* (Section 3.5) closes the loop by using the calibrated signal to guide a residual-fitting weak learner and deriving the threshold from the same posterior trajectory, addressing **C2** by coupling score refinement and threshold selection. Algorithm 1 summarizes the overall procedure.

3.3 Posterior Estimation via Mixture Modeling

The feedback loop first maps raw scores to a scale-invariant probabilistic representation. At iteration t, we standardize the current scores as $\mathbf{s}^{(t)} = (\mathbf{f}^{(t-1)} - \mu)/\sigma$, removing detector-specific location and scale, where $\mathbf{f}^{(0)}$ contains the min–max normalized base detector scores. We then fit a two-component Gaussian mixture model (GMM), $p(s_i^{(t)}) = \sum_{k\in\{0,1\}} \pi_k \mathcal{N}(s_i^{(t)}; \mu_k, \sigma_k^2)$, and designate the higher-mean component as anomalous, $k^* = \arg\max_k \mu_k$. The anomaly posterior at each time step i is $q_i^{(t)} = P(z_i = 1 \mid s_i^{(t)}; \boldsymbol{\theta}^{(t)})$, where $z_i \in \{0, 1\}$ indicates membership in the anomalous component and $\boldsymbol{\theta}^{(t)} = \{\pi_k, \mu_k, \sigma_k^2\}_{k\in\{0,1\}}$ denotes the GMM parameters fitted at iteration t.

By combining z-score normalization with a two-component mixture, the resulting posterior $\mathbf{q}^{(t)} \in [0, 1]^n$ depends only on the *relative* structure of the score distribution rather than its absolute scale, making it comparable across heterogeneous detectors and datasets. The two-component design

is principled for three reasons: (i) the components represent soft normal and anomalous classes, matching the binary task and identifying anomalies by the higher mean; (ii) avoiding fragmentation preserves the direct mapping from mixture mass to anomaly probability required for gating and thresholding; and (iii) restricting mixture complexity regularizes against overfitting to real-world TSAD score characteristics, such as heavy tails and quasi-periodic secondary modes. Consistently, a Dirichlet-process mixture with adaptive component counts does not improve score quality in our ablation (Section 4.2.2). We reduce fitting costs by warm-starting EM from the previous iteration. The resulting posterior $\mathbf{q}^{(t)}$ serves two roles in the rest of the framework: it gates the variance signal in Section 3.4, and its trajectory across iterations directly informs the final threshold in Section 3.5.

3.4 Posterior-Calibrated Variance

The second component of our feedback loop converts the posterior $\mathbf{q}^{(t)}$ into a refinement signal for the underlying scores. We build this signal from prediction variance: anomalous patterns are intrinsically harder to model and tend to exhibit higher variance under successive refinement, while normal patterns stabilize quickly (Tifrea et al., 2022; Ye et al., 2023). Concretely, at iteration t, we measure the per-step variance across the trajectory of predictions accumulated so far via $v_i^{(t)} = \mathrm{Std}\Big(\{h_i^{(\tau)}\}_{\tau=0}^{t-1}\Big)$, where $h_i^{(0)} := f_i^{(0)}$ is the (normalized) base detector score, and $h_i^{(\tau)}$ for $\tau \geq 1$ is the weak-learner output at iteration τ (formally introduced in Section 3.5). For $t = 1$ the trajectory contains a single element and we set $v_i^{(1)} = 0$, deferring the variance signal to subsequent iterations.

Variance alone, however, is a double-edged signal (**C3**): it surfaces genuine model disagreement on anomalous points, but is equally inflated by spurious fluctuations and weak-learner noise on normal points. To use it selectively, we gate the variance by the anomaly posterior. To avoid letting a single noisy posterior estimate dominate the gating, we use an exponential moving average across iterations:

$$\bar{q}_i^{(t)} = \beta\, \bar{q}_i^{(t-1)} + (1-\beta)\, q_i^{(t)}, \qquad \bar{q}_i^{(1)} = q_i^{(1)}, \tag{1}$$

where $\beta \in [0, 1)$ controls the smoothing strength. The smoothed posterior $\bar{q}^{(t)}$ directly gates the variance, yielding the calibrated refinement signal $\bar{q}_i^{(t)} v_i^{(t)}$. This probabilistically grounded mechanism suppresses variance as $\bar{q}_i^{(t)} \to 0$ (confidently normal), preventing noise amplification, and fully preserves it as $\bar{q}_i^{(t)} \to 1$ (confidently anomalous). EMA smoothing stabilizes the gate across iterations: sustained confidence in normality drives weights toward zero, while ambiguous points retain positive weights and thus nonzero refinement signals when their variance is positive. The resulting signal emphasizes informative uncertainty while suppressing noise.

3.5 Iterative Boosting and Thresholding

The third component of our framework closes the feedback loop: it uses the posterior-calibrated variance to drive score refinement, and derives the decision threshold from the same posterior trajectory. At iteration t, we construct a pseudo-target by injecting the calibrated refinement signal into the current score:

$$u_i^{(t)} = f_i^{(t-1)} + \bar{q}_i^{(t)} \odot v_i^{(t)}. \tag{2}$$

We standardize $\mathbf{u}^{(t)}$ and pass it through a sigmoid to obtain a bounded pseudo-target $\hat{y}_i^{(t)} = \sigma\big((u_i^{(t)} - \mu_u)/\sigma_u\big) \in [0, 1]$, where μ_u and σ_u are the mean and standard deviation of $\mathbf{u}^{(t)}$, and define the residual as $r_i^{(t)} = \hat{y}_i^{(t)} - f_i^{(t-1)}$. A weak learner $h^{(t)}$, a small MLP on sliding windows of the raw series (Appendix A.2), is then trained to fit $\mathbf{r}^{(t)}$, and the score is updated additively via $\mathbf{f}^{(t)} = \mathbf{f}^{(t-1)} + \eta\, \mathbf{h}^{(t)}$, where $\eta \in (0, 1]$ is the learning rate.

Residual fitting distinguishes PVBoost from booster-style frameworks such as UADB (Ye et al., 2023), which train each weak learner to regress the teacher score directly. By learning the incremental correction $\hat{\mathbf{y}}^{(t)} - \mathbf{f}^{(t-1)}$, PVBoost avoids repeatedly reconstructing the entire score surface and concentrates learner capacity on discrepancies between the current scores and the posterior–variance pseudo-target. Points already aligned with the pseudo-target have near-zero residual targets, whereas those with larger discrepancies receive stronger corrections. This gradient-style update uses η to control the magnitude of each refinement step.

The two normalization steps support this process in complementary ways: z-score normalization before GMM fitting removes detector-specific location and scale while preserving distributional

shape, whereas standardization followed by a sigmoid gives the pseudo-target a consistent, bounded range across iterations. Together with posterior estimation (Section 3.3), the score update forms a self-reinforcing feedback loop: refined scores can sharpen the posteriors, enabling more selective variance gating and, in turn, more targeted refinement.

Stabilizing the loop. A naive instantiation of this loop is fragile under unsupervised conditions. We incorporate three mechanisms to ensure reliable convergence. *(i) Posterior momentum.* The EMA-smoothed posterior $\bar{\mathbf{q}}^{(t)}$ introduced in Section 3.4 prevents a single noisy GMM fit, particularly in early iterations when scores are still poorly calibrated, from destabilizing the gating signal. *(ii) Variance-reduced weak learners.* Each $h^{(t)}$ is obtained as the average prediction of a K-fold ensemble trained on $\mathbf{r}^{(t)}$, which reduces the variance of the weak-learner output and prevents the boosting trajectory from amplifying overfitting noise into the variance signal $v_i^{(t)}$. *(iii) Posterior-drift early stopping.* We monitor $\Delta^{(t)} = \frac{1}{n}\sum_i |q_i^{(t)} - q_i^{(t-1)}|$ and terminate when $\Delta^{(t)}$ falls below a tolerance for several consecutive iterations, indicating that the loop has reached a fixed point.

Thresholding from the posterior trajectory. A central feature of PVBOOST is that thresholding is not a separate post-hoc step but a direct readout of the same posterior that drove refinement, addressing **C2** by construction. After the boosting loop terminates at iteration T, we obtain a final posterior by blending two complementary estimates: the EMA posterior $\bar{\mathbf{q}}^{(T)}$ accumulated along the boosting trajectory, and a snapshot posterior $\mathbf{q}^{\text{snap}}$ obtained by re-fitting a GMM on the final scores $\mathbf{f}^{(T)}$, giving $q_i^{\text{final}} = \beta\, \bar{q}_i^{(T)} + (1-\beta)\, q_i^{\text{snap}}$. The trajectory term integrates evidence across all iterations and is robust to local fluctuations, while the snapshot term reflects the sharpest separation achievable on the final refined scores; blending both avoids over-relying on either.

The posterior mass $\hat{\pi}_{\text{mass}} = \text{clip}\big(\frac{1}{n}\sum_i q_i^{\text{final}},\, 0.001,\, 0.15\big)$ is a natural estimate of the anomaly proportion, but on its own it has no notion of *rarity*: at EM convergence it equals the mixing weight of the anomaly component, and a two-component fit to a unimodal score histogram simply bisects the bulk (on pure Gaussian noise it returns the 0.15 cap). We therefore cap it with a temporal-coherence test on the refined score. Let $C(r)$ be the number of contiguous runs of $\{i : f_i^{(T)} \geq \text{Quantile}(\mathbf{f}^{(T)}, 1-r)\}$ in time order, and $E_0(r) = k(n-k+1)/n$ with $k = \lceil rn \rceil$ the expected number of runs when the same k points are placed at random. Then

$$\hat{\pi} = \min\big(\hat{\pi}_{\text{mass}},\, r_{\text{frag}}\big), \qquad r_{\text{frag}} = \arg\min_{r \in \mathcal{R}}\, C(r)/E_0(r), \tag{3}$$

where $\mathcal{R}$ is a log-spaced grid of 36 rates in $[0.002, 0.4]$. The posterior proposes a mass; the test caps it at the rate beyond which newly flagged points stop extending segments and start landing as isolated noise. On i.i.d. noise $C(r)/E_0(r) \approx 1$ at every rate: the test finds no segment structure and prefers no particular rate, so it is informative only when anomalies form contiguous segments. The decision threshold is $\tau = \text{Quantile}(\mathbf{f}^{(T)},\, 1-\hat{\pi})$, yielding binary predictions $p_i = \mathbf{1}[f_i^{(T)} \geq \tau]$.

4 EXPERIMENTS

In this section, we first describe the experimental setup in Section 4.1, followed by a comprehensive evaluation of the proposed method in Section 4.2.

4.1 EXPERIMENTAL SETUP

Datasets. To mitigate mislabeling, bias, and limited realism in TSAD benchmarks (Wu & Keogh, 2021; Liu & Paparrizos, 2024), we use TSB-AD (Liu & Paparrizos, 2024), a large-scale, curated benchmark covering diverse anomaly patterns and data characteristics. Its evaluation set contains 350 univariate (TSB-AD-U) and 180 multivariate (TSB-AD-M) series across nine domains: web services (Wu & Keogh, 2021; Ahmad et al., 2017), medical (Goldberger et al., 2000; Greenwald, 1990), facility (Filonov et al., 2016; Mathur & Tippenhauer, 2016), synthetic (Laptev et al., 2015; Lai et al., 2021), human activity (Bachlin et al., 2009; Roggen et al., 2010), sensor (Hundman et al., 2018; Ahmed et al., 2017), environment (Wu & Keogh, 2021; tao), finance (Tran et al., 2016; Sharafaldin et al., 2018), and traffic (Ahmad et al., 2017). Details appear in Appendix B.

TSAD Algorithms. To assess the model-agnostic applicability of PVBOOST, we evaluate representative detectors spanning major TSAD categories (Schmidl et al., 2022; Liu & Paparrizos, 2024; Paparrizos et al., 2025). Classical methods include proximity-based LOF (Breunig et al., 2000) and

Table 1: VUS-PR and point-wise F-score on TSB-AD-U (350 series) and TSB-AD-M (180 series). PVBOOST reports three-seed means (standard deviations: Table 12). Averages cover 15 detectors; 'Imp.' denotes relative change over Src (VUS-PR) or 3σ (F-score). Additional thresholding comparisons appear in Table 9; domain-level results in Tables 6 and 7.

	Method	TSB-AD-U								TSB-AD-M							
		VUS-PR			F					VUS-PR			F				
		Src	UADB	**PVBOOST**	3σ	POT	Top-15%	γGMM	**PVBOOST**	Src	UADB	**PVBOOST**	3σ	POT	Top-15%	γGMM	**PVBOOST**
Stats	LOF	0.268	0.211	0.288	0.113	0.096	0.111	0.094	0.175	0.146	0.230	0.264	0.052	0.057	0.092	0.087	0.208
	KNN	0.362	0.266	0.357	0.138	0.147	0.163	0.101	0.233	0.190	0.273	0.278	0.069	0.091	0.121	0.121	0.215
	KMeansAD	0.374	0.287	0.373	0.119	0.155	0.161	0.116	0.210	0.303	0.317	0.339	0.131	0.135	0.167	0.145	0.247
	OCSVM	0.235	0.224	0.238	0.080	0.088	0.113	0.068	0.156	0.270	0.291	0.268	0.098	0.093	0.173	0.090	0.190
	HBOS	0.233	0.233	0.229	0.095	0.095	0.131	0.100	0.157	0.195	0.193	0.204	0.085	0.078	0.139	0.076	0.137
	IForest	0.227	0.250	0.223	0.077	0.079	0.139	0.074	0.157	0.206	0.221	0.217	0.099	0.074	0.181	0.104	0.166
	PCA	0.426	0.326	0.422	0.235	0.171	0.234	0.215	0.311	0.312	0.303	0.310	0.179	0.091	0.249	0.222	0.291
NN	LSTMAD	0.329	0.262	0.328	0.169	0.139	0.179	0.179	0.223	0.309	0.290	0.291	0.153	0.116	0.224	0.216	0.236
	DeepAnT	0.351	0.284	0.350	0.183	0.154	0.177	0.185	0.238	0.317	0.271	0.289	0.202	0.111	0.222	0.222	0.252
	PatchTST	0.286	0.281	0.305	0.090	0.071	0.145	0.166	0.191	0.252	0.269	0.293	0.046	0.037	0.182	0.168	0.210
	AutoEncoder	0.273	0.236	0.267	0.100	0.100	0.138	0.108	0.163	0.298	0.309	0.304	0.134	0.073	0.220	0.177	0.271
	OmniAnomaly	0.296	0.268	0.297	0.147	0.107	0.174	0.153	0.214	0.323	0.276	0.318	0.127	0.110	0.199	0.157	0.221
	TimesNet	0.283	0.282	0.302	0.093	0.068	0.122	0.142	0.174	0.191	0.231	0.257	0.046	0.035	0.145	0.119	0.200
	AnomalyTrans.	0.126	0.275	0.262	0.024	0.025	0.083	0.041	0.181	0.123	0.235	0.227	0.020	0.032	0.093	0.043	0.169
	TranAD	0.264	0.255	0.278	0.101	0.084	0.147	0.128	0.186	0.295	0.280	0.295	0.119	0.093	0.189	0.166	0.220
Avg		0.289	0.263	**0.301**	0.118	0.105	0.148	0.125	**0.198**	0.249	0.266	**0.277**	0.104	0.082	0.173	0.141	**0.215**
Imp.			-9.0%	**+4.3%**		-10.5%	+25.7%	+6.0%	**+68.3%**		+6.9%	**+11.4%**		-21.3%	+66.5%	+35.5%	**+107.3%**

KNN (Ramaswamy et al., 2000), clustering-based KMeansAD (Yairi et al., 2001), distribution-based OCSVM (Schölkopf et al., 1999) and HBOS (Goldstein & Dengel, 2012), tree-based IForest (Liu et al., 2008), and encoding-based PCA (Aggarwal, 2017). Neural methods include forecasting-based DeepAnT (Munir et al., 2018), LSTMAD (Malhotra et al., 2015), and PatchTST (Nie et al., 2022), alongside reconstruction-based AutoEncoder (Sakurada & Yairi, 2014), OmniAnomaly (Su et al., 2019), TranAD (Tuli et al., 2022), TimesNet (Wu et al., 2022), and AnomalyTransformer (Xu et al., 2021). Together, these models span diverse statistical approaches and neural architectures, including CNNs, LSTMs, and Transformers. Further details appear in Appendix C. We select representative rather than exhaustive TSAD detectors spanning diverse assumptions and architectures to systematically assess whether boosting and thresholding consistently improve performance.

Baselines. For score refinement, we compare against original scores (Src) and UADB (Ye et al., 2023), a variance-based booster. Thresholding baselines include the widely used 3σ (Barnett et al.) rule, which sets the threshold three standard deviations above the mean; POT (Siffer et al., 2017), which models extreme score tails; fixed quantiles flagging the top 1%, 5%, 10%, or 15% of scores, with 15% matching PVBOOST's anomaly-rate cap; and γGMM (Perini et al., 2023), a Bayesian estimator of the contamination factor that thresholds at its posterior-mean rate.

Evaluation Measures. Following recent TSAD evaluation practices, we adopt VUS-PR (Paparrizos et al., 2022; Boniol et al., 2025) as the primary metric for assessing scoring quality. VUS-PR is a threshold-independent measure that accounts for temporal misalignment via buffer regions around anomaly boundaries, enabling a more robust and less biased evaluation than AUC-PR (Davis & Goadrich, 2006) in time-series settings. For binary detection performance, we report the F-score, defined as the harmonic mean of precision and recall, to capture the trade-off between detection accuracy and false positives under a given threshold.

4.2 EXPERIMENTAL RESULTS

We organize the evaluation around four research questions: **(RQ1)** Does PVBOOST improve detection performance across diverse base detectors? **(RQ2)** Does the posterior-derived threshold yield more reliable decisions than common heuristic thresholds? **(RQ3)** Is each component of the posterior-variance feedback loop necessary? **(RQ4)** How sensitive is PVBOOST to its main hyperparameters? Section 4.2.1 addresses RQ1 and RQ2 across 15 detectors in the univariate and multivariate settings of TSB-AD. Section 4.2.2 examines RQ3 through five component ablations and RQ4 by varying the learning rate η and EMA momentum β. Section 4.2.3 provides qualitative examples, while Appendix D reports additional results, including runtime analysis.

4.2.1 PERFORMANCE OVERVIEW

Table 1 reports VUS-PR for score refinement and F-score for thresholding on TSB-AD-U and TSB-AD-M, across 15 base detectors spanning statistical and neural categories.

RQ1: PVBOOST improves average score quality across both benchmarks. On TSB-AD-U, PVBOOST increases mean VUS-PR from 0.289 (Src) to 0.301 ($+4.3\%$), whereas UADB declines

Table 2: Ablation on TSB-AD-M. F uses the threshold of Eq. (3) in every row except Source, which is thresholded with 3σ as in Table 1.

	LOF		KMeansAD		PatchTST		TimesNet		TranAD		**Average**	
	VUS-PR	F	VUS-PR	F	VUS-PR	F	VUS-PR	F	VUS-PR	F	VUS-PR	F
Source	0.146	0.052	0.303	0.131	0.252	0.046	0.191	0.046	0.295	0.119	0.237	0.079
NoBoost	0.146	0.068	0.303	0.213	0.252	0.181	0.191	0.103	0.295	0.219	0.237	0.157
NoPG	0.214	0.159	0.341	0.250	0.286	0.219	0.245	0.201	0.290	0.215	0.275	0.209
DPGMM	0.211	0.040	0.328	0.162	0.297	0.029	0.232	0.025	0.298	0.068	0.273	0.065
NoVar	0.255	0.200	0.299	0.204	0.288	0.203	0.251	0.204	0.286	0.210	0.276	0.204
FitTeacher	0.203	0.155	0.226	0.162	0.271	0.210	0.269	0.229	0.246	0.181	0.243	0.187
PVBOOST	0.264	0.208	0.339	0.247	0.293	0.210	0.257	0.200	0.295	0.220	**0.290**	**0.217**

by 9.0% and underperforms Src on 12 of 15 detectors. On TSB-AD-M, PVBOOST improves the mean from 0.249 to 0.277 ($+11.4\%$), exceeding UADB's $+6.9\%$. Overall, PVBOOST outperforms UADB on 12 (U) and 11 (M) detectors. Gains concentrate on weak or noisy source scores, including AnomalyTransformer ($+0.136$, U), LOF ($+0.118$, M), PatchTST, and TimesNet. These improvements span statistical and neural detectors, supporting the model-agnostic design (Section 3). Strong source scores generally remain stable: the eight regressions on TSB-AD-U are at most 0.006, while the largest on TSB-AD-M are DeepAnT (-0.028) and LSTMAD (-0.018).

Paired per-series analysis (Appendix D.7) confirms detector-dependent refinement, with zero median AUC-PR change and the greatest benefits when anomalies form a few contiguous segments. Domain-level results (Appendix D.2) likewise concentrate VUS-PR gains in *Facility*, *HumanActivity*, and *Finance*, whereas the thresholding gains reported below extend to every domain.

RQ2: Posterior-derived thresholds substantially outperform heuristic baselines. Thresholding yields larger gains: on TSB-AD-U, PVBOOST achieves an F-score of 0.198, improving over 3σ (0.118) by 68.3%, whereas POT declines by 10.5%. On TSB-AD-M, PVBOOST reaches 0.215, approximately $2.1\times$ and $2.6\times$ the scores of 3σ (0.104) and POT (0.082). Fixed top-15%, corresponding to PVBOOST's rate when $\hat{\pi}$ reaches its cap, is stronger ($+25.7\%$ U; $+66.5\%$ M over 3σ), yet trails PVBOOST by 0.050 (U) and 0.042 (M), showing that the cap alone does not explain the gains.

PVBOOST outperforms 3σ and POT for all 15 detectors in both suites, including HBOS and IForest, whose source scores are mediocre. It also exceeds top-15% for all 15 detectors on TSB-AD-U and 13 on TSB-AD-M. Fixed-rate comparisons at 1%, 5%, and 10% on source and refined scores (Appendix D.4) confirm the gain: even the strongest baseline, top-10% on refined scores, trails PVBOOST by 0.030 (U) and 0.014 (M). PVBOOST further outperforms γGMM for all 15 detectors in both suites, whether γGMM uses individual source scores (0.125 U / 0.141 M; Table 1), all 15 detectors' scores jointly (0.160 / 0.156), or PVBOOST-refined scores (0.120 / 0.142; Appendix D.4, Table 9). This supports the **C1**-motivated design (Section 3.1): adapting to score distributions without the Gaussian-tail or extreme-value assumptions of 3σ and POT. PVBOOST slightly exceeds an oracle that uses each series' ground-truth anomaly ratio to threshold source scores on TSB-AD-M (0.215 vs. 0.213), but trails it on TSB-AD-U (0.198 vs. 0.222; Appendix D.3). Gains persist under range-based F1, held-out seeds, and permutation controls (Appendices D.4 and D.5).

4.2.2 Ablation Study

Key Design Choices. We ablate five components of the feedback loop on five representative detectors. In Table 2, every variant is below the full PVBOOST on both average VUS-PR and average F, so each component contributes to the final performance.

NoBoost fits one GMM without boosting, retaining source VUS-PR (0.237). F increases from 0.079 (source, 3σ) to 0.157, so thresholding alone (Eq. 3) supplies over half the gain to 0.217; boosting supplies the remainder and all scoring gains. *NoPG* replaces $\bar{\mathbf{q}} \odot \mathbf{v}$ with $\mathbf{v}$, reducing VUS-PR/F from $0.290/0.217$ to $0.275/0.209$ and supporting gating to suppress noise in normal regions (**C3**). *DPGMM* uses a Dirichlet Process GMM, lowering mean VUS-PR to 0.273 despite marginal gains on PatchTST and TranAD; LOF drops from 0.264 to 0.211. F collapses to 0.065, below source 3σ (0.079). Variable components fragment the high-score mass: across LOF, PatchTST, and TimesNet, the highest-mean component contains a median of 0.09% of points and under 1% on 83% of series, pushing $\hat{\pi}_{\text{mass}}$ toward its floor and under-detecting anomalies. Thus, greater mixture flexibility offers no average scoring benefit, while two components preserve a well-defined anomaly mass (Section 3.3). *NoVar* uses posterior-only targets, reducing VUS-PR/F to $0.276/0.204$, comparable to NoPG. The

posterior supplies most refinement gains, while variance adds a complementary benefit. *FitTeacher* directly regresses $\hat{\mathbf{y}}^{(t)}$ using UADB-style updates, producing the largest scoring loss among boosted variants (VUS-PR/F: $0.243/0.187$). Its KMeansAD regression (0.226 vs. source 0.303) supports residual fitting to limit over-correction (Section 3.5).

Sensitivity Analysis. We analyze the sensitivity of PVBOOST to its two main hyperparameters: the learning rate η and the EMA momentum β (default $\eta = 0.1$, $\beta = 0.7$). We sweep $\eta \in \{0.1, 0.3, 0.5, 0.7\}$ with β fixed, and $\beta \in \{0.1, 0.3, 0.5, 0.7\}$ with η fixed, on the same five representative detectors as in Table 2. Figure 3 reports ΔVUS-PR relative to the default on the 180 TSB-AD-M series. Larger learning rates mostly hurt: at $\eta = 0.7$ all five detectors lose VUS-PR, from 0.004 (PatchTST) to 0.040 (KMeansAD), consistent with the gradient-style framing in Section 3.5 where large steps overshoot the calibrated pseudo-target. The decline is monotone for KMeansAD, LOF and TranAD, whereas TimesNet gains slightly at $\eta \in \{0.3, 0.5\}$ ($+0.006$) before dropping at $\eta = 0.7$. PVBOOST is more robust to β: the worst case over all β configurations is -0.007 VUS-PR (PatchTST, $\beta = 0.1$), about six times smaller than the largest η drop. For $\beta \in [0.5, 0.7]$ every detector stays within 0.003 of the default, so posterior momentum stabilizes the loop without being a brittle knob. Together, these results show that PVBOOST's gains do not hinge on aggressive hyperparameter tuning: a small learning rate combined with moderate posterior smoothing is a robust choice. Beyond hyperparameter robustness, PVBOOST also runs faster than UADB on average and scales gracefully with time-series length (Appendix D.8).

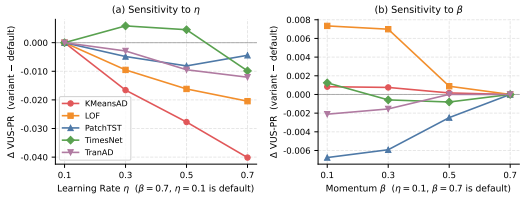

Figure 3: Sensitivity of PVBOOST to the (a) learning rate η and (b) EMA momentum β.

4.2.3 CASE STUDY

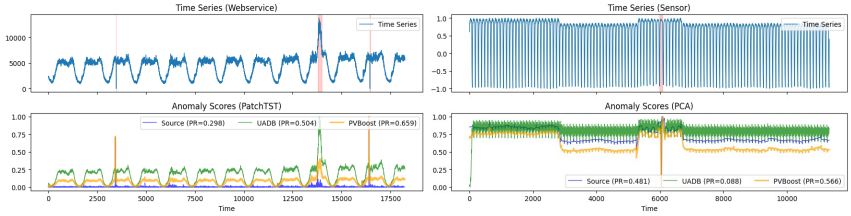

Figure 4: Case studies illustrating the impact of the proposed boosting strategy on anomaly scores. The red shaded regions in the time series plots indicate ground-truth anomalies.

Figure 4 illustrates two representative boosting behaviors. On Webservice, PatchTST's noisy source scores poorly separate normal and anomalous regions. UADB amplifies the middle anomaly but misses both flanking anomalies; PVBOOST sharpens all three and suppresses normal regions, increasing VUS-PR from 0.298 to 0.659, consistent with posterior-gated variance. On Sensor, PCA already performs well: UADB flattens its score profile, reducing VUS-PR to 0.088, whereas PVBOOST closely follows the source while further suppressing normal regions and sharpening the anomaly, reaching 0.566. This illustrates residual fitting (Section 3.5): agreement between source scores and the posterior–variance pseudo-target yields near-zero weak-learner corrections. Together, the two cases instantiate the asymmetric behavior that drives PVBOOST's stability advantage: it lifts where the source is weak and stays where the source is strong.

5 CONCLUSION

We revisited the *last mile* of unsupervised TSAD: jointly refining black-box detector scores and deriving reliable thresholds, addressing three challenges previously treated separately. Our model-agnostic framework, PVBOOST, couples both tasks through a self-reinforcing posterior–variance loop: mixture-model posteriors gate prediction variance into calibrated refinement signals, while the same posterior trajectory yields prior-free thresholds. Across nine TSB-AD domains and 15 base detectors, PVBOOST improves score quality across detector categories and substantially outperforms common thresholding methods, with ablations supporting each component's contribution. These findings motivate explicit modeling of the often-overlooked thresholding stage over ad hoc heuristics. Future work will extend PVBOOST to richer mixture families and streaming settings.

AI USE STATEMENT

We used generative AI tools solely to polish the manuscript's language and correct grammatical errors. We did not use generative AI tools for any other aspect of the research. We reviewed all AI-assisted edits to ensure accuracy and preserve the intended meaning. We take full responsibility for the final content of this work, including all text, claims, and artifacts.

REPRODUCIBILITY STATEMENT

Code and experiment scripts are available at `https://anonymous.4open.science/r/PVBoost`. PVBOOST is specified in Section 3 and Algorithm 1; implementation details are in Appendix A.2, and the default hyperparameters, shared across all detectors and time series, are in Table 3. All experiments use the public TSB-AD benchmark (Appendix B); the 15 detectors are described in Appendix C, and baselines and metrics in Section 4.1. Appendix D.4 shows that suite averages vary by at most 0.001 across three seeds. Hardware is described in Appendix D.1.

REFERENCES

`http://iops.ai/dataset_detail/?id=10`.

Tao. `https://www.pmel.noaa.gov/`. URL `https://www.pmel.noaa.gov/`.

Charu C. Aggarwal. *Outlier Analysis*. Springer International Publishing, 2 edition, 2017. ISBN 978-3-319-47577-6. doi: 10.1007/978-3-319-47578-3. URL `https://www.springer.com/gp/book/9783319475776`.

Charu C Aggarwal and Charu C Aggarwal. *An introduction to outlier analysis*. Springer, 2017.

Charu C Aggarwal and Saket Sathe. Theoretical foundations and algorithms for outlier ensembles. *Acm sigkdd explorations newsletter*, 17(1):24–47, 2015.

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

Chuadhry Mujeeb Ahmed, Venkata Reddy Palleti, and Aditya P Mathur. Wadi: a water distribution testbed for research in the design of secure cyber physical systems. In *Proceedings of the 3rd international workshop on cyber-physical systems for smart water networks*, pp. 25–28, 2017.

Marc Bachlin, Meir Plotnik, Daniel Roggen, Inbal Maidan, Jeffrey M Hausdorff, Nir Giladi, and Gerhard Troster. Wearable assistant for parkinson's disease patients with the freezing of gait symptom. *IEEE Transactions on Information Technology in Biomedicine*, 14(2):436–446, 2009.

Vic Barnett, Toby Lewis, et al. *Outliers in statistical data*, volume 3.

Ane Blázquez-García, Angel Conde, Usue Mori, and Jose A Lozano. A review on outlier/anomaly detection in time series data. *ACM Computing Surveys (CSUR)*, 54(3):1–33, 2021.

Paul Boniol, Michele Linardi, Federico Roncallo, Themis Palpanas, Mohammed Meftah, and Emmanuel Remy. Unsupervised and scalable subsequence anomaly detection in large data series. *The VLDB Journal*, pp. 1–23, 2021a.

Paul Boniol, John Paparrizos, Themis Palpanas, and Michael J Franklin. Sand: streaming subsequence anomaly detection. *Proceedings of the VLDB Endowment*, 14(10):1717–1729, 2021b.

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.

Paul Boniol, Ashwin K Krishna, Marine Bruel, Qinghua Liu, Mingyi Huang, Themis Palpanas, Ruey S Tsay, Aaron Elmore, Michael J Franklin, and John Paparrizos. Vus: effective and efficient accuracy measures for time-series anomaly detection. *The VLDB Journal*, 34(3):32, 2025.

Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104, 2000.

Lei Cao, Yizhou Yan, Yu Wang, Samuel Madden, and Elke A Rundensteiner. Autood: Automatic outlier detection. *Proceedings of the ACM on Management of Data*, 1(1):1–27, 2023.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pp. 233–240, 2006.

Pavel Filonov, Andrey Lavrentyev, and Artem Vorontsov. Multivariate industrial time series with cyber-attack simulation: Fault detection using an lstm-based predictive data model, 2016.

Philippe Fleith. Controlled anomalies time series (cats) dataset, September 2023. Accessed: 2023-09-01.

Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.

Markus Goldstein and Andreas Dengel. Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm. *KI-2012: poster and demo track*, 9, 2012.

Scott David Greenwald. *Improved detection and classification of arrhythmias in noise-corrupted electrocardiograms using contextual information*. Thesis, Massachusetts Institute of Technology, 1990. URL https://dspace.mit.edu/handle/1721.1/29206. Accepted: 2005-10-07T20:45:22Z.

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.

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.

Vincent Jacob, Fei Song, Arnaud Stiegler, Bijan Rad, Yanlei Diao, and Nesime Tatbul. Exathlon: A benchmark for explainable anomaly detection over time series. *Proceedings of the VLDB Endowment*, 14(11):2613–2626, 2021.

Kwei-Herng Lai, Daochen Zha, Junjie Xu, Yue Zhao, Guanchu Wang, and Xia Hu. Revisiting time series outlier detection: Definitions and benchmarks. In *NeurIPS Track on Datasets and Benchmarks*, 2021.

N. Laptev, S. Amizadeh, and Y. Billawala. S5 - A Labeled Anomaly Detection Dataset, version 1.0(16M), March 2015. URL https://webscope.sandbox.yahoo.com/catalog.php?datatype=s&did=70.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 eighth ieee international conference on data mining*, pp. 413–422. IEEE, 2008.

Qinghua Liu and John Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. *Advances in Neural Information Processing Systems*, 37:108231–108261, 2024.

Qinghua Liu, Seunghak Lee, and John Paparrizos. Easyad: A demonstration of automated solutions for time-series anomaly detection. *Proceedings of the VLDB Endowment*, 18(12):5431–5434, 2025a.

Qinghua Liu, Seunghak Lee, and John Paparrizos. Tsb-autoad: Towards automated solutions for time-series anomaly detection. *PVLDB*, 18(11):4364–4379, 2025b.

Pankaj Malhotra, L. Vig, Gautam M. Shroff, and Puneet Agarwal. Long Short Term Memory Networks for Anomaly Detection in Time Series. In *ESANN*, 2015.

Aditya P Mathur and Nils Ole Tippenhauer. Swat: A water treatment testbed for research and training on ics security. In *2016 international workshop on cyber-physical systems for smart water networks (CySWater)*, pp. 31–36. IEEE, 2016.

Steffen Moritz, Frederik Rehbach, Sowmya Chandrasekaran, Margarita Rebolledo, and Thomas Bartz-Beielstein. Gecco industrial challenge 2018 dataset: A water quality dataset for the 'internet of things: Online anomaly detection for drinking water quality'competition at the genetic and evolutionary computation conference 2018, kyoto, japan. *Kyoto, Japan*, 2018.

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, 2018.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The Eleventh International Conference on Learning Representations*, 2022.

John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S Tsay, Aaron Elmore, and Michael J Franklin. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. *Proceedings of the VLDB Endowment*, 15(11):2774–2787, 2022.

John Paparrizos, Paul Boniol, Qinghua Liu, and Themis Palpanas. Advances in time-series anomaly detection: Algorithms, benchmarks, and evaluation measures. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 6151–6161, 2025.

Lorenzo Perini, Paul-Christian Bürkner, and Arto Klami. Estimating the contamination factor's distribution in unsupervised anomaly detection. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, 2023.

Xiangfei Qiu, Zhe Li, Wanghui Qiu, Shiyan Hu, Lekui Zhou, Xingjian Wu, Zhengyu Li, Chenjuan Guo, Aoying Zhou, Zhenli Sheng, et al. Tab: Unified benchmarking of time series anomaly detection methods. *arXiv preprint arXiv:2506.18046*, 2025.

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.

Shebuti Rayana and Leman Akoglu. Less is more: Building selective anomaly ensembles. *Acm transactions on knowledge discovery from data (tkdd)*, 10(4):1–33, 2016.

Daniel Roggen, Alberto Calatroni, Mirco Rossi, Thomas Holleczek, Kilian Förster, Gerhard Tröster, Paul Lukowicz, David Bannach, Gerald Pirkl, Alois Ferscha, et al. Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh international conference on networked sensing systems (INSS)*, pp. 233–240. IEEE, 2010.

Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, pp. 4–11, 2014.

Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. Anomaly detection in time series: a comprehensive evaluation. *Proceedings of the VLDB Endowment*, 15(9):1779–1797, 2022.

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.

Bernhard Schölkopf, Robert Williamson, Alex Smola, John Shawe-Taylor, and John Platt. Support vector method for novelty detection. In *Proceedings of the 12th International Conference on Neural Information Processing Systems*, NIPS'99, pp. 582–588, Cambridge, MA, USA, November 1999. MIT Press.

Iman Sharafaldin, Arash Habibi Lashkari, Ali A Ghorbani, et al. Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, 1:108–116, 2018.

Haotian Si, Changhua Pei, Hang Cui, Jingwen Yang, Yongqian Sun, Shenglin Zhang, Jingjing Li, Haiming Zhang, Jing Han, Dan Pei, et al. Timeseriesbench: An industrial-grade benchmark for time series anomaly detection models. *arXiv preprint arXiv:2402.10802*, 2024.

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.

Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '19, pp. 2828–2837, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450362016. doi: 10.1145/3292500.3330672. URL https://doi.org/10.1145/3292500.3330672.

Emmanouil Sylligardos, Paul Boniol, John Paparrizos, Panos Trahanias, and Themis Palpanas. Choose wisely: An extensive evaluation of model selection for anomaly detection in time series. *Proceedings of the VLDB Endowment*, 16(11):3418–3432, 2023.

Markus Thill, Wolfgang Konen, and Thomas Bäck. MarkusThill/MGAB: The Mackey-Glass Anomaly Benchmark, https://doi.org/10.5281/zenodo.3762385, April 2020.

Alexandru Tifrea, Eric Stavarache, and Fanny Yang. Semi-supervised novelty detection using ensembles with regularized disagreement. In *Uncertainty in Artificial Intelligence*, pp. 1939–1948. PMLR, 2022.

Luan Tran, Liyue Fan, and Cyrus Shahabi. Distance-based outlier detection in data streams. *Proceedings of the VLDB Endowment*, 9(12):1089–1100, 2016.

Shreshth Tuli, Giuliano Casale, and Nicholas R Jennings. Tranad: deep transformer networks for anomaly detection in multivariate time series data. *Proceedings of the VLDB Endowment*, 15(6): 1201–1214, 2022.

Alexander von Birgelen and Oliver Niggemann. Anomaly detection and localization for cyber-physical production systems with self-organizing maps. *IMPROVE-Innovative Modelling Approaches for Production Systems to Raise Validatable Efficiency: Intelligent Methods for the Factory of the Future*, pp. 55–71, 2018.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *The Eleventh International Conference on Learning Representations*, 2022.

Renjie Wu and Eamonn J Keogh. Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress. *IEEE transactions on knowledge and data engineering*, 35(3): 2421–2429, 2021.

Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly transformer: Time series anomaly detection with association discrepancy. In *International Conference on Learning Representations*, 2021.

Takehisa Yairi, Yoshikiyo Kato, and Koichi Hori. Fault detection by mining association rules from house-keeping data. In *proceedings of the 6th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, volume 18, pp. 21. Citeseer, 2001.

Hangting Ye, Zhining Liu, Xinyi Shen, Wei Cao, Shun Zheng, Xiaofan Gui, Huishuai Zhang, Yi Chang, and Jiang Bian. Uadb: Unsupervised anomaly detection booster. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*, pp. 2593–2606. IEEE, 2023.

Zahra Zamanzadeh Darban, Geoffrey I Webb, Shirui Pan, Charu Aggarwal, and Mahsa Salehi. Deep learning for time series anomaly detection: A survey. *ACM Computing Surveys*, 57(1):1–42, 2024.

Shenglin Zhang, Zhenyu Zhong, Dongwen Li, Qiliang Fan, Yongqian Sun, Man Zhu, Yuzhi Zhang, Dan Pei, Jiyan Sun, Yinlong Liu, et al. Efficient kpi anomaly detection through transfer learning for large-scale web services. *IEEE Journal on Selected Areas in Communications*, 40(8):2440–2455, 2022.

SUPPLEMENTARY MATERIAL FOR PVBOOST

A MORE DETAILS ON PVBOOST

A.1 PSEUDO ALGORITHM

Algorithm 1 PVBOOST: Posterior-Calibrated Variance Boosting

Require: Base detector scores $\mathbf{f}^{(0)} \in \mathbb{R}^n$; max iterations T; EMA momentum β; learning rate η; patience P; drift tolerance ϵ.
Ensure: Refined scores $\mathbf{f}^{(T)}$, threshold τ, predictions $\mathbf{p}$.

$\mathcal{H} \leftarrow [\mathbf{f}^{(0)}]$; $c \leftarrow 0$
for $t = 1, 2, \ldots, T$ **do**
  $\mathbf{v}^{(t)} \leftarrow \mathrm{Std}(\mathcal{H})$ ▷ Per-step variance across the trajectory
  $\mathbf{s}^{(t)} \leftarrow \mathrm{ZScore}(\mathbf{f}^{(t-1)})$; fit 2-component GMM on $\mathbf{s}^{(t)}$
  $\mathbf{q}^{(t)} \leftarrow P(z = 1 \mid \mathbf{s}^{(t)}; \boldsymbol{\theta}^{(t)})$ ▷ Anomaly posterior, Eq. 1
  $\bar{\mathbf{q}} \leftarrow \mathbf{q}^{(1)}$ **if** $t = 1$ **else** $\beta\,\bar{\mathbf{q}} + (1-\beta)\,\mathbf{q}^{(t)}$ ▷ EMA-smoothed posterior
  if $t > 1$ **and** $\frac{1}{n}\|\mathbf{q}^{(t)} - \mathbf{q}_{\text{prev}}\|_1 < \epsilon$ **then**
    $c \leftarrow c + 1$; **if** $c \geq P$ **break**
  else $c \leftarrow 0$
  end if
  $\mathbf{q}_{\text{prev}} \leftarrow \mathbf{q}^{(t)}$
  $\hat{\mathbf{y}}^{(t)} \leftarrow \sigma\big(\mathrm{ZScore}(\mathbf{f}^{(t-1)} + \bar{\mathbf{q}} \odot \mathbf{v}^{(t)})\big)$ ▷ Calibrated pseudo-target
  Train weak learner $h^{(t)}$ on residual $\hat{\mathbf{y}}^{(t)} - \mathbf{f}^{(t-1)}$
  $\mathbf{f}^{(t)} \leftarrow \mathbf{f}^{(t-1)} + \eta\,\mathbf{h}^{(t)}$; append $\mathbf{h}^{(t)}$ to $\mathcal{H}$
end for
Fit fresh GMM on $\mathbf{f}^{(T)}$ to obtain snapshot posterior $\mathbf{q}^{\text{snap}}$
$\mathbf{q}^{\text{final}} \leftarrow \beta\,\bar{\mathbf{q}} + (1-\beta)\,\mathbf{q}^{\text{snap}}$; $\hat{\pi}_{\text{mass}} \leftarrow \mathrm{clip}(\frac{1}{n}\sum_i q_i^{\text{final}}, 0.001, 0.15)$
$r_{\text{frag}} \leftarrow \arg\min_{r \in \mathcal{R}} C(r)/E_0(r)$ on $\mathbf{f}^{(T)}$; $\hat{\pi} \leftarrow \min(\hat{\pi}_{\text{mass}}, r_{\text{frag}})$ ▷ Eq. 3
$\tau \leftarrow \mathrm{Quantile}(\mathbf{f}^{(T)}, 1 - \hat{\pi})$; $\mathbf{p} \leftarrow \mathbf{1}[\mathbf{f}^{(T)} \geq \tau]$
return $\mathbf{f}^{(T)}, \tau, \mathbf{p}$

A.2 IMPLEMENTATION DETAIL

Weak-learner inputs. PVBOOST needs no access to the base detector's internals, but its weak learners do see the raw series: each $h^{(t)}$ maps the input at time step i to a correction of f_i. Each channel is min-max scaled, and time step i is represented by the window $[x_{i-w+1}, \ldots, x_i]$ flattened into a vector of length wd, with zero padding for $i < w$, so there is one window per time step. The window length w is the dominant period of the series: the highest local maximum of the autocorrelation function over lags 3 to 400, computed on the first channel (and on at most the first 20,000 points), with $w = 125$ when no such maximum exists or it exceeds 300. The GMM and the threshold operate only on the one-dimensional score and do not use these windows.

Weak learner and training. Each weak learner is a three-layer MLP (hidden width 128, GELU activations) with a linear output, trained with the mean squared error on the residual $\mathbf{r}^{(t)}$ using Adam (learning rate 10^{-3}, batch size 256). The K-fold ensemble of Section 3.5 uses $K = 3$ contiguous, unshuffled folds of the series. For each fold, one MLP is trained on the other two folds for 10 epochs of 10,000 sampled points each, and the epoch with the lowest validation loss on the held-out fold is kept. $h^{(t)}$ is the average of the three MLPs' predictions on all time steps. A new ensemble is trained at every boosting iteration. The GMM of each iteration is fit with at most 30 EM steps, warm-started from the previous iteration; the snapshot GMM of the final posterior is fit from scratch with at most 100 steps.

UADB baseline. For fair comparison, we obtain UADB results by re-running the official implementation[1] with the same preprocessing pipeline as PVBOOST, including identical normalization and windowing strategies. This ensures that performance differences are attributed to the boosting mechanism rather than discrepancies in data processing.

Thresholding details. For thresholding baselines, we adopt the Peak-Over-Threshold (POT) method and tune its hyperparameters on the TSB-AD tuning set. The best-performing configuration is then fixed and applied across all evaluation datasets. This protocol follows standard practice to ensure a fair and robust comparison under unsupervised settings.

Reproducibility. More detailed implementation choices regarding PVBOOST and baselines, including hyperparameters (Table 3) and training procedures, are provided in the released codebase[2].

Table 3: Default hyperparameter settings for PVBOOST.

Symbol	Description	Default
T	Maximum boosting iterations	10
β	EMA momentum for posterior smoothing	0.7
η	Learning rate for additive update	0.1
P	Patience for posterior-drift early stopping	3
ϵ	Drift tolerance	10^{-3}
$[\pi_{\min}, \pi_{\max}]$	Clip bounds on the posterior mass $\hat{\pi}_{\text{mass}}$	$[0.001, 0.15]$
$\mathcal{R}$	Rate grid for r_{frag} (Eq. equation 3)	36 log-spaced in $[0.002, 0.4]$

B DATASET DESCRIPTION

We evaluate the proposed framework on the **TSB-AD benchmark** (Liu & Paparrizos, 2024), a large-scale benchmark for time-series anomaly detection.

Benchmark Overview. The evaluation set of TSB-AD contains two subsets: **TSB-AD-U** (350 univariate time series) and **TSB-AD-M** (180 multivariate time series), covering diverse real-world scenarios.

Dataset Domains. The benchmark aggregates datasets from multiple domains, summarized as follows:

- **Web service:** NAB (Ahmad et al., 2017), WSD (Zhang et al., 2022), IOPS (IOP), NEK (Si et al., 2024), Yahoo (Laptev et al., 2015)
- **Facility:** NAB (Ahmad et al., 2017), SMD (Su et al., 2019), UCR (Wu & Keogh, 2021), Exathlon (Jacob et al., 2021),
- **Sensor:** SMAP (Hundman et al., 2018), MSL (Hundman et al., 2018), UCR (Wu & Keogh, 2021), WADI (Ahmed et al., 2017), Genesis (von Birgelen & Niggemann, 2018), GHL (Filonov et al., 2016), GECCO (Moritz et al., 2018), SWaT (Mathur & Tippenhauer, 2016)
- **Medical:** MITDB (Goldberger et al., 2000), SVDB (Greenwald, 1990), LTDB (Goldberger et al., 2000), UCR (Wu & Keogh, 2021)
- **Human activity:** UCR (Wu & Keogh, 2021), Daphnet (Bachlin et al., 2009), OPPORTUNITY (Roggen et al., 2010)
- **Environment:** TAO (tao), UCR (Wu & Keogh, 2021)
- **Finance:** Stock (Tran et al., 2016), CreditCard (Sharafaldin et al., 2018)
- **Traffic:** NAB (Ahmad et al., 2017)
- **Synthetic and controlled datasets:** UCR (Wu & Keogh, 2021), TODS (Lai et al., 2021), CATSv2 (Fleith, 2023), MGAB (Thill et al., 2020), SED (Boniol et al., 2021b)

Evaluation Protocol. All datasets are evaluated in a fully unsupervised setting, where ground-truth labels are used only for evaluation and are not available during training or threshold selection.

[1] https://github.com/HangtingYe/UADB

[2] https://anonymous.4open.science/r/PVBoost

Table 4: Domain distribution of the TSB-AD benchmark.

Domain	# Time Series
Facility	133
Sensor	110
Medical	90
WebService	64
HumanActivity	51
Synthetic	39
Environment	29
Finance	9
Traffic	5
Total	530

Discussion. The diversity of domains in TSB-AD introduces substantial variability in temporal dynamics, noise characteristics, and anomaly patterns. This heterogeneity is essential for evaluating model-agnostic frameworks, particularly in the last-mile setting where anomaly score distributions differ significantly across datasets.

C TSAD Algorithm Description

We briefly describe the underlying TSAD algorithms used in our experiments. These methods are widely adopted in the literature and cover a diverse set of modeling paradigms, including distance-based, density-based, clustering-based, statistical, and deep learning approaches. As such, they serve as *representative* baselines for evaluating model-agnostic frameworks.

- **LOF** (Breunig et al., 2000) measures the local density deviation of a sample with respect to its neighbors, where points in sparse regions are considered anomalous.
- **KNN** (Ramaswamy et al., 2000) defines anomaly scores based on the distance to the k-th nearest neighbor, with larger distances indicating higher abnormality.
- **KMeansAD** (Yairi et al., 2001) assigns each instance to a cluster and computes anomaly scores as the distance to its cluster centroid.
- **OCSVM** (Schölkopf et al., 1999) learns a decision boundary that encloses normal data in a high-dimensional feature space, treating points outside the boundary as anomalies.
- **HBOS** (Goldstein & Dengel, 2012) constructs histograms for each feature independently and assigns anomaly scores based on the inverse density of each bin.
- **IForest** (Liu et al., 2008) isolates anomalies by randomly partitioning the data; anomalies tend to have shorter path lengths in the isolation trees.
- **PCA** (Aggarwal, 2017) projects data into a low-dimensional subspace and detects anomalies via reconstruction error or deviation from the principal components.
- **DeepAnT** (Munir et al., 2018) is a CNN-based forecasting model that detects anomalies by measuring prediction errors over future time steps.
- **LSTMAD** (Malhotra et al., 2015) employs LSTM networks to model temporal dependencies and flags anomalies based on deviations between predicted and observed values.
- **PatchTST** (Nie et al., 2022) is a transformer-based model that processes time series as patch sequences, capturing long-range dependencies for forecasting-based anomaly detection.
- **AutoEncoder** (Sakurada & Yairi, 2014) learns a compressed representation of normal patterns and identifies anomalies via large reconstruction errors.
- **OmniAnomaly** (Su et al., 2019) is a stochastic recurrent model that captures temporal dynamics with latent variables and detects anomalies via reconstruction probability.
- **TimesNet** (Wu et al., 2022) models time series by transforming them into 2D representations to capture multi-periodicity and complex temporal patterns.
- **AnomalyTransformer** (Xu et al., 2021) introduces an anomaly-attention mechanism to measure association discrepancy for anomaly detection.
- **TranAD** (Tuli et al., 2022) is a transformer-based architecture with self-conditioning and adversarial training to amplify anomaly signals.

D Additional Experimental Results

D.1 Hardware and Software.

All experiments are conducted on a server equipped with two AMD EPYC 7713 processors (64 cores each) and two NVIDIA A100 GPUs, running Ubuntu 22.04.3 LTS (64-bit).

Table 5: Performance comparison on TSB-AD-U and TSB-AD-M across AUC-PR, VUS-PR, and F-score (same protocol as Table 1; PVB = PVBoost, mean over three seeds; T10 = fixed top-10% threshold on the source score).

	Method	TSB-AD-U										TSB-AD-M									
		AUC-PR			VUS-PR			F				AUC-PR			VUS-PR			F			
		Src	UADB	PVB	Src	UADB	PVB	3σ	POT	T10	PVB	Src	UADB	PVB	Src	UADB	PVB	3σ	POT	T10	PVB
Stats	LOF	0.162	0.166	0.183	0.268	0.211	0.288	0.113	0.096	0.114	0.175	0.096	0.207	0.215	0.146	0.230	0.264	0.052	0.057	0.092	0.208
	KNN	0.273	0.213	0.269	0.362	0.266	0.357	0.138	0.147	0.171	0.233	0.138	0.231	0.226	0.190	0.273	0.278	0.069	0.091	0.127	0.215
	KMeansAD	0.321	0.237	0.316	0.374	0.287	0.373	0.119	0.155	0.172	0.210	0.247	0.275	0.287	0.303	0.317	0.339	0.131	0.135	0.183	0.247
	OCSVM	0.157	0.174	0.161	0.235	0.224	0.238	0.080	0.088	0.113	0.156	0.227	0.238	0.223	0.270	0.291	0.268	0.098	0.093	0.193	0.190
	HBOS	0.175	0.180	0.172	0.233	0.233	0.229	0.095	0.095	0.134	0.157	0.161	0.152	0.162	0.195	0.193	0.204	0.085	0.078	0.125	0.137
	IForest	0.165	0.198	0.161	0.227	0.250	0.223	0.077	0.079	0.136	0.157	0.189	0.190	0.189	0.206	0.221	0.217	0.099	0.074	0.170	0.166
	PCA	0.374	0.279	0.368	0.426	0.326	0.422	0.235	0.171	0.256	0.311	0.308	0.292	0.304	0.312	0.303	0.310	0.179	0.091	0.263	0.291
NN	LSTMAD	0.315	0.229	0.314	0.329	0.262	0.328	0.169	0.139	0.190	0.223	0.312	0.274	0.286	0.309	0.290	0.291	0.153	0.116	0.245	0.236
	DeepAnT	0.330	0.252	0.332	0.351	0.284	0.350	0.183	0.154	0.188	0.238	0.317	0.267	0.287	0.317	0.271	0.289	0.202	0.111	0.245	0.252
	PatchTST	0.189	0.227	0.231	0.286	0.281	0.305	0.090	0.071	0.152	0.191	0.175	0.221	0.231	0.252	0.269	0.293	0.046	0.037	0.182	0.210
	AutoEncoder	0.191	0.186	0.189	0.273	0.236	0.267	0.100	0.100	0.141	0.163	0.296	0.282	0.300	0.298	0.309	0.304	0.134	0.073	0.247	0.271
	OmniAnomaly	0.266	0.218	0.266	0.296	0.268	0.297	0.147	0.107	0.185	0.214	0.266	0.246	0.262	0.323	0.276	0.318	0.127	0.110	0.222	0.221
	TimesNet	0.179	0.224	0.214	0.283	0.282	0.302	0.093	0.068	0.124	0.174	0.128	0.188	0.204	0.191	0.231	0.257	0.046	0.035	0.141	0.200
	AnomalyTrans.	0.077	0.227	0.206	0.126	0.275	0.262	0.024	0.025	0.083	0.181	0.073	0.197	0.177	0.123	0.235	0.227	0.020	0.032	0.090	0.169
	TranAD	0.202	0.206	0.219	0.264	0.255	0.278	0.101	0.084	0.157	0.186	0.249	0.237	0.252	0.295	0.280	0.295	0.119	0.093	0.204	0.220
Avg		0.225	0.214	**0.240**	0.289	0.263	**0.301**	0.118	0.105	0.154	**0.198**	0.212	0.233	**0.240**	0.249	0.266	**0.277**	0.104	0.082	0.182	**0.215**

D.2 Per Domain Analysis.

Table 6: Per Domain Comparison on VUS-PR. SO = Source, PV = PVBoost, Δ = PV $-$ SO. Each cell averages the TSB-AD-U and TSB-AD-M per-domain means; PVBoost values are three-seed means.

AD	Environment			Facility			Finance			HumanActivity			Medical			Sensor			Synthetic			Traffic			WebService		
	SO	PV	Δ	SO	PV	Δ	SO	PV	Δ	SO	PV	Δ	SO	PV	Δ	SO	PV	Δ	SO	PV	Δ	SO	PV	Δ	SO	PV	Δ
LOF	0.586	0.590	0.004	0.160	0.414	0.254	0.459	0.499	0.041	0.116	0.155	0.039	0.181	0.188	0.006	0.241	0.249	0.007	0.492	0.482	-0.009	0.197	0.189	-0.008	0.201	0.200	-0.002
KNN	0.645	0.644	-0.001	0.295	0.462	0.167	0.441	0.443	0.002	0.148	0.154	0.007	0.299	0.301	0.002	0.305	0.297	-0.008	0.487	0.474	-0.013	0.221	0.216	-0.005	0.223	0.218	-0.005
KMeansAD	0.696	0.688	-0.008	0.316	0.398	0.081	0.450	0.450	0.001	0.152	0.151	-0.001	0.335	0.326	-0.009	0.432	0.429	-0.003	0.568	0.581	0.013	0.170	0.153	-0.017	0.232	0.231	-0.001
OCSVM	0.590	0.573	-0.017	0.373	0.332	-0.041	0.435	0.438	0.003	0.096	0.106	0.010	0.132	0.142	0.011	0.287	0.308	0.021	0.359	0.354	-0.005	0.183	0.173	-0.009	0.172	0.172	-0.000
HBOS	0.534	0.527	-0.007	0.263	0.271	0.008	0.426	0.426	-0.000	0.119	0.121	0.001	0.208	0.203	-0.004	0.223	0.221	-0.002	0.300	0.300	-0.000	0.214	0.204	-0.010	0.098	0.096	-0.001
IForest	0.549	0.537	-0.012	0.275	0.284	0.009	0.415	0.415	0.000	0.116	0.124	0.008	0.210	0.202	-0.008	0.207	0.211	0.003	0.324	0.321	-0.003	0.149	0.139	-0.010	0.091	0.089	-0.002
PCA	0.571	0.570	-0.000	0.621	0.614	-0.007	0.521	0.533	0.012	0.448	0.519	0.071	0.252	0.231	-0.020	0.250	0.246	-0.005	0.240	0.241	0.001	0.303	0.294	-0.009	0.303	0.291	-0.013
LSTMAD	0.564	0.561	-0.003	0.528	0.522	-0.006	0.489	0.519	0.030	0.284	0.271	-0.013	0.142	0.102	-0.040	0.170	0.171	0.001	0.403	0.426	0.023	0.113	0.114	0.001	0.289	0.285	-0.004
DeepAnT	0.567	0.567	-0.000	0.492	0.498	0.006	0.513	0.558	0.046	0.228	0.262	0.033	0.185	0.113	-0.072	0.219	0.211	-0.007	0.452	0.448	-0.004	0.155	0.153	-0.002	0.371	0.330	-0.041
PatchTST	0.501	0.500	-0.001	0.459	0.602	0.143	0.461	0.493	0.032	0.087	0.118	0.032	0.159	0.112	-0.047	0.243	0.233	-0.010	0.293	0.291	-0.002	0.188	0.176	-0.012	0.221	0.214	-0.006
AutoEncoder	0.572	0.570	-0.002	0.436	0.412	-0.024	0.425	0.426	0.000	0.123	0.120	-0.002	0.180	0.174	-0.006	0.237	0.245	0.007	0.409	0.401	-0.008	0.199	0.184	-0.015	0.286	0.284	-0.002
OmniAnomaly	0.506	0.507	0.001	0.499	0.503	0.004	0.478	0.477	-0.001	0.303	0.294	-0.009	0.202	0.179	-0.023	0.126	0.141	0.014	0.214	0.225	0.012	0.108	0.107	-0.002	0.289	0.278	-0.011
TimesNet	0.489	0.491	0.002	0.363	0.522	0.159	0.469	0.505	0.037	0.070	0.117	0.047	0.110	0.100	-0.010	0.155	0.191	0.036	0.390	0.377	-0.012	0.151	0.165	0.013	0.325	0.273	-0.051
AnomalyTrans.	0.476	0.502	0.026	0.110	0.395	0.286	0.441	0.488	0.048	0.063	0.216	0.153	0.089	0.110	0.022	0.068	0.138	0.071	0.202	0.246	0.045	0.122	0.165	0.043	0.097	0.177	0.080
TranAD	0.508	0.508	0.000	0.481	0.506	0.025	0.473	0.505	0.032	0.285	0.275	-0.010	0.113	0.106	-0.006	0.107	0.121	0.013	0.300	0.314	0.014	0.110	0.107	-0.003	0.255	0.256	0.001
Average	0.557	0.556	-0.001	0.378	0.449	0.071	0.460	0.478	0.019	0.176	0.200	0.024	0.186	0.173	-0.014	0.218	0.227	0.009	0.362	0.365	0.003	0.172	0.169	-0.003	0.230	0.226	-0.004

To complement the aggregate results in Table 1, we provide a per-domain breakdown of PVBoost's behavior across the nine domains in TSB-AD. Tables 6 and 7 report VUS-PR and F-score, respectively, for every (detector, domain) combination, with Δ columns showing the change relative to the source detector (VUS-PR) or the 3σ baseline (F-score).

Score refinement. Table 6 shows that VUS-PR improvements are unevenly distributed across domains. The largest gains occur on *Facility* ($+0.071$ on average), *HumanActivity* ($+0.024$), and *Finance* ($+0.019$), where source scores leave substantial headroom for refinement (averaging 0.176 – 0.460). In contrast, on *Environment* and *Synthetic* the source scores are already strong (0.557 and 0.362 on average), and PVBoost stays close to the source rather than overcorrecting (-0.001 and $+0.003$). The only domain-average regression larger than 0.005 is *Medical* (-0.014). The

Table 7: Per Domain Comparison on F1. PV = PVBOOST, Δ = PV $-$ 3σ. Each cell averages the TSB-AD-U and TSB-AD-M per-domain means; PVBOOST values are three-seed means.

AD	Environment			Facility			Finance			HumanActivity			Medical			Sensor			Synthetic			Traffic			WebService		
	3σ	PV	Δ	3σ	PV	Δ	3σ	PV	Δ	3σ	PV	Δ	3σ	PV	Δ	3σ	PV	Δ	3σ	PV	Δ	3σ	PV	Δ	3σ	PV	Δ
LOF	0.117	0.165	0.048	0.076	0.396	0.320	0.008	0.043	0.035	0.070	0.144	0.074	0.102	0.148	0.046	0.153	0.175	0.022	0.140	0.203	0.063	0.071	0.136	0.065	0.053	0.076	0.024
KNN	0.105	0.163	0.058	0.110	0.417	0.307	0.005	0.049	0.044	0.023	0.157	0.134	0.167	0.234	0.067	0.207	0.220	0.013	0.168	0.263	0.094	0.067	0.164	0.097	0.034	0.074	0.040
KMeansAD	0.091	0.167	0.076	0.165	0.350	0.185	0.003	0.027	0.024	0.010	0.144	0.134	0.161	0.240	0.080	0.229	0.263	0.034	0.100	0.225	0.125	0.044	0.074	0.030	0.035	0.074	0.039
OCSVM	0.065	0.117	0.052	0.125	0.307	0.182	0.001	0.038	0.037	0.014	0.084	0.071	0.038	0.104	0.066	0.234	0.252	0.018	0.122	0.169	0.047	0.000	0.158	0.158	0.011	0.070	0.058
HBOS	0.087	0.154	0.067	0.121	0.213	0.091	0.006	0.039	0.034	0.017	0.136	0.119	0.094	0.172	0.079	0.177	0.180	0.002	0.088	0.159	0.071	0.053	0.103	0.049	0.017	0.049	0.032
IForest	0.246	0.201	-0.045	0.107	0.276	0.169	0.006	0.030	0.023	0.009	0.140	0.131	0.098	0.160	0.063	0.134	0.165	0.030	0.077	0.157	0.080	0.025	0.054	0.029	0.017	0.037	0.020
PCA	0.231	0.520	0.289	0.366	0.527	0.160	0.076	0.079	0.002	0.206	0.488	0.282	0.168	0.191	0.023	0.175	0.211	0.036	0.057	0.094	0.037	0.090	0.233	0.143	0.190	0.190	-0.000
LSTMAD	0.296	0.456	0.159	0.249	0.421	0.171	0.177	0.160	-0.017	0.110	0.228	0.119	0.091	0.101	0.010	0.095	0.132	0.037	0.153	0.073	-0.080	0.022	0.066	0.043	0.179	0.169	-0.010
DeepAnT	0.383	0.518	0.135	0.258	0.421	0.163	0.318	0.346	0.028	0.113	0.211	0.098	0.122	0.104	-0.018	0.144	0.160	0.016	0.182	0.097	-0.085	0.064	0.136	0.072	0.191	0.174	-0.017
PatchTST	0.037	0.076	0.039	0.111	0.508	0.397	0.025	0.054	0.029	0.037	0.079	0.043	0.083	0.111	0.028	0.084	0.131	0.047	0.046	0.080	0.034	0.077	0.097	0.020	0.040	0.125	0.085
AutoEncoder	0.242	0.392	0.150	0.178	0.372	0.194	0.017	0.030	0.013	0.023	0.086	0.064	0.072	0.156	0.084	0.172	0.202	0.030	0.083	0.134	0.051	0.042	0.065	0.023	0.077	0.176	0.099
OmniAnomaly	0.015	0.099	0.084	0.237	0.427	0.190	0.013	0.101	0.088	0.118	0.254	0.136	0.132	0.151	0.019	0.072	0.113	0.041	0.018	0.049	0.031	0.026	0.036	0.011	0.207	0.172	-0.035
TimesNet	0.030	0.158	0.129	0.142	0.454	0.312	0.025	0.063	0.037	0.014	0.093	0.079	0.042	0.091	0.049	0.077	0.122	0.045	0.038	0.061	0.023	0.056	0.107	0.051	0.076	0.102	0.026
AnomalyTrans.	0.017	0.065	0.047	0.019	0.385	0.366	0.005	0.015	0.010	0.012	0.151	0.139	0.023	0.103	0.080	0.032	0.103	0.071	0.017	0.086	0.069	0.011	0.140	0.129	0.030	0.114	0.084
TranAD	0.030	0.058	0.028	0.226	0.450	0.224	0.011	0.050	0.039	0.110	0.229	0.120	0.062	0.101	0.039	0.056	0.102	0.046	0.016	0.050	0.034	0.024	0.061	0.037	0.122	0.153	0.031
Average	0.133	0.221	0.088	0.166	0.395	0.229	0.046	0.075	0.029	0.059	0.175	0.116	0.097	0.145	0.048	0.136	0.169	0.033	0.087	0.127	0.040	0.045	0.109	0.064	0.085	0.117	0.032

largest single-cell gain is AnomalyTransformer on *Facility* ($+0.286$), where a noisy source detector benefits dramatically from posterior-gated refinement; the largest regression (-0.072 for DeepAnT on *Medical*) is small in absolute terms and limited to detectors whose source scores are themselves unstable on that domain.

Thresholding. Table 7 reveals a much more uniform pattern: PVBOOST's posterior-derived threshold improves F-score over 3σ on *every* domain on average, with gains ranging from $+0.029$ on *Finance* to $+0.229$ on *Facility*. Four of the nine domains exhibit average improvements above $+0.05$, and the largest single-cell gain (PatchTST on *Facility*, $+0.397$) exceeds the largest VUS-PR gain ($+0.286$). This asymmetry between thresholding and scoring – also visible at the aggregate level in Section 4.2.1 – confirms that the dominant practical bottleneck of the unsupervised TSAD pipeline is decision-making rather than score quality, and that adapting the threshold to each detector's score distribution provides substantially larger headroom than refining the scores themselves.

Failure modes. Only 8 of the 135 (detector, domain) cells in Table 7 are negative. They are concentrated on the forecasting detectors LSTMAD and DeepAnT (5 cells, the largest being -0.085 and -0.080 on *Synthetic*) and on *WebService* (3 cells, including OmniAnomaly at -0.035). These failures are bounded in magnitude – no domain shows an average regression – and they are far outweighed by the consistent improvements on the remaining domain-detector combinations.

D.3 THRESHOLDING COMPARISON WITH SUPERVISED RATIO

To further contextualize the quality of PVBOOST's posterior-derived threshold, we compare it against a *supervised* oracle baseline ("Ratio") that uses the ground-truth anomaly ratio of each time series to compute the threshold via $\tau = \mathrm{Quantile}(\mathbf{f}, 1 - \pi^*)$, where π^* is the true anomaly proportion. This baseline cannot be deployed in practice – it requires labeled anomalies for every test series – but serves as an upper-bound reference on how well a quantile-based threshold could perform if the contamination ratio were known. Table 8 reports F-score across all 15 base detectors.

Table 8 reports two further references: π^*-Ref applies the true ratio to the *refined* score, and r_{opt} picks the F-optimal rate of each series, an upper bound for any rate-based rule.

PVBOOST matches the source-score oracle on the multivariate suite. On TSB-AD-M, PVBOOST's average F-score (0.215) slightly exceeds the π^*-Src oracle (0.213), despite operating without any access to ground-truth labels. PVBOOST outperforms this oracle on 5 of 15 detectors, with the largest wins on LOF (0.208 vs 0.108, $+0.100$), AnomalyTransformer (0.169 vs 0.094, $+0.075$), KNN (0.215 vs 0.147, $+0.068$), and TimesNet (0.200 vs 0.140, $+0.060$). These wins do not mean that the estimated π is better than the true one: at the true ratio on the refined score, the π^*-Ref oracle (0.243) is above PVBOOST on every detector. They come from refinement repairing the score ranking – a phenomenon we discuss below.

Table 8: PVBOOST's unsupervised threshold against label-supervised oracles (F-score). π^*-Src / π^*-Ref: threshold at the true anomaly ratio of each series, on the source / refined score. r_{opt}: the F-optimal rate of each series on the refined score – the ceiling of any rate-based rule.

	TSB-AD-U				TSB-AD-M			
Detector	PVBOOST	π^*-Src	π^*-Ref	r_{opt}	PVBOOST	π^*-Src	π^*-Ref	r_{opt}
LOF	0.175	0.155	0.181	0.249	0.208	0.108	0.217	0.261
KNN	0.233	0.262	0.262	0.325	0.215	0.147	0.225	0.265
KMeansAD	0.210	0.303	0.302	0.358	0.247	0.250	0.284	0.328
OCSVM	0.156	0.152	0.154	0.214	0.190	0.225	0.238	0.279
HBOS	0.157	0.176	0.172	0.218	0.137	0.142	0.160	0.248
IForest	0.157	0.164	0.160	0.205	0.166	0.184	0.192	0.272
PCA	0.311	0.360	0.354	0.401	0.291	0.308	0.307	0.362
LSTMAD	0.223	0.307	0.306	0.349	0.236	0.302	0.287	0.332
DeepAnT	0.238	0.320	0.328	0.371	0.252	0.301	0.283	0.342
PatchTST	0.191	0.207	0.240	0.280	0.210	0.191	0.236	0.286
AutoEncoder	0.163	0.185	0.185	0.241	0.271	0.298	0.299	0.349
OmniAnomaly	0.214	0.257	0.259	0.296	0.221	0.263	0.264	0.302
TimesNet	0.174	0.185	0.214	0.255	0.200	0.140	0.213	0.270
AnomalyTrans.	0.181	0.093	0.215	0.257	0.169	0.094	0.187	0.234
TranAD	0.186	0.200	0.215	0.252	0.220	0.250	0.257	0.298
Avg	0.198	0.222	0.237	0.285	0.215	0.213	0.243	0.295

Oracle advantage on the univariate suite. On TSB-AD-U, the π^*-Src oracle (0.222) outperforms PVBOOST (0.198) on average, and PVBOOST is ahead only on AnomalyTransformer, LOF and OCSVM. The gap concentrates on detectors with already-strong source scores: KMeansAD (-0.093), LSTMAD (-0.084), DeepAnT (-0.082), and PCA (-0.049). When source scores already rank anomalies well, an exact π estimate amplifies the existing ranking, and any error in PVBOOST's posterior-derived π directly costs F-score. The shorter typical length of univariate series in TSB-AD-U also gives the GMM fewer points to fit a stable posterior, widening the gap relative to the multivariate suite.

Takeaway. PVBOOST matches the source-score oracle on multivariate data and provides a robust unsupervised alternative on univariate data. The remaining headroom is in estimating π itself: PVBOOST is 0.028 (M) and 0.039 (U) below π^*-Ref, and the per-series ceiling r_{opt} (0.295 and 0.285) is higher still. More fundamentally, the cases where PVBOOST surpasses the oracle reveal a complementary benefit that motivates our entire framework: thresholding quality is bounded by score quality, and treating refinement and thresholding as a single coupled procedure (**C2**) can rescue detectors whose raw scores are misordered – a regime where supervised oracle thresholds inherently fail.

D.4 THRESHOLDING BASELINES, RANGE-BASED METRICS AND SEED VARIABILITY

Thresholding baselines. Table 9 widens the comparison of Table 1 to fixed-rate rules on both the source and the refined score. The strongest fixed rule is Top-10% on the refined score (0.168 U / 0.201 M); PVBOOST exceeds it by $+0.030$ (U) and $+0.014$ (M) per pair (Wilcoxon $p = 2.7 \times 10^{-36}$ and 4.8×10^{-5}). The constant cap $\min(\hat{\pi}_{\text{mass}}, 0.12)$ is the most demanding competitor because 0.12 was chosen on the test series; PVBOOST is ahead of it by $+0.018$ on TSB-AD-U ($p = 1.8 \times 10^{-7}$), but on TSB-AD-M the difference ($+0.008$) is not significant ($p = 0.93$). Relative to the posterior mass alone, the fragmentation term adds $+0.021$ (U) and $+0.013$ (M) and raises F on 15 and 14 of the 15 detectors.

Range-based metrics. Point-wise F can be dominated by a few long anomalies, so Table 10 also reports the range-based R-F1 and the affiliation F-score at the same thresholds. On R-F1, PVBOOST improves over the posterior-mass rule by $+0.043$ (U) and $+0.033$ (M) and over refined Top-10% by $+0.042$ and $+0.030$ (all $p < 10^{-15}$); the surrogate-null variant is slightly higher still. Affiliation F is nearly flat across the refined-score rules on TSB-AD-U (0.679 – 0.699), and on TSB-AD-M

Table 9: Thresholding rules compared on identical scores (point-wise F, average over 15 detectors; mean $\pm$ std over three seeds where the rule depends on the PVBOOST run). †The constant 0.12 is the best cap over a sweep *on the test series*, i.e. an optimistic tuned baseline. ‡When γGMM finds no admissible prior after 100 refits it returns $\hat{\gamma} = 0$ and flags nothing, as in the reference implementation (43% of U / 31% of M pairs on the source score). §One $\hat{\gamma}$ per series from the 15 source scores jointly, applied to each detector's score.

Rule	TSB-AD-U	TSB-AD-M
Source score		
3σ	0.118	0.104
POT	0.105	0.082
Top-1%	0.113	0.102
Top-5%	0.151	0.167
Top-10%	0.154	0.182
Top-15%	0.148	0.173
γGMM (Perini et al., 2023) (posterior-mean contamination)		
Source score, single detector‡	0.125	0.141
Source score, stack of all 15 detectors§	0.160	0.156
Refined score (seed 0)	0.120	0.142
Refined score		
3σ	0.128 ± 0.001	0.119 ± 0.001
Top-1%	0.111 ± 0.001	0.097 ± 0.000
Top-5%	0.158 ± 0.001	0.182 ± 0.000
Top-10%	0.168 ± 0.001	0.201 ± 0.001
Top-15%	0.160 ± 0.001	0.189 ± 0.001
Tuned constant cap $\min(\hat{\pi}_{\text{mass}}, 0.12)$†	0.180 ± 0.001	0.208 ± 0.002
Posterior mass $\hat{\pi}_{\text{mass}}$ only	0.177 ± 0.000	0.203 ± 0.002
PVBOOST ($\hat{\pi}$ with surrogate null)	0.193 ± 0.001	0.213 ± 0.001
PVBOOST ($\hat{\pi}$, Eq. equation 3)	0.198 ± 0.000	0.215 ± 0.001
Oracle (uses labels)		
True ratio π^*, source score	0.222	0.213
True ratio π^*, refined score	0.237 ± 0.001	0.243 ± 0.001
Best rate per series r_{opt}, refined	0.285 ± 0.001	0.295 ± 0.000

Table 10: Threshold-dependent metrics at the threshold each rule selects (average over 15 detectors, three seeds). Point-F1 is the F of Table 1; R-F1 and Aff-F are the range-based and affiliation F-scores of TSB-AD (Liu & Paparrizos, 2024). Src/Ref: rule applied to the source / PVBOOST-refined score. Bold: best non-oracle rule. †Tuned on the test series.

	TSB-AD-U			TSB-AD-M		
Rule	Point-F1	R-F1	Aff-F	Point-F1	R-F1	Aff-F
Src, 3σ	0.118	0.127	0.549	0.104	0.127	0.543
Src, POT	0.105	0.129	0.630	0.082	0.114	0.577
Src, Top-10%	0.154	0.087	0.690	0.182	0.119	0.708
Ref, Top-10%	0.168	0.097	0.692	0.201	0.126	**0.710**
Ref, tuned cap†	0.180	0.108	**0.699**	0.208	0.134	**0.710**
Ref, $\hat{\pi}_{\text{mass}}$	0.177	0.097	**0.699**	0.203	0.123	**0.710**
Ref, $\hat{\pi}$ (surrogate)	0.193	**0.149**	0.679	0.213	**0.162**	0.667
PVBOOST	**0.198**	0.139	**0.699**	**0.215**	0.155	0.690
Oracle π^* (ref)	0.237	0.151	0.692	0.243	0.167	0.690

PVBOOST is *lower* than the posterior-mass rule by 0.020 ($p = 0.005$): the fragmentation term lowers the flagged rate, and the affiliation metric, which rewards proximity to every anomalous event, penalises the missed events more than it rewards the removed false alarms. The gain of the rule is therefore in point and range F, not in affiliation F.

Table 11: Where the F gain comes from: a 2×2 factorial of score (source vs. PVBOOST-refined) $\times$ threshold (3σ vs. PVBOOST's $\hat{\pi}$). A: source + 3σ; B: source + $\hat{\pi}$; C: refined + 3σ; D: refined + $\hat{\pi}$ (full PVBOOST). Shares are of the total gain D$-$A. The last column isolates refinement under a fixed top-10% threshold.

	A	B	C	D	Threshold (B$-$A)	Refinement (C$-$A)	Interaction	Top-10%: source $\rightarrow$ refined
TSB-AD-U	0.118	0.179	0.128	0.198	+0.062 (77%)	+0.010 (13%)	+0.008 (10%)	0.154 $\rightarrow$ 0.168
TSB-AD-M	0.104	0.190	0.119	0.215	+0.086 (77%)	+0.015 (14%)	+0.010 (9%)	0.182 $\rightarrow$ 0.201

Table 12: Seed variability of PVBOOST: mean $\pm$ std over seeds 0/1/2 of the per-detector average (source scores are fixed; only PVBOOST's weak learners and GMM initialisation are reseeded). F uses the threshold rule of Eq. 3.

	TSB-AD-U		TSB-AD-M	
Detector	VUS-PR	F	VUS-PR	F
LOF	0.288 ± 0.001	0.175 ± 0.000	0.264 ± 0.004	0.208 ± 0.003
KNN	0.357 ± 0.000	0.233 ± 0.002	0.278 ± 0.002	0.215 ± 0.005
KMeansAD	0.373 ± 0.001	0.210 ± 0.000	0.339 ± 0.001	0.247 ± 0.004
OCSVM	0.238 ± 0.000	0.156 ± 0.004	0.268 ± 0.005	0.190 ± 0.010
HBOS	0.229 ± 0.000	0.157 ± 0.001	0.204 ± 0.001	0.137 ± 0.002
IForest	0.223 ± 0.000	0.157 ± 0.001	0.217 ± 0.000	0.166 ± 0.002
PCA	0.422 ± 0.000	0.311 ± 0.002	0.310 ± 0.001	0.291 ± 0.003
LSTMAD	0.328 ± 0.000	0.223 ± 0.002	0.291 ± 0.000	0.236 ± 0.003
DeepAnT	0.350 ± 0.001	0.238 ± 0.001	0.289 ± 0.001	0.252 ± 0.007
PatchTST	0.305 ± 0.001	0.191 ± 0.002	0.293 ± 0.004	0.210 ± 0.003
AutoEncoder	0.267 ± 0.000	0.163 ± 0.001	0.304 ± 0.001	0.271 ± 0.010
OmniAnomaly	0.297 ± 0.000	0.214 ± 0.001	0.318 ± 0.002	0.221 ± 0.005
TimesNet	0.302 ± 0.001	0.174 ± 0.003	0.257 ± 0.001	0.200 ± 0.003
AnomalyTrans.	0.262 ± 0.011	0.181 ± 0.012	0.227 ± 0.002	0.169 ± 0.004
TranAD	0.278 ± 0.001	0.186 ± 0.001	0.295 ± 0.000	0.220 ± 0.002
Avg	0.301 ± 0.001	0.198 ± 0.000	0.277 ± 0.001	0.215 ± 0.001

Refinement versus thresholding. Table 11 crosses the two contributions of PVBOOST. About 77% of the F gain over source + 3σ comes from the threshold ($+0.062$ U / $+0.086$ M), 13–14% from refinement and 9–10% from their interaction, on both suites. Refinement also helps independently of our threshold: with a fixed Top-10% rule it raises F from 0.154 to 0.168 (U) and from 0.182 to 0.201 (M).

Seed variability. Table 12 reruns PVBOOST with seeds 0/1/2; source scores are fixed, so only the weak learners and the GMM initialisation change. Suite-level averages move by at most 0.001 (VUS-PR 0.301 ± 0.001 U, 0.277 ± 0.001 M; F 0.198 ± 0.000 U, 0.215 ± 0.001 M). Per series, the median across-seed std of VUS-PR is 0.0002 (U) and 0.0007 (M), and 6.8% (U) and 10.9% (M) of pairs exceed 0.01. The most variable detector is AnomalyTransformer on TSB-AD-U (0.262 ± 0.011), the weakest source in that suite (AUC-PR 0.077).

Comparison with a Bayesian contamination estimator. γGMM (Perini et al., 2023) places a Dirichlet-process GMM on log-transformed, standardised anomaly scores, derives the posterior of the contamination factor, and thresholds at its posterior mean $\hat{\gamma}$. We run the reference implementation with its default hyperparameters ($p_0 = p_{\text{high}} = 0.01$, $\gamma_{\text{high}} = 0.15$, up to 100 refits). Because TSB-AD series reach 9×10^5 points, the DPGMM is fit on a uniform subsample of $2{,}000$ points per series and the full series is thresholded. The DP truncation is $K = 100$ on the 15-detector stack and $K = 20$ on single scores, where at most 9 components are active; on 150 random pairs rerun with $K = 100$, point F changes from 0.130 to 0.121, so the smaller truncation does not handicap γGMM. When no admissible prior is found after 100 refits, the reference implementation returns $\hat{\gamma} = 0$ and nothing is flagged; we keep this behaviour. The γGMM rows of Table 9 report three uses. (i) On each detector's source score, γGMM reaches 0.125 (U) and 0.141 (M) against 0.198 / 0.215 for PVBOOST; it fails on 43% (U) and 31% (M) of pairs. (ii) Given all 15 detectors' scores at once, its intended multi-detector use, it fails far less often on TSB-AD-U (2%; 19% on M) and reaches

0.160 / 0.156, although it then sees 15 detectors where PVBOOST sees one. (iii) Holding the score fixed isolates the rate: on the PVBOOST-refined score (seed 0), γGMM gives 0.120 / 0.142, failing on 52% / 47% of pairs. PVBOOST is higher than every variant on all 15 detectors of both suites, by 0.038–0.078 per pair (paired Wilcoxon $p < 10^{-28}$). The gap comes from reliability rather than from rate accuracy: where γGMM does return an estimate on the source score, its rate is closer to the true ratio than $\hat{\pi}$ (mean absolute error 0.052 / 0.043 vs. 0.070 / 0.069 on the same pairs), but $\hat{\pi}$ always yields a threshold.

D.5 CONTROLS FOR THE FRAGMENTATION TERM

Table 13: Controls for the fragmentation term of Eq. equation 3 (point F, macro over the 15 detectors, read off the per-series F-vs-rate curve). PVBOOST: $\hat{\pi}$ as shipped. Best cap: the best constant cap $\min(\hat{\pi}_{\text{mass}}, c)$ over $c \in [0.02, 0.15]$, chosen on the test series (value of c in parentheses). Permuted: r_{frag} shuffled across series within each (suite, detector), which keeps its distribution and destroys only the pairing with the series; mean $\pm$ std over 20 shuffles, and z of PVBOOST against it.

	TSB-AD-U				TSB-AD-M			
	PVBOOST	Best cap (c)	Permuted	z	PVBOOST	Best cap (c)	Permuted	z
Seed 0	0.196	0.181 (0.125)	0.166 ± 0.001	22.8	0.216	0.210 (0.120)	0.189 ± 0.002	12.8
Seed 1	0.196	0.180 (0.120)	0.165 ± 0.001	35.3	0.217	0.208 (0.120)	0.188 ± 0.002	17.9
Seed 2	0.196	0.181 (0.125)	0.165 ± 0.001	24.3	0.215	0.206 (0.120)	0.185 ± 0.001	30.5

Two alternative explanations for the gain of Eq. equation 3 are ruled out in Table 13, on all three seeds. *A better constant?* Capping the posterior mass at the best constant, chosen on the test series ($c = 0.12$ – 0.125 every time), reaches 0.180–0.181 (U) and 0.206–0.210 (M); PVBOOST is ahead by $+0.015$ to $+0.016$ (U) and $+0.007$ to $+0.009$ (M) without any tuning. *Any series-independent rate distribution?* Shuffling r_{frag} across the series of each (suite, detector) keeps its distribution but breaks its link to the series; F falls to 0.165 (U) and 0.186–0.189 (M), $z = 12.8$ to 35.3 standard deviations below the real rule. The gain therefore comes from reading each series' own temporal structure. The operator $\min(\cdot, \cdot)$ was selected on seed 0; seeds 1 and 2 and both controls were run afterwards and are held-out in that sense.

D.6 HOW WELL IS THE ANOMALY RATIO ESTIMATED?

Table 14: How well $\hat{\pi}$ (Eq. equation 3) estimates the true anomaly ratio π^*, by which term of the rule binds. Share: fraction of (detector, series) pairs. MAE and bias of $\hat{\pi} - \pi^*$; ρ: Spearman correlation of $\hat{\pi}$ with π^* and with the per-series F-optimal rate r_{opt}. Three-seed means. The last two rows score the posterior-mass rule and a constant rate the same way.

	TSB-AD-U					TSB-AD-M				
Regime	Share	MAE	Bias	$\rho(\cdot, \pi^*)$	$\rho(\cdot, r_{\text{opt}})$	Share	MAE	Bias	$\rho(\cdot, \pi^*)$	$\rho(\cdot, r_{\text{opt}})$
Cap binds ($\hat{\pi} = 0.15$)	56%	0.113	+0.111	+nan	+nan	53%	0.104	+0.103	+nan	+nan
Fragmentation binds	34%	0.042	+0.004	+0.25	+0.29	30%	0.037	+0.004	+0.29	+0.25
Posterior mass binds	9%	0.050	+0.016	+0.23	+0.17	17%	0.049	+0.013	+0.18	+0.16
All (PVBOOST)	–	0.083	+0.066	-0.10	+0.15	–	0.074	+0.058	+0.00	+0.22
$\hat{\pi}_{\text{mass}}$ only	–	0.097	+0.091	-0.06	+0.13	–	0.088	+0.080	-0.07	+0.13
Constant 0.10	–	0.067	+0.055	+nan	+nan	–	0.060	+0.050	+nan	+nan

True anomaly ratios in TSB-AD are small and spread over two orders of magnitude: the 5th/50th/95th percentiles of π^* are 0.07%/2.2%/12.9% on TSB-AD-U and 0.48%/3.8%/12.7% on TSB-AD-M, and 41% of univariate series have $\pi^* < 1\%$. A per-series rate is therefore needed, and Table 14 and Figure 5 show how far $\hat{\pi}$ is from providing an accurate one.

Where the rule binds. On 57% (U) and 53% (M) of pairs, neither the posterior mass nor the fragmentation test goes below the 0.15 cap, and $\hat{\pi} = 0.15$ (the horizontal band in Figure 5); with the posterior mass alone this happens on 80% and 73%. In these pairs $\hat{\pi}$ carries no information about π^*

Table 15: Point F by true anomaly ratio π^*, pooled over (detector, series) pairs and three seeds. n: pairs per seed. 3σ and Top-10% threshold the source score; $\hat{\pi}_{\text{mass}}$, PVBOOST and the oracle (π^*) threshold the refined score. Bold: best non-oracle rule.

	TSB-AD-U						TSB-AD-M					
π^*	n	3σ	Top-10%	$\hat{\pi}_{\text{mass}}$	PVBOOST	Oracle	n	3σ	Top-10%	$\hat{\pi}_{\text{mass}}$	PVBOOST	Oracle
$< 0.5\%$	1335	**0.053**	0.012	0.016	0.032	0.086	180	**0.070**	0.018	0.025	0.046	0.086
0.5–1%	840	**0.158**	0.054	0.075	0.108	0.189	390	**0.088**	0.031	0.047	0.074	0.121
1–2%	405	**0.188**	0.086	0.115	0.176	0.211	375	**0.103**	0.062	0.061	0.083	0.132
2–5%	885	0.157	0.165	0.171	**0.200**	0.222	735	0.111	0.143	0.146	**0.151**	0.162
5–10%	855	0.135	0.265	0.271	**0.284**	0.298	585	0.145	0.305	0.323	**0.344**	0.361
$\geq 10\%$	930	0.089	0.366	**0.449**	0.445	0.464	435	0.066	0.390	**0.475**	0.463	0.493

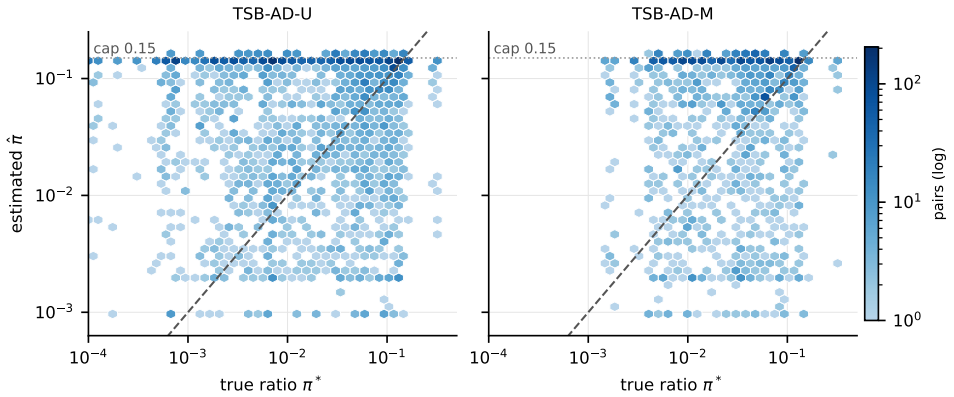

Figure 5: Estimated $\hat{\pi}$ against the true ratio π^* per (detector, series) pair, seed 0, log scales; colour is the number of pairs per cell. Dashed: $\hat{\pi} = \pi^*$; dotted: the 0.15 cap of $\hat{\pi}_{\text{mass}}$.

and over-estimates it by 0.11 (U) and 0.10 (M) on average. Where the fragmentation test binds (34% U / 30% M), the estimate is nearly unbiased (bias $+0.004$) with MAE 0.04, and it is rank-correlated with both π^* ($\rho = 0.25$ / 0.29) and r_{opt} ($\rho = 0.29$ / 0.25). Pooled over all pairs, $\hat{\pi}$ is *not* a good estimate of π^*: its MAE (0.083 U / 0.074 M) is larger than that of the constant 0.10 (0.067 / 0.060), and its rank correlation with π^* is -0.10 (U) and 0.00 (M). It is better aligned with r_{opt} ($\rho = 0.15$ / 0.22), the rate that maximises F on each series.

Tolerance. F is flat around its optimum: the set of rates reaching 90% of the per-series best F spans a factor of 1.8 in the median. $\hat{\pi}$ falls inside this band on 29% (U) and 30% (M) of pairs, versus 25%/26% for the posterior mass, 19%/22% for Top-10%, and 32%/32% for the true ratio π^* itself.

Rare anomalies are the failure case. Table 15 breaks F down by π^*. For $2\% \leq \pi^* < 10\%$, PVBOOST is the best non-oracle rule and close to the oracle; above 10% the posterior mass alone is slightly better (0.449 vs 0.445 U, 0.475 vs 0.463 M). For $\pi^* < 2\%$ – 49% of univariate and 35% of multivariate pairs – the plain 3σ rule on the source score is *better* than PVBOOST (e.g. 0.158 vs 0.108 for 0.5–1% on TSB-AD-U), because $\hat{\pi}$ stays at or near the cap (0.12–0.15 in the median) and flags 7.5× or more too many points. PVBOOST's aggregate advantage comes from the series with more anomalous mass, and the 0.15 cap of $\hat{\pi}_{\text{mass}}$ is the main open limitation of the threshold.

Synthetic sanity checks. On 20 replicates of series of length 20,000 with the posterior q taken from a two-component GMM on the z-scored series, the posterior mass returns the cap 0.15 on i.i.d. Gaussian noise with no anomaly, on AR(0.95) noise, and on noise with 5 planted $+3\sigma$ segments at every tested rate from 0.5% to 10%. With the fragmentation term, $\hat{\pi}$ falls to 0.021 on i.i.d. noise and tracks the planted rates (0.0034, 0.0078, 0.017, 0.048, 0.102 for 0.5, 1, 2, 5, 10%). On AR(0.95) noise the analytic null of Eq. equation 3 does not bind and $\hat{\pi}$ stays at 0.15, whereas the surrogate null returns 0.005; the surrogate variant (Table 9) is the correct choice when the score is strongly autocorrelated.

D.7 WHERE DOES SCORE REFINEMENT HELP?

Table 16: Paired, threshold-free effect of score refinement: AUC-PR of the source score and of PVBOOST (three-seed mean) on the same series. Δ: mean paired change; $^{\uparrow}$/$^{\downarrow}$: significant increase / decrease under a two-sided Wilcoxon signed-rank test at $p < 0.05$ (uncorrected). Improved: share of series whose AUC-PR rises.

	TSB-AD-U				TSB-AD-M			
Detector	Src	PVBOOST	Δ	Improved	Src	PVBOOST	Δ	Improved
LOF	0.162	0.183	+0.021	43%	0.096	0.215	+0.119$^{\uparrow}$	68%
KNN	0.273	0.269	-0.004$^{\downarrow}$	47%	0.138	0.226	+0.088$^{\uparrow}$	60%
KMeansAD	0.321	0.316	-0.004$^{\downarrow}$	44%	0.247	0.287	+0.040$^{\uparrow}$	36%
OCSVM	0.157	0.161	+0.004	51%	0.227	0.223	-0.004	39%
HBOS	0.175	0.172	-0.003$^{\downarrow}$	47%	0.161	0.162	+0.001	44%
IForest	0.165	0.161	-0.003$^{\downarrow}$	42%	0.189	0.189	+0.000	51%
PCA	0.374	0.368	-0.006$^{\downarrow}$	47%	0.308	0.304	-0.004$^{\downarrow}$	34%
LSTMAD	0.315	0.314	-0.001	44%	0.312	0.286	-0.026$^{\downarrow}$	27%
DeepAnT	0.330	0.332	+0.002	42%	0.317	0.287	-0.029$^{\downarrow}$	24%
PatchTST	0.189	0.231	+0.041	48%	0.175	0.231	+0.057$^{\uparrow}$	60%
AutoEncoder	0.191	0.189	-0.002$^{\downarrow}$	43%	0.296	0.300	+0.004	48%
OmniAnomaly	0.266	0.266	-0.000	49%	0.266	0.262	-0.004$^{\downarrow}$	39%
TimesNet	0.179	0.214	+0.034	51%	0.128	0.204	+0.076$^{\uparrow}$	58%
AnomalyTrans.	0.077	0.206	+0.129$^{\uparrow}$	69%	0.073	0.177	+0.103$^{\uparrow}$	65%
TranAD	0.202	0.219	+0.017	45%	0.249	0.252	+0.003	47%
All detectors	0.225	0.240	+0.015	47%	0.212	0.240	+0.028	47%
Excl. AnomalyTrans.	0.236	0.242	+0.007	46%	0.222	0.245	+0.023	45%

Table 17: Where PVBOOST helps, by the number of contiguous anomaly segments in the series. n: (detector, series) pairs. ΔAP: paired change in AUC-PR from source to PVBOOST (three-seed mean). ΔF: PVBOOST minus 3σ on the source score.

	TSB-AD-U			TSB-AD-M		
Segments	n	ΔAP	ΔF	n	ΔAP	ΔF
1	2415	+0.003	+0.027	705	+0.018	+0.031
2–3	1290	+0.058	+0.210	705	+0.074	+0.216
4–10	540	+0.003	+0.025	465	+0.027	+0.127
11–50	435	-0.012	+0.050	345	+0.002	+0.042
> 50	570	-0.001	+0.090	480	-0.003	+0.112

Paired, threshold-free effect. Table 16 compares the AUC-PR of the source and of PVBOOST on the same series. Averaged over detectors, refinement raises AUC-PR from 0.225 to 0.240 (U, $+6.6\%$) and from 0.212 to 0.240 (M, $+13.3\%$), but the median per-series change is zero on both suites and only 47% of series improve. On TSB-AD-U the gain is carried by AnomalyTransformer (source AUC-PR 0.077, $+0.129$): without it the U gain halves to $+0.007$, and six detectors (KNN, KMeansAD, HBOS, IForest, PCA, AutoEncoder) show small but significant decreases. On TSB-AD-M six detectors improve significantly (LOF, KNN, KMeansAD, PatchTST, TimesNet, AnomalyTransformer) and four decrease (PCA, LSTMAD, DeepAnT, OmniAnomaly). Score refinement is thus beneficial on average but detector-dependent, and it cannot create anomaly evidence that the source score does not contain.

What the mixture models. The GMM of Section 3.3 models the scalar *detector score*, not the raw observations or individual anomaly types: spikes, level shifts and variance changes can all fall in the same high-score component, and the mixture estimates the anomalous tail mass rather than distinguishing anomaly types.

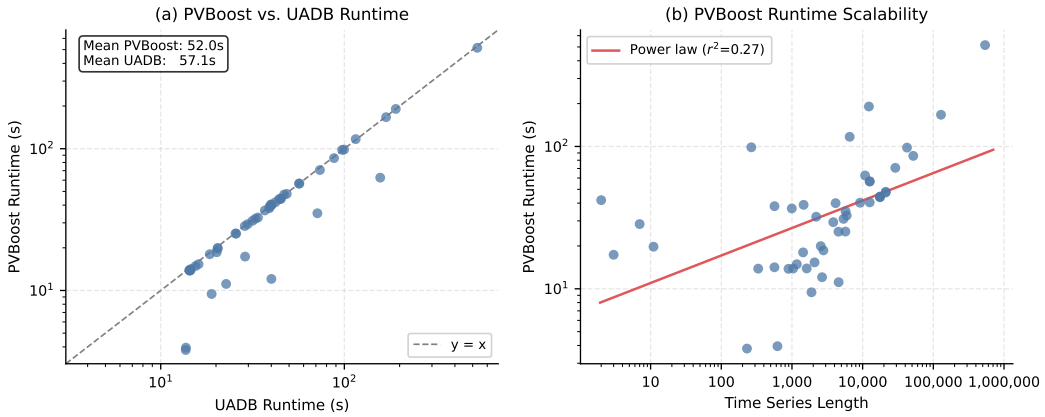

Figure 6: Runtime analysis. (a) Per-series runtime of PVBOOST versus UADB; each point is one (detector, dataset) combination, and the dashed line denotes $y = x$. (b) PVBOOST runtime as a function of time-series length, with a power-law fit ($r^2 = 0.27$).

Fragmented anomalies. Table 17 groups series by the number of contiguous anomaly segments. Refinement helps most on series with 2–3 segments ($+0.058$ U / $+0.074$ M AUC-PR, and the largest F gains, $+0.21$ / $+0.22$) and is neutral or slightly negative beyond 10 segments. This is the same structure that the fragmentation test of Eq. equation 3 reads: PVBOOST works best when anomalies form a few contiguous segments, and highly fragmented anomaly patterns remain a limitation for both refinement and the threshold.

D.8 RUNTIME ANALYSIS.

We assess the practical efficiency of PVBOOST along two axes: (i) end-to-end runtime relative to UADB, the closest booster-style baseline, and (ii) scalability with respect to time-series length. All measurements are conducted on the hardware described in Appendix D.1.

Runtime versus UADB. Figure 6(a) plots per-series runtime of PVBOOST against UADB, where each point corresponds to one (detector, dataset) combination. Despite incorporating an additional GMM fit and posterior update at every iteration, PVBOOST averages 52.0 seconds per series, slightly faster than UADB's 57.1 seconds. Most points cluster near the $y = x$ diagonal, indicating that the two methods have comparable end-to-end cost across detectors and datasets. The advantage of PVBOOST stems from two design choices: warm-starting the GMM with the previous EM solution (Section 3.3) amortizes posterior estimation cost across iterations, and posterior-drift early stopping (Section 3.5) terminates the loop as soon as the posterior converges, often well before the maximum $T = 10$ iterations. The runtime parity confirms that the benefits of posterior calibration come without an efficiency penalty relative to existing booster frameworks.

Scalability with time-series length. Figure 6(b) shows PVBOOST's runtime as a function of the input time-series length, ranging over six orders of magnitude (from ~ 10 to $\sim 10^6$ time steps). A power-law fit yields a slope consistent with sub-linear empirical scaling ($r^2 = 0.27$): runtimes on million-length series remain within an order of magnitude of those on thousand-length series. The weak correlation indicates that runtime is shaped more by the base detector's complexity and per-iteration weak learner training cost than by the series length itself, since GMM fitting and quantile-based thresholding scale linearly in the number of points and contribute negligibly to the total cost. This makes PVBOOST practical for the long sequences common in real-world TSAD applications.

E LIMITATIONS AND FUTURE WORK

While PVBOOST achieves strong empirical performance, we discuss several directions for future investigation.

Bimodal score assumption. PVBOOST relies on a two-component Gaussian mixture to estimate the anomaly posterior, which implicitly assumes that the score distribution can be reasonably separated into a normal majority and an anomalous minority. This assumption holds for most detectors in our experiments and is deliberately restrictive (Section 3.3) – in our DPGMM ablation (Section 4.2.2), letting the data determine the number of components does not improve score quality (VUS-PR 0.273 vs. 0.290) and badly hurts thresholding (F 0.065 vs. 0.217). However, scores from detectors with multi-modal failure modes, or from heavily contaminated data where the "normal" mode itself is fragmented, may violate this assumption. Extending the framework to structured mixture families that preserve a well-defined anomaly component while admitting richer normal structure is a promising direction.

Rare anomalies and the rate cap. The threshold inherits the 0.15 cap of the posterior mass. On more than half of the (detector, series) pairs neither term of Eq. equation 3 goes below it, and on series with $\pi^* < 2\%$ this makes PVBOOST flag far too many points and fall behind a plain 3σ rule (Appendix D.6). A rate estimate that remains reliable when the anomaly ratio is well below 1% is the most direct way to improve the threshold further.

Offline batch setting. The current loop runs offline over the full time series: each iteration refits the GMM on all scores and trains a weak learner on the entire sequence. Many practical TSAD applications (e.g., real-time monitoring, streaming fraud detection) require incremental processing under constraints on memory and latency. Adapting PVBOOST to a streaming setting would require online posterior updates, sliding-window variance estimation, and incremental weak-learner training – all non-trivial extensions of the current design. Investigating how the posterior-variance feedback loop behaves under concept drift is a particularly interesting open question.