DEPLOYMENT PERFORMANCE ESTIMATION FOR TIME-SERIES ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Time-series anomaly detectors can exhibit substantial instance-to-instance performance variation even within the same data source. Yet conventional benchmark evaluations summarize performance over labeled collections and provide little indication of how a given detector will perform on a particular deployment instance. We define *deployment-time performance estimation* (DPE) for time-series anomaly detection as the task of estimating a detector's instance-level performance on an unlabeled deployment instance, given its historical evaluation records. To address this challenge, we introduce TIMEDPE, which constructs a compact, label-free surrogate representation from the detector's anomaly-score sequence and the raw time series to capture performance-relevant information, and learns its relationship with detector performance from historical evaluations. Across 17 data sources and 10 anomaly detectors, TIMEDPE reduces the macro-averaged mean absolute error of instance-level VUS-PR estimation by 40.2% relative to the conventional practice of predicting each target with the mean performance from historical benchmark evaluations. Moreover, we show that the surrogate representation plays a key role in TimeDPE's sample efficiency, enabling accurate performance estimation from limited historical evaluations.

1 INTRODUCTION

Time-series anomaly detection (TSAD), one of the major downstream tasks for time-series data, has gained considerable attention in applications such as industrial monitoring, healthcare, and finance (Blázquez-García et al., 2021; Liu et al., 2024). Yet, evaluating and deploying anomaly detectors is complicated by substantial instance-to-instance performance variation, even within the same data source: an anomaly detector with strong average performance may still miss crucial anomalies or generate costly false alarms on a particular deployment instance. Figure 1 illustrates this for KMeansAD (Yairi et al., 2001) on 259 Yahoo time-series instances (Liu & Paparrizos, 2024), where instance-level VUS-PR spans nearly the full range from 0 to 1. At deployment time, what matters is an accurate estimate of performance on the current instance, which could help flag unreliable detector outputs and trigger additional review.

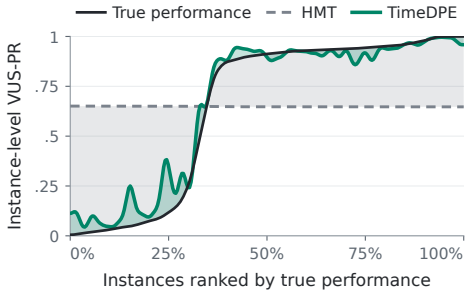


Figure 1: **Motivation for instance-level DPE.** For KMeansAD on 259 YAHOO time-series instances, true VUS-PR ranges from near 0 to near 1. Historical mean transfer (HMT) provides a constant estimate, hence missing much of this instance-level variation, whereas TIMEDPE produces instance-level performance estimates that closely track the true performance. Shading highlights the HMT–truth discrepancy.

Conventional benchmark evaluations assess detector performance on labeled historical instances, typically summarizing results using average scores or overall rankings (Paparrizos et al., 2022b; Schmidl et al., 2022; Liu et al., 2025). Such aggregate results characterize overall detector performance rather than its performance on the current instance. Figure 1 illustrates this limitation: historical

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.

mean transfer (HMT), which predicts the performance on a target instance as the mean over historical records, produces constant estimates across instances, missing the true instance-level variation. This leaves a gap between the instance-level performance that matters at deployment and the aggregate information that conventional evaluation provides. We therefore formulate *deployment-time performance estimation* (DPE) for TSAD as the task of estimating a detector's instance-level performance on an unlabeled deployment instance, given its historical evaluation records.

Performance estimation on unlabeled target data has been studied in image classification (Garg et al., 2022; Pouget et al., 2025). These methods share a paradigm in which label-free *surrogate signals* are derived from classifier outputs, such as confidence and predictive uncertainty, and used as approximate indicators of deployment performance. However, these methods cannot be applied directly to TSAD, where detector outputs are variable-length anomaly-score sequences and performance is defined over an entire time-series instance rather than at a single timestamp. By contrast, DPE has received little attention in TSAD. The closest line of work is unsupervised model selection, which ranks detectors without labels (Goswami et al., 2023; Sylligardos et al., 2023; Liu et al., 2025). However, model selection provides only a relative comparison among candidate detectors: it always recommends one model, even if every candidate performs poorly, and does not estimate the absolute performance of a given detector on a particular unlabeled instance.

To bridge this gap, we propose TIMEDPE, which reuses historical evaluation records to estimate the performance of a given detector on an unlabeled deployment instance. As previewed in Figure 1, TIMEDPE produces instance-level estimates that capture much of the performance variation missed by historical averaging. A central challenge in achieving such instance-level estimation is the limited historical supervision available for DPE in TSAD: each historically evaluated time-series instance contributes only one scalar performance observation, regardless of sequence length. For example, the data sources in TSB-AD-U contain only between 1 and 259 time-series instances (Liu & Paparrizos, 2024), providing at most a few hundred historical performance observations for DPE. TIMEDPE addresses this limited-data setting in two stages. First, it constructs a compact, label-free *surrogate representation* by combining observable signals derived from the anomaly-score sequence and raw time series to expose performance-relevant structure and thereby support sample-efficient estimation. The representation captures three complementary aspects: the distributional characteristics of the anomaly scores, such as dispersion and upper-tail concentration; their temporal organization, such as persistence and fragmentation; and the relational structure between the anomaly scores and the raw time series, which characterizes how high-score regions correspond to distinctive patterns or deviations in the observed time series (Lubba et al., 2019; Li et al., 2025; Fox, 1972; Tsay, 1988). Second, TIMEDPE learns the relationship between the surrogate representation and detector performance using a sample-efficient estimator. In our default implementation, we use Gaussian-process regression (GPR) as a regularized nonlinear estimator suitable for limited historical observations.

Our contributions are threefold:

1. **Problem formulation.** We define *deployment-time performance estimation* (DPE) for TSAD as the task of estimating a detector's instance-level performance on an unlabeled deployment instance, given its historical evaluation records.
2. **Method.** We introduce TIMEDPE, a DPE framework that constructs a compact, label-free, TSAD-specific surrogate representation and learns its relationship with detector performance under limited historical supervision.
3. **Empirical evaluation and analysis.** Across 17 data sources and 10 representative anomaly detectors, TIMEDPE reduces instance-level VUS-PR estimation error by 40.2% relative to historical mean transfer. Further analyses highlight the importance of the surrogate representation for sample-efficient estimation under limited historical supervision.

An anonymized repository containing the core implementation is available at this link.

2 RELATED WORK

The two lines of work most closely related to DPE for TSAD are unsupervised performance estimation in image classification and unsupervised model selection for TSAD.

Unsupervised performance estimation. Prior work has studied estimating classifier performance on unlabeled target data using label-free surrogate signals derived from model outputs, including confidence, entropy, logits, and predictive uncertainty (Hendrycks & Gimpel, 2017; Guillory et al., 2021; Garg et al., 2022; Deng et al., 2023; Lu et al., 2023). More recent work further combines multiple surrogate signals to improve performance estimation (Pouget et al., 2025). DPE for TSAD follows the same general principle of inferring performance from observable model behavior, but considers TSAD, where model outputs are variable-length anomaly-score sequences and performance is defined over an entire time-series instance. Consequently, surrogate signals developed for image classification do not transfer to DPE for TSAD and require a representation tailored to time-series detector outputs. Relatedly, Baudzus et al. (2023) propose an information-theoretic index of univariate time-series data as a proxy for anomaly-detection task difficulty and study its association with detector AUC, but do not estimate the realized performance of a given detector on an individual unlabeled instance.

Unsupervised model selection in TSAD. Unsupervised model selection aims to select or rank anomaly detectors and configurations without anomaly labels on the target series (Goswami et al., 2023; Sylligardos et al., 2023; Liu et al., 2025). Some meta-learning-based selectors regress the performance of each candidate detector from generic meta-features of the raw series, such as catch22, and recommend the detector with the highest predicted performance (Liu et al., 2025). Although these predictions are performance estimates, they serve only to rank candidates and are not evaluated as absolute estimates. Our catch22 + GP-RQ baseline instantiates this paradigm, which is based on raw time-series features, under the DPE setting. Related work in anomaly detection has also used surrogate signals derived from anomaly-score outputs for unsupervised hyperparameter selection (Li et al., 2025). This is conceptually close to TIMEDPE in using observable detector responses as proxies for detector quality, but the objective remains ranking or selecting configurations rather than estimating instance-level detector performance.

3 PROBLEM FORMULATION

Instance-level TSAD performance. A time-series instance is a complete time series $T = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathbb{R}^{n \times c}$, where n denotes the length of the instance and c is the number of channels; n may vary across instances. Let $f : \mathbb{R}^{n \times c} \to \mathbb{R}^n$ denote an anomaly-detection pipeline, including its model configuration and prescribed fitting and scoring procedures, which is held fixed across the considered instances. Applied to T, the detector produces the time-aligned anomaly-score sequence $f(T) \in \mathbb{R}^n$, where larger values indicate stronger evidence of anomalousness. When pointwise ground-truth anomaly labels $y \in \{0, 1\}^n$ are available, the instance-level performance of f under a prespecified evaluation metric $q : \mathbb{R}^n \times \{0, 1\}^n \to [0, 1]$ is $q(f(T), y)$.

Deployment-time performance estimation (DPE) for TSAD. For a given detector f, suppose that N historical time-series instances T_i, with ground-truth anomaly labels y_i, from the same deployment setting have previously been evaluated. Their evaluation records are

$$\mathcal{R}_f = \{(T_i,\ f(T_i),\ q(f(T_i), y_i))\}_{i=1}^{N}\,, \tag{1}$$

We consider a deployment target T_* from the same deployment setting, whose ground-truth labels y_* are unavailable.

At deployment time, T_* and its anomaly-score sequence $f(T_*)$ are observable, whereas its realized performance $q(f(T_*), y_*)$ cannot be directly evaluated. Deployment-time performance estimation (DPE) seeks an estimator h_f that uses the observable information from the unlabeled target together with the historical evaluation records to estimate this quantity:

$$\widehat{q}_* = h_f(T_*, f(T_*); \mathcal{R}_f)\,, \tag{2}$$

In practice, DPE for TSAD must accommodate deployment observations of instance-dependent length while learning their relationship to performance from only N historical scalar evaluations, where N is often small.

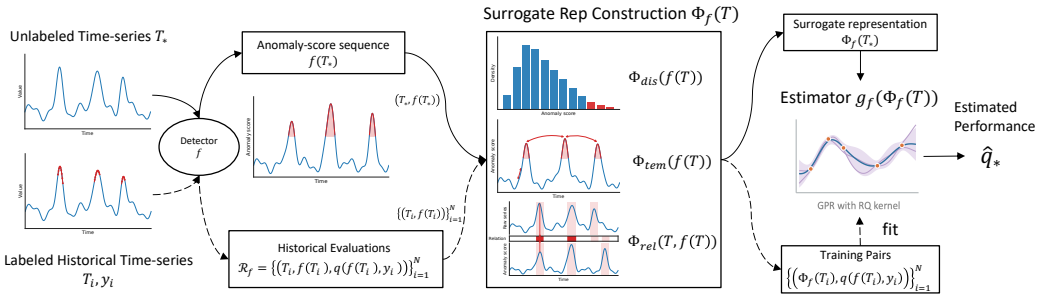


Figure 2: An overview of TIMEDPE.

4 METHODOLOGY

We propose TIMEDPE for DPE in TSAD. For a given detector f, it consists of two components:

1. A representation function $\Phi_f : \mathbb{R}^{n \times c} \to \mathbb{R}^d$ that maps a time-series instance T to a compact, fixed-dimensional surrogate representation $\Phi_f(T)$, designed to capture performance-relevant information without ground-truth labels. The surrogate representation may use the detector response $f(T)$.
2. An estimator $g_f : \mathbb{R}^d \to \mathbb{R}$ that learns from historical evaluations how the surrogate representation relates to detector performance and, for a deployment target T_*, produces $\widehat{q}_* = g_f(\Phi_f(T_*))$ without access to the target labels y_*.

TIMEDPE instantiates the predictor h_f in Equation 2 by first mapping each instance to $\Phi_f(T)$ and then applying the estimator g_f. From the historical evaluation records, g_f is fitted on

$$\{(\Phi_f(T_i),\, q(f(T_i), y_i))\}_{i=1}^{N},$$

and is then used to estimate the performance of an unlabeled deployment target. Figure 2 provides a schematic overview of the framework.

4.1 SURROGATE REPRESENTATION

Prior work on classifier reliability and unlabeled performance estimation suggests that multiple observable signals can provide complementary information about model behavior, supporting more accurate performance estimation (Hendrycks & Gimpel, 2017; Garg et al., 2022; Pouget et al., 2025). Following this work, and given the limited historical supervision available for DPE in TSAD which makes learning a representation directly from data challenging, we construct such signals from handcrafted statistical and structural descriptors of the raw time series T and the detector response $f(T)$. Specifically, we construct $\Phi_f(T)$ by concatenating three complementary families: the *distributional* characteristics of the anomaly scores Φ_{dist}, the *temporal* structure of the anomaly scores Φ_{temp}, and the *relational* structure between the raw time series and anomaly scores Φ_{rel}:

$$\Phi_f(T) = [\Phi_{\text{dist}}(f(T)), \Phi_{\text{temp}}(f(T)), \Phi_{\text{rel}}(T, f(T))]\,. \tag{3}$$

Distributional surrogate signals $\Phi_{\text{dist}}(f(T))$. Detector performance depends in part on how well anomaly scores separate anomalous from normal observations. Although this separation cannot be measured without labels, statistics of the score distribution such as dispersion, upper-tail concentration, and high-score contrasts, may still provide indirect evidence about it. Related score-distribution properties have been used for label-free model selection and hyperparameter optimization in anomaly detection (Goix, 2016; Xu et al., 2019; Li et al., 2025; Dai & Fan, 2025).

Temporal surrogate signals $\Phi_{\text{temp}}(f(T))$. The temporal organization of detector responses is relevant to TSAD performance, particularly when anomalies span contiguous intervals. Range-aware metrics explicitly account for properties such as event coverage, overlap, and fragmentation (Tatbul et al., 2018; Huet et al., 2022; Paparrizos et al., 2022a). We therefore use Φ_{temp} to characterize persistence, fragmentation, temporal dependence, and local variation in the anomaly scores, using descriptors such as run lengths, autocorrelation, and successive differences (Lubba et al., 2019).

Relational surrogate signals $\Phi_{\mathrm{rel}}(T, f(T))$. The two preceding families characterize the detector response $f(T)$ alone. However, $f(T)$ is a detector-specific transformation of T: different detectors may preserve, emphasize, or suppress different structures in the raw series, and the scores alone do not reveal whether high responses correspond to meaningful raw-series structure. Φ_{rel} therefore relates the scores back to the raw series in two complementary ways. First, drawing on residual-based characterizations of temporal deviation (Fox, 1972; Tsay, 1988; Hundman et al., 2018), it examines whether high-score regions coincide with departures from simple temporal behavior: we compute raw-series residuals under global-mean, persistence, and autoregressive reference models and summarize their coding costs inside and outside high-score regions. Second, it examines whether the score response helps describe raw-series structure, following the description-length principle that a useful explanation shortens the total description of the data (Rissanen, 1978; Grünwald, 2007): we summarize each window by its raw-series coding complexity and score level, and measure the cost of describing raw complexity given the score, as well as the net gain after also accounting for the cost of describing the score states themselves, across multiple window lengths. Complete definitions of all three families are provided in Appendix A.

4.2 Instance-level Performance Estimator

Given the surrogate representation $\Phi_f(T)$, we require an estimator g_f that maps $\Phi_f(T)$ to an estimate $\hat{q}$ of the performance of f on the time series T. Since only a limited number of historical instances are available for DPE in TSAD, this estimator must be sample-efficient. Additionally, the relationship between surrogate signals and detector performance may be nonlinear and involve interactions among signals, which the estimator should be able to capture.

Based on these requirements, we use Gaussian process regression (GPR) (Williams & Rasmussen, 2006) as the default estimator, as it can model nonlinear relationships while remaining suitable for regression on limited samples. Among the kernels compared in Table 2 of Appendix C, the Rational Quadratic, Matérn, and RBF kernels (GP-RQ, GP-Matérn, GP-RBF) perform very closely; we adopt GP-RQ by default. To reduce estimator sensitivity to differences in scale and distributional shape across signals, each of the 29 surrogate signals is rank-transformed through its empirical CDF over the historical instances and then standardized before fitting.

5 Experiments

We evaluate TimeDPE by answering the following experimental questions:

- **Q1:** How effectively does TimeDPE estimate instance-level detector performance across data sources and detectors?
- **Q2:** How sample-efficient is TimeDPE as the amount of historical supervision varies?
- **Q3:** How does the surrogate representation contribute to DPE performance, and what makes complementary surrogate signals effective?

We additionally present a controlled monitoring use case in which variations in electromagnetic disturbance (EMD) induce instance-level shifts in the monitored signals, which in turn affect detector performance. We evaluate whether TimeDPE can estimate this performance variation, demonstrating its potential to improve the deployment reliability of anomaly detectors in a real-world setting.

5.1 Experimental Setup

Benchmark data sources and detector pool. We use the TSB-AD-U benchmark (Liu & Paparrizos, 2024) and treat each data source as a collection of related time-series instances and within-source variation as a proxy for deployment-time variation. After excluding sources with fewer than four instances, the benchmark contains 860 time-series instances from 17 data sources. We instantiate the detector f from Section 3 with ten TSAD methods spanning subspace, clustering, matrix-profile, shape-based, polynomial, convolutional, reconstruction-based, and zero-shot foundation-model approaches, and retain their default hyperparameters, fitting and scoring pipelines from TSB-AD. SubPCA (Aggarwal, 2016), KMeansAD (Yairi et al., 2001), MatrixProfile (Yeh et al., 2016), KShapeAD

(Paparrizos & Gravano, 2015), POLY (Li et al., 2007), and HydraAD (Carpentier et al., 2026) operate on the complete unlabeled series; USAD (Audibert et al., 2020) and CNN (Munir et al., 2018) use the benchmark-designated training prefix before scoring the complete series; and MOMENT (Goswami et al., 2024) and TSPulse (Ekambaram et al., 2026) operate in a zero-shot manner. Pairing the 17 data sources with the 10 detectors yields 170 pairs of a data source and a detector (source–detector pairs for short), with 8 to 259 instances per pair.

Deployment evaluation protocol. Within each source–detector pair containing m instances, we perform exact leave-one-instance-out evaluation. Each instance in turn serves as the unseen deployment target T_* in Equation 2, while the remaining $m-1$ instances form the historical evaluation records $\mathcal{R}_f$ used to fit the TIMEDPE estimator, as defined in Equation 1. Historical labels enter only through their recorded performance values, whereas the target labels y_* remain unavailable during fitting and prediction and are used only to assess prediction accuracy. Repeating this procedure over all source–detector pairs produces 8,600 deployment-performance predictions.

Evaluation metrics. We instantiate q as VUS-PR (Paparrizos et al., 2022a), a threshold-independent and range-aware TSAD metric. DPE accuracy is measured by the MAE between estimated and true performance within each source–detector pair, followed by macro-averaging across the 170 pairs so that each pair contributes equally. We refer to this as the *macro-averaged MAE*. We additionally report relative gain and win/loss counts with respect to HMT, Acc@.10, the fraction of predictions within 0.10 of the true performance, and substantial overestimation (Over), the fraction of predictions that overestimate true performance by more than 0.10; overestimation is of particular concern in deployment, as it can lead practitioners to over-trust an unreliable detector and miss true anomalies. Results with AUC-PR, AUC-ROC, and range-based F_1 (Tatbul et al., 2018) as alternative instantiations of q also show consistent improvements over HMT; see Table 5 in Appendix C.

5.2 COMPARISON METHODS

To answer the three questions above, we evaluate six comparison methods against the default TIMEDPE configuration, which uses the 29-dimensional surrogate representation with GP-RQ. Historical mean transfer (HMT) serves as the conventional baseline. The other comparison methods vary along three dimensions: replacing the surrogate representation while keeping GP-RQ (Best single and catch22); using a surrogate representation learned from data (JNSR-DKL and TinyMLP); and replacing the GP-RQ regressor (MLP).

HMT. Historical mean transfer estimates a target instance's performance as the mean performance of the historical instances in the same source–detector pair. It serves as the baseline corresponding to conventional evaluation by transferring aggregate historical performance to the current instance.

Best single + GP-RQ. This baseline retrospectively selects the best-performing signal from the 29 single-signal GP-RQ predictors and uses the same GP-RQ estimator as TIMEDPE. It tests how much is gained by combining complementary surrogate signals rather than relying on a single indicator.

catch22 + GP-RQ. We replace the surrogate representation Φ_f of TIMEDPE with the generic catch22 representation (Lubba et al., 2019) extracted from the raw time series. catch22 has been used as a meta-feature representation for model selection in automated TSAD (Liu et al., 2025), making it a natural generic baseline for our 29-dimensional surrogate representation.

JNSR-DKL. Joint Neural Surrogate Representation with Deep Kernel Learning (JNSR-DKL) replaces the handcrafted surrogate representation learned jointly with a Gaussian-process objective, following deep kernel learning (Wilson et al., 2016). It tests whether the representation can instead be learned from limited historical supervision.

TinyMLP. Tiny Multilayer Perceptron (TinyMLP) uses a small-capacity MLP to directly regress detector performance from the raw time series and anomaly-score sequence, without an explicit surrogate representation. It provides a direct end-to-end baseline for assessing the difficulty of learning the DPE mapping from limited historical supervision without task-specific surrogate construction.

TIMEDPE + MLP. This variant retains the 29-dimensional surrogate representation but replaces GP-RQ with a small MLP containing one hidden layer of four tanh units, allowing nonlinear relationships and interactions among surrogate signals. It tests the difficulty of learning the surrogate-to-performance mapping under limited historical supervision when the representation is fixed.

Table 1: **Deployment performance estimation on TSB-AD-U.** Results are reported over 170 source–detector pairs using VUS-PR. Brackets show 95% percentile bootstrap confidence intervals obtained from 20,000 resamples of the 170 source–detector pairs with replacement. Gain and W/L are measured relative to HMT. Acc@.10 is the fraction of predictions within 0.10 of the true performance ($|\hat{q} - q| \leq 0.10$), while Over is the fraction that overestimate it by more than 0.10.

Method	Macro MAE ↓	Gain	W/L	Acc@.10 ↑	Over ↓
Historical mean transfer (HMT)	0.1637 [.148,.180]	—	—	40.3%	30.5%
Best single + GP-RQ (1 D)	0.1314 [.119,.145]	19.7%	130/40	62.3%	17.1%
catch22 + GP-RQ (raw, 22 D)	0.1298 [.115,.146]	20.7%	124/46	64.2%	16.4%
JNSR-DKL (29 D)	0.1359 [.121,.152]	17.0%	106/64	64.6%	17.5%
TinyMLP (raw + score)	0.1554 [.142,.169]	5.1%	87/83	58.1%	18.7%
TimeDPE (29 D + MLP)	0.1073 [.096,.119]	34.5%	129/41	68.2%	15.4%
TimeDPE (29 D + GP-RQ)	**0.0979** [.088,.108]	**40.2%**	**151/19**	**71.9%**	**12.7%**

Additional implementation details for comparison methods are provided in Appendix B.2.

5.3 Results

Q1: Overall effectiveness.

Table 1 summarizes the overall DPE performance. Compared with HMT, the default TimeDPE reduces macro-averaged MAE from 0.1637 to 0.0979, corresponding to a 40.2% relative improvement. It also improves Acc@.10 from 40.3% to 71.9% and reduces substantial overestimation from 30.5% to 12.7%. Among all comparison methods, TimeDPE achieves the strongest overall performance across the reported evaluation metrics. Figure 3 provides an instance-level view of this improvement. Overall, TimeDPE predictions cluster more tightly around the diagonal representing perfect estimation than HMT predictions do. The improvement is especially important in the low-performance region, where TimeDPE is able to identify poorly performing detector–instance pairs that HMT, by construction, cannot. Appendix Tables 3 and 4 further show that the improvement is consistent across detectors and data sources. TimeDPE is the only method with positive aggregate gain over HMT for all 10 detectors and all 17 data sources, with gains ranging from 32.4% to 53.6% across detectors and from 10.9% to 78.4% across data sources.

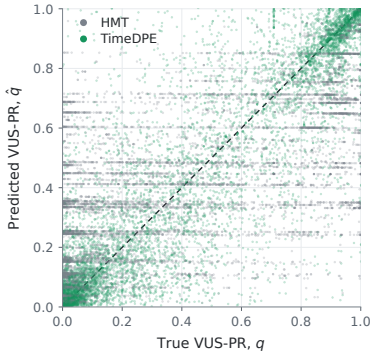


Figure 3: Predicted versus true instance-level VUS-PR over 8,600 deployment targets. The dashed diagonal denotes perfect estimation.

Q2: Learning with limited historical supervision.

To assess sample efficiency, we vary the number of historical instances available for fitting and compare the resulting learning curves. Figure 4 shows representative small-, medium-, and large-scale data sources—LTDB ($N = 9$), Opportunity ($N = 29$), and UCR ($N = 228$), respectively. A common pattern across three regimes is that both TimeDPE variants converge smoothly and stably as historical instances accumulate, in contrast to the fluctuating behavior of several alternatives. On the LTDB, this advantage is not yet apparent with only one or two historical instances, where TimeDPE performs comparably to the other methods; as more instances become available, TimeDPE separates from the alternatives and reaches the lowest error. This separation emerges earlier and is more pronounced on Opportunity and UCR, where TimeDPE is consistently among the lowest-error methods across nearly the entire curve. Additionally, Table 4 shows that both TimeDPE variants achieve positive gain on every data source with fewer than 10 instances, whereas every alternative method exhibits negative gain on at least one source.

We further examine the sensitivity of TimeDPE to the choice of estimator, given the fixed surrogate representation. The MLP variant is generally comparable to GP-RQ, but in extremely data-limited cases such as IOPS and SVDB, its gains are only -0.75% and 1.96%, respectively, whereas GP-RQ

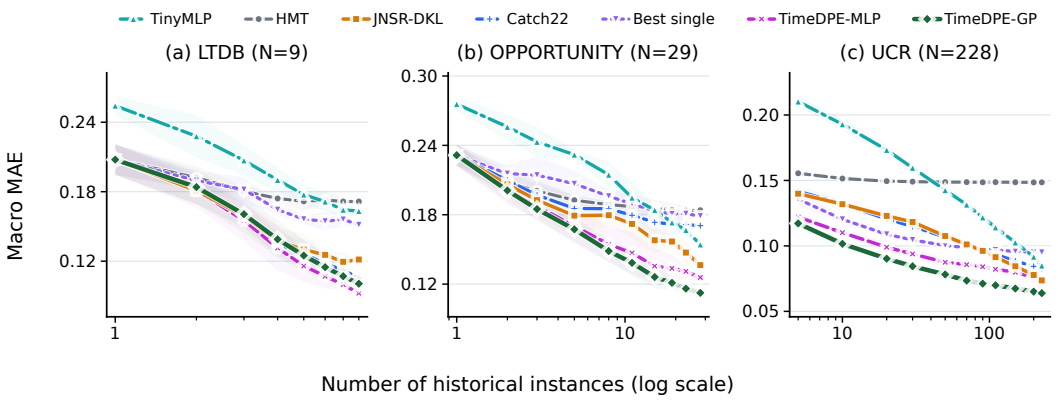
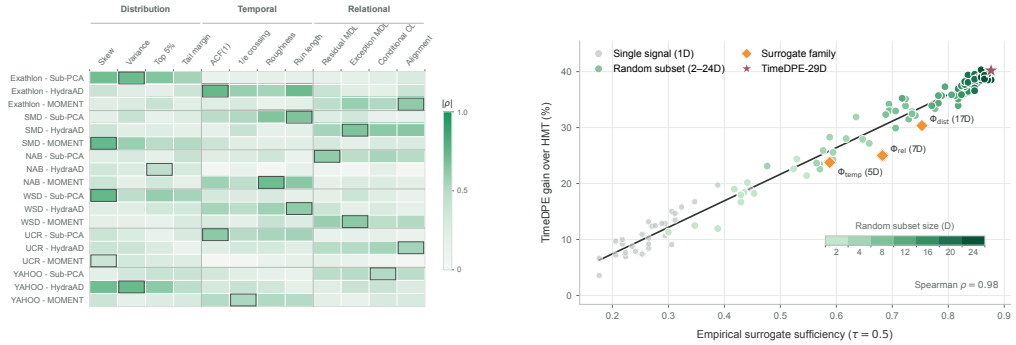


Figure 4: **Learning under limited historical supervision.** The x-axis denotes the number of historical instances available for fitting, and the y-axis reports macro-averaged MAE over the ten detectors for each data source. Detailed learning-curve construction settings are provided in Appendix B.3.



(a) Pair-level associations **(b)** Representation sufficiency vs. DPE gain

Figure 5: **Surrogate representation analysis.** **(a)** Pair-level association strength (absolute Spearman correlation) between each surrogate signal and true detector performance; rows denote source–detector pairs, columns denote surrogate signals, and outlined cells mark the strongest signal per pair. **(b)** Empirical surrogate sufficiency ($\tau = 0.5$) vs. TIMEDPE gain over HMT, across all 117 evaluated representations.

achieves gains of 11.64% and 10.88%. Across the 17 data sources, GP-RQ achieves the best result on 11, while the MLP variant does so on 5. Nevertheless, their overall gains remain relatively close, at 40.2% for GP-RQ and 34.5% for MLP. The broader estimator comparison in Appendix Table 2 supports the same pattern: with the 29-dimensional surrogate representation fixed, several estimators achieve strong performance, and even linear Ridge attains an overall gain of 32.4%. Overall, these results indicate that the surrogate representation substantially reduces the difficulty of learning the surrogate-to-performance mapping, thereby contributing to the sample efficiency of TIMEDPE.

Q3: Role and complementarity of the surrogate representation.

The learning curves in Figure 4 suggest that representation choice plays an important role in the sample efficiency observed in Q2: methods using different representations exhibit substantially different learning behavior, whereas the two TIMEDPE variants with the same 29-dimensional representation remain relatively close despite using different estimators. To examine which surrogate signals carry performance-relevant information, we compute, within each source–detector pair, the absolute Spearman correlation between each surrogate signal and the true instance-level detector performance across instances. Figure 5(a) visualizes these pair-level associations for selected source–detector pairs, with outlined cells marking the strongest association within each pair. The most strongly associated signal varies substantially across pairs and spans all three signal families, indicating that no single signal is uniformly informative. This complementarity is also reflected in the surrogate-

family ablations in Table 2, where the full representation achieves lower macro-averaged MAE than every single-family and two-family variant. To explore how the surrogate representation relates to TIMEDPE performance, we define *empirical surrogate sufficiency* S_τ as the fraction of source–detector pairs for which a representation contains at least one signal whose absolute Spearman correlation with true performance is at least τ. We use $\tau = 0.5$ in the main analysis; Appendix B.4 provides the formal definition and a sensitivity analysis for $\tau \in \{0.6, 0.7, 0.8, 0.9\}$. Across all 117 evaluated representations shown in Figure 5(b) — individual signals, random subsets of varying size, the three surrogate families, and the complete representation — S_τ is strongly associated with gain over HMT ($\rho = 0.98$). Together, these results answer Q3: different surrogate signals are informative for different source–detector pairs, and their combination yields a representation that is broadly informative across conditions. This is reflected in both the family ablations and the observed relationship between surrogate sufficiency and TimeDPE gain.

Failure analysis. The link between surrogate informativeness and DPE accuracy is also reflected in the failure cases of TIMEDPE. TIMEDPE underperforms HMT on 19 of the 170 source–detector pairs. Five of these are MGAB pairs where HMT already achieves an MAE of only 0.001–0.003, leaving little room for improvement. All 19 failures occur in pairs with at most 20 instances. However, limited history alone does not explain these failures: within this low-instance regime, successful pairs exhibit substantially stronger surrogate–performance associations than failed pairs (median maximum absolute Spearman correlation 0.806 vs. 0.583). Thus, these results further emphasize the importance of a surrogate representation that captures performance-relevant information.

5.4 USE CASE: MONITORING UNDER ELECTROMAGNETIC DISTURBANCES

The benchmark used in our main experiments treats each data source as a deployment setting and its instances as observations from that setting, without assuming a specific source of instance-level variation. In this EMD-inspired use case, motivated by SDR-based electromagnetic-disturbance monitoring (Xue et al., 2026), varying background interference provides an explicit source of such variation. We consider a monitoring task in which the underlying anomaly-detection objective remains unchanged, while different interference conditions alter the observed time-series instances and can affect the performance of a given detector. To realize this setting, we generate synthetic device signals with injected anomalies and synthetic background interference, add them together digitally, and pass the combined signal through a transmit–receive link formed by two ADALM-PLUTO radios (see Appendix D for the full setup). Across different interference configurations and independent realizations, this procedure produces 240 time-series instances. We use the same ten detectors and evaluation protocol as in the main experiments. Across the ten detectors, the macro-averaged MAE decreases from 0.1360 for HMT to 0.0524 for TIMEDPE, corresponding to a 61.4% reduction (see Table 6 for full results). These results show that TIMEDPE can capture substantial instance-level performance variation in this controlled monitoring setting, illustrating its potential to support reliability assessment of anomaly detectors under changing interference conditions.

6 CONCLUSION

Strong aggregate benchmark performance does not ensure that a time-series anomaly detector performs well on a particular deployment instance. We formulate instance-level deployment-time performance estimation (DPE) to address this gap and introduce TIMEDPE, which learns the relationship between a complementary surrogate representation and detector performance from limited historical evaluation records, without requiring target labels. Across 170 source–detector pairs, TIMEDPE reduces the macro-averaged MAE of instance-level VUS-PR estimation by 40.2% relative to historical mean transfer. A controlled EMD-motivated monitoring use case further illustrates the potential of TIMEDPE for reliability assessment in real-world settings.

Our experiments and analyses further highlight the importance of the surrogate representation for sample-efficient DPE. In particular, the effectiveness of TIMEDPE depends on whether the representation captures performance-relevant information for the deployment setting, suggesting that designing domain-specific surrogate representations will be key to applying TIMEDPE to particular application domains. The 29 surrogate signals considered in this work, spanning three complementary families, provide a useful starting point for such domain-specific representation design.

AI USE STATEMENT

In this work, we used generative AI tools to provide feedback on research methodology and experimental design, assist with method implementation, clean and reformat datasets, and support the interpretation of experimental results. We did not use generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, propose or refine hypotheses, or assist with translation. Providing critical ingredients for proving mathematical claims, assisting with the writing of proofs, and supporting qualitative or thematic data analysis are not applicable to this work.

Additionally, we used generative AI tools to assist with creating or modifying scientific figures using Python, suggest experimental parameters, create or edit software code, draft and edit parts of the manuscript, summarize and analyze existing literature, support brainstorming and information search, identify relevant literature, improve the readability of the manuscript, and format references.

We reviewed all AI-assisted work. AI-assisted code and data-processing procedures were inspected, executed, and verified by the authors. AI-assisted manuscript text, literature summaries, figures, experimental suggestions, and interpretations of results were checked against the underlying experiments, source material, and relevant literature. The authors independently evaluated the scientific validity of all analyses, interpretations, and conclusions and made all final research decisions. We take responsibility for the final content of this work, including text, claims, code, analyses, figures, or other artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies deployment-time performance estimation for time-series anomaly detection. The main experiments use the publicly available TSB-AD benchmark (Liu & Paparrizos, 2024), which consists of previously released time-series collections and involves no human subjects and no personally identifiable information. The EMD monitoring use case (Appendix D) uses synthetic signals transmitted over a coaxial link between two software-defined radios; no over-the-air emission and no external device are involved, and the results are exploratory rather than evidence of industrial EMC compliance. TIMEDPE produces estimates of detector performance, not detection decisions; because these estimates are imperfect, they should support rather than replace human review when anomaly detectors are deployed in safety- or cost-critical monitoring settings. We declare no conflicts of interest.

REPRODUCIBILITY STATEMENT

The paper and appendix specify all components needed to reproduce our experiments, and the accompanying code provides the exact configurations. The problem setting and the deployment evaluation protocol are defined in Sections 3 and 5.1; the benchmark, detector pool, and evaluation metrics are listed in Section 5.1. The complete definitions of the 29-dimensional surrogate representation are given in Appendix A, and the TIMEDPE preprocessing, target transformation, and prediction procedure in Appendix B.1. The main comparison methods are described in Appendix B.2, and the learning-curve and surrogate-sufficiency protocols in Appendices B.3 and B.4. The data-generation and acquisition procedure for the EMD-motivated use case is described in Appendix D. The main results use exact leave-one-instance-out prediction, so no train/test split sampling is involved; all randomized components (learning-curve history sampling, random surrogate subsets, and bootstrap intervals) use fixed seeds. An anonymized repository containing the full implementation, including estimator hyperparameters and the configurations of all compared estimators, is available at this link.

REFERENCES

Charu C Aggarwal. An introduction to outlier analysis. In *Outlier analysis*, pp. 1–34. Springer, 2016.

Analog Devices. ADALM-PLUTO: Software-defined radio active learning module. `https://www.analog.com/en/resources/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html`, n.d.a. Accessed September 11, 2026.

Analog Devices. Analog devices hardware python interfaces: pyadi-iio documentation. https://analogdevicesinc.github.io/pyadi-iio/, n.d.b. Accessed September 11, 2026.

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

Arn Baudzus, Bin Li, Adnane Jadid, and Emmanuel Muller. On model performance estimation in time series anomaly detection. In *Proceedings of the 2023 6th International Conference on Computational Intelligence and Intelligent Systems*, pp. 106–117, 2023.

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.

Louis Carpentier, Laurens Devos, Wannes Meert, and Mathias Verbeke. HydraAD: Exceptionally fast and accurate time series anomaly detection using competing random kernels. *Data Mining And Knowledge Discovery*, 2026.

Wei Dai and Jicong Fan. AutoUAD: Hyper-parameter optimization for unsupervised anomaly detection. In *International Conference on Learning Representations*, volume 2025, pp. 76859–76887, 2025.

Weijian Deng, Yumin Suh, Stephen Gould, and Liang Zheng. Confidence and dispersity speak: Characterizing prediction matrix for unsupervised accuracy estimation. In *International Conference on Machine Learning*, pp. 7658–7674. PMLR, 2023.

Vijay Ekambaram, Subodh Kumar, Arindam Jati, Sumanta Mukherjee, Tomoya Sakai, Pankaj Dayama, Wesley Gifford, and Jayant Kalagnanam. TSPulse: Tiny pre-trained models with disentangled representations for rapid time-series analysis. In *International Conference on Learning Representations*, volume 2026, pp. 62090–62133, 2026.

Anthony J Fox. Outliers in time series. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 34(3):350–363, 1972.

Saurabh Garg, Sivaraman Balakrishnan, Zachary Lipton, Behnam Neyshabur, and Hanie Sedghi. Leveraging unlabeled data to predict out-of-distribution performance. In *International Conference on Learning Representations (ICLR)*, 2022.

Nicolas Goix. How to evaluate the quality of unsupervised anomaly detection algorithms? *arXiv preprint arXiv:1607.01152*, 2016.

Mononito Goswami, Cristian Challu, Laurent Callot, Lenon Minorics, and Andrey Kan. Unsupervised model selection for time-series anomaly detection. *International Conference on Learning Representations*, 2023.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In *International Conference on Machine Learning*, 2024.

Peter D Grünwald. *The minimum description length principle*. The MIT press, 2007.

Devin Guillory, Vaishaal Shankar, Sayna Ebrahimi, Trevor Darrell, and Ludwig Schmidt. Predicting with confidence on unseen distributions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1134–1144, 2021.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*, 2017.

Alexis Huet, Jose Manuel Navarro, and Dario Rossi. Local evaluation of time series anomaly detection algorithms. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 635–645, 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.

Zhi Li, Hong Ma, and Yongbing Mei. A unifying method for outlier and change detection from data streams based on local polynomial fitting. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 150–161. Springer, 2007.

Zhong Li, Yuhang Wang, and Matthijs van Leeuwen. Towards automated self-supervised learning for truly unsupervised graph anomaly detection. *Data Mining and Knowledge Discovery*, 39(5): 44, 2025.

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, Paul Boniol, Themis Palpanas, and John Paparrizos. Time-series anomaly detection: Overview and new trends. *Proceedings of the VLDB Endowment (PVLDB)*, 17(12):4229–4232, 2024.

Qinghua Liu, Seunghak Lee, and John Paparrizos. TSB-AutoAD: Towards automated solutions for time-series anomaly detection. *Proceedings of the VLDB Endowment*, 18(11):4364–4379, 2025.

Yuzhe Lu, Zhenlin Wang, Runtian Zhai, Soheil Kolouri, Joseph Campbell, and Katia Sycara. Predicting out-of-distribution error with confidence optimal transport. *arXiv preprint arXiv:2302.05018*, 2023.

Carl H. Lubba, Sarab S. Sethi, Philip Knaute, Simon R. Schultz, Ben D. Fulcher, and Nick S. Jones. catch22: CAnonical Time-series CHaracteristics: Selected through highly comparative time-series analysis. *Data Mining and Knowledge Discovery*, 33(6):1821–1852, 2019. doi: 10.1007/s10618-019-00647-x.

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.

John Paparrizos and Luis Gravano. k-Shape: Efficient and accurate clustering of time series. In *Proceedings of the 2015 ACM SIGMOD international conference on management of data*, pp. 1855–1870, 2015.

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

John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S Tsay, Themis Palpanas, and Michael J Franklin. TSB-UAD: An end-to-end benchmark suite for univariate time-series anomaly detection. *Proc. VLDB Endow.*, 15(8):1697–1711, 2022b.

Angéline Pouget, Mohammad Yaghini, Stephan Rabanser, and Nicolas Papernot. Suitability filter: A statistical framework for classifier evaluation in real-world deployment settings. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267. PMLR, 2025.

Jorma Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978.

Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. Anomaly detection in time series: A comprehensive evaluation. *Proc. VLDB Endow.*, 15(9):1779–1797, 2022. doi: 10.14778/3538598.3538602.

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

Nesime Tatbul, Tae Jun Lee, Stan Zdonik, Mejbah Alam, and Justin Gottschlich. Precision and recall for time series. *Advances in neural information processing systems*, 31, 2018.

Ruey S Tsay. Outliers, level shifts, and variance changes in time series. *Journal of forecasting*, 7 (1):1–20, 1988.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. Deep kernel learning. In *Artificial intelligence and statistics*, pp. 370–378. PMLR, 2016.

Zekun Xu, Deovrat Kakde, and Arin Chaudhuri. Automatic hyperparameter tuning method for local outlier factor, with applications to anomaly detection. In *2019 IEEE International Conference on Big Data (Big Data)*, pp. 4201–4207. IEEE, 2019.

Xinting Xue, Miriam Gonzalez-Atienza, Tiago Nunes, Tim Claeys, Hugo Gamboa, and Davy Pissoort. Design and evaluation of an SDR-based platform for monitoring electromagnetic disturbances in a biosignals acquisition device. *IEEE Transactions on Electromagnetic Compatibility*, 68(4):813–824, 2026. doi: 10.1109/TEMC.2026.3697853.

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

Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. Matrix Profile I: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In *2016 IEEE 16th international conference on data mining (ICDM)*, pp. 1317–1322. Ieee, 2016.

A Full Surrogate Signal Representation

This section provides details of the 29 surrogate signals introduced in Section 4.1. For a univariate time-series instance $T = (x_1, \ldots, x_n)$, let

$$\mathbf{s} = f(T) = (s_1, \ldots, s_n) \in \mathbb{R}^n$$

denote the anomaly-score sequence produced by detector f. Sequence lengths may vary across instances.

The representation is extracted solely from the observable pair $(T, \mathbf{s})$. Ground-truth labels y and the corresponding performance value $q(f(T), y)$ are not used in any surrogate signal. The representation comprises three families:

$$\Phi_f(T) = [\Phi_{\text{dist}}(\mathbf{s}), \Phi_{\text{temp}}(\mathbf{s}), \Phi_{\text{rel}}(T, \mathbf{s})] \in \mathbb{R}^{29},$$

where

$$\begin{aligned}
\Phi_{\text{dist}}(\mathbf{s}) &= (S_{01}, \ldots, S_{10}, C_{01}, \ldots, C_{07}) \in \mathbb{R}^{17}, \\
\Phi_{\text{temp}}(\mathbf{s}) &= (T_{01}, \ldots, T_{05}) \in \mathbb{R}^{5}, \\
\Phi_{\text{rel}}(T, \mathbf{s}) &= (A_{01}, \ldots, A_{07}) \in \mathbb{R}^{7}.
\end{aligned}$$

The three families respectively characterize anomaly-score distributional characteristics, anomaly-score temporal structure, and raw–score relational structure. The coordinates are concatenated in the order shown above.

A.1 Distributional surrogate signals

The distributional family contains 17 signals that characterize the anomaly-score distribution independently of temporal order. Inspired by prior work using score-distribution and upper-tail properties as label-free indicators of anomaly-detector behavior (Dai & Fan, 2025), it consists of ten marginal score statistics (S01–S10) describing score level, dispersion, upper-tail magnitude, and distributional shape, and seven concentration and contrast signals (C01–C07) describing how detector response is distributed across timestamps.

Marginal score statistics (S01–S10). The first ten signals summarize marginal properties of the anomaly-score sequence, including location, dispersion, range, upper-tail magnitude, and distributional shape. Let

$$\bar{s} = \frac{1}{n}\sum_{t=1}^{n} s_t, \qquad m_k = \frac{1}{n}\sum_{t=1}^{n}(s_t - \bar{s})^k,$$

and let $Q_\alpha(\mathbf{s})$ denote the α-quantile of $\mathbf{s}$. The ten statistics are defined as follows:

Signal	Statistic	Definition
S_{01}	Mean	$\bar{s}$
S_{02}	Median	$Q_{0.50}(\mathbf{s})$
S_{03}	Variance	m_2
S_{04}	Interquartile range	$Q_{0.75}(\mathbf{s}) - Q_{0.25}(\mathbf{s})$
S_{05}	Minimum	$\min_t s_t$
S_{06}	Maximum	$\max_t s_t$
S_{07}	90th percentile	$Q_{0.90}(\mathbf{s})$
S_{08}	95th percentile	$Q_{0.95}(\mathbf{s})$
S_{09}	Skewness	$m_3/m_2^{3/2}$
S_{10}	Excess kurtosis	$m_4/m_2^2 - 3$

The first eight statistics describe the score distribution on its original scale, while S_{09} and S_{10} characterize its shape. For a constant score sequence, S_{09} and S_{10} are set to zero.

Score concentration and contrast (C01–C07). The remaining seven distributional signals are motivated by prior work that uses properties of the anomaly-score distribution for unsupervised hyperparameter selection (Li et al., 2025). They characterize how strongly the detector response is concentrated among high-scoring timestamps and how clearly the upper tail is separated from the rest of the score distribution.

We first robustly standardize the score sequence within each instance, using the median as center and a robust scale estimate, and clip the result to a bounded range:

$$\widetilde{s}_t = \mathrm{clip}\left(\frac{s_t - \mathrm{median}(\mathbf{s})}{\sigma_s}, -8, 8\right),$$

where σ_s is a within-instance scale estimate based primarily on the interquartile range, with fallbacks for degenerate cases. We then convert the standardized responses into normalized score mass,

$$p_t = \frac{\exp(\widetilde{s}_t)}{\sum_{j=1}^{n}\exp(\widetilde{s}_j)}, \qquad \sum_{t=1}^{n} p_t = 1.$$

Here, p_t represents the relative allocation of detector response across timestamps rather than an anomaly probability.

Peak and top-tail concentration (C01–C02). Let $k = \lceil 0.05n \rceil$ and let $\mathcal{I}_k$ denote the indices of the k largest values of p_t. We define

$$C_{01} = \frac{n\max_t p_t - 1}{n-1}, \qquad C_{02} = \frac{\sum_{t\in\mathcal{I}_k} p_t - k/n}{1 - k/n}.$$

C_{01} describes concentration at the single strongest response, whereas C_{02} summarizes the mass assigned collectively to the highest-scoring 5% of timestamps.

Global concentration (C03–C05). We further summarize how score mass is distributed across the complete instance using Shannon entropy, Rényi-2 entropy, and Hoyer sparsity:

$$C_{03} = -\frac{\sum_{t=1}^{n} p_t \log p_t}{\log n},$$

$$C_{04} = -\frac{\log\left(\sum_{t=1}^{n} p_t^2\right)}{\log n},$$

$$C_{05} = \frac{\sqrt{n} - \|\mathbf{p}\|_1/\|\mathbf{p}\|_2}{\sqrt{n}-1}.$$

The entropy measures decrease as the response becomes more concentrated, whereas Hoyer sparsity increases.

Score energy (C06). Aggregate prominence of large standardized responses is summarized by

$$C_{06} = -\,\mathrm{logsumexp}(\widetilde{\mathbf{s}}) + \log n = -\log\left(\frac{1}{n}\sum_{t=1}^{n}\exp(\widetilde{s}_t)\right).$$

Upper-tail margin (C07). Let

$$\widetilde{s}_{(1)} \geq \widetilde{s}_{(2)} \geq \cdots \geq \widetilde{s}_{(n)}$$

denote the standardized scores in descending order, and define

$$k_7 = \min\left\{\max\left(1, \lfloor 0.05n \rfloor\right), \left\lfloor \frac{n}{2} \right\rfloor\right\}.$$

We define

$$C_{07} = \frac{1}{k_7}\sum_{j=1}^{k_7}\widetilde{s}_{(j)} - \frac{1}{k_7}\sum_{j=k_7+1}^{2k_7}\widetilde{s}_{(j)}.$$

This signal describes the separation between the strongest detector responses and the immediately following part of the upper tail.

A.2 TEMPORAL SURROGATE SIGNAL

The temporal family contains five signals that characterize the organization of detector responses over time. Inspired by classical time-series descriptors and range-aware evaluation in TSAD (Lubba et al., 2019; Tatbul et al., 2018; Huet et al., 2022), the signals describe local dependence, temporal variation, and the persistence and fragmentation of high-score regions.

All five signals use the standardized score sequence $\widetilde{\mathbf{s}}$ defined above. Let

$$b_t = \widetilde{s}_t - \frac{1}{n}\sum_{j=1}^{n} \widetilde{s}_j,$$

and define the normalized autocorrelation at lag ℓ as

$$\rho(\ell) = \frac{\sum_{t=1}^{n-\ell} b_t b_{t+\ell}}{\sum_{t=1}^{n} b_t^2}.$$

Temporal dependence and variation (T01–T03). Lag-one autocorrelation captures local dependence between adjacent responses,

$$T_{01} = \rho(1).$$

To summarize longer-range dependence, let

$$\ell_* = \min\left\{\ell \geq 1 : \rho(\ell) < e^{-1}\right\},$$

and define

$$T_{02} = \frac{\ell_*}{n-1},$$

with $T_{02} = 1$ when no such crossing occurs. Successive variation is measured by

$$T_{03} = \frac{1}{n-1}\sum_{t=1}^{n-1} \left|\widetilde{s}_{t+1} - \widetilde{s}_t\right|.$$

High-score persistence and fragmentation (T04–T05). Define the high-score indicator

$$h_t = \mathbb{I}\left[\widetilde{s}_t > Q_{0.90}(\widetilde{\mathbf{s}})\right],$$

and let $\mathcal{R}_h$ denote the set of contiguous high-score runs. The longest-run ratio is

$$T_{04} = \frac{\max_{r\in\mathcal{R}_h} |r|}{n},$$

while fragmentation is

$$T_{05} = \frac{|\mathcal{R}_h|}{\sum_{t=1}^{n} h_t}.$$

Thus, T_{04} describes the persistence of the longest high-response region, whereas T_{05} describes how dispersed high responses are across separate regions. If no high-score timestamp exists, T_{04} and T_{05} are set to zero. For a constant score sequence, all five temporal signals are defined as zero.

A.3 RAW–SCORE RELATIONAL SIGNALS

The relational family relates the anomaly scores back to the raw series, addressing the two questions introduced in Section 4.1: whether high-score regions coincide with departures from simple temporal behavior (A01–A03), and whether the score response helps describe raw-series structure (A04–A07). Both constructions use coding costs to summarize how easily raw-series residuals or window-level complexity states can be described. For both, we median-center the raw values and scores separately and divide each by its median absolute deviation, with a small numerical stabilizer; this normalization is specific to the relational family. All quantities are computed within the current instance, without labels.

A01–A03: Do high-score regions coincide with departures from simple temporal behavior? If high scores reflect genuine deviations in the raw series, raw values in high-score regions should be poorly described by simple temporal models. We use three reference models: a global-mean model, a persistence model whose residuals are first differences, and a ridge-regularized AR(3) model. Their residuals capture deviations from the overall level, the preceding value, and a short autoregressive prediction, respectively. Coding cost is measured by the zero-mean Gaussian negative log-likelihood of residuals in bits.

For each model b and score quantile τ, the high-score region contains timestamps at or above the τ-quantile of the standardized scores, and the background contains the remaining timestamps. Let $D_B(b, \tau)$ and $D_H(b, \tau)$ denote the average per-timestamp coding costs of residuals in the background and the high-score region, respectively. Background costs use the variance of the background residuals, so D_B reflects how well the model describes typical behavior. High-score costs use a fixed reference variance, equal to four times the variance of the model's full residual sequence, so D_H grows as high-score residuals become large relative to the instance's typical residual scale. Finally, let b_* be the reference model that best describes the instance as a whole, i.e., the one with the lowest total coding cost over its full residual sequence, where each model's cost uses its own residual variance.

The three signals are

$$A_{01} = D_B(\mathrm{ar}, 0.95), \qquad A_{02} = D_H(\mathrm{ar}, 0.90), \qquad A_{03} = D_H(b_*, 0.99).$$

A_{01} characterizes how predictable the background is under AR(3), providing a reference level; A_{02} and A_{03} measure how strongly the raw series departs from AR(3) and from the best-fitting reference model within high-score and extreme-score regions, respectively.

A04–A07: Does the score response help describe raw-series structure? Following the description-length principle that a useful explanation shortens the total description of the data, we treat the score response as a candidate explanation of local raw-series complexity. Knowing the score states may make raw complexity cheaper to describe. We measure both the remaining cost and whether any reduction outweighs the cost of describing the score states themselves.

We divide the instance into overlapping windows of length $W \in \{50, 100, 200\}$ and summarize each window j in two ways. The raw-complexity summary U_j is the Gaussian coding cost of the window's levels and first differences, divided by its length; at each scale, the reference level and difference statistics are estimated from relatively low-variation windows of the same instance, selected by the combined log-variance of their levels and first differences. The score summary E_j is the average of the highest 10% of standardized scores in the window. We discretize U_j into eight quantile-based raw-complexity states, and E_j into either two score states, separated at its 90th percentile, or four quantile-based score states. We denote the number of score states by $B \in \{2, 4\}$ and the score-state sequence by C.

Using smoothed empirical symbol frequencies, we compare two ways of describing the raw-complexity states. Without the scores, their coding cost is $L_U(W)$. With the scores, the description consists of the score-state sequence itself, with run-length coding cost $L_C(W, B)$, plus the raw-complexity states given the score states, with cost $L_{U|C}(W, B)$. The net coding gain and its normalized form are

$$\Gamma(W, B) = L_U(W) - \big[L_C(W, B) + L_{U|C}(W, B)\big],$$
$$\gamma(W, B) = \frac{\Gamma(W, B)}{|L_U(W)| + \eta},$$

where η is a small numerical stabilizer. A positive gain means that the score states are worth describing: they shorten the total description of raw complexity.

The four signals are

$$A_{04} = L_{U|C}(100, 2), \quad A_{05} = \Gamma(100, 4),$$
$$A_{06} = \max_{W,B} \gamma(W, B), \quad A_{07} = \mathrm{mean}_{W,B}\, \gamma(W, B),$$

where the maximum and mean range over the six combinations of W and B. A_{04} measures the raw-complexity cost that remains after conditioning on score states, A_{05} the net gain at one fixed configuration, and A_{06} and A_{07} summarize normalized gains across temporal scales and score resolutions.

B Additional Experimental Details

B.1 TimeDPE Implementation Details

Surrogate preprocessing. For each leave-one-instance-out fold, each surrogate signal is transformed with respect to its empirical distribution over the historical instances. For feature j, given n historical values $v_{1j}, \ldots, v_{nj}$, a value x is mapped to

$$R_j(x) = \frac{1}{n+1} \sum_{i=1}^{n} \mathbb{I}[v_{ij} \leq x],$$

which corresponds to an empirical-CDF-based rank transformation. The transformed values are then standardized using statistics computed from the historical instances, and the same transformations are applied to the held-out deployment instance. This preprocessing is shared across estimators operating on the handcrafted surrogate representation.

Performance transformation. For performance metrics in $[0, 1]$, we first define

$$q_i' = \mathrm{clip}(q_i, \epsilon, 1 - \epsilon), \qquad \epsilon = 10^{-6},$$

and transform the historical performance values to log-odds,

$$\widetilde{q}_i = \mathrm{logit}(q_i') = \log \frac{q_i'}{1 - q_i'}.$$

Predictions are subsequently mapped back to the original performance scale using the inverse logit.

Training and prediction. Let $Z(\Phi_f(T_i))$ denote the preprocessed surrogate representation. TimeDPE fits the GP-RQ estimator to the historical pairs

$$\{(Z(\Phi_f(T_i)), \widetilde{q}_i)\}_{i=1}^{n}.$$

Let m_f denote the fitted GP posterior mean in logit space. For a deployment instance T_*, the logit-scale prediction is

$$\widehat{\widetilde{q}}_* = m_f(Z(\Phi_f(T_*))),$$

which is mapped back to the original performance scale as

$$\widehat{q}_* = \sigma\left(\widehat{\widetilde{q}}_*\right),$$

where $\sigma(\cdot)$ denotes the logistic sigmoid. Thus, the estimator g_f in the main text includes preprocessing and the inverse-logit transformation: $g_f(z) = \sigma(m_f(Z(z)))$.

B.2 Additional Comparison Method Details

This subsection provides additional implementation details for comparison methods that require further specification beyond Section 5.2.

Best single + GP-RQ. To construct the best-single baseline, we evaluate each of the 29 surrogate signals independently using the same GP-RQ estimator and preprocessing as TimeDPE. The reported baseline uses the signal with the lowest macro-averaged MAE over the complete 170 source–detector conditions, which is the population skewness of the anomaly-score sequence (S09, `score_skewness`). Signal selection is performed retrospectively after completing the leave-one-instance-out evaluation for all 29 candidates.

Catch22 + GP-RQ. For Catch22 + GP-RQ, each complete raw time series is represented by the 22 canonical catch22 features (Lubba et al., 2019), computed using `pycatch22` 0.4.5 with Catch24 disabled. No anomaly-score information is used in this representation. The resulting 22-dimensional feature vector is passed through the same fold-local preprocessing and GP-RQ estimator as the default TimeDPE configuration.

Shared neural input preprocessing. JNSR-DKL and TinyMLP both take as input the observable raw time series T and its aligned anomaly-score sequence $f(T)$. The two sequences are independently linearly interpolated onto a common temporal grid of length 256, producing

$$U \in \mathbb{R}^{2\times 256}.$$

Within each leave-one-instance-out fold, normalization statistics are estimated separately for the two channels using only the historical instances. Each channel is centered by its historical median and scaled by its historical interquartile range. When the interquartile range is at most 10^{-12}, the scale falls back successively to 1.4826 times the median absolute deviation, the population standard deviation, and finally 1. The resulting transformation is applied to the held-out instance, and normalized values are clipped to $[-10, 10]$. Both models are initialized from scratch in every fold using seed 42 and trained with full-batch AdamW, without a validation split or early stopping.

Joint Neural Surrogate Representation with Deep Kernel Learning (JNSR-DKL). JNSR-DKL replaces the handcrafted surrogate representation with a learned 29-dimensional representation optimized jointly with a Gaussian-process prediction objective. Following deep kernel learning (Wilson et al., 2016), a neural encoder g_θ maps the preprocessed input U to a learned representation that defines the inputs to an RQ kernel:

$$k_{\theta,\phi}(U_i, U_j) = k_{\mathrm{RQ},\phi}\big(g_\theta(U_i), g_\theta(U_j)\big).$$

The predictor is an exact Gaussian process with zero mean, a scaled isotropic RQ kernel, and a Gaussian observation likelihood. The encoder, kernel parameters, and likelihood parameters are optimized jointly by minimizing the negative exact log marginal likelihood.

The encoder first applies a 1×1 Conv1D projection from two input channels to eight channels. This is followed by four residual depthwise-separable Conv1D blocks with kernel size 5 and dilations $1, 2, 4, 8$. Each block uses bias-free depthwise and pointwise convolutions, four-group GroupNorm, GELU activations, length-preserving padding, and a residual connection. Global mean pooling, global maximum pooling, and four-bin adaptive-average pooling are concatenated into a 48-dimensional summary, which is projected through a linear layer followed by $\tanh$ to obtain

$$g_\theta(U) \in [-1, 1]^{29}.$$

The encoder contains 1,989 trainable parameters, excluding the GP and likelihood parameters.

Historical VUS-PR values are clipped to $[10^{-6}, 1 - 10^{-6}]$, logit-transformed, and standardized using the historical mean and population standard deviation within each fold. A standard deviation below 10^{-12} is replaced by 1. The RQ shape parameter, covariance amplitude, and likelihood-noise variance are initialized to 1, 1, and 0.05, respectively. The initial length scale is set to the median strictly positive pairwise distance between the initial historical embeddings.

Training uses 100 full-batch AdamW updates with learning rate 10^{-3}. Weight decay is 10^{-3} for encoder parameters and zero for GP and likelihood parameters, and gradients are clipped to unit norm. Prediction uses the posterior mean of the jointly trained GP, followed by inverse target standardization and the sigmoid transformation. Final predictions are clipped to $[0, 1]$.

Direct Neural Regression (TinyMLP). TinyMLP directly regresses detector performance from the observable raw time series and anomaly-score sequence, without constructing an explicit surrogate representation. The preprocessed input U is flattened in raw-then-score order into a 512-dimensional vector and passed through an MLP with widths

$$512 \to 32 \to 16 \to 1.$$

The two hidden layers use ReLU activations, and all linear layers include bias terms. The resulting network contains 16,961 trainable parameters.

Training minimizes mean squared error directly on the historical VUS-PR values for 200 full-batch AdamW updates with learning rate 10^{-3} and weight decay 10^{-3} on weight parameters. No target transformation, output sigmoid, dropout, learning-rate scheduling, or checkpoint selection is used. Predictions are obtained from the final network parameters and clipped to $[0, 1]$ at inference.

B.3 Learning Curve

The learning-curve experiments evaluate how DPE performance changes with the number of historically evaluated instances available for fitting. The main figure reports three sources: LTDB ($n = 9$), Opportunity ($n = 29$), and UCR ($n = 228$), with history sizes

$$\begin{aligned}\mathcal{M}_{\mathrm{LTDB}} &= \{1, 2, 3, 4, 5, 6, 7, 8\},\\ \mathcal{M}_{\mathrm{Opportunity}} &= \{1, 2, 3, 5, 8, 11, 15, 19, 23, 28\},\\ \mathcal{M}_{\mathrm{UCR}} &= \{5, 10, 20, 30, 50, 70, 90, 110, 150, 200, 227\}.\end{aligned}$$

The horizontal axis shows the history size on a logarithmic scale. For each source, the largest history size is $n - 1$ and therefore corresponds to the standard leave-one-instance-out setting.

For each source, detector, held-out target, and sampling replicate, the target instance is first removed from the candidate history pool. The remaining $n - 1$ instances are randomly permuted using a deterministic sampling procedure with base seed 42, and the first m instances define the history set of size m. History sets are therefore nested as m increases within each target and replicate. All methods use the same sampled historical instances for a given source, detector, target, history size, and replicate.

For method a, absolute errors are averaged over all n_s held-out targets within each source–detector condition:

$$E^{(r)}_{s,d,a}(m) = \frac{1}{n_s}\sum_{i=1}^{n_s}\left|\widehat{q}^{(r)}_{s,d,i,a}(m) - q_{s,d,i}\right|,$$

where n_s is the number of instances in source s. These errors are then averaged equally across the ten detectors:

$$E^{(r)}_{s,a}(m) = \frac{1}{10}\sum_{d=1}^{10} E^{(r)}_{s,d,a}(m).$$

All three sources use five fixed sampling replicates, $r \in \{0, \ldots, 4\}$, and the plotted curves report their mean. The shaded regions denote two-sided 95% Student-t intervals across the five replicate-level MAEs:

$$\overline{E}_{s,a}(m) \pm t_{0.975,4}\frac{s_{s,a}(m)}{\sqrt{5}},$$

where $\overline{E}_{s,a}(m)$ and $s_{s,a}(m)$ are the mean and sample standard deviation of the five replicate-level MAEs, respectively. At the largest history size, all replicates use the same complete history, so the interval collapses to zero width.

B.4 Surrogate Sufficiency Analysis

Here, we define the empirical surrogate sufficiency measure S_τ used in the surrogate representation analysis in the main text and describe the evaluated representations.

Empirical surrogate sufficiency. For a source–detector pair c and surrogate signal j, let

$$\rho_{c,j} = \mathrm{Spearman}(\{\phi_j(T_i, f(T_i))\}_{i\in c}, \{q_i\}_{i\in c})$$

denote the Spearman correlation between the signal values and true instance-level VUS-PR values within that pair. For a representation specified by a signal index set $\mathcal{J} \subseteq \{1, \ldots, 29\}$, define

$$S_\tau(\mathcal{J}) = \frac{1}{C}\sum_{c=1}^{C}\mathbb{I}\left[\max_{j\in\mathcal{J}}|\rho_{c,j}| \geq \tau\right],$$

where $C = 170$. We use $\tau = 0.5$ in the main-text analysis and examine $\tau \in \{0.6, 0.7, 0.8, 0.9\}$ below. Thus, $S_\tau(\mathcal{J})$ measures the fraction of source–detector pairs in which at least one included

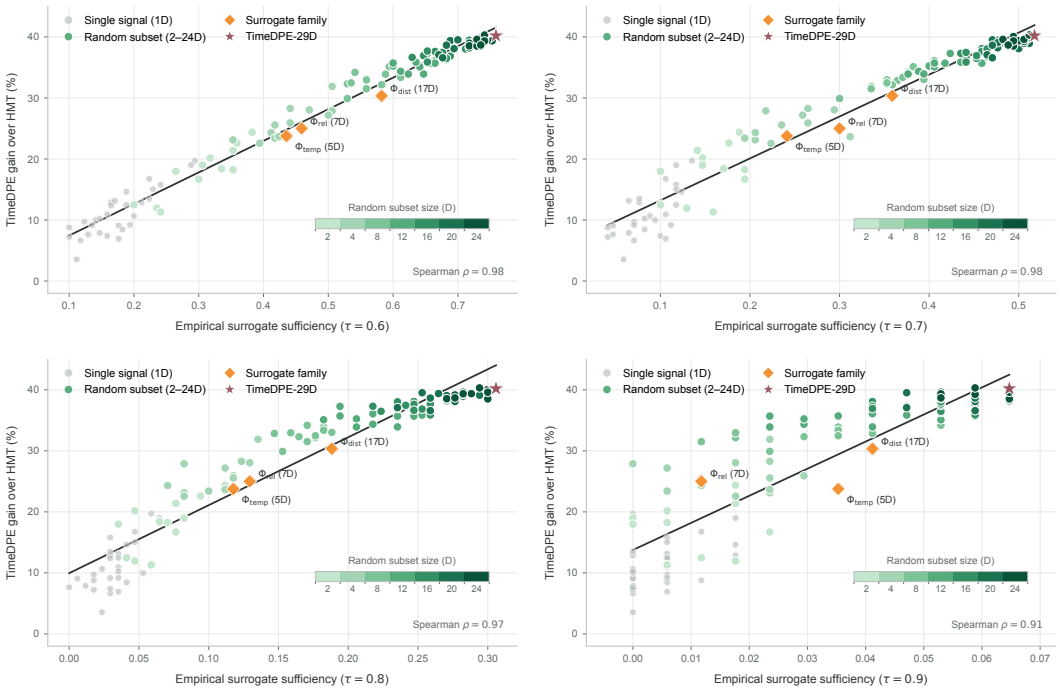


Figure 6: Sensitivity of empirical surrogate sufficiency to the correlation threshold τ. The four panels show all 117 evaluated representations at $\tau = 0.6$, 0.7, 0.8, and 0.9, respectively. The vertical axis is the unchanged gain over HMT; each panel displays the Spearman correlation across representations and a fitted linear trend. The horizontal scale is adjusted to the coverage range at each threshold.

signal has an absolute Spearman correlation of at least τ with detector performance. This retrospective coverage measure is descriptive and can increase with the number of included signals.

Evaluated representations. The analysis uses 117 unique representations: 29 individual signals; the three surrogate families Φ_{dist}, Φ_{temp}, and Φ_{rel}; 12 random subsets at each size $d \in \{2, 4, 8, 12, 16, 20, 24\}$; and the complete 29-dimensional representation. Random subsets are sampled without replacement using seed 42, with duplicate subsets excluded. All representations use the same preprocessing, GP-RQ estimator, and exact leave-one-instance-out protocol as the default TIMEDPE configuration. Gain over HMT is computed from macro-averaged MAE.

As a sensitivity check for Fig. 5(b), Fig. 6 repeats the representation analysis at four higher thresholds. The association remains strong at $\tau = 0.6$, 0.7, and 0.8 (Spearman $\rho = 0.984$, 0.977, and 0.972, versus 0.980 at 0.5), but weakens at $\tau = 0.9$ ($\rho = 0.914$): coverage becomes sparse (17 representations have zero coverage, and only 11 of 170 conditions are covered by the complete 29D representation, 10 of which have at most 20 instances), producing many ties and making the result more sensitive to representation size and small source–detector pairs.

C ADDITIONAL EXPERIMENTAL RESULTS

This section supplements the main results with four tables. Table 2 reports extended method comparisons, surrogate-family ablations, and estimator comparisons. Tables 3 and 4 break down the seven-method comparison by detector and data source, respectively. Table 5 evaluates TIMEDPE using alternative metrics of detector performance.

Table 2: **Extended comparisons and ablation results on TSB-AD-U.** All methods estimate VUS-PR under the same within-source, fixed-detector leave-one-instance-out protocol as Table 1, yielding 8,600 predictions over 170 source–detector pairs. Macro MAE is the unweighted mean of the 170 within-pair MAEs. Brackets give its 95% percentile bootstrap confidence interval, obtained by resampling source–detector pairs 20,000 times with seed 42. Gain is the percentage reduction in Macro MAE relative to HMT; W/L counts pairs with lower/higher MAE than HMT. Acc@.10 and Over are pooled percentages over all 8,600 predictions: the former measures $|\widehat{q} - q| \leq 0.10$, and the latter measures $\widehat{q} - q > 0.10$. The comparison block and the final GP-RQ row contain the seven methods in Table 1. Catch22 uses raw-series features with GP-RQ. Surrogate-family ablations fix GP-RQ and vary the representation; estimator comparisons fix the complete 29-dimensional representation and vary the estimator, with the MLP variant listed in the comparison block. Bold highlights the default TIMEDPE configuration. The 95% paired bootstrap intervals for Macro MAE differences among the three GP kernels all include zero. † The best single surrogate (score skewness) is selected retrospectively from the 29 single-signal GP-RQ predictors by overall Macro MAE and is held fixed across all source–detector pairs.

Method	Macro MAE ↓	Gain	W/L	Acc@.10 ↑	Over ↓
Comparison methods					
Historical mean transfer (HMT)	0.1637 [.148,.180]	—	—	40.3%	30.5%
Best single + GP-RQ† (1 D)	0.1314 [.119,.145]	19.7%	130/40	62.3%	17.1%
Catch22 + GP-RQ (raw, 22 D)	0.1298 [.115,.146]	20.7%	124/46	64.2%	16.4%
JNSR-DKL (29 D)	0.1359 [.121,.152]	17.0%	106/64	64.6%	17.5%
TinyMLP (raw + score)	0.1554 [.142,.169]	5.1%	87/83	58.1%	18.7%
TIMEDPE (29 D + MLP)	0.1073 [.096,.119]	34.5%	129/41	68.2%	15.4%
Surrogate-family ablations (GP-RQ)					
Φ_{dist} (17 D)	0.1140 [.102,.127]	30.3%	135/35	68.0%	14.9%
Φ_{temp} (5 D)	0.1248 [.112,.138]	23.8%	132/38	64.7%	15.7%
Φ_{rel} (7 D)	0.1228 [.108,.139]	25.0%	130/40	67.3%	14.8%
$\Phi_{\text{dist}} + \Phi_{\text{temp}}$ (22 D)	0.1040 [.094,.115]	36.4%	144/26	70.2%	13.4%
$\Phi_{\text{dist}} + \Phi_{\text{rel}}$ (24 D)	0.1017 [.091,.113]	37.9%	143/27	71.0%	13.4%
$\Phi_{\text{temp}} + \Phi_{\text{rel}}$ (12 D)	0.1128 [.100,.127]	31.1%	139/31	68.8%	13.7%
Complete 29 D representation: estimator comparison					
+ GP-Matérn	0.0978 [.088,.108]	40.2%	153/17	72.0%	12.9%
+ GP-RBF	0.0979 [.088,.108]	40.2%	148/22	71.8%	12.7%
+ KRR-RBF	0.0991 [.089,.109]	39.5%	139/31	70.8%	13.3%
+ Local linear ridge	0.1062 [.095,.118]	35.1%	131/39	68.3%	14.2%
+ kNN	0.1067 [.095,.119]	34.8%	150/20	71.6%	13.3%
+ Ridge	0.1107 [.099,.123]	32.4%	129/41	67.5%	14.6%
+ Huber	0.1353 [.121,.150]	17.3%	100/70	66.6%	15.2%
TIMEDPE (29 D + GP-RQ)	**0.0979** [.088,.108]	**40.2%**	**151/19**	**71.9%**	**12.7%**

Table 3: **Gain rate (%) relative to HMT, by detector.** Each cell is computed from the unweighted mean of the MAEs over the 17 source–detector pairs involving that detector. Negative values indicate performance worse than HMT.

Method	CNN	HydraAD	KMeansAD	KShapeAD	MOMENT	MatrixProfile	POLY	Sub-PCA	TSPulse	USAD
Historical mean transfer (HMT)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Best single + GP-RQ (1 D)	6.74	22.11	19.72	26.38	21.28	23.72	19.80	25.81	10.62	16.29
Catch22 + GP-RQ (raw, 22 D)	15.27	8.74	27.74	28.20	24.88	22.53	13.79	20.95	22.82	24.88
JNSR-DKL	13.90	8.87	27.57	27.88	22.50	9.06	9.99	10.33	24.54	21.78
TinyMLP (raw + score)	0.62	-6.86	11.67	8.69	7.20	18.08	14.61	-1.64	-2.83	-2.83
TIMEDPE (29 D + MLP)	26.71	24.36	42.77	51.91	24.65	34.00	35.37	36.94	19.02	41.85
TIMEDPE (29 D + GP-RQ)	**35.36**	**32.46**	**46.71**	**53.60**	**33.43**	**36.27**	**37.44**	**44.69**	**32.40**	**45.69**

Table 4: **Gain rate (%) relative to HMT, by data source.** Each cell is computed from the unweighted mean of the MAEs over the 10 source–detector pairs involving that data source. The number of instances in each data source is shown in parentheses. Sources are ordered by increasing instance count. Negative values indicate performance worse than HMT. The best value in each column is shown in bold.

Method	MITDB (8)	LTDB (9)	MGAB (9)	MSL (9)	NEK (9)	TODS (15)	IOPS (17)	SMAP (19)	SVDB (20)	Stock (20)	NAB (28)	OPPORTUNITY (29)	Exathlon (32)	SMD (38)	WSD (111)	UCR (228)	YAHOO (259)
Historical mean transfer (HMT)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Best single + GP-RQ (1 D)	4.91	11.75	-15.27	15.50	26.85	4.38	2.21	30.04	0.63	24.86	13.62	2.76	32.62	35.52	30.43	36.01	32.62
Catch22 + GP-RQ (raw, 22 D)	1.25	38.58	18.88	-18.65	**37.85**	36.16	7.12	-4.11	8.47	54.08	-1.30	7.25	57.32	43.46	31.88	44.64	38.70
JNSR-DKL	-36.43	29.21	-6.76	-15.58	31.58	-5.07	-14.04	12.23	-11.87	18.55	4.44	25.80	70.64	48.50	29.28	50.43	33.03
TinyMLP (raw + score)	-21.60	4.82	-570.94	2.46	11.06	-10.99	-25.05	8.48	-23.93	-185.36	0.36	16.11	48.95	43.96	19.60	42.78	37.71
TimeDPE (29 D + MLP)	23.84	**46.16**	**26.60**	14.11	29.12	**42.91**	-0.75	23.47	1.96	**66.20**	26.68	31.63	**79.64**	45.25	40.65	49.29	45.21
TimeDPE (29 D + GP-RQ)	**25.85**	41.36	17.33	**19.28**	36.77	41.94	**11.64**	**30.47**	**10.88**	62.71	**36.38**	**38.86**	78.39	**51.74**	**51.47**	**56.99**	**57.17**

Table 5: **Performance estimation under alternative target metrics.** We follow the same experimental protocol and reporting conventions as Table 1, changing only the detector performance metric q. HMT and all reported statistics are computed using the corresponding target metric.

Target q	HMT MAE	TimeDPE MAE	Gain	W/L	Acc@.10	Over
VUS-PR	0.1637	0.0979 [.088,.108]	40.2%	151/19	71.9%	12.7%
AUC-PR	0.1589	0.1006 [.089,.112]	36.7%	147/23	72.8%	12.7%
AUC-ROC	0.1385	0.0978 [.089,.107]	29.4%	141/29	64.1%	20.4%
Range-F1	0.1394	0.1026 [.094,.112]	26.4%	133/37	64.5%	15.4%

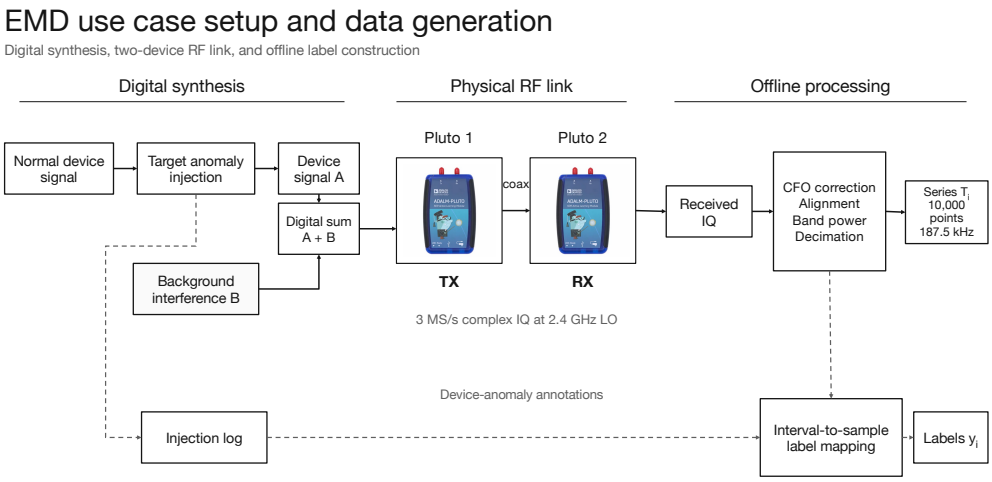


Figure 7: **SDR-based data-generation pipeline.** Synthetic device signals with injected anomalies are digitally combined with background interference and transmitted through a coaxial link between two ADALM-PLUTO radios. Offline processing converts received IQ samples into univariate monitoring series. Ground-truth labels are obtained by aligning the device-anomaly injection log with the processed series; background interference does not define additional anomaly labels.

D Details of the EMD-motivated Use Case

This appendix provides additional details on the controlled EMD-motivated monitoring setup and data-generation procedure, together with the full experimental results across the ten detectors.

D.1 TSAD Task Construction

We construct the monitoring task by separately generating the device signal and background interference, digitally combining them, and acquiring the resulting signal through the SDR link.

Synthetic monitored signal and anomaly injection. We represent the monitored process using a synthetic complex carrier with a periodically varying amplitude envelope and a slow drift component. Each instance contains three injected anomaly events that perturb this signal: *interruption* locally attenuates the envelope, *local amplitude* locally amplifies it, and *local rhythm* replaces it with a faster oscillating envelope. The corresponding injection intervals define the ground-truth anomaly labels. To generate related but nonidentical instances of the same monitoring task, the signal and anomaly parameters are varied across independent realizations.

Background interference. Background interference is generated independently of the device signal using three patterns: a continuous narrowband tone, band-limited noise, and an intermittent tone, representing persistent narrowband, broadband stochastic, and temporally intermittent interference backgrounds, respectively. For each interference type, we use four digital RMS amplitude levels $\{0.03, 0.08, 0.16, 0.28\}$ and generate twenty independent realizations at each level. This yields $3 \times 4 \times 20 = 240$ monitoring instances. The interference is digitally superposed with the device signal and does not define additional ground-truth anomalies.

SDR acquisition and preprocessing. We use two ADALM-PLUTO software-defined radios (Analog Devices, n.d.a), controlled through `pyadi-iio` (Analog Devices, n.d.b). The digitally combined device and interference signal is transmitted by one radio over a coaxial link and received by the other (Figure 7). Both radios use a 2.4 GHz RF center frequency and a 3 MHz IQ sam-

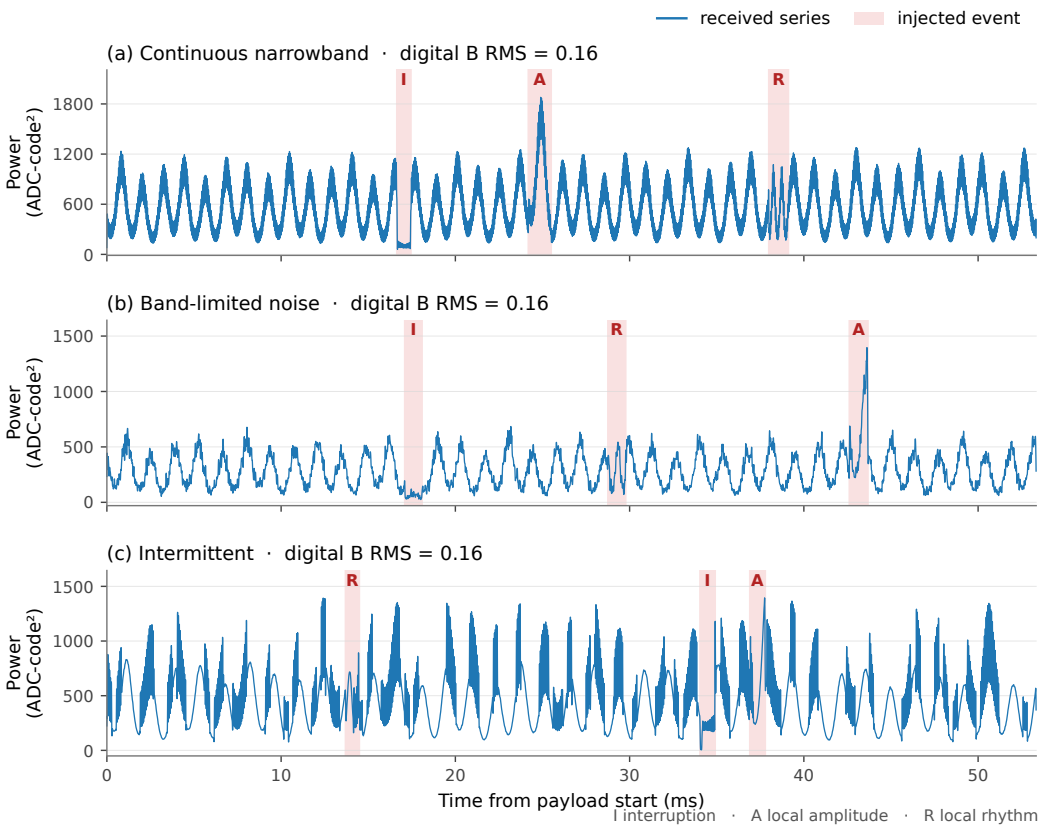


Figure 8: **Example monitoring signals under three interference types.** (a) Continuous narrowband interference; (b) band-limited noise; and (c) intermittent interference. All three instances use digital background RMS amplitude $\alpha = 0.16$. Shaded intervals indicate injected device anomalies: interruption, local amplitude, and local rhythm. Instances are independently generated rather than paired versions of the same waveform. Vertical-axis ranges differ across panels; no additional smoothing or normalization is applied.

pling rate. Received complex IQ samples undergo fixed frequency-offset correction, marker-based synchronization, and payload extraction. We then select the monitored frequency band, compute squared-magnitude power, and apply anti-aliasing filtering followed by sixteen-fold downsampling. Each processed instance contains 10,000 univariate samples at 187.5 kHz, spanning approximately 53.33 ms, without per-instance normalization. The recorded anomaly-injection intervals are mapped to the processed samples by temporal overlap. Figure 8 shows representative resulting instances under the three interference types.

D.2 Detector Performance on the EMD-Motivated Data

The resulting TSAD task is to detect the injected anomalies in the monitored signal under varying background interference. We apply the same ten anomaly detectors as in the main experiments, retaining their configurations and fitting and scoring procedures. Detector performance is evaluated using VUS-PR on each of the 240 time-series instances, yielding 2,400 detector–instance performance records.

Figure 9 shows the resulting instance-level performance distributions. Several detectors exhibit substantial performance variation across the monitoring instances, providing a concrete setting in which aggregate detector performance does not fully characterize performance on an individual instance.

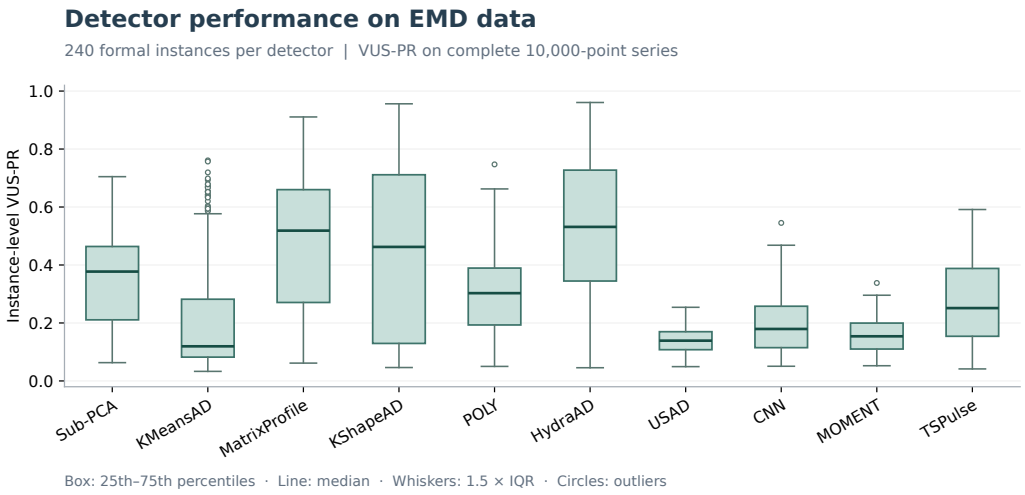


Figure 9: **Instance-level detector performance on the EMD-motivated monitoring dataset.** VUS-PR distributions for ten detectors evaluated on the same 240 monitoring instances. Boxes span the 25th–75th percentiles, and central lines indicate medians. Several detectors exhibit substantial instance-to-instance performance variation.

Table 6: **Exploratory TimeDPE estimation results by detector on the saved controlled SDR EMD records.** Each detector has 240 exact leave-one-instance-out VUS-PR predictions from the frozen 29-dimensional representation and GP-RQ estimator. Brackets are 95% bootstrap confidence intervals over the 240 independent instances (20,000 draws; seed 42). Gain and W/L are relative to the same-detector historical mean transfer (HMT) baseline. Acc@.10 is the fraction estimated within 0.10 of true VUS-PR; Over is the fraction overestimated by more than 0.10. The final row is the unweighted mean of the ten detector MAEs; its W/L counts all 2,400 paired targets.

Detector	TimeDPE MAE ↓	Gain	W/L	Acc@.10 ↑	Over ↓
Sub-PCA	0.0604 [0.054,0.067]	54.2%	188/52	81.7%	7.5%
KMeansAD	0.0336 [0.029,0.038]	79.8%	220/20	93.3%	2.1%
MatrixProfile	0.0681 [0.060,0.077]	66.0%	197/43	78.3%	12.5%
KShapeAD	0.0710 [0.061,0.082]	72.2%	207/33	75.4%	12.1%
POLY	0.0665 [0.059,0.074]	40.1%	159/81	75.8%	10.0%
HydraAD	0.0778 [0.070,0.086]	62.5%	186/54	70.8%	15.0%
USAD	0.0243 [0.022,0.027]	29.9%	157/83	99.6%	0.0%
CNN	0.0548 [0.048,0.061]	32.9%	163/77	81.7%	6.7%
MOMENT	0.0231 [0.021,0.026]	52.5%	183/57	100.0%	0.0%
TSPulse	0.0446 [0.040,0.049]	63.6%	199/41	90.8%	3.3%
Macro over 10 detectors	**0.0524** [0.050,0.055]	**61.4%**	1859/541	**84.8%**	6.9%

D.3 Deployment Performance Estimation

We follow the same exact leave-one-instance-out DPE protocol as in the main experiments. For each detector, each of the 240 instances serves once as the deployment target, while the remaining 239 instances form the historical evaluation records. We use the default TimeDPE configuration with the 29-dimensional surrogate representation and GP-RQ estimator, and use HMT as the baseline.

Table 6 reports the full detector-level results. Across the ten detectors, the macro-averaged MAE decreases from 0.1360 for HMT to 0.0524 for TimeDPE, corresponding to a 61.4% reduction.