# WHEN NORMAL-ONLY UPDATES ARE NOT ENOUGH: TEST-TIME ADAPTATION IN RECONSTRUCTION-BASED TIME-SERIES ANOMALY DETECTION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Normal operating conditions can evolve after deployment, motivating test-time adaptation (TTA) of time-series anomaly detectors using incoming data. Yet the same stream may contain anomalies, so TTA with online updates must decide which test samples should be used for adaptation. This leaves a more basic question: even if updates are restricted to samples labeled as normal in the benchmark, does TTA reliably improve anomaly detection? We study this question with a controlled intervention that replaces the practical update selector with a rule that uses only samples labeled as normal in the benchmark, while keeping the rest of the online adaptation procedure unchanged. Across nine benchmark families, four optimizer configurations, and five random seeds, we obtain three main findings: (1) using only samples labeled as normal in the benchmark does not reliably improve detection; (2) practical TTA failures exhibit two distinct responses to this intervention, with 63 of 180 settings recovering a clear detection gain while 90 still show no clear detection gain; and (3) the updates reduce their immediate reconstruction objective across the evaluated settings, yet these reductions do not consistently lead to better anomaly ranking. Both failure patterns persist across alternative practical adaptation variants and modest perturbations of the benchmark labels. Together, these findings show that improving sample selection alone is not a complete solution to reconstruction-based TSAD TTA.

## 1 INTRODUCTION

Time series anomaly detection (TSAD) is widely used to monitor complex systems and detect departures from normal operating behavior (Cook et al., 2020; Wen et al., 2022). Many TSAD detectors learn or calibrate a notion of normality from historical observations, yet the normal operating conditions encountered after deployment may continue to evolve (Blázquez-García et al., 2022). Under distribution shift after deployment, a fixed detector may assign high anomaly scores to shifted but benign observations, producing false positives (Kim et al., 2024; 2026). This deployment setting motivates test-time adaptation (TTA), which updates a deployed detector during inference using incoming test data, allowing adaptation without waiting for a separate offline refitting cycle (Wang et al., 2021; 2025; Kim et al., 2024; 2026).

However, the same test stream that enables adaptation also creates a basic challenge. It contains both evolving normal observations that may support adaptation and the anomalies the detector is expected to detect. Indiscriminate updates therefore risk adapting the detector to anomalous observations rather than benign changes in normal behavior. Recent TTA methods for reconstruction-based TSAD therefore make explicit choices about which test samples drive adaptation, and some also constrain how the detector is updated. For example, M2N2 (Kim et al., 2024) updates the detector using samples predicted as normal, whereas CANDI (Kim et al., 2026) further curates informative candidates using anomaly scores and similarity in the latent space and restricts adaptation to a lightweight residual module. These designs show that careful sample selection addresses one important source of risk, but does not by itself determine whether the resulting updates improve anomaly detection. This leaves a basic question unresolved: if test-time updates are restricted to samples labeled as normal in the benchmark, would adaptation reliably improve anomaly detection?



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.

To answer this question, we construct an oracle setting using benchmark labels. In this setting, the practical selection rule is replaced by one that uses only samples labeled as normal in the benchmark, while keeping the source initialization, updater, optimizer configuration, and online protocol unchanged. Our first finding is that adaptation using only samples labeled as normal in the benchmark can lead to very different detection outcomes. Across nine benchmark families, four optimizer configurations, and five random seeds, it produces substantial gains in some settings, little change in others, and degradation in some cases.

Our second finding is that practical TTA failures do not all respond to this intervention in the same way. Under our primary practical selector, which selects samples predicted as normal using a score threshold, 63 of the 180 matched cells show no clear detection gain under practical TTA but recover a clear gain when updates are restricted to samples labeled as normal in the benchmark. In contrast, 90 cells still show no clear detection gain even after the same replacement. Both patterns persist across alternative practical adaptation variants and modest perturbations of the benchmark labels. Taken together, these results show that changing sample selection can yield clear detection gains in some cases where practical TTA fails, while leaving others unresolved, motivating a closer examination of the adaptation behavior in the latter cases.

Failures that remain after restricting updates to samples labeled as normal in the benchmark raise a deeper question: could they simply reflect a failure to improve the reconstruction objective used for adaptation? Our third finding is that improving the reconstruction objective does not reliably imply better anomaly detection: the updates reduce their immediate reconstruction objective across the evaluated cells, while detection gains vary substantially. An analysis across all 180 cells further shows that larger detection gains are associated with more favorable correction of difficult normal–anomaly score orderings. This association is driven primarily by differences across datasets and adaptation settings, rather than by a consistent relationship across optimizers or seeds within individual datasets.

Together, our study yields three main empirical insights into reconstruction-based TSAD TTA. (1) Benchmark-labeled normal-only updates do not guarantee improved anomaly detection. Even when updates are restricted to samples labeled as normal in the benchmark, detection can improve substantially, change little, or degrade. (2) Practical TTA failures are not homogeneous. Some failures are rescued when practical selection is replaced by benchmark-labeled normal-only selection, whereas others persist under the same intervention. Thus, changing sample selection alone does not resolve all practical-TTA failures. (3) Improving the reconstruction objective does not reliably imply improved anomaly ranking. Across datasets and adaptation settings, larger detection gains are associated with more favorable changes in the relative scores of difficult normal and anomalous samples, although this relationship is primarily a between-dataset/regime pattern. To establish these insights, we use a matched label-conditioned intervention that changes update-sample selection while keeping the remaining online adaptation configuration fixed.

## 2 Related Work

**TSAD under distribution shift and test-time adaptation.** Most TSAD methods are trained under a relatively stable notion of normality, but deployment-time drift can cause shifted yet benign patterns to receive high anomaly scores. Reconstruction-based detectors are a common TSAD family, ranging from autoencoders (Sakurada & Yairi, 2014) to recurrent encoder–decoders (Malhotra et al., 2016) and adversarial reconstruction models such as USAD (Audibert et al., 2020). Because reconstruction error naturally provides both an anomaly score and an update objective, reconstruction-based detectors also provide a direct basis for test-time model adaptation. Earlier streaming anomaly-detection work considered adaptation under drift, including prediction-error-based incremental RNN adaptation and FITNESS-style fine-tuning on recent similar samples in streams with drift and outliers (Saurav et al., 2018; Sankararaman et al., 2022). More recent TSAD-TTA methods use explicit test-time update protocols with reconstruction-based detectors. M2N2 updates detector parameters using samples predicted as normal after trend-based normalization (Kim et al., 2024), whereas CANDI curates informative adaptation samples using anomaly scores and latent-space similarity and restricts updates to a lightweight residual module (Kim et al., 2026). Despite their different selection and update designs, both methods perform reconstruction-based test-time adaptation using selected test samples. Our work is complementary: rather than proposing another selector or update rule, we isolate the role of update-sample selection by replacing the evaluated selector with benchmark-labeled normal-only

selection under a matched adaptation configuration, and study when this intervention does and does not recover detection gains.

**General test-time adaptation.** Test-time adaptation (TTA) addresses distribution shift by adapting a trained model using unlabeled test data during inference. Prior online TTA methods, including TENT, CoTTA, and EATA, show that adaptation behavior can depend strongly on sample filtering, optimization stability, continual updating, and evaluation protocol (Wang et al., 2021; 2022; Niu et al., 2022; Wang et al., 2025). These issues are particularly relevant to TSAD because the same incoming stream supplies adaptation data while also containing the anomalies to be detected. Consequently, selecting which test samples drive updates is intertwined with how online parameter updates reshape the detector. Our study focuses on this interaction in reconstruction-based TSAD, where the update-sample selector can be changed while the remaining adaptation configuration is held fixed.

**Benchmarking and evaluation in TSAD.** Recent TSAD benchmark studies show that empirical conclusions can be sensitive to dataset choice, post-processing, windowing protocols, evaluation metrics, and benchmark-label quality. TAB documents inconsistencies in dataset splitting, threshold selection, point adjustment, and output protocols (Qiu et al., 2025), while TSB-AD highlights flaws in existing datasets and evaluation measures and advocates more reliable benchmarking practices (Liu & Paparrizos, 2024). These observations are particularly relevant to our label-conditioned intervention: benchmark labels provide a controlled way to define normal-only update samples, but they should not be interpreted as a perfect semantic notion of normality. Our work does not introduce a new benchmark; instead, it uses a matched intervention across existing TSAD benchmarks to study whether replacing practical update selection with benchmark-labeled normal-only selection is sufficient to recover detection gains, and how adaptation behavior relates to the downstream anomaly-ranking objective.

## 3 Problem Setup and Matched Intervention

We consider reconstruction-based TSAD with online test-time parameter updates. Let $X_t = (x_{t,1}, \ldots, x_{t,w})$ denote the test window processed at online step $t$, viewed as an element of $\mathbb{R}^{w \times D}$ by stacking the $w$ observations along the time dimension, where $w$ is the window length and $D$ is the number of variables. Here, $x_{t,i} \in \mathbb{R}^D$ is the multivariate observation at position $i \in \{1, \ldots, w\}$ within the window. Each point has a label $y_{t,i} \in \{0, 1\}$, where 0 denotes normal and 1 denotes anomalous. A reconstruction detector with parameters $\theta$ outputs $\hat{x}_{t,i}(\theta) \in \mathbb{R}^D$ and assigns the pointwise anomaly score

$$s_{t,i}(\theta) = \ell(x_{t,i}, \hat{x}_{t,i}(\theta)),$$

where $\ell$ denotes a scalar pointwise reconstruction loss. In the main autoencoder (AE) experiments, we instantiate it as the mean squared reconstruction error over the $D$ variables,

$$\ell(x_{t,i}, \hat{x}_{t,i}(\theta)) = \frac{1}{D} \sum_{d=1}^{D} \left(\hat{x}_{t,i,d}(\theta) - x_{t,i,d}\right)^2 .$$

This pointwise loss is used directly as the anomaly score, with larger values indicating higher anomaly likelihood. Detection is evaluated by ranking test points according to these scalar scores.

Test-time adaptation updates the detector using selected test points. We represent this selection by a point-level binary mask $m_t = (m_{t,1}, \ldots, m_{t,w}) \in \{0, 1\}^w$, where $m_{t,i} = 1$ means that position $i$ in window $X_t$ contributes to the update loss. The mask selects time points rather than individual variables. Given $m_t$, the test-time reconstruction objective is

$$\mathcal{L}_{\mathrm{TTA}}(\theta; X_t, m_t) = \frac{1}{w} \sum_{i=1}^{w} m_{t,i}\, \ell(x_{t,i}, \hat{x}_{t,i}(\theta)).$$

Unselected points contribute zero loss but remain in the window average; equivalently, point losses are zero-filled by the mask before averaging over the full window. The selector therefore determines which points contribute to the update objective and how much selected data contributes to the averaged update loss.

We compare three matched settings that differ in whether and how test points are selected for online updates. The frozen detector keeps the source parameters fixed, $\theta_t^F = \theta_0$ for all $t$. Practical TTA, denoted by $P$, uses our primary score-thresholded predicted-normal selector,

$$m_{t,i}^P = \mathbf{1}\{s_{t,i}(\theta_t^P) < \tau_{\text{sel}}\},$$

where $\tau_{\text{sel}}$ is calibrated separately for each physical entity and seed as the 95th percentile of point-level reconstruction scores produced by the frozen source checkpoint on the chronological source-validation split. The threshold is fixed throughout online testing and is never recalibrated using test data. This selector serves as the primary deployable baseline for the matched comparison and is not intended to represent all possible practical selection strategies. Oracle normal-only TTA, denoted by $O$, replaces this mask with the benchmark-label-conditioned normal mask,

$$m_{t,i}^O = \mathbf{1}\{y_{t,i} = 0\}\,.$$

The oracle is non-deployable and is not a performance upper bound. It controls whether update points are benchmark-labeled normal while leaving the remaining adaptation procedure matched; it does not make the selected normal points maximally informative or the update rule optimal.

For $r \in \{P, O\}$, both adaptive settings apply the same online update rule,

$$\theta_{t+1}^r = \text{Update}(\theta_t^r, \nabla_\theta \mathcal{L}_{\text{TTA}}(\theta_t^r; X_t, m_t^r))\,,$$

with the same source initialization, optimizer, learning rate, online order, updatable parameter set, and optimizer-step schedule. At each online step $t$, anomaly scores for $X_t$ are computed and recorded using the current parameters $\theta_t^r$ before any update on that window. The update then produces $\theta_{t+1}^r$, which is used only for subsequent test windows; scores already recorded for $X_t$ are not recomputed after the update. Thus, the controlled difference between Practical and Oracle TTA is the update mask produced by the selector, which also changes the total contribution of selected points to the masked update loss.

We evaluate detection using the area under the precision–recall curve (AUPRC) without point adjustment. For multi-entity datasets, the main paired analysis uses a global common physical support. Let $\mathcal{E}_d$ denote the retained entities for dataset $d$, shared by Frozen, Practical, and Oracle across the main optimizer–seed grid. For adaptive setting $r \in \{P, O\}$, optimizer configuration $c$, and seed $s$, we define the paired detection effect as

$$\Delta_{r,d,c,s} = \text{median}_{e \in \mathcal{E}_d}\left[\text{AUPRC}_{r,d,e,c,s} - \text{AUPRC}_{F,d,e,s}\right].$$

For a single-entity dataset, this reduces to the corresponding paired difference. Thus, physical entities are paired before dataset-level aggregation; we do not subtract independently aggregated arm medians.

AUPRC is expressed on the $[0, 1]$ scale, and $\Delta$ is the corresponding signed paired effect. When reporting effects in percentage points, we use $100\Delta$. We classify a cell as showing a clear detection gain only when its full-precision paired effect satisfies $\Delta > 0.005$, corresponding to more than $0.5$ percentage points.

Writing $\Delta_{\text{prac}} = \Delta_P$ and $\Delta_{\text{orc}} = \Delta_O$, the paired comparison yields four outcomes: both improve (BI), when both effects exceed $0.005$; oracle-only gain (OOG), when $\Delta_{\text{orc}} > 0.005$ and $\Delta_{\text{prac}} \leq 0.005$; neither improves (NI), when both effects are at most $0.005$; and practical-only gain (POG), when $\Delta_{\text{prac}} > 0.005$ and $\Delta_{\text{orc}} \leq 0.005$. NI denotes the absence of a clear gain under either adaptive setting; it includes near-zero changes as well as deterioration and does not imply that every NI cell degrades detection. An OOG outcome is consistent with the evaluated practical selector being limiting in that cell. An NI outcome indicates only that the failure persists under the tested benchmark-labeled normal-only replacement; it does not identify which remaining component of the adaptation procedure is responsible.

## 4 Experiments and Results

**Experimental setup.** The main study evaluates the AE backbone on nine benchmark families, four adaptation optimizer configurations, and five random seeds, yielding 180 dataset–optimizer–seed cells. Within each cell, Frozen, Practical, and Oracle share the source checkpoint. Practical and Oracle use a matched online adaptation configuration, with the update mask produced by the selector as the only controlled difference. Detailed dataset and protocol settings are provided in Appendix A.

### 4.1 ORACLE NORMAL-ONLY UPDATES DO NOT GUARANTEE DETECTION GAINS

We first examine whether benchmark-labeled normal-only adaptation reliably improves over the corresponding frozen detector. Across the 180 matched dataset–optimizer–seed cells, Oracle normal-only TTA produces a clear AUPRC gain in 90 cells and no clear gain in the remaining 90. The median paired Oracle-minus-Frozen AUPRC effect is 0.0033, with cell-level effects ranging from $-0.2169$ to $+0.5840$. Thus, even when practical selection is replaced by benchmark-labeled normal-only selection, adaptation does not yield a uniformly beneficial detection effect.

Table 1: Oracle-minus-Frozen paired AUPRC effects by dataset. For each dataset, results are summarized over the 20 combinations of four optimizer configurations and five random seeds. For each combination, the paired effect is the median of the effects across physical entities in the common support. Displayed values are in percentage points.

| Dataset | Mean | Median | Minimum | Maximum |
|---|---|---|---|---|
| Daphnet | $-8.75$ | $-7.16$ | $-21.69$ | $-0.03$ |
| GECCO | $+44.66$ | $+45.46$ | $+12.26$ | $+58.40$ |
| Genesis | $-0.21$ | $-0.04$ | $-0.72$ | $+0.00$ |
| MSL | $-0.03$ | $-0.01$ | $-0.11$ | $+0.00$ |
| PSM | $+12.35$ | $+13.46$ | $-2.29$ | $+24.70$ |
| PUMP | $+32.76$ | $+31.91$ | $+24.99$ | $+41.27$ |
| SMAP | $-0.03$ | $0.00$ | $-0.11$ | $0.00$ |
| SMD | $+2.06$ | $+1.65$ | $+0.02$ | $+6.07$ |
| SWAT | $+12.21$ | $+13.09$ | $+7.17$ | $+15.03$ |

Table 1 shows pronounced heterogeneity across datasets and adaptation configurations. GECCO, PUMP, PSM, and SWAT have positive average Oracle-minus-Frozen effects, whereas Daphnet is negative and Genesis, MSL, and SMAP remain near zero. SMD shows a smaller positive effect. Overall, benchmark-labeled normal-only adaptation can substantially improve detection in some settings, while providing little gain or degrading detection in others.

Together, these results show that access to benchmark-labeled normal update samples is not sufficient to guarantee successful TTA under the studied reconstruction objective and matched update protocol. Even under benchmark-labeled normal-only selection, detection effects remain heterogeneous across datasets and adaptation configurations.

### 4.2 ORACLE NORMAL-ONLY TTA REVEALS OUTCOMES HIDDEN BY PRACTICAL TTA ALONE

Comparing only the frozen detector with practical TTA shows whether a deployed TTA procedure helps, but it does not explain why it succeeds or fails. We therefore compare Practical TTA with its matched Oracle normal-only counterpart, differing in the update mask produced by the selector while keeping the remaining online adaptation configuration matched. This comparison distinguishes oracle-only-gain cases, where the benchmark-labeled normal-only replacement recovers a detection gain, from neither-improves cases, where the failure persists under the same replacement. An NI outcome does not identify the remaining failure mechanism; it establishes only that replacing the evaluated practical selector with benchmark-labeled normal-only selection is insufficient to produce a clear detection gain in that setting, motivating the objective-level analysis in the next subsection. We use the paired effects $\Delta_{\mathrm{prac}}$ and $\Delta_{\mathrm{orc}}$ and the four outcome categories defined above.

Figure 1 shows that the matched comparison separates practical-TTA failures into distinct outcomes. Out of 180 matched cells, 63 (35.0%) are oracle-only-gain cases: Practical TTA does not obtain a clear gain, whereas replacing its selector with benchmark-labeled normal-only selection does. These cases are consistent with the evaluated practical selector being limiting. In contrast, 90 cells (50.0%) are neither-improves cases, where neither Practical nor Oracle TTA obtains a clear gain over the Frozen detector. These failures therefore persist under the tested benchmark-labeled normal-only replacement. The remaining 27 cells (15.0%) improve under both Practical and Oracle TTA, and no cell falls into the practical-only-gain category.

The exact outcome proportions depend on the practical adaptation variant, but the two qualitatively different failure regimes remain visible under substantially different selection and preprocessing choices. In the 45-setting robustness evaluation, the primary Static-P95/B1 configuration yields

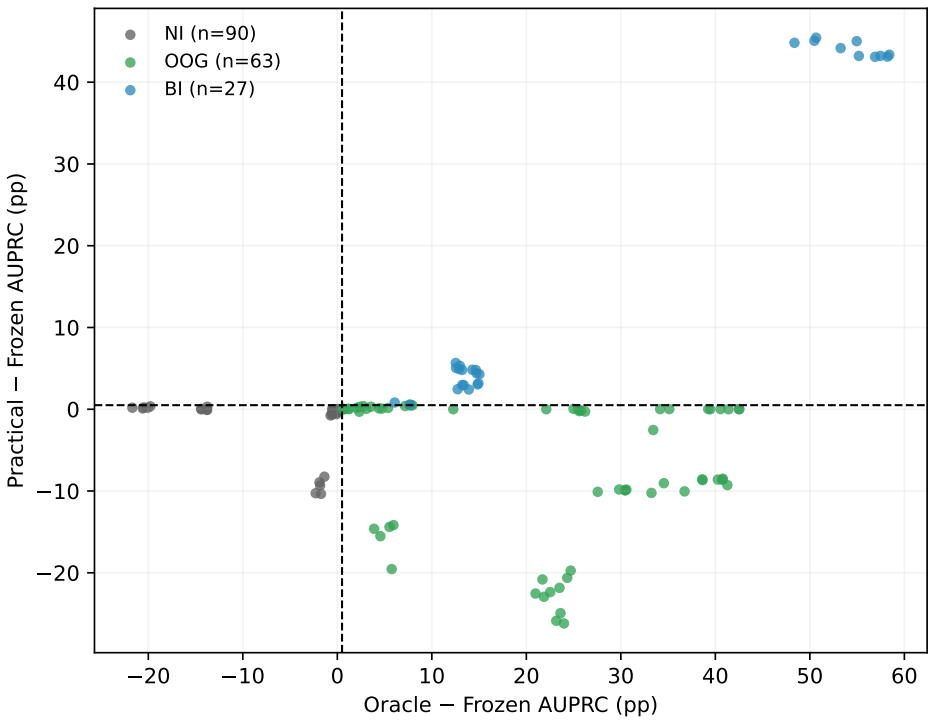



Figure 1: Matched-cell outcome categories from comparing practical TTA with oracle normal-only TTA. The figure plots paired cell-level gains over the frozen detector, with $\Delta_{\text{orc}}$ on the x-axis and $\Delta_{\text{prac}}$ on the y-axis. Dashed lines indicate the clear-gain threshold of $+0.5$ percentage points (pp). BI denotes a clear gain under both Practical and Oracle, OOG denotes a clear gain only under Oracle, NI denotes no clear gain under either setting, and POG denotes a clear gain only under Practical. No POG case occurs in the main grid.

BI/OOG/NI/POG counts of 10/15/20/0. The corresponding counts are 9/21/15/0 for an M2N2-style exponential moving average (EMA) detrending variant and 13/12/20/0 for a CANDI-style False Positive Mining (FPM) whole-window selector. Thus, changing the practical selection or preprocessing mechanism changes the exact balance between oracle-rescuable and other outcomes, but both oracle-only-gain and neither-improves cases remain substantial.

Overall, the matched comparison reveals that practical-TTA failure is not a single regime. Some failures are oracle-rescuable and therefore consistent with the evaluated selector being limiting, whereas others persist under benchmark-labeled normal-only selection. The latter cases do not identify a unique remaining cause, but they show that improving sample selection alone is not a complete solution under the evaluated adaptation configurations. We next examine whether these persistent failures can be explained simply by failure to improve the reconstruction objective.

### 4.3 Objective-level gains do not reliably imply detection gains

Sections 4.1 and 4.2 show that benchmark-labeled normal-only updates do not reliably improve detection. We next ask whether these persistent failures simply arise because adaptation fails to improve the reconstruction objective it optimizes. Test-time adaptation minimizes reconstruction loss on selected normal points, whereas anomaly detection depends on the relative ranking of normal and anomalous scores over the full test stream. We focus on Oracle normal-only TTA to examine this question after replacing the evaluated practical selector with benchmark-labeled normal-only selection.

For each Oracle run, we measure objective improvement by the immediate decrease in reconstruction loss induced by the online updates. At online step $t$, let $\theta_t^O$ and $\theta_{t+1}^O$ denote the Oracle parameters immediately before and after the update on $X_t$, respectively. For each benchmark-labeled normal

point contributing to that update, we define the pre–post loss decrease as

$$\delta^{O}_{t,i} = \ell\big(x_{t,i}, \hat{x}_{t,i}(\theta^{O}_{t})\big) - \ell\big(x_{t,i}, \hat{x}_{t,i}(\theta^{O}_{t+1})\big)\,.$$

Within each physical entity, we average $\delta^{O}_{t,i}$ over the benchmark-labeled normal points contributing to Oracle updates. For multi-entity datasets, we then take the median of the finite entity-level values to obtain one objective-reduction value for each dataset–optimizer–seed cell. Positive values indicate that the Oracle updates decrease their immediate reconstruction objective. Detection improvement is measured as $\Delta_{\mathrm{orc}}$, the effect of Oracle normal-only TTA relative to the Frozen detector. To understand when objective improvement does or does not translate into detection gain, we first examine the cell-level relationship and then analyze how adaptation reshapes difficult normal–anomaly score orderings.

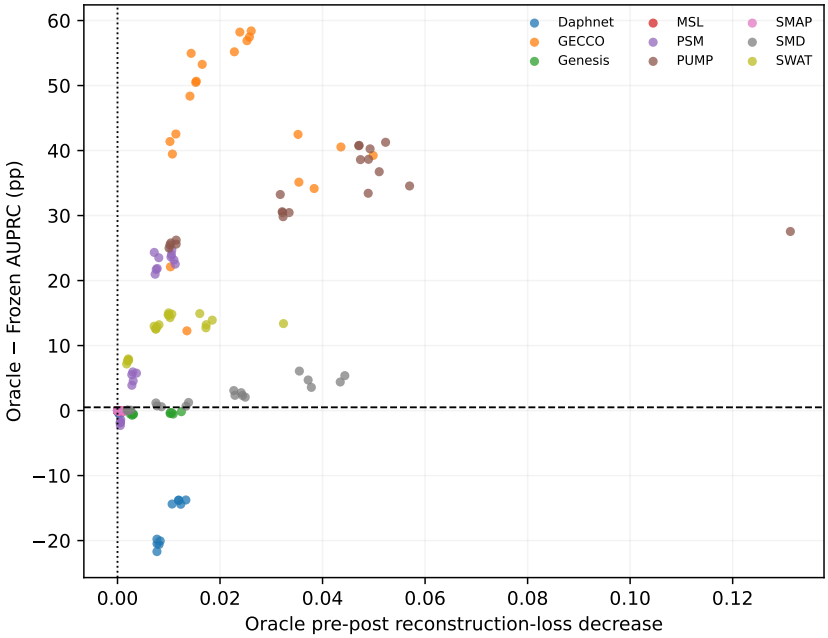



Figure 2: Objective-level improvement versus detection-level improvement under Oracle normal-only TTA. Each point represents one dataset–optimizer–seed cell. The x-axis reports the immediate pre–post decrease in reconstruction loss on benchmark-labeled normal points contributing to Oracle updates; positive values indicate that the online updates decrease their immediate reconstruction objective. Objective decreases are averaged over benchmark-labeled normal update points within each physical entity and then aggregated by the median over finite entities; this diagnostic support can differ from the common physical support used for the paired detection effect. The y-axis reports the paired Oracle-minus-Frozen AUPRC effect in percentage points (pp). The dotted vertical line marks zero objective decrease, and the dashed horizontal line marks the +0.5 pp clear-gain threshold for AUPRC. Colors denote datasets.

Figure 2 shows that the Oracle online updates decrease their immediate reconstruction objective across all 180 cells, but these objective-level improvements do not reliably imply detection success. All 180 objective reductions are positive, whereas 75 of the 180 Oracle-minus-Frozen detection effects are negative. Objective reduction and detection effect have a positive rank association (Spearman $\rho = 0.605$), but its dataset-cluster 95% confidence interval is wide ($[-0.020, 0.811]$). Thus, reconstruction-objective improvement alone does not provide a reliable criterion for detection-level improvement.

Reducing reconstruction error on benchmark-labeled normal update samples does not directly constrain how anomalous scores move relative to normal scores. We therefore examine difficult parts of the score distribution: hard normals, which have high Frozen anomaly scores, and boundary anomalies, which have low Frozen anomaly scores. Hard normals are benchmark-normal points in the upper Frozen-score tail, whereas boundary anomalies are anomalous points in the lower

Frozen-score tail. Selective compression measures whether Oracle reduces hard-normal scores more than boundary-anomaly scores, while net pair gain measures the change in the fraction of correctly ordered pairs of hard normal and boundary anomaly points.

Across the full 180-cell grid, larger Oracle detection gains are associated with more favorable score correction. The pooled Spearman correlations with Oracle-minus-Frozen AUPRC are $\rho = 0.736$ for selective compression and $\rho = 0.745$ for net pair gain. The same relationship is visible across the nine dataset medians ($\rho = 0.667$ and $0.728$, respectively). In contrast, after ranking cells within each dataset and pooling those within-dataset ranks, the associations are much weaker ($\rho = 0.144$ and $0.106$). Thus, the systematic relationship is primarily a between-dataset/regime pattern rather than a universal optimizer- or seed-level law within each dataset. These statistics are associative rather than causal and should not be interpreted as necessary or sufficient conditions for adaptation success.

Appendix C provides representative visual examples: PUMP illustrates a setting with favorable hard-normal compression and a large detection gain, whereas Daphnet illustrates that favorable-looking local score shifts need not be sufficient for improved anomaly ranking.

### 4.4 How robust are these findings?

**Cross-backbone robustness.** We first examine whether the two failure regimes are specific to the main AE backbone. Table 2 extends the matched comparison to USAD and LSTMED, with both full-model adaptation and separately audited decoder-only variants where available. Both oracle-only-gain (OOG) and neither-improves (NI) outcomes remain present in every evaluated backbone and update scope. Thus, the coexistence of oracle-rescuable and persistent failures is not confined to the main AE configuration.

Table 2: Cross-backbone and update-scope outcome decomposition under matched Practical–Oracle adaptation comparisons. Each row contains 180 logical dataset–optimizer–seed cells. Entries report counts with row percentages in parentheses. BI denotes cells where both Practical and Oracle show a clear gain; OOG denotes cells with no clear Practical gain but a clear Oracle gain; NI denotes cells with no clear gain under either setting; and POG denotes cells with a clear Practical gain but no clear Oracle gain. A clear gain requires $\Delta > 0.005$. Each row is aggregated on its own matched Frozen–Practical–Oracle physical support. LSTMED denotes the LSTM-based encoder–decoder backbone.

| Backbone | Update scope | Cells | BI | OOG | NI | POG |
|---|---|---|---|---|---|---|
| AE | Full model | 180 | 27 (15.0%) | 63 (35.0%) | 90 (50.0%) | 0 (0.0%) |
| USAD | Full model | 180 | 29 (16.1%) | 52 (28.9%) | 99 (55.0%) | 0 (0.0%) |
| USAD | Decoder only | 180 | 33 (18.3%) | 42 (23.3%) | 105 (58.3%) | 0 (0.0%) |
| LSTMED | Full model | 180 | 12 (6.7%) | 28 (15.6%) | 140 (77.8%) | 0 (0.0%) |
| LSTMED | Decoder only | 180 | 8 (4.4%) | 26 (14.4%) | 146 (81.1%) | 0 (0.0%) |

**Alternative adaptation variants.** Both oracle-only-gain (OOG) and neither-improves (NI) cases remain present under the evaluated alternative adaptation variants. We also evaluate a CANDI-style configuration combining False Positive Mining (FPM) with Spatiotemporally-Aware Normality Adaptation (SANA), the two core components of CANDI. In this setting, the AE backbone is frozen and only the SANA input/output adaptation modules are updated. We compare Practical TTA using FPM-selected windows with two Oracle variants: one updates on every window whose benchmark labels are all normal, while the other retains the FPM candidate set but removes windows containing benchmark-labeled anomalies. These comparisons yield BI/OOG/NI/POG counts of 7/18/20/0 and 7/1/37/0, respectively.

The two regimes are also substantially stable under perturbations of the benchmark labels. We consider entity-wise, segment-relative 10% boundary dilation and erosion, together with a fixed 10% false-negative contamination realization. Each perturbation is applied either to the Oracle update mask (mask-only), to the evaluation labels for all three arms (evaluation-only), or to both (joint). Boundary dilation preserves the outcome category of all 45/45 cells in all three channels. Boundary erosion preserves 45/45 categories in the mask-only channel and 41/45 in each of the evaluation-only

and joint channels. False-negative contamination preserves 40/45, 44/45, and 39/45 categories in the three channels, respectively.

Changing the online update cadence likewise retains both regimes. For cadence $B \in \{1, 2, 4, 8\}$, every chronological window is scored and contributes its masked loss, while gradients are accumulated over blocks of $B$ consecutive windows before each optimizer step. The corresponding BI/OOG/NI/POG counts for B1, B2, B4, and B8 are 10/15/20/0, 11/14/20/0, 12/13/20/0, and 7/13/25/0, respectively. Thus, the coexistence of oracle-rescuable and persistent failures is not restricted to the primary update cadence.

Finally, uncertainty estimates reflect substantial heterogeneity across the nine benchmark families. A 10,000-replicate dataset-cluster bootstrap gives an OOG proportion of $0.35$ with a 95% confidence interval of $[0.122,\ 0.600]$, and an NI proportion of $0.50$ with a 95% confidence interval of $[0.194,\ 0.806]$. These intervals emphasize that the 180 optimizer–seed cells should not be interpreted as 180 independent dataset-level observations.

Overall, these checks change the exact outcome proportions but preserve both oracle-rescuable and persistent failures.

## 5 Discussion and Conclusion

We studied test-time adaptation for reconstruction-based time-series anomaly detection under deployment-time distribution shift, focusing on the role of update-sample selection. Our study establishes three main empirical insights into reconstruction-based TSAD TTA. First, replacing a practical update selector with benchmark-labeled normal-only selection does not guarantee improved anomaly detection under the same matched adaptation configuration. Second, paired practical and normal-only adaptation reveals qualitatively different failure regimes: some practical failures are rescued by the normal-only intervention, whereas others persist even after this replacement. Third, improving the reconstruction objective on benchmark-labeled normal update samples does not reliably translate into improved anomaly ranking. Our score-level analysis further shows that larger detection gains under normal-only adaptation are associated across datasets and adaptation regimes with more favorable correction of difficult normal–anomaly score orderings, although this relationship is primarily a between-dataset/regime pattern rather than a universal within-dataset law.

Together, these findings show that update-sample selection matters, but changing selection alone is not sufficient to ensure detection gains in the adaptation settings studied here. When replacing practical selection with benchmark-labeled normal-only selection recovers detection gains, the result is consistent with the evaluated selector being limiting. When failure persists under the same intervention, however, the normal-only replacement alone is insufficient to recover detection improvement, motivating examination of the broader adaptation mechanism, including the update objective, optimization dynamics, update scope, model behavior, and their interaction with anomaly-score distributions. The matched normal-only intervention should therefore be interpreted as a diagnostic tool for studying the role of update-sample selection, rather than as a deployable TTA method or a performance upper bound.

Our conclusions are deliberately scoped to the evaluated reconstruction-based TSAD settings with online parameter updates. Forecasting-based, density-based, or other adaptation paradigms may use different objectives, score semantics, and update mechanisms, so the empirical findings here should not be assumed to transfer directly to those settings. In addition, the normal-only intervention is conditioned on benchmark labels. If those labels miss subtle anomalies, conflate benign shifts with anomalies, or reflect dataset-specific annotation conventions, the resulting update mask inherits these limitations. Thus, the intervention provides access to benchmark-labeled normal samples, not to a perfect semantic notion of normality.

Overall, understanding reconstruction-based TSAD TTA requires considering not only which samples are used for updating, but also how the resulting updates reshape the relative scores of normal and anomalous observations.

### AI USE STATEMENT

Large language models and AI-assisted coding tools were used to support research ideation and execution, including experimental planning, code development and debugging, analysis planning, and interpretation of results. They were also used to assist with drafting and polishing portions of the manuscript. The authors take responsibility for the final content of this work, including its experimental designs, implementations, analyses, scientific claims, citations, and manuscript text.

### REPRODUCIBILITY STATEMENT

We specify the datasets, preprocessing and window settings, optimizer configurations, random seeds, online update protocol, evaluation metrics, and aggregation procedures used in our experiments. Appendix A further documents the compute resources, common support and paired aggregation, and matched practical/oracle adaptation protocol. We will release the implementation, experiment configurations, and result-processing scripts upon publication.

## REFERENCES

Ahmed Abdulaal, Zhuanghua Liu, and Tomer Lancewicki. Practical approach to asynchronous multivariate time series anomaly detection and localization. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 2485–2494, 2021.

Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A. Zuluaga. USAD: unsupervised anomaly detection on multivariate time series. In Rajesh Gupta, Yan Liu, Jiliang Tang, and B. Aditya Prakash (eds.), *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pp. 3395–3404. ACM, 2020. doi: 10.1145/3394486.3403392. URL https://doi.org/10.1145/3394486.3403392.

Marc Bächlin, Meir Plotnik, Daniel Roggen, Inbal Maidan, Jeffrey M. Hausdorff, Nir Giladi, and Gerhard Tröster. Wearable assistant for Parkinson's disease patients with the freezing of gait symptom. *IEEE Transactions on Information Technology in Biomedicine*, 14(2):436–446, 2010.

Ane Blázquez-García, Angel Conde, Usue Mori, and José Antonio Lozano. A review on outlier/anomaly detection in time series data. *ACM Comput. Surv.*, 54(3):56:1–56:33, 2022. doi: 10.1145/3444690. URL https://doi.org/10.1145/3444690.

Andrew A. Cook, Goksel Misirli, and Zhong Fan. Anomaly detection for IoT time-series data: A survey. *IEEE Internet of Things Journal*, 7(7):6481–6494, 2020. doi: 10.1109/JIOT.2019.2958185. URL https://doi.org/10.1109/JIOT.2019.2958185.

Cheng Feng and Pengwei Tian. Time series anomaly detection for cyber-physical systems via neural system identification and Bayesian filtering. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 2858–2867, 2021.

Jonathan Goh, Sridhar Adepu, Khurum Nazir Junejo, and Aditya Mathur. A dataset to support research in the design of secure water treatment systems. In *International conference on critical information infrastructures security*, pp. 88–99. Springer, 2016.

Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Söderström. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 387–395, 2018.

Dongmin Kim, Sunghyun Park, and Jaegul Choo. When model meets new normals: Test-time adaptation for unsupervised time-series anomaly detection. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (eds.), *Thirty-Eighth AAAI Conference on Artificial Intelligence*, pp. 13113–13121. AAAI Press, 2024.

HyunGi Kim, Jisoo Mok, Hyungyu Lee, Juhyeon Shin, and Sungroh Yoon. CANDI: curated test-time adaptation for multivariate time-series anomaly detection under distribution shift. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor (eds.), *Fortieth AAAI Conference on Artificial Intelligence*, pp. 15018–15026. AAAI Press, 2026.

Qinghua Liu and John Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. In Amir Globerson, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10–15, 2024*, pp. 108231–108261, 2024.

Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. LSTM-based encoder-decoder for multi-sensor anomaly detection. *CoRR*, abs/1607.00148, 2016. URL http://arxiv.org/abs/1607.00148.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, Proceedings of Machine Learning Research, pp. 16888–16905. PMLR, 2022.

Xiangfei Qiu, Zhe Li, Wanghui Qiu, Shiyan Hu, Lekui Zhou, Xingjian Wu, Zhengyu Li, Chenjuan Guo, Aoying Zhou, Zhenli Sheng, Jilin Hu, Christian S. Jensen, and Bin Yang. TAB: unified benchmarking of time series anomaly detection methods. *Proc. VLDB Endow.*, 18(9):2775–2789, 2025. doi: 10.14778/3746405.3746407. URL https://www.vldb.org/pvldb/vol18/p2775-hu.pdf.

Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In Ashfaqur Rahman, Jeremiah D. Deng, and Jiuyong Li (eds.), *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, Gold Coast, Australia, QLD, Australia, December 2, 2014*, pp. 4. ACM, 2014. doi: 10.1145/2689746.2689747. URL https://doi.org/10.1145/2689746.2689747.

Abishek Sankararaman, Balakrishnan Narayanaswamy, Vikramank Y. Singh, and Zhao Song. FITNESS: (fine tune on new and similar samples) to detect anomalies in streams with drift and outliers. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, Proceedings of Machine Learning Research, pp. 19153–19177. PMLR, 2022.

Sakti Saurav, Pankaj Malhotra, Vishnu TV, Narendhar Gugulothu, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. Online anomaly detection with concept drift adaptation using recurrent neural networks. In Sayan Ranu, Niloy Ganguly, Raghu Ramakrishnan, Sunita Sarawagi, and Shourya Roy (eds.), *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data, COMAD/CODS 2018, Goa, India, January 11-13, 2018*, pp. 78–87. ACM, 2018.

SpotSeven. GECCO challenge 2018 (GECCO dataset). https://www.spotseven.de/gecco/gecco-challenge/gecco-challenge-2018/, 2018.

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)*, pp. 2828–2837, 2019.

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

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno A. Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=uXl3bZLkr3c.

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 7191–7201. IEEE, 2022.

Zixin Wang, Yadan Luo, Liang Zheng, Zhuoxiao Chen, Sen Wang, and Zi Huang. In search of lost online test-time adaptation: A survey. *Int. J. Comput. Vis.*, 133(3):1106–1139, 2025. doi: 10.1007/S11263-024-02213-5. URL https://doi.org/10.1007/s11263-024-02213-5.

Qingsong Wen, Linxiao Yang, Tian Zhou, and Liang Sun. Robust time series analysis and applications: An industrial perspective. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 4836–4837, 2022. doi: 10.1145/3534678.3542612. URL https://doi.org/10.1145/3534678.3542612.

# A Experimental and Protocol Details

## A.1 Dataset Details

We evaluate this oracle-based comparison on nine TSAD benchmarks: Daphnet (Bächlin et al., 2010), MSL and SMAP (Hundman et al., 2018), PSM (Abdulaal et al., 2021), PUMP (Feng & Tian, 2021), SMD (Su et al., 2019), GECCO (SpotSeven, 2018), Genesis (von Birgelen & Niggemann, 2018), and SWAT (Goh et al., 2016). These datasets span diverse application domains, including water treatment, server monitoring, spacecraft telemetry, health monitoring, and industrial or cyber-physical monitoring. Together, they provide a heterogeneous testbed for studying test-time adaptation under train–test distribution shift. Summary statistics of the datasets used in our experiments are provided in Table A.1.

For multi-entity benchmarks such as SMD, MSL, and SMAP, the reported lengths, anomaly ratios, and main results are aggregated at the benchmark level unless otherwise stated. In Table A.1, *Obtained from* indicates the repository or benchmark package from which the data were obtained for our experiments, rather than the original dataset paper. This distinction is important because some datasets were obtained from their original public repositories, whereas others were obtained from benchmark packages. When needed, we analyze variation across subsets separately in additional experiments, rather than using subsets as the default reporting unit in the main evaluation.

Table A.1 reports benchmark-level channel counts. In the model description, however, $D$ denotes the processed feature count of the current physical entity. Because preprocessing may remove features and multi-entity benchmarks may have entity-specific feature dimensions, $D$ need not equal the benchmark-level channel count reported in Table A.1.

The reported train and test lengths refer to the full processed splits before incomplete-window truncation. Test anomaly ratios are computed from the processed point-level test labels; for multi-entity benchmarks, anomalous and total test points are pooled across entities before computing the ratio.

Table A.1: Dataset properties used in the main experiments. *Obtained from* indicates the repository or benchmark package from which the data were obtained for our experiments.

| Dataset | Domain | Entities/ Subsets | Channels | Train Len | Test Len | Test Anomaly Ratio (%) | Obtained from |
|---|---|---|---|---|---|---|---|
| SWAT | Water treatment | 1 | 51 | 495,000 | 449,919 | 12.14 | TAB |
| SMD | Server machine | 28 | 38 | 708,405 | 708,420 | 4.16 | OmniAnomaly |
| MSL | Spacecraft | 27 | 55 | 58,317 | 73,729 | 10.48 | Telemanom |
| SMAP | Spacecraft | 55 | 25 | 140,825 | 444,035 | 12.83 | Telemanom |
| PSM | Server machine | 1 | 25 | 132,481 | 87,841 | 27.76 | RANSynCoders |
| Daphnet | Health | 1 | 9 | 9,693 | 29,081 | 7.93 | TSB-AD |
| PUMP | Water treatment | 1 | 44 | 110,151 | 110,151 | 9.40 | TAB |
| GECCO | Water treatment | 1 | 9 | 16,165 | 122,356 | 1.41 | TSB-AD |
| Genesis | Machinery | 1 | 18 | 4,055 | 12,165 | 0.41 | TSB-AD |

**Implementation basis and data sources.** Our implementation is built by modifying the open-source Time-Series-Library codebase, which is released under the MIT License. For benchmark sources, TSB-AD is released under Apache-2.0, OmniAnomaly under MIT, Telemanom under a BSD-style Caltech/JPL license, and RANSynCoders under a BSD License. For TAB, we did not find an explicit repository license in the GitHub repository at access time; we therefore report it as an acquisition source rather than assigning a license ourselves. The benchmark datasets used in our experiments were obtained from the corresponding sources listed in Table A.1.

## A.2 Windowing Protocol

Across all nine benchmark families, we use a window size of 12 and a test stride of 12. Test windows are therefore non-overlapping and are processed sequentially in chronological order during online adaptation.

### A.3 Backbone and Optimization Details

The main results in the paper are based on a reconstruction-based AE backbone. The model is first trained on the training split and then evaluated under the frozen, practical TTA, and oracle normal-only TTA settings using the same pre-trained source model. Reconstruction losses and anomaly scores are computed in standardized feature space, using the feature-wise mean squared reconstruction error defined in Section 3; no inverse transformation is applied before scoring.

**AE architecture.** A batch of input windows has shape $[B, 12, D]$, where $B$ is the batch size and $D$ is the processed feature count of the current physical entity. Each window is flattened to $F = 12D$ features before encoding. The encoder has dimensions

$$F \to H_1 \to H_2 \to L,$$

where

$$H_1 = \max(1, \lfloor F/2 \rfloor), \qquad H_2 = \max(1, \lfloor F/4 \rfloor), \qquad L = \min(64, \max(8, \lfloor F/8 \rfloor)).$$

A ReLU follows each encoder linear layer. The decoder reverses these dimensions as

$$L \to H_2 \to H_1 \to F,$$

with ReLU activations after its first two linear layers and no output activation, and reshapes the reconstruction back to $[B, 12, D]$. During the main full-model TTA protocol, all encoder and decoder parameters are updated; Frozen, Practical, and Oracle use the same entity- and seed-specific source-trained checkpoint.

For adaptation, we evaluate four optimizer configurations: SGD with learning rates $10^{-3}$ and $10^{-4}$, and AdamW with learning rates $10^{-4}$ and $3 \times 10^{-4}$. These configurations are applied consistently across practical and oracle TTA within each optimizer setting, so practical-vs-oracle comparisons are not confounded by different updater choices. Comparisons against the frozen detector still measure the combined effect of online updating and the selected update mask. All experiments are repeated over five random seeds. The main analysis uses the resulting dataset–optimizer–seed cells to compute paired Practical-minus-Frozen and Oracle-minus-Frozen detection effects as defined in Section 3.

We report the backbone and optimization settings needed to interpret the comparisons reported in the main text.

**Common support and paired aggregation.** For each multi-entity dataset, the main analysis uses one global retained physical-entity support shared by Frozen, Practical, and Oracle across the optimizer–seed grid. Paired detection effects are first computed at the physical-entity level as Practical-minus-Frozen and Oracle-minus-Frozen differences, and are then aggregated by the median within each dataset–optimizer–seed cell. For single-entity datasets, this reduces to the corresponding paired difference.

### A.4 Protocol Implementation Details

The frozen, practical TTA, and oracle normal-only TTA settings all start from the same source checkpoint for each physical entity and seed. The frozen detector performs no test-time parameter updates. Practical TTA and Oracle normal-only TTA use the matched update protocol: the same model initialization, updatable parameter set, optimizer family, learning rate, update rule, and online order.

Practical TTA uses the score-thresholded predicted-normal selector defined in Section 3. For each physical entity and seed, the official source-training series is split chronologically: the first 80% is used to train the source model and the final 20% is used as the source-validation split. The frozen source checkpoint is selected by minimum reconstruction loss on this validation split. Using that checkpoint, we compute point-level reconstruction scores on non-overlapping length-12 validation windows with stride 12, discarding an incomplete final window, and set $\tau_{\mathrm{sel}}$ to the 95th percentile of these scores. Calibration is performed separately for each physical entity and seed; multi-entity datasets therefore use entity-specific thresholds. No test data or test labels are used for calibration.

During online testing, $\tau_{\mathrm{sel}}$ remains fixed. For each arriving test window, Practical TTA computes point-level reconstruction scores using the current pre-update model and selects points whose scores are strictly below $\tau_{\mathrm{sel}}$. The threshold is not recalibrated from past or future test scores.

For both Practical and Oracle TTA, the point-level anomaly scores retained for detection evaluation are the scores computed before the update on the current window. After these scores are recorded, the corresponding update mask is constructed and the online parameter update is applied. The updated parameters are first used when processing the next test window; the current window is not rescored after adaptation.

Oracle normal-only TTA keeps the matched update protocol but replaces the practical predicted-normal mask with a benchmark-label-conditioned normal mask. For each arriving test window, only benchmark-labeled normal points contribute to the reconstruction update loss; all other point losses are set to zero before averaging over the full window. If the window contains no benchmark-labeled normal points, its masked data loss is zero, but the scheduled optimizer step is still executed; retained optimizer state can therefore change the parameters. Benchmark labels are used only for this diagnostic Oracle update selection and for final evaluation; they are not used to tune the threshold, learning rate, optimizer choice, update scope, or any other hyperparameter.

This implementation keeps the adaptation procedure matched between Practical and Oracle TTA except for the update mask. The optimizer, parameter update rule, learning rate, online order, update schedule, and source initialization are shared. Thus, the comparison isolates the effect of replacing the Practical predicted-normal selector with benchmark-normal selection, including the resulting change in how much selected data contributes to the update loss.

### A.5 Cross-Backbone and Update-Scope Robustness

**Cross-backbone protocol.** We additionally evaluate USAD, a dual-decoder reconstruction model, and LSTMED, a recurrent encoder–decoder reconstruction model. Each reported backbone–scope setting uses the same nine benchmark families, four optimizer configurations, and five random seeds as the main AE study, yielding 180 dataset–optimizer–seed cells. The Practical and Oracle comparisons follow the same intervention semantics as in Section 3: Practical selection is calibrated from the corresponding frozen source model, whereas Oracle selection uses benchmark-normal labels. Detection scores are recorded before each online update, and outcome categories use the same $\Delta > 0.005$ clear-gain criterion.

**Update scope.** Full-model adaptation updates all trainable detector parameters. Decoder-only adaptation freezes the encoder and updates only decoder-side reconstruction parameters. For USAD, this freezes the shared encoder and updates the decoder modules; for LSTMED, it freezes the recurrent encoder and updates the corresponding reconstruction decoder. Within each backbone–scope setting, Practical and Oracle TTA use the same parameter scope, optimizer configuration, online order, scoring procedure, and evaluation protocol; their controlled difference remains the update-sample selection rule.

**Windowing and matched support.** The reported USAD and LSTMED studies use the same non-overlapping test windowing protocol as the main analysis, with window length 12 and stride 12. Within each backbone and update scope, Frozen, Practical, and Oracle are evaluated on matched physical-entity support before aggregation to dataset–optimizer–seed cells. All five reported backbone–scope settings retain all 180 logical cells, although the retained physical-entity support need not be identical across backbone or update-scope settings.

### A.6 Compute Resources

All experiments were run on NVIDIA V100 GPUs with 32GB memory, with each run using a single GPU. The main AE study covers nine datasets, four optimizer configurations, and five random seeds, yielding 180 dataset–optimizer–seed cells. Additional selector, cadence, label-perturbation, and score-level analyses use the analysis-specific evaluation grids reported in the main text and appendix.

## B Additional Outcome and Robustness Results

This appendix summarizes the outcome and robustness results supporting the main analysis. Table B.1 reports the canonical 180-cell Practical–Oracle outcome counts, Table B.2 summarizes the evaluated

Table B.1: Main-grid outcome counts for the matched Practical–Oracle comparison. The 180 cells correspond to nine datasets, four optimizer configurations, and five random seeds. BI: both improve; OOG: oracle-only gain; NI: neither improves; POG: practical-only gain. A clear gain means a paired AUPRC improvement over Frozen greater than $0.005$ (0.5 percentage points).

| Analysis | Cells | BI | OOG | NI | POG |
|---|---|---|---|---|---|
| Main AE grid | 180 | 27 | 63 | 90 | 0 |

selector, adaptation, and cadence variants, and Table B.3 reports outcome-category retention under benchmark-label perturbations. These results complement the corresponding analyses in Sections 4.2 and 4.4.

Unless otherwise stated, the robustness analyses use the same 45 logical dataset–optimizer–seed cells: the nine benchmark families, AdamW with learning rate $10^{-4}$, and seeds 42–46. Static-P95/B1 is the corresponding 45-cell slice of the main 180-cell AE experiment and reuses its Frozen, Practical, and Oracle runs. The robustness analyses share these logical cell identities, although branch-specific analyses may use different matched physical-entity support.

For cadence $B \in \{1, 2, 4, 8\}$, every chronological test window is scored and masked, and its masked loss is scaled by $1/B$ and backpropagated. Gradients are accumulated over each complete block of $B$ windows and an optimizer step is taken at the end of the block. With zero-based indexing, steps therefore occur when $(t + 1) \bmod B = 0$. Model and optimizer state are retained between block boundaries, and gradients from an incomplete final block are discarded without an optimizer step. Practical and Oracle TTA use the same cadence schedule.

**Robustness variants.** Static-P95/B1 is the AdamW-$10^{-4}$ slice of the main protocol and uses the point-level source-validation P95 selector with full-AE updates on every test window. The M2N2-style variant maintains an online exponential moving mean, reconstructs the detrended current window, restores the mean before scoring, and uses the same P95 rule for Practical updates; its matched baseline and Oracle use the same detrending procedure. Both adaptive arms update the full AE.

**CANDI-style FPM selector.** False Positive Mining (FPM) is one of the two core components of CANDI (Kim et al., 2026). Our FPM-only variant transplants its score- and latent-similarity based sample-selection principle into the main full-AE adaptation protocol. The Practical point-level Static-P95 selector is replaced by a label-free whole-window selector constructed from source-validation reconstruction-score strata and proximity to frozen source-encoder latent references. Selected windows update the full AE, while the Frozen and benchmark-normal point Oracle controls are retained from the core protocol.

**CANDI-style FPM+SANA variants.** Spatiotemporally-Aware Normality Adaptation (SANA) is the second core component of CANDI. In our FPM+SANA variants, the source AE is frozen and only the input/output SANA modules are trainable. Practical TTA updates these modules on FPM-selected windows. One Oracle variant updates on every window whose benchmark labels are all normal, whereas the other preserves the FPM candidate set and removes candidates containing any benchmark-labeled anomaly.

Table B.2: Outcome counts for selector, adaptation, and cadence variants over 45 matched cells per analysis, comprising nine datasets, AdamW at learning rate $10^{-4}$, and seeds 42–46. BI denotes both improve, OOG denotes Oracle-only gain, NI denotes neither improves, and POG denotes Practical-only gain; a clear gain requires a paired AUPRC improvement over Frozen greater than $0.005$ (0.5 percentage points). For cadence $B$, gradients are accumulated over blocks of $B$ consecutive test windows before each optimizer step. Physical-entity support may differ across variants. EMA denotes exponential moving average. FPM denotes False Positive Mining and SANA denotes Spatiotemporally-Aware Normality Adaptation, the two evaluated CANDI components.

| Analysis | BI | OOG | NI | POG |
|---|---|---|---|---|
| Static-P95 / B1 | 10 | 15 | 20 | 0 |
| M2N2-style EMA detrending | 9 | 21 | 15 | 0 |
| CANDI-style FPM-only | 13 | 12 | 20 | 0 |
| FPM+SANA / all-normal-window Oracle | 7 | 18 | 20 | 0 |
| FPM+SANA / label-cleaned FPM Oracle | 7 | 1 | 37 | 0 |
| Cadence B2 | 11 | 14 | 20 | 0 |
| Cadence B4 | 12 | 13 | 20 | 0 |
| Cadence B8 | 7 | 13 | 25 | 0 |

Across these selector, adaptation, and cadence variants, the exact outcome proportions change, but both OOG and NI remain present throughout the evaluated settings.

**Benchmark-label perturbations.** Perturbations are constructed independently for each physical entity from its point-level benchmark labels. An anomaly segment is a maximal contiguous run of anomalous points in the full test sequence, independent of window boundaries. For a segment of length $L$, the boundary width used for both dilation and erosion is

$$w(L) = \begin{cases} 0, & L \leq 1, \\ \min\left\{\max(1, \lceil 0.1L \rceil), \left\lfloor \frac{L-1}{2} \right\rfloor\right\}, & L > 1. \end{cases}$$

Dilation relabels up to $w(L)$ adjacent normal points on each side of the segment as anomalous, stopping at the sequence boundaries. Erosion relabels the first and last $w(L)$ anomalous points of each segment as normal. The cap prevents complete removal of a segment; segments of length one or two are unchanged.

For false-negative contamination, we relabel

$$K = \lceil 0.1 N_{\text{anom}} \rceil$$

anomalous points as normal within each entity, where $N_{\text{anom}}$ is the number of benchmark-labeled anomalous points in that entity. Using a fixed NumPy random seed of 20260725, we randomly permute anomaly segments and consume them in that order: whole segments are relabeled while they fit within the remaining budget, and, when necessary, a contiguous block is sampled uniformly from the final partially used segment. We use one fixed realization per entity label sequence, reused across model seeds and perturbation channels.

In the mask-only channel, the perturbed labels modify only the Oracle update mask, while evaluation uses the original benchmark labels. In the evaluation-only channel, Oracle update selection uses the original benchmark labels, while Frozen, Practical, and Oracle are all evaluated against the same perturbed label vector. In the joint channel, the same perturbed vector is used both to define the Oracle update mask and to evaluate all three arms.

Table B.3: Outcome-category retention under benchmark-label perturbations on the same 45-cell AdamW-$10^{-4}$, seed-42–46 robustness grid. Boundary dilation and erosion use segment-relative 10% perturbations. For false-negative contamination, anomalous points are relabeled as normal using a 10% anomaly-point budget and the fixed realization generated with seed 20260725. Each entry reports the number of cells whose outcome category matches the corresponding unperturbed analysis. Retention denotes category agreement, not sample availability. Mask-only perturbs only the Oracle update mask; evaluation-only perturbs the evaluation labels for all three arms; joint applies the same perturbed labels to both the Oracle update mask and the evaluation of all three arms.

| Perturbation | Mask-only | Evaluation-only | Joint |
|---|---|---|---|
| Boundary dilation | 45/45 | 45/45 | 45/45 |
| Boundary erosion | 45/45 | 41/45 | 41/45 |
| False-negative contamination | 40/45 | 44/45 | 39/45 |

**Dataset-cluster uncertainty.** To characterize uncertainty without treating the 180 optimizer–seed cells as independent dataset-level observations, we use a dataset-cluster bootstrap over the nine benchmark families. We draw 10,000 bootstrap replicates by resampling datasets as clusters and recompute the corresponding outcome proportions within each replicate. This procedure gives an OOG proportion of $0.35$ with a 95% confidence interval of $[0.122,\ 0.600]$ and an NI proportion of $0.50$ with a 95% confidence interval of $[0.194,\ 0.806]$.

## C Additional Results for Section 4.3: Score-Shift Examples

**Objective–detection association.** Across the 180 main-grid cells, the immediate pre–post reconstruction-loss decrease on benchmark-labeled normal points contributing to Oracle updates has a Spearman rank correlation of $\rho = 0.605$ with the Oracle-minus-Frozen AUPRC effect. For each physical entity, we average the point-level loss decreases from immediately before to immediately after Oracle updates over the benchmark-labeled normal points contributing to those updates, and then take the median of the finite entity-level values within each dataset–optimizer–seed cell.

The objective diagnostic uses the finite physical-entity support on which the pre–post loss quantities are defined, whereas the paired detection effect uses the canonical common physical support defined in Section 3. To reflect dependence among optimizer–seed cells from the same benchmark family, uncertainty is estimated by clustering at the dataset level. The resulting 95% confidence interval is $[-0.020, 0.811]$. Thus, although the point estimate is positive, immediate reconstruction-objective improvement alone is not a reliable criterion for detection improvement.

**Full-grid score-shift association.** For each physical entity $e$, let

$$\mathcal{N}_e = \{i : y_i = 0\}, \qquad \mathcal{A}_e = \{j : y_j = 1\}.$$

Under the primary tail setting, hard normals and boundary anomalies are defined from Frozen scores as

$$\mathcal{H}_e = \left\{ i \in \mathcal{N}_e : s_i^F \geq Q_{0.85}(s_{\mathcal{N}_e}^F) \right\},$$

and

$$\mathcal{B}_e = \left\{ j \in \mathcal{A}_e : s_j^F \leq Q_{0.30}(s_{\mathcal{A}_e}^F) \right\}.$$

Tail membership therefore depends only on benchmark labels and Frozen scores. Quantile thresholds use linear interpolation, and points equal to a threshold are included.

Let

$$d_i = s_i^F - s_i^O$$

denote the Frozen-to-Oracle score decrease. We define selective compression as

$$\mathrm{SC}_e = \operatorname*{median}_{i \in \mathcal{H}_e} d_i - \operatorname*{median}_{j \in \mathcal{B}_e} d_j.$$

Thus, $\mathrm{SC}_e > 0$ means that hard-normal scores are preferentially reduced relative to boundary-anomaly scores.

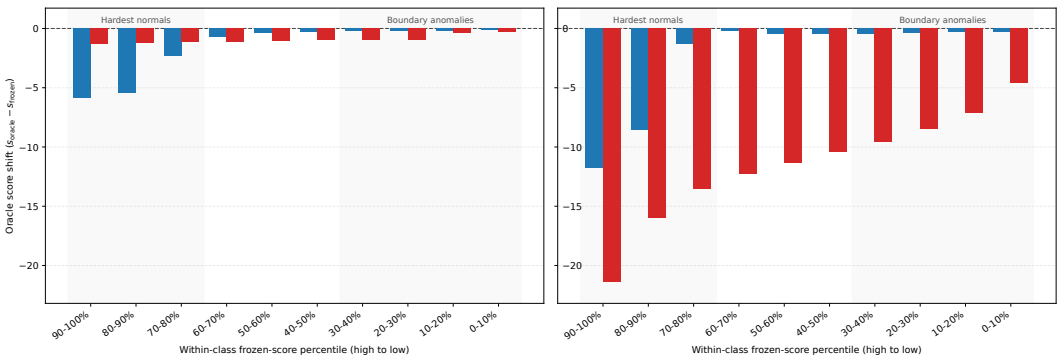



Figure C.1: Representative Oracle normal-only score-shift cases for the AE backbone with AdamW at learning rate $10^{-4}$ and seeds 42–46. Within each seed, we compute the median score shift $s_{\text{Oracle}} - s_{\text{Frozen}}$ in class-specific Frozen-score decile bins, and each bar reports the median of these values across seeds. Blue bars denote normal points and red bars denote anomalous points; more negative values indicate stronger score reduction after Oracle normal-only adaptation. PUMP (left) illustrates a gain case, where high-scoring normal points are strongly compressed while low-scoring boundary anomalies are largely preserved. Daphnet (right) illustrates a failure case, where anomalous scores also decrease and detection worsens despite reconstruction-objective improvement. These illustrative decile bins are distinct from the 15%/30% tail definition used for the systematic selective-compression and net-pair-gain analysis.

For each pair $(i, j) \in \mathcal{H}_e \times \mathcal{B}_e$, where $i$ is normal and $j$ is anomalous, define

$$C_{ij}^{M} = \mathbf{1}\left[s_j^M > s_i^M\right], \qquad M \in \{F, O\}.$$

Score ties are treated as incorrectly ordered. Net pair gain is

$$\mathrm{NPG}_e = \frac{1}{|\mathcal{H}_e||\mathcal{B}_e|} \sum_{i \in \mathcal{H}_e} \sum_{j \in \mathcal{B}_e} \left(C_{ij}^{O} - C_{ij}^{F}\right).$$

It therefore measures the change in the fraction of pairs of hard normal and boundary anomaly points for which the anomalous point is ranked above the normal point. Positive values indicate more wrong-to-correct than correct-to-wrong pair transitions after Oracle adaptation.

Both statistics are computed independently within each physical entity. Net pair gain uses the exact Cartesian product of the selected tails, without pair sampling. For each dataset–optimizer–seed cell, entity-level statistics are aggregated by the median over finite matched Frozen–Oracle entities. The finite entity support used for the score-shift statistics is not always identical to the canonical common support used for the paired detection effect, because entities with non-finite score arrays are excluded from the score-shift analysis.

Using the primary 15%/30% tail setting, we evaluate the association between Oracle-minus-Frozen AUPRC and two score-ordering statistics across the 180 dataset–optimizer–seed cells. The pooled Spearman correlations are $\rho = 0.736$ for selective compression and $\rho = 0.745$ for net pair gain. Across the nine dataset medians, the corresponding correlations are $\rho = 0.667$ and $0.728$. After ranking the detection effects and score-shift statistics within each dataset and pooling those within-dataset ranks, the associations decrease to $\rho = 0.144$ and $0.106$, respectively. These results support a primarily between-dataset/regime association rather than a universal within-dataset optimizer–seed relationship, and should be interpreted as associative rather than causal.

Figure C.1 provides representative score-shift examples for the objective–detection mismatch discussed in Section 4.3. The examples are illustrative rather than statistical. They show how oracle normal-only updates can reduce reconstruction loss while producing different effects on anomaly ranking. In PUMP, high-scoring normal points show larger score reductions than low-scoring boundary anomalies, alongside improved detection. In Daphnet, anomalous scores also decrease substantially, while detection worsens despite reconstruction-objective improvement.