ZERO-AD: LEARNING NORMALITY AS A ZERO-LEVEL SET FOR MULTIVARIATE TIME-SERIES ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multivariate time-series anomaly detection (MTSAD) often models normal behavior indirectly, using objectives such as reconstruction, forecasting, representation similarity, or normal prototypes. However, these approaches do not explicitly describe what normal temporal behavior is expected to satisfy. As a result, it can be difficult to distinguish true anomalous deviations from errors or uncertainty introduced by the model itself.

We introduce **ZERO-AD**, a geometric framework that models normal temporal behavior using a small set of learned implicit constraints, $F_\theta(Z) = [f_1(Z), \ldots, f_K(Z)]^\top$, such that normal windows satisfy $F_\theta(Z) \approx 0$. The common zero-level set of these constraints therefore provides a data-driven representation of normality, while anomalies are identified directly through violations of the learned constraints. To prevent degenerate solutions, ZERO-AD regularizes the local constraint geometry by encouraging complementary Jacobian directions, consistency across observed normal temporal variations, and finite-step sensitivity when moving away from the zero set. During inference, the constraint violations are calibrated using held-out normal data and then combined using reliability weighting, temporal persistence, and dense–sparse aggregation to produce the final anomaly score.

ZERO-AD does not require anomalous training samples or synthetic anomaly generation, and it avoids the need for a reconstruction decoder or forecasting head. On TSB-AD-M, it achieves the best aggregate performance on five of the six reported metrics, including a VUS-PR of 0.437 and a Point-F1 of 0.470, while using only about 0.07M parameters. These results show that learning the constraints that characterize normal temporal behavior can provide a compact and geometrically interpretable approach to normal-only MTSAD.

1 INTRODUCTION

Multivariate time-series anomaly detection (MTSAD) is widely used for monitoring industrial infrastructure, cyber-physical systems, autonomous platforms, healthcare applications, and scientific sensing systems. In many of these settings, anomalies are rare and expensive to label, making it important to learn normal behavior from available observations and identify deviations during testing. Classical one-class methods characterize normal data through its support or compactness (Schölkopf et al., 2001; Ruff et al., 2018), while more recent MTSAD methods increasingly rely on learned temporal representations. Most existing methods, however, capture normal behavior only indirectly. Reconstruction-based approaches identify anomalies through reconstruction error (Su et al., 2019; Zhang et al., 2023; Du et al., 2023), while other methods rely on temporal associations, contrastive representations, graph-based dependencies, or frequency- and channel-level information (Xu et al., 2022; Yang et al., 2023; Zhao et al., 2020; Wu et al., 2025). More recent methods, such as PaAno (Park & Kang, 2026) and KAN-AD (Zhou et al., 2025), further improve the learned representations, but the anomaly signal is still derived from reconstruction, prediction, representation distance, or discrepancy. As a result, the way normal behavior is learned and the way anomalies are identified often remain separate.

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

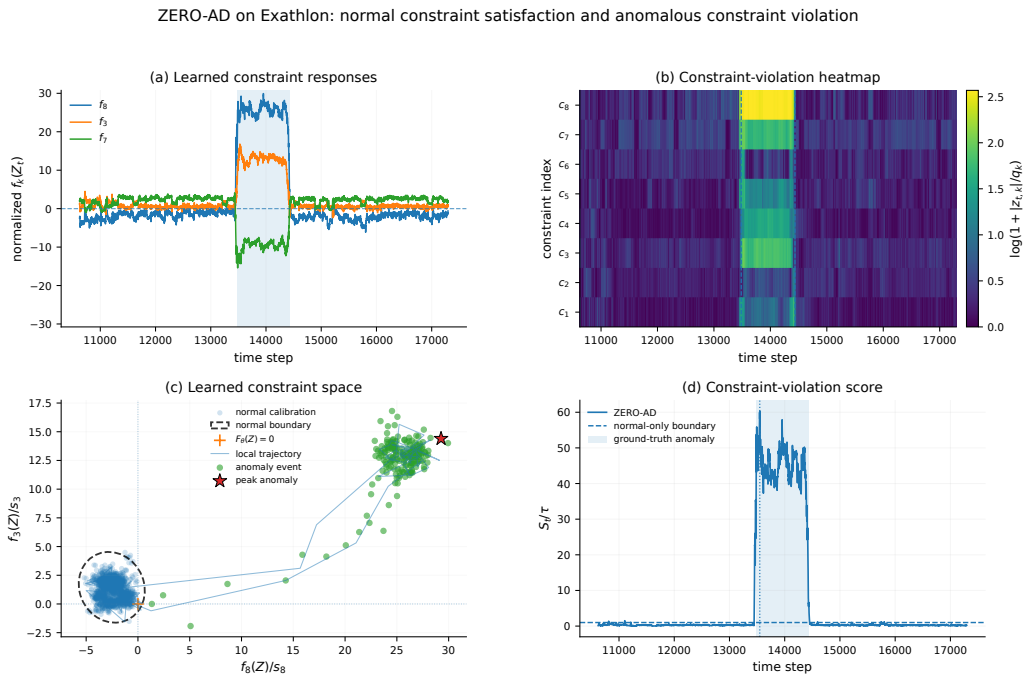


Figure 1: **Qualitative visualization of ZERO-AD on Exathlon.** (a) Representative learned constraint responses, showing clear deviations during the anomalous interval. (b) Boundary-normalized constraint violations, where brighter regions indicate stronger departures from normality. (c) Learned constraint space, where the anomaly trajectory moves away from the normal zero-level region $F_\theta(Z) = 0$. (d) Normalized ZERO-AD score, which rises sharply over the ground-truth anomalous interval.

Highly expressive models may also fit or explain abnormal inputs, while prediction errors can become large even when the observed behavior is valid but naturally stochastic. In addition, benchmark studies have shown that the choice of anomaly scoring and evaluation protocol can noticeably affect the reported results and conclusions (Liu & Paparrizos, 2024).

These observations lead to a more direct question:

> *Can we learn data-driven constraints that normal temporal behavior should satisfy and identify anomalies directly when these constraints are violated?*

To address this question, we introduce **ZERO-AD**, a framework that represents normal temporal behavior through the common zero-level set of a small set of learned constraint functions. For a window $Z \in \mathbb{R}^{L \times D}$,

$$\mathcal{M}_\theta = \{Z : F_\theta(Z) = 0\}, \qquad F_\theta(Z) = [f_1(Z), \ldots, f_K(Z)]^\top. \tag{1}$$

The model is trained so that normal windows remain close to these implicit normality constraints, while anomalous windows are expected to violate one or more of them. Our contribution is therefore not the zero-level representation itself, which is related to existing ideas in one-class and implicit-manifold learning, but its use for normal-only MTSAD together with a structured local departure geometry and an anomaly signal defined directly by constraint violations. Simply enforcing $F_\theta(Z) \approx 0$ is not sufficient, since it admits the trivial solution $F_\theta \equiv 0$. To avoid this collapse, ZERO-AD regularizes the Jacobian

$$J_\theta(Z) = \frac{\partial F_\theta(Z)}{\partial \operatorname{vec}(Z)} \in \mathbb{R}^{K \times LD}$$

so that

$$J_\theta(Z) J_\theta(Z)^\top \approx I_K, \qquad J_\theta(Z) \operatorname{vec}(t_Z) \approx 0, \tag{2}$$

where t_Z denotes an empirical local temporal direction estimated from nearby normal windows. The first condition encourages the learned constraints to capture complementary and locally informative departure directions, while the second encourages invariance along observed normal temporal evolution. Finite-step objectives further promote consistency along normal trajectories and controlled growth when moving away from the learned zero set.

This structure also provides a direct local interpretation of constraint violation. For a small perturbation δ around a normal point, $F_\theta(Z+\delta) \approx J_\theta(Z)\delta$. When the Jacobian rows are approximately orthonormal and the required local rank condition holds, $\|F_\theta(Z+\delta)\|_2^2$ approximates the energy of the perturbation projected onto the local normal space. In this sense, ZERO-AD is designed to be sensitive to departures away from normal temporal behavior while remaining comparatively insensitive to locally tangent variation. A formal statement and the corresponding conditions are provided in Appendix B.1, while the interpretation of K as the codimension of the learned representation is discussed in Appendix B.2.

Figure 1 provides an example of this behavior on Exathlon. During normal operation, the constraint responses stay within their calibrated normal range. When the anomalous interval starts, several constraints are violated, the trajectory moves away from the learned zero-level region, and the anomaly score rises noticeably. In this way, ZERO-AD detects anomalies directly through violations of the learned normality constraints, rather than relying only on reconstruction or prediction errors.

Contributions. Our main contributions are:

- We formulate normal-only MTSAD as learning a compact set of **implicit temporal normality constraints**, whose common zero set represents nominal behavior and whose violations directly provide anomaly evidence.
- We introduce a **geometry-aware learning objective** that combines Jacobian orthogonality, empirical temporal-direction consistency, and finite-step regularization to prevent collapse and learn complementary local departure directions.
- We provide a **local geometric interpretation** showing that, under suitable regularity conditions, constraint-violation energy measures displacement along the local normal space of the learned zero set.
- We evaluate ZERO-AD on the TSB-AD-M benchmark across diverse multivariate time-series datasets and evaluation metrics, showing competitive aggregate performance with a compact model and stable behavior across random seeds.

2 Related Work

Multivariate time-series anomaly detection. MTSAD has been explored through a wide range of statistical, reconstruction-based, forecasting-based, and representation-learning approaches. Classical methods such as One-Class SVM Schölkopf et al. (2001), Isolation Forest Liu et al. (2008), and LOF Breunig et al. (2000) model normal behavior using support estimation, isolation, or local density. Deep learning methods include LSTM-based approaches Malhotra et al. (2015; 2016), OmniAnomaly Su et al. (2019), autoencoder-based models Zhang et al. (2023), and generative methods Du et al. (2023). These approaches generally detect anomalies through prediction or reconstruction errors, likelihood estimates, or other deviations from the learned normal behavior.

Temporal and cross-variable representation learning. Recent work has increasingly focused on capturing richer temporal patterns and dependencies across variables. MTAD-GAT Zhao et al. (2020) and SARAD Dai et al. (2024) explicitly model interactions between channels, while Anomaly Transformer Xu et al. (2022) and DCdetector Yang et al. (2023) use association discrepancy and contrastive learning, respectively. Other architectures, such as TimesNet Wu et al. (2023), ModernTCN donghao & wang xue (2024), FITS Xu et al. (2024), Peri-midFormer Wu et al. (2024), and CATCH Wu et al. (2025), explore temporal, convolutional, frequency-based, periodic, and channel-aware representations. More recent methods, including PaAno Park & Kang (2026), KAN-AD Zhou et al. (2025), and CARLA Darban et al. (2025), further improve representation learning through patch-based modeling, Kolmogorov–Arnold Networks, and contrastive self-supervision. Despite these advances, the final

anomaly signal is still usually derived through reconstruction, prediction, representation distance, discrepancy, or a separate learned scoring function.

Relation to one-class and implicit-manifold learning. ZERO-AD is closely related to one-class learning and implicit manifold modeling, but differs in how normality is represented and directly connected to anomaly detection. Deep SVDD Ruff et al. (2018), for example, learns a compact hyperspherical region for normal samples, while neural implicit-manifold methods describe data manifolds as zero sets of learned functions. In ZERO-AD, the zero-level representation itself is not the main novelty. Instead, we formulate normal-only MTSAD as learning a compact set of temporal constraints,

$$F_\theta(Z) \approx 0,$$

where violations of these constraints directly serve as evidence of anomalous behavior. The constraint Jacobian is explicitly regularized to encourage complementary local departure directions, while observed normal temporal evolution is encouraged to remain locally tangent to the learned zero set. In this way, ZERO-AD jointly learns the normality constraints, the local geometry around the normal region, and the resulting anomaly signal, rather than relying on a separate reconstruction, distance, density, or discrepancy objective for detection.

3 METHODOLOGY

3.1 PROBLEM SETUP

Let $X = \{x_t\}_{t=1}^{T}$, with $x_t \in \mathbb{R}^D$, denote a D-variate time series. We consider normal-only anomaly detection, where only a normal training prefix is used for training, validation, calibration, and anomaly scoring. For a temporal window $Z = [z_1, \dots, z_L] \in \mathbb{R}^{L\times D}$, ZERO-AD learns

$$F_\theta(Z) = [f_1(Z), \dots, f_K(Z)]^\top, \qquad \mathcal{M}_\theta = \{Z : F_\theta(Z) = 0\}. \tag{3}$$

Thus, normality is represented directly as a common zero-level set, and anomalies are identified through violations of the learned constraints. In contrast, a Deep-SVDD-style detector defines a compact region $\mathcal{C} = \{Z : \|\phi_\theta(Z) - c\|_2^2 \le R^2\}$ and scores samples by their distance to c; ZERO-AD instead structures its local normal space through $J_\theta(Z)J_\theta(Z)^\top \approx I_K$ and $J_\theta(Z)t_Z \approx 0$. The overall architecture of ZERO-AD is illustrated in Figure 2.

3.2 TEMPORAL CONSTRAINT NETWORK

For each window, we compute the first temporal difference

$$\Delta z_\ell = \begin{cases} 0, & \ell = 1, \\ z_\ell - z_{\ell-1}, & \ell > 1, \end{cases} \qquad U = [Z, \Delta Z]. \tag{4}$$

A point-wise projection maps U to a hidden representation, followed by dilated residual temporal blocks containing normalization, depth-wise temporal convolution, gated point-wise channel mixing, and residual connections. Temporal mean and maximum pooling are then concatenated and passed through a small MLP to obtain $F_\theta(Z) \in \mathbb{R}^K$.

3.3 LEARNING THE NORMAL ZERO SET

For a normal window Z, ZERO-AD first minimizes

$$\mathcal{L}_{\text{zero}} = \frac{1}{K}\|F_\theta(Z)\|_2^2, \tag{5}$$

which places normal windows near the learned zero set. Since $F_\theta \equiv 0$ is a trivial solution, we additionally regularize the local geometry of F_θ.

Independent constraint directions. Let

$$J_\theta(Z) = \frac{\partial F_\theta(Z)}{\partial \,\mathrm{vec}(Z)} \in \mathbb{R}^{K\times LD}. \tag{6}$$

To avoid locally flat or redundant constraints, we encourage approximately unit-norm and mutually distinct Jacobian rows:

$$\mathcal{L}_{\text{orth}} = \|J_\theta(Z)J_\theta(Z)^\top - I_K\|_F^2. \tag{7}$$

Preserving observed normal temporal evolution. Rather than estimating the full tangent space, ZERO-AD uses the observed transition between nearby normal windows as an empirical local direction,

$$t_Z = \frac{Z^+ - Z}{\|Z^+ - Z\|_F + \epsilon}, \qquad \mathcal{L}_{\text{tan}} = \|J_\theta(Z)\,\text{vec}(t_Z)\|_2^2. \tag{8}$$

This encourages invariance along observed normal temporal evolution without assuming that a single t_Z spans the complete tangent space.

Finite-step consistency. The first-order tangent condition is complemented by

$$\mathcal{L}_{\text{tz}} = \|F_\theta(Z + \epsilon_t t_Z)\|_2^2\,, \tag{9}$$

which keeps a finite step along the observed temporal direction close to the zero set.

To control finite-step behavior away from the zero set, for a sampled constraint f_k we define

$$n_k = \frac{\nabla_Z f_k(Z)}{\|\nabla_Z f_k(Z)\|_F + \epsilon}, \qquad \mathcal{L}_{\text{dist}} = \left(f_k(Z + \epsilon_n n_k) - \epsilon_n\right)^2. \tag{10}$$

A second-order expansion,

$$f_k(Z + \epsilon_n n_k) = f_k(Z) + \epsilon_n \|\nabla_Z f_k(Z)\|_F + \frac{\epsilon_n^2}{2} n_k^\top H_k(Z) n_k + O(\epsilon_n^3), \tag{11}$$

shows that $\mathcal{L}_{\text{dist}}$ complements $\mathcal{L}_{\text{orth}}$ by constraining finite-step growth along a local normal direction.

Overall objective. The complete training objective is

$$\boxed{\mathcal{L} = \lambda_0 \mathcal{L}_{\text{zero}} + \lambda_o \mathcal{L}_{\text{orth}} + \lambda_t \mathcal{L}_{\text{tan}} + \lambda_d \mathcal{L}_{\text{dist}} + \lambda_{tz} \mathcal{L}_{\text{tz}}}. \tag{12}$$

Here, $\mathcal{L}_{\text{zero}}$ places normal windows on the zero set, $\mathcal{L}_{\text{orth}}$ prevents flat or redundant constraints, $\mathcal{L}_{\text{tan}}$ and $\mathcal{L}_{\text{tz}}$ preserve local and finite-step normal temporal evolution, and $\mathcal{L}_{\text{dist}}$ encourages predictable growth away from the zero set. A formal local geometric interpretation is given in Appendix B.1, with further discussion of K in Appendix B.2.

3.4 Normal Calibration and Anomaly Scoring

The normal prefix is chronologically divided into fitting, validation, and calibration subsets. For each constraint k, the calibration subset provides the robust center and scale

$$m_k = \text{median}(f_k), \qquad s_k = c_{\text{MAD}}\,\text{median}(|f_k - m_k|)\,, \tag{13}$$

and the standardized violation energy

$$z_{t,k} = \frac{f_k(Z_t) - m_k}{s_k + \epsilon}, \qquad E_{t,k} = z_{t,k}^2. \tag{14}$$

Constraints with high normal variability receive smaller reliability weights. Let

$$\nu_k = \text{RobustScale}(E_{t,k}), \qquad r_k = \frac{1}{\nu_k^2 + \epsilon}, \qquad w_k = (1-\eta)\frac{1}{K} + \eta \widetilde{r}_k, \quad \sum_{k=1}^{K} \widetilde{r}_k = 1. \tag{15}$$

To favor persistent violations, constraint energies are temporally smoothed:

$$\overline{E}_{t,k} = \sum_{\tau=-R}^{R} h_\tau E_{t+\tau,k}, \qquad E^*_{t,k} = (1-\rho)E_{t,k} + \rho \overline{E}_{t,k}, \qquad \sum_{\tau=-R}^{R} h_\tau = 1. \tag{16}$$

Finally, ZERO-AD combines reliability-weighted dense and sparse violations,

$$S_t^{\text{dense}} = \sum_{k=1}^{K} w_k E^*_{t,k}, \qquad S_t^{\text{sparse}} = \max_{1 \le k \le K} E^*_{t,k}, \tag{17}$$

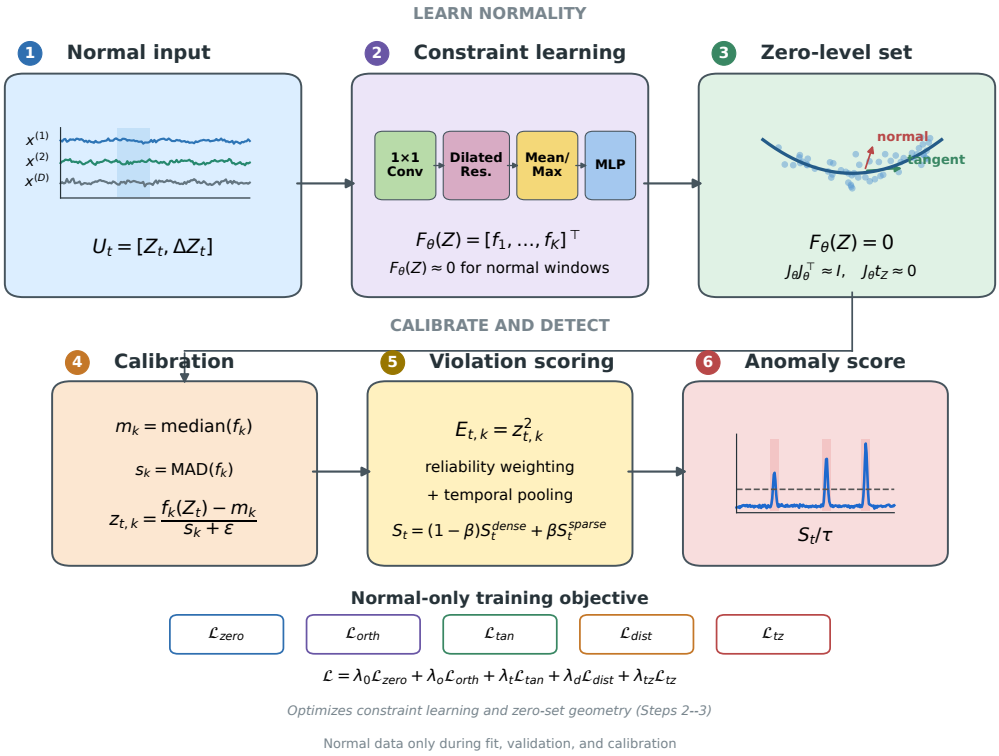


Figure 2: **Overall architecture of ZERO-AD.** A normal multivariate window and its temporal difference are mapped to K implicit normality constraints. Geometry-aware objectives shape the learned zero-level set during normal-only training; normal-only calibration and reliability-weighted temporal dense–sparse aggregation then produce the final anomaly score.

to form

$$\boxed{S_t = (1-\beta)S_t^{\text{dense}} + \beta S_t^{\text{sparse}}}. \tag{18}$$

All calibration statistics and score-combination parameters are estimated from normal data only. The network is trained on the fitting subset, checkpoint selection uses the normal validation subset, and the calibration subset is reserved for (m_k, s_k) and reliability estimation. At inference, each window requires a single forward pass followed by deterministic calibration and score aggregation. Complete optimization, calibration, evaluation, and implementation details are provided in Appendix C; computational complexity is discussed in Appendix B.3.

4 Experiments

4.1 Experimental Setup

We evaluate ZERO-AD on TSB-AD-M Liu & Paparrizos (2024), comprising 200 multivariate time series from 17 dataset families. Following the official benchmark protocol and PaAno Park & Kang (2026), 20 series are used for hyperparameter tuning and the remaining 180 series for final evaluation; details are provided in Appendix C.1. We compare against the same statistical/ML, neural-network, and Transformer-based baselines considered by PaAno and report VUS-PR, VUS-ROC, Range-F1, AUC-PR, AUC-ROC, and Point-F1. Complete implementation and evaluation details are provided in Appendix C.

Table 1: Comparison on TSB-AD-M. Results of competing methods are taken from PaAno. For each detection metric, the best result is shown in **bold** and the second-best is underlined.

Type	Method	VUS-PR	VUS-ROC	Range-F1	AUC-PR	AUC-ROC	Point-F1	# Params	Run Time
Stat./ML	KMeansAD (2001)	0.29	0.73	0.33	0.25	0.69	0.31	–	62.0s
	PCA (2017)	0.31	0.74	0.29	0.31	0.70	0.37	–	0.1s
Linear	DLinear (2023)	0.29	0.70	0.21	0.27	0.66	0.32	$<$ 0.1M	14.8s
	NLinear (2023)	0.30	0.70	0.21	0.26	0.65	0.31	$<$ 0.1M	15.0s
NN-based	DeepAnT (2018)	0.31	0.76	0.37	0.32	0.73	0.37	$<$ 0.1M	9.5s
	OmniAnomaly (2019)	0.31	0.69	0.37	0.27	0.65	0.32	$<$ 0.1M	9.1s
	TimesNet (2022)	0.19	0.64	0.17	0.13	0.56	0.20	$<$ 0.1M	52.1s
	FITS (2023)	0.21	0.66	0.16	0.15	0.58	0.22	$<$ 0.1M	16.7s
	DADA (2025)	0.31	0.73	0.25	0.31	0.69	0.35	1.84M	2.1s
	KAN-AD (2025)	0.40	0.76	0.41	0.35	0.73	0.40	$<$ 0.1M	31.9s
Transformer	AnomalyTransformer (2021)	0.12	0.57	0.14	0.07	0.52	0.12	4.8M	55.8s
	DCdetector (2023)	0.10	0.54	0.09	0.05	0.48	0.10	0.8M	15.0s
	OFA (2023)	0.21	0.63	0.17	0.15	0.55	0.21	81.9M	532.9s
	PatchTST (2023)	0.30	0.71	0.22	0.28	0.67	0.33	6.9M	66.9s
	iTransformer (2024)	0.30	0.71	0.18	0.24	0.64	0.29	6.4M	24.4s
	CATCH (2025)	0.30	0.72	0.18	0.23	0.65	0.29	210.8M	40.1s
Patch-based	PaAno (2026)	0.43	**0.79**	0.41	0.38	0.76	0.43	0.3M	9.7s
Geometry-based	**ZERO-AD (Ours)**	**0.437**	0.78	**0.435**	**0.396**	**0.782**	**0.47**	0.07M	19.5s

Table 2: **Dataset-wise comparison with PaAno on all 17 TSB-AD-M datasets.** For each metric, PaAno and ZERO-AD are shown in separate columns and the better value is highlighted in **bold**. The final column reports the number of metrics on which ZERO-AD outperforms PaAno.

Dataset	VUS-PR		VUS-ROC		Range-F1		AUC-PR		AUC-ROC		Point-F1		Wins
	PaAno	Ours	PaAno	Ours	PaAno	Ours	PaAno	Ours	PaAno	Ours	PaAno	Ours	
CATSv2	0.0690	**0.1892**	**0.6881**	0.6396	0.1381	**0.3469**	0.0744	**0.2482**	**0.6813**	0.6467	0.1285	**0.3261**	4/6
CreditCard	0.0343	**0.0577**	0.5558	**0.7142**	**0.0351**	0.0237	**0.0068**	0.0058	0.5652	**0.6802**	**0.0416**	0.0253	3/6
Daphnet	0.2706	**0.2969**	**0.8925**	0.8874	**0.3117**	0.3018	0.2564	**0.2852**	**0.9052**	0.8876	**0.4189**	0.4152	2/6
Exathlon	0.7769	**0.9324**	0.9545	**0.9725**	0.6045	**0.8111**	0.7482	**0.9186**	0.9518	**0.9705**	0.8357	**0.9220**	**6/6**
GECCO	0.1601	**0.3155**	0.8727	**0.9278**	**0.2268**	0.0996	0.2192	**0.3174**	0.9050	**0.9105**	0.2959	**0.3955**	5/6
GHL	0.0085	**0.0296**	0.3391	**0.4284**	0.0240	**0.0346**	0.0073	**0.0297**	0.3090	**0.4231**	0.0241	**0.0587**	**6/6**
Genesis	0.2964	**0.6780**	0.9902	**0.9985**	**0.2930**	0.2740	0.1945	**0.2098**	0.9867	**0.9939**	0.3023	**0.3412**	5/6
LTDB	**0.6066**	0.4218	**0.8640**	0.7508	**0.5965**	0.4175	**0.5856**	0.3924	**0.8327**	0.7120	**0.5861**	0.4875	0/6
MITDB	**0.3841**	0.1981	**0.8803**	0.7352	**0.4647**	0.3494	**0.4264**	0.2681	**0.8567**	0.6996	**0.4864**	0.3307	0/6
MSL	0.2416	**0.3842**	0.7768	**0.8296**	0.3630	**0.3671**	0.1969	**0.3106**	0.7498	**0.7781**	0.2937	**0.4045**	**6/6**
OPPORTUNITY	**0.1701**	0.1245	**0.6069**	0.4853	**0.3463**	0.2212	**0.1646**	0.1107	**0.5810**	0.4840	**0.2134**	0.2126	0/6
PSM	0.2033	**0.2613**	0.6597	**0.7115**	**0.2749**	0.1985	0.1839	**0.2743**	0.6571	**0.7435**	0.2621	**0.3460**	5/6
SMAP	**0.5363**	0.4141	**0.9009**	0.8365	**0.5860**	0.4583	**0.5220**	0.3477	**0.9055**	0.8344	**0.5428**	0.4934	0/6
SMD	**0.3545**	0.3045	**0.8479**	0.8227	**0.4044**	0.3712	**0.3850**	0.2215	**0.8728**	0.8000	**0.4445**	0.3658	0/6
SVDB	**0.5574**	0.3152	**0.9050**	0.7744	**0.5436**	0.3107	**0.5259**	0.3316	**0.8814**	0.7221	**0.5496**	0.3675	0/6
SWaT	0.2428	**0.4246**	0.6864	**0.7085**	**0.3421**	0.2642	0.2174	**0.5357**	0.6679	**0.8111**	0.3213	**0.6626**	5/6
TAO	0.7273	**0.7351**	0.8535	**0.8648**	**0.1848**	0.1487	**0.0849**	0.0819	**0.4966**	0.4831	**0.1591**	0.1583	2/6

5 Results and Discussion

Overall performance. Table 1 compares ZERO-AD with statistical, linear, neural, Transformer-based, and recent anomaly-detection methods on TSB-AD-M. ZERO-AD achieves the best result on five of six metrics: VUS-PR 0.437, Range-F1 0.435, AUC-PR 0.396, AUC-ROC 0.782, and Point-F1 0.470. Relative to PaAno (Park & Kang, 2026), these correspond to absolute gains of 0.007, 0.025, 0.016, 0.022, and 0.040, respectively, while VUS-ROC is slightly lower (0.780 vs. 0.790). Thus, the learned constraint violations provide competitive anomaly evidence across both threshold-free and threshold-dependent metrics.

Dataset-wise behavior. Table 2 compares ZERO-AD with PaAno across all 17 dataset families. ZERO-AD improves at least one metric on 11 datasets and all six metrics on Exathlon, GHL, and MSL. Particularly strong gains include VUS-PR from 0.777 to 0.932 and Range-F1 from 0.605 to 0.811 on Exathlon; VUS-PR from 0.296 to 0.678 on Genesis; AUC-PR from 0.184 to 0.274 on PSM; and AUC-PR from 0.217 to 0.536 and Point-F1 from 0.321 to 0.663 on SWaT. Gains are also

observed on CATSv2, CreditCard, Daphnet, GECCO, and TAO, indicating that the learned constraints generalize across diverse temporal systems.

The comparison also reveals clear limitations. PaAno remains stronger on all six metrics for LTDB, MITDB, OPPORTUNITY, SMAP, SMD, and SVDB, suggesting that a fixed compact constraint geometry is not uniformly optimal when normal dynamics are highly heterogeneous or anomalous behavior remains close to the normal regime. Complete per-metric results are provided in Appendix D.2.

Metric-dependent behavior. Improvements in anomaly ranking do not always imply equally accurate temporal localization. On Genesis, for example, ZERO-AD increases VUS-PR from 0.296 to 0.678 and VUS-ROC from 0.990 to 0.999, while Range-F1 decreases from 0.293 to 0.274. A similar pattern appears on GECCO, PSM, and SWaT, where ZERO-AD improves several score-based and point-level measures but PaAno retains higher Range-F1. This suggests that ZERO-AD can assign strong anomaly evidence to the relevant temporal region while precise event boundaries remain dataset-dependent. The ablation and sensitivity analyses further examine the role of temporal pooling and score aggregation in this behavior.

Compactness and computational cost. ZERO-AD uses approximately 0.07M parameters, compared with 0.3M for PaAno and substantially larger Transformer-based models such as CATCH and OFA. Its runtime is higher (19.5 s vs. 9.7 s for PaAno), primarily because Jacobian-based geometric regularization adds training cost. This overhead is restricted to training; inference requires only a forward pass followed by deterministic calibration and score aggregation. A detailed complexity analysis is given in Appendix B.3.

Summary. Overall, ZERO-AD achieves the strongest aggregate performance on five of six TSB-AD-M metrics with a compact model, while the dataset-wise results show both substantial gains and identifiable failure regimes. These findings support the central idea of detecting anomalies through violations of learned normality constraints, while also showing that heterogeneous normal regimes and precise temporal localization remain challenging. Further qualitative, ablation, sensitivity, and failure-case analyses are presented next and in Appendix E.

5.1 Qualitative Analysis

Figure 3 illustrates the detection process of ZERO-AD on Genesis. Representative constraint responses remain relatively stable during normal operation but deviate strongly within the anomalous interval (a). The violation heatmap shows that several constraints respond jointly, while a smaller subset exhibits particularly strong activations (b), supporting the complementary dense–sparse aggregation used by ZERO-AD. In the learned constraint space, normal windows remain close to the calibrated zero-level region, whereas the anomalous trajectory moves away from it toward a clearly separated peak-anomaly point (c). Correspondingly, the final anomaly score rises sharply during this departure and decreases after the event (d). Together, these views show that the final detection can be traced back to specific violations of the learned normality constraints rather than only to an opaque scalar anomaly score.

5.2 Ablation Study

Table 3 evaluates the main components of ZERO-AD on all 180 TSB-AD-M evaluation series. Removing any component degrades all six metrics, indicating that both the learned zero-set geometry and the constraint-violation scoring contribute to performance.

Zero-set geometry. Removing Jacobian orthogonality reduces VUS-PR from 0.437 to 0.393 and AUC-PR from 0.396 to 0.330, while removing temporal consistency lowers VUS-PR to 0.390 and Range-F1 to 0.362. Removing distance consistency reduces AUC-ROC from 0.782 to 0.725, and removing all geometric terms yields AUC-PR of 0.328 and AUC-ROC of 0.720. These results show that enforcing $F_\theta(Z) \approx 0$ alone is insufficient; structuring the local geometry around the learned zero set is important for effective anomaly detection.

Constraint aggregation. Removing reliability weighting lowers VUS-PR from 0.437 to 0.394, while removing temporal pooling causes the largest Range-F1 drop, from 0.435 to 0.349. Eliminating

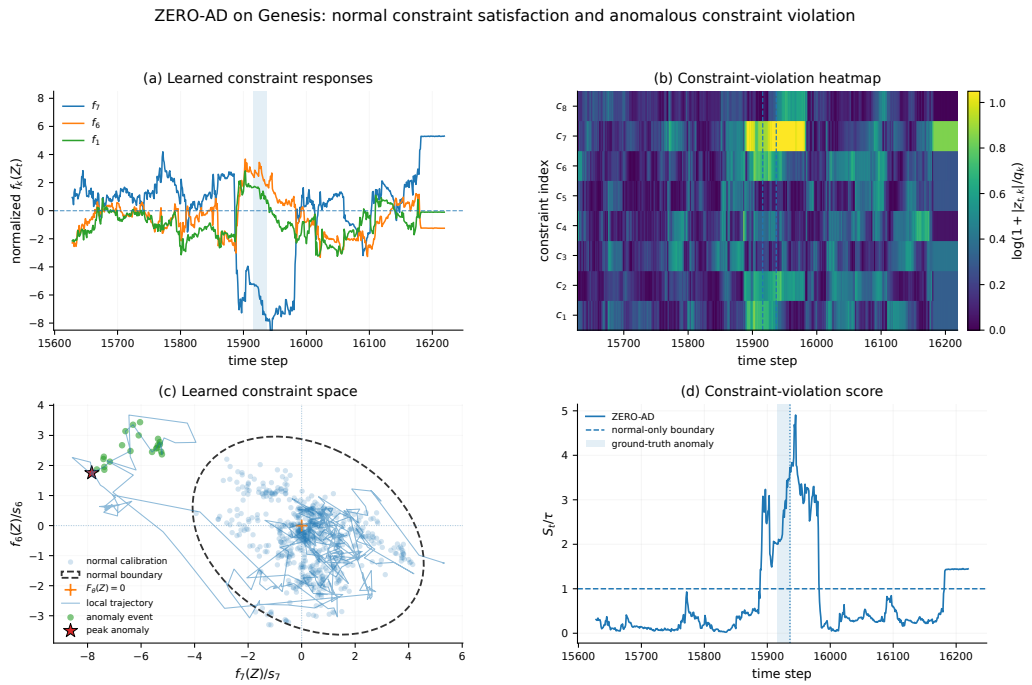


Figure 3: **Qualitative analysis of ZERO-AD on Genesis.** (a) Representative constraint responses. (b) Constraint-wise violation heatmap. (c) Constraint-space projection showing anomalies departing from the normal zero-level region. (d) Normalized anomaly score S_t/τ, which increases sharply over the anomalous interval.

Table 3: **Component ablation of ZERO-AD on TSB-AD-M.** ✓ denotes an enabled component and ✗ denotes its removal. Results are averaged over the 180 evaluation series.

Variant	Model Components							Performance					
	Zero Level	Orth. Geom.	Temporal Consistency	Distance Consistency	Reliability Weighting	Temporal Pooling	Sparse Score	VUS-PR	VUS-ROC	Range-F1	AUC-PR	AUC-ROC	Point-F1
Full ZERO-AD	✓	✓	✓	✓	✓	✓	✓	**0.437**	**0.78**	**0.435**	**0.396**	**0.782**	**0.47**
w/o Orthogonality	✓	✗	✓	✓	✓	✓	✓	0.3932	0.7679	0.3760	0.3299	0.7261	0.4075
w/o Temporal Consistency	✓	✓	✗	✓	✓	✓	✓	0.3903	0.7652	0.3619	0.3297	0.7243	0.4064
w/o Distance Consistency	✓	✓	✓	✗	✓	✓	✓	0.3954	0.7656	0.3646	0.3328	0.7247	0.4088
w/o Zero-Set Geometry	✓	✗	✗	✗	✓	✓	✓	0.3938	0.7631	0.3847	0.3279	0.7201	0.4067
w/o Reliability Weighting	✓	✓	✓	✓	✗	✓	✓	0.3942	0.7652	0.3661	0.3336	0.7241	0.4074
w/o Temporal Pooling	✓	✓	✓	✓	✓	✗	✓	0.3927	0.7651	0.3490	0.3318	0.7233	0.4061
w/o Sparse Score	✓	✓	✓	✓	✓	✓	✗	0.3956	0.7656	0.3708	0.3336	0.7241	0.4083

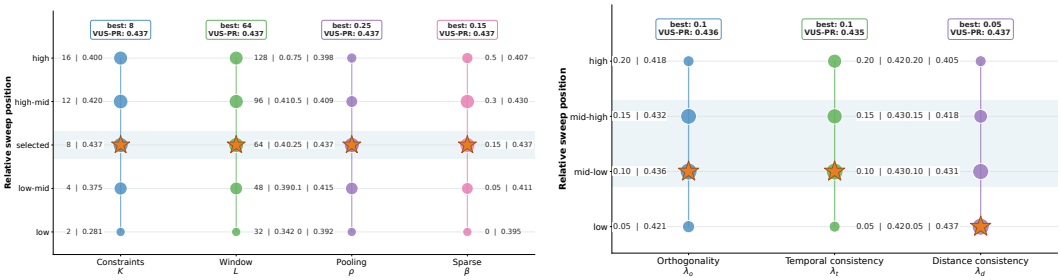
Temporal Consistency includes both $\mathcal{L}_{\text{tan}}$ and $\mathcal{L}_{\text{tz}}$. The "w/o Zero-Set Geometry" variant jointly removes $\mathcal{L}_{\text{orth}}$, temporal-consistency losses, and $\mathcal{L}_{\text{dist}}$.

the sparse component reduces Point-F1 from 0.470 to 0.408. Thus, reliability weighting suppresses unstable constraints, temporal pooling supports persistent anomaly evidence, and sparse aggregation preserves strong localized violations.

Summary. The ablation results show complementary roles for the two stages of ZERO-AD: geometry-aware learning produces informative departures from normality, while calibration and aggregation convert those departures into a robust anomaly score. The complete model achieves the strongest result across all six metrics.

5.3 Parameter Sensitivity

Protocol. All hyperparameters used in the main experiments were fixed before final test evaluation without using test anomaly labels. The sweeps in Figure 4 are post-hoc analyses used only to examine robustness around this reference configuration.



(a) Sensitivity to model and scoring parameters. (b) Sensitivity to geometric regularization weights.

Figure 4: **Parameter sensitivity of ZERO-AD on TSB-AD-M using VUS-PR.** Bubble size is proportional to VUS-PR and stars indicate the best tested configuration. (a) Sensitivity to the number of constraints K, window length L, temporal pooling weight ρ, and sparse-score weight β. (b) Sensitivity to the orthogonality, temporal-consistency, and distance-consistency weights λ_o, λ_t, and λ_d. Overall, ZERO-AD exhibits a clear high-performing region around moderate parameter values rather than requiring extreme settings.

Model and scoring parameters. Figure 4a shows that performance is strongest around the selected settings. VUS-PR increases from 0.281 at $K = 2$ to 0.437 at $K = 8$ and then decreases for larger K, suggesting diminishing benefit from additional constraints. Similarly, $L = 64$ provides the best trade-off between temporal context and localization. Moderate temporal pooling performs best at $\rho = 0.25$, while stronger pooling degrades performance, and the sparse-score weight remains relatively stable around $\beta = 0.15$–0.30.

Geometric regularization. Figure 4b shows a similar pattern for the geometry weights. Performance peaks around $\lambda_o = \lambda_t = 0.10$, while the distance-consistency term performs best at the smaller value $\lambda_d = 0.05$, indicating that it acts as a complementary regularizer rather than the dominant geometric objective.

Summary. Overall, performance varies smoothly around the reference configuration, indicating that ZERO-AD does not rely on a narrowly tuned set of hyperparameters.

6 Conclusion

We presented **ZERO-AD**, a geometric framework for multivariate time-series anomaly detection that represents normal behavior as the zero-level set of learned nonlinear constraints. Instead of relying on reconstruction or prediction errors to define anomaly evidence, ZERO-AD identifies departures directly through violations of these constraints. Jacobian orthogonality, temporal consistency, and distance-based regularization are used to shape the local geometry around normal behavior, while normal-only calibration and dense–sparse aggregation convert the resulting violations into the final anomaly score.

On TSB-AD-M, ZERO-AD achieves the best aggregate performance on five of the six reported metrics while maintaining a compact model size. Ablation studies further show that both the geometry-aware learning objectives and the scoring components make meaningful contributions to the overall performance. These results suggest that directly learning *what normal temporal behavior should satisfy* provides a useful alternative to conventional reconstruction- and prediction-based MTSAD approaches.

ZERO-AD also has several limitations. Its performance can vary across datasets, and precise event localization remains weaker for some time series. Jacobian-based regularization also introduces additional training cost. Moreover, using a fixed number of constraints may be less suitable when normal behavior contains highly heterogeneous or non-stationary regimes. Future work could investigate adaptive constraint dimensionality, regime-aware calibration, and online adaptation to better handle evolving patterns of normal behavior.

AI USE STATEMENT

In this work, we used generative AI tools for paraphrasing, rephrasing, and formatting assistance. All AI-assisted content was carefully reviewed by the authors. We take full responsibility for the final content of the paper, including its text, claims, and any artifacts produced with the assistance of generative AI.

REFERENCES

Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, SIGMOD '00, pp. 93–104, New York, NY, USA, 2000. Association for Computing Machinery. ISBN 1581132174. doi: 10.1145/342009.335388. URL https://doi.org/10.1145/342009.335388.

Zhihao Dai, Ligang He, Shuanghua Yang, and Matthew Leeke. SARAD: Spatial association-aware anomaly detection and diagnosis for multivariate time series. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=gmf5Aj01Hz.

Zahra Zamanzadeh Darban, Geoffrey I Webb, Shirui Pan, Charu C Aggarwal, and Mahsa Salehi. Carla: Self-supervised contrastive representation learning for time series anomaly detection. *Pattern Recognition*, 157:110874, 2025.

Luo donghao and wang xue. ModernTCN: A modern pure convolution structure for general time series analysis. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=vpJMJerXHU.

Bowen Du, Xuanxuan Sun, Junchen Ye, Ke Cheng, Jingyuan Wang, and Leilei Sun. Gan-based anomaly detection for multivariate time series using polluted training set. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12208–12219, 2023. doi: 10.1109/TKDE.2021.3128667.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422, 2008. doi: 10.1109/ICDM.2008.17.

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

Pankaj Malhotra, Lovekesh Vig, Gautam M. Shroff, and Puneet Agarwal. Long short term memory networks for anomaly detection in time series. In *The European Symposium on Artificial Neural Networks*, 2015. URL https://api.semanticscholar.org/CorpusID:43680425.

Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam M. Shroff. Lstm-based encoder-decoder for multi-sensor anomaly detection. *ArXiv*, abs/1607.00148, 2016. URL https://api.semanticscholar.org/CorpusID:9286983.

Jinju Park and Seokho Kang. Paano: Patch-based representation learning for time-series anomaly detection. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=NXThkM7Iym.

Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4393–4402. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/ruff18a.html.

Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471, 2001. doi: 10.1162/089976601750264965.

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 & Data Mining*, KDD '19, pp. 2828–2837, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450362016. doi: 10.1145/3292500.3330672. URL https://doi.org/10.1145/3292500.3330672.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

Qiang Wu, Gechang Yao, Zhixi Feng, and Shuyuan Yang. Peri-midformer: Periodic pyramid transformer for time series analysis. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=5iUxMVJVEV.

Xingjian Wu, Xiangfei Qiu, Zhengyu Li, Yihang Wang, Jilin Hu, Chenjuan Guo, Hui Xiong, and Bin Yang. CATCH: Channel-aware multivariate time series anomaly detection via frequency patching. In *ICLR*, 2025.

Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly transformer: Time series anomaly detection with association discrepancy. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=LzQQ89U1qm_.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling time series with \$10k\$ parameters. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=bWcnvZ3qMb.

Yiyuan Yang, Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. Dcdetector: Dual attention contrastive representation learning for time series anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, pp. 3033–3045, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701030. doi: 10.1145/3580305.3599295. URL https://doi.org/10.1145/3580305.3599295.

Yuxin Zhang, Yiqiang Chen, Jindong Wang, and Zhiwen Pan. Unsupervised deep anomaly detection for multi-sensor time-series signals. *IEEE Transactions on Knowledge and Data Engineering*, 35 (2):2118–2132, 2023. doi: 10.1109/TKDE.2021.3102110.

Hang Zhao, Yujing Wang, Juanyong Duan, Congrui Huang, Defu Cao, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, and Qi Zhang. Multivariate time-series anomaly detection via graph attention network. *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 841–850, 2020. URL https://api.semanticscholar.org/CorpusID:221507538.

Quan Zhou, Changhua Pei, Fei Sun, HanJing, Zhengwei Gao, Haiming Zhang, Gaogang Xie, Dan Pei, and Jianhui li. KAN-AD: Time series anomaly detection with kolmogorov–arnold networks. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=LWQ4zu9SdQ.

A Appendix

B Theoretical Foundations and Computational Properties

This section provides the formal local interpretation of the learned zero-level set, clarifies the role of the number of constraints, and summarizes the computational cost of the Jacobian-based training objective.

B.1 Local Geometric Interpretation of ZERO-AD

We provide a local geometric justification for the ZERO-AD anomaly signal. Let $x = \mathrm{vec}(Z) \in \mathbb{R}^{LD}$ and let $F_\theta(x) \in \mathbb{R}^K$ denote the learned constraint map.

Proposition 1 (Local geometry of the learned zero set)**.** *Assume that F_θ is twice differentiable, that a normal point x satisfies $F_\theta(x) = 0$, and that the Jacobian $J_\theta(x)$ has full row rank K. If*

$$\left\| J_\theta(x) J_\theta(x)^\top - I_K \right\|_2 \leq \varepsilon, \tag{19}$$

then, locally around x, the zero-level set

$$\mathcal{M}_\theta = \{x' : F_\theta(x') = 0\}$$

is a smooth manifold of codimension K, with normal space

$$\mathcal{N}_x = \operatorname{range}\left(J_\theta(x)^\top\right). \tag{20}$$

Moreover, for a sufficiently small perturbation δ,

$$\|F_\theta(x+\delta)\|_2^2 = \|\Pi_{\mathcal{N}_x}\delta\|_2^2 + O\big(\varepsilon\|\delta\|_2^2 + \|\delta\|_2^3\big), \tag{21}$$

where $\Pi_{\mathcal{N}_x}$ denotes projection onto the local normal space.

Proof sketch. By the implicit function theorem, the full-row-rank condition implies that $\mathcal{M}_\theta$ is locally a smooth manifold of codimension K. A first-order Taylor expansion gives

$$F_\theta(x+\delta) = J_\theta(x)\delta + O(\|\delta\|_2^2). \tag{22}$$

Since the rows of $J_\theta(x)$ are approximately orthonormal, $J_\theta(x)^\top J_\theta(x)$ approximates the projection onto $\mathcal{N}_x$. Hence,

$$\|F_\theta(x+\delta)\|_2^2 \approx \delta^\top J_\theta(x)^\top J_\theta(x)\delta \approx \|\Pi_{\mathcal{N}_x}\delta\|_2^2, \tag{23}$$

which gives Eq. equation 21.

Implication for anomaly detection. The proposition provides a direct interpretation of constraint violation. Perturbations that remain approximately tangent to the learned normal set produce little first-order change in F_θ, whereas perturbations with a component along the local normal space produce larger constraint violations. Thus, locally, the ZERO-AD violation energy measures departure away from normal temporal behavior rather than arbitrary variation along it.

Connection to the ZERO-AD objective. The individual training terms support this geometry in complementary ways. $\mathcal{L}_{\text{zero}}$ places normal windows near the zero-level set, while $\mathcal{L}_{\text{orth}}$ encourages complementary and well-conditioned local normal directions. $\mathcal{L}_{\text{tan}}$ encourages observed normal temporal evolution to remain approximately tangent to the learned set. Finally, $\mathcal{L}_{\text{tz}}$ and $\mathcal{L}_{\text{dist}}$ extend these properties beyond purely infinitesimal perturbations by enforcing finite-step temporal consistency and predictable growth away from the zero set.

B.2 Interpretation of the Number of Constraints

The number of constraints K has a geometric meaning beyond its role as an architectural hyperparameter. For a temporal window $Z \in \mathbb{R}^{L\times D}$, the learned constraint map $F_\theta(Z) \in \mathbb{R}^K$ defines the zero-level set

$$\mathcal{M}_\theta = \{Z : F_\theta(Z) = 0\}.$$

When the constraint Jacobian $J_\theta(Z)$ has full row rank, this set has local codimension K, and hence local dimension $LD - K$. In other words, the K constraint gradients define K locally independent directions along which the learned notion of normality can be violated.

This gives a useful interpretation of ZERO-AD. Instead of explicitly modeling all degrees of freedom associated with normal temporal behavior, the model describes normality through a compact collection of complementary local departure directions. It is important, however, to interpret K as the codimension of the *learned* zero-level representation rather than as an exact estimate of the intrinsic codimension of the underlying data-generating process.

The sensitivity analysis is also consistent with this interpretation. When K is too small, the model has only a limited number of departure directions with which to characterize deviations from normal behavior. In contrast, a very large K may introduce redundant or unnecessarily restrictive constraints. Empirically, performance improves as K increases from 2 to 8, and then gradually decreases for larger values. This suggests that a moderate number of constraints is sufficient for the datasets considered in our experiments. We therefore use $K = 8$ across all datasets as a fixed and compact constraint budget, rather than estimating a separate intrinsic dimension for each system.

B.3 Computational Complexity

Let $C_{\mathrm{fwd}}(L, D)$ denote the cost of one forward pass of the constraint network for a window $Z \in \mathbb{R}^{L \times D}$. The network produces K constraint values, so ordinary inference requires

$$O(C_{\mathrm{fwd}}(L, D) + K)$$

operations, where the additional $O(K)$ term corresponds to calibration and constraint-score aggregation. Importantly, no Jacobian is required at inference.

The additional cost of ZERO-AD occurs only during geometry-aware training. The exact constraint Jacobian has size

$$J_\theta(Z) \in \mathbb{R}^{K \times LD}.$$

Using reverse-mode differentiation, constructing the complete Jacobian requires approximately K gradient evaluations, giving a computational cost of roughly

$$O(K\, C_{\mathrm{fwd}}(L, D))\,,$$

in addition to the ordinary training cost. If explicitly materialized, the Jacobian requires $O(KLD)$ memory. Once the Jacobian is available, $J_\theta J_\theta^\top$ costs $O(K^2 LD)$, while Jacobian–vector products used by the tangent objective cost $O(KLD)$.

In our setting, $K = 8$ is small and the Jacobian-based geometric terms are evaluated only once every four minibatches, substantially reducing their amortized training overhead. Nevertheless, the $O(KLD)$ Jacobian size implies that exact geometric regularization can become more expensive for very large window lengths or numbers of variables. This affects training only: after training, ZERO-AD retains its lightweight forward-pass inference without Jacobian computation.

C Reproducibility and Evaluation Protocol

This section summarizes the implementation, optimization, calibration, evaluation, and randomness settings used for ZERO-AD. Unless stated otherwise, the same configuration is applied across all datasets.

The complete hyperparameter configuration is reported in Table 4. Unless explicitly noted, no dataset-specific architecture or optimization settings are used.

C.1 Benchmark Split and Hyperparameter Selection

TSB-AD-M Liu & Paparrizos (2024) contains 200 multivariate time series from 17 dataset families. Following the official TSB-AD protocol and PaAno Park & Kang (2026), the benchmark is divided into a designated *tuning* set containing 20 series and an *evaluation* set containing 180 series. We use the tuning set only for selecting the global hyperparameters. Once selected, the configuration is fixed and applied unchanged to all 180 evaluation series. All aggregate results reported in the paper are therefore computed over these 180 held-out evaluation series.

Each series also includes a predefined split point that separates the training prefix from the subsequent test segment. ZERO-AD uses the available normal training prefix for fitting, validation, and calibration. Within this prefix, we further apply a chronological $70\%/15\%/15\%$ fit/validation/calibration split. No anomaly labels from the 180 evaluation series are used for hyperparameter selection, model fitting, checkpoint selection, or calibration.

Data preprocessing. For each multivariate time series, only the available normal prefix is used for model fitting and calibration. Each channel is independently min–max normalized using statistics estimated from this normal prefix, and the same transformation is then applied to the full series. Anomalous samples and anomaly labels are not used during training, validation, or calibration.

The normal prefix is divided chronologically into 70% fitting, 15% validation, and 15% calibration subsets. Using a chronological split helps avoid leakage from future observations.

Model inputs are constructed using sliding windows of length $L = 64$. Training and validation windows are sampled with strides of 4 and 8, respectively, with a maximum of 10,000 training

Table 4: **ZERO-AD reproducibility configuration.** Unless otherwise stated, the same architecture, optimization, normal-only calibration, and scoring parameters are used across all datasets.

Hyperparameter	**Value**
Input and architecture	
Window length L	64
Hidden dimension H	64
Number of constraints K	8
Number of temporal blocks	4
Temporal dilations	$\{1, 2, 4, 8\}$
Depth-wise convolution kernel	5
Dropout	0.05
Input representation	$[Z, \Delta Z]$
Data splitting and sampling	
Fit / validation / calibration	$70\%/15\%/15\%$
Training-window stride	4
Validation-window stride	8
Temporal tangent shift	1
Maximum training windows	10,000
Maximum validation windows	2,000
Optimization	
Optimizer	AdamW
Training epochs	5
Batch size	32
Learning rate	3×10^{-4}
Weight decay	1×10^{-5}
Gradient clipping	1.0
Early-stopping patience	3 epochs
Minimum improvement	1×10^{-5}
Zero-set objective	
λ_0	1.00
λ_o	0.10
λ_t	0.10
λ_d	0.05
λ_{tz}	0.10
Normal displacement ϵ_n	0.05
Tangent displacement ϵ_t	0.03
Jacobian mode	Exact
Jacobian update interval	Every 4 minibatches
Calibration and scoring	
MAD scale factor c_{MAD}	1.4826
Constraint scale floor	1×10^{-3}
Reliability blend η	0.50
Reliability clipping range	$[0.35, 2.50]$
Temporal pooling radius R	2
Temporal pooling weight ρ	0.25
Sparse-score weight β	0.15
Inference batch size	512
Random seeds	$\{42, 123, 2027\}$

windows and 2,000 validation windows per series. For each window $Z \in \mathbb{R}^{L \times D}$, the network receives the augmented representation

$$X = [Z, \Delta Z],$$

where ΔZ denotes the first-order temporal difference. The empirical temporal tangent used by the geometric objective is computed using a one-step temporal shift.

Constraint network. ZERO-AD uses a lightweight temporal constraint network with hidden dimension $H = 64$ and $K = 8$ learned constraints. The network consists of four temporal blocks

with dilations $\{1, 2, 4, 8\}$ and depth-wise convolution kernels of size 5. Dropout is set to 0.05. For a temporal window Z, the network produces

$$F_\theta(Z) = [f_1(Z), \ldots, f_K(Z)]^\top \in \mathbb{R}^K,$$

whose common zero set defines the learned representation of normal behavior.

Training objective. The full ZERO-AD training objective is

$$\mathcal{L} = \lambda_0 \mathcal{L}_0 + \lambda_o \mathcal{L}_{\mathrm{orth}} + \lambda_t \mathcal{L}_{\tan} + \lambda_d \mathcal{L}_{\mathrm{dist}} + \lambda_{tz} \mathcal{L}_{\mathrm{tz}},$$

with

$$(\lambda_0, \lambda_o, \lambda_t, \lambda_d, \lambda_{tz}) = (1.0, 0.10, 0.10, 0.05, 0.10).$$

Here, $\mathcal{L}_0$ encourages normal windows to remain close to the zero-level set, $\mathcal{L}_{\mathrm{orth}}$ regularizes the local Jacobian geometry, $\mathcal{L}_{\tan}$ preserves observed temporal tangent directions, $\mathcal{L}_{\mathrm{dist}}$ encourages informative responses when moving away from the normal set, and $\mathcal{L}_{\mathrm{tz}}$ encourages zero-set consistency under tangent perturbations.

The normal and tangent perturbation magnitudes are fixed to $\epsilon_n = 0.05$ and $\epsilon_t = 0.03$, respectively. The Jacobian is computed exactly rather than approximated using stochastic Jacobian probes. Because exact Jacobian regularization is more computationally expensive than the remaining objective terms, the Jacobian-based geometric terms are updated once every four minibatches.

Optimization and model selection. All models are optimized with AdamW for at most five epochs. Checkpoint selection uses only the held-out normal validation windows. Specifically, at each epoch we evaluate the same normal-only objective used for training,

$$\mathcal{L}_{\mathrm{val}} = \lambda_0 \mathcal{L}_{\mathrm{zero}} + \lambda_o \mathcal{L}_{\mathrm{orth}} + \lambda_t \mathcal{L}_{\tan} + \lambda_d \mathcal{L}_{\mathrm{dist}} + \lambda_{tz} \mathcal{L}_{\mathrm{tz}}, \tag{24}$$

and select

$$\theta^* = \arg\min_{\theta_e} \frac{1}{|\mathcal{V}|} \sum_{Z \in \mathcal{V}} \mathcal{L}_{\mathrm{val}}(Z; \theta_e), \tag{25}$$

where $\mathcal{V}$ contains only normal validation windows. Early stopping uses patience 3 with a minimum improvement of 10^{-5}. Neither anomaly labels nor test-set detection metrics are used for checkpoint selection.

Normal-only calibration. After training, the held-out normal calibration subset is passed through the selected constraint network. For each constraint k, we estimate its robust center and scale as

$$m_k = \mathrm{median}_t\, f_k(Z_t), \qquad s_k = 1.4826\ \mathrm{median}_t\, |f_k(Z_t) - m_k| .$$

A minimum scale of 10^{-3} is imposed for numerical stability. At inference, constraint responses are standardized as

$$z_{t,k} = \frac{f_k(Z_t) - m_k}{s_k + \epsilon}, \qquad E_{t,k} = z_{t,k}^2.$$

Constraints exhibiting large variability on normal calibration data receive smaller reliability weights. The reliability contribution is blended with uniform weighting using $\eta = 0.5$ and clipped to the range $[0.35, 2.50]$ before normalization. All calibration quantities are therefore estimated exclusively from normal observations.

Temporal and sparse scoring. To reduce sensitivity to isolated fluctuations, the per-constraint energies are temporally aggregated using a centered triangular filter with radius $R = 2$. The original and pooled energies are combined with temporal-pooling weight $\rho = 0.25$. ZERO-AD then forms a reliability-weighted dense score and a sparse maximum-violation score,

$$S_t^{\mathrm{dense}} = \sum_{k=1}^{K} w_k E_{t,k}^*, \qquad S_t^{\mathrm{sparse}} = \max_k E_{t,k}^*.$$

The final continuous anomaly score is

$$S_t = (1 - \beta) S_t^{\mathrm{dense}} + \beta S_t^{\mathrm{sparse}}, \qquad \beta = 0.15.$$

Thus, the dense term captures coordinated deviations across multiple constraints, whereas the sparse term preserves strong violations localized to only a few constraint directions.

Inference. At test time, the trained model parameters and all calibration statistics are kept fixed. Windows are evaluated with an inference batch size of 512, and each window requires only a forward pass through the constraint network followed by the deterministic calibration and score aggregation described above. No fine-tuning, test-time adaptation, or anomaly-label information is used during inference.

Evaluation protocol. We use the official TSB-AD-M evaluation implementation for all reported metrics. AUC-PR, AUC-ROC, VUS-PR, and VUS-ROC are computed directly from the continuous anomaly scores and therefore do not require the selection of an operating threshold. For threshold-dependent metrics, we follow the oracle evaluation protocol used by the benchmark. Point-F1 is reported as the maximum F1 obtained over the precision–recall thresholds of the test sequence. Range-F1 is computed by evaluating 100 uniformly spaced thresholds between the minimum and maximum anomaly scores and reporting the best range-based F1.

These oracle thresholds are used only by the benchmark evaluator after inference to summarize detection performance. Test anomaly labels are not used during model fitting, checkpoint selection, normal calibration, anomaly-score construction, or test-time adaptation. ZERO-AD therefore produces a label-free continuous anomaly score, while ground-truth test labels are used only by the standard benchmark evaluation procedure for threshold-dependent metrics.

Randomness and implementation. All experiments are repeated using three random seeds, $\{42, 123, 2027\}$. For each run, the same seed is used consistently for model initialization, data sampling, and other stochastic operations during training. The configuration reported in Table 4 is kept fixed across datasets and seeds unless explicitly stated otherwise. We report the mean performance over the three runs, together with the corresponding standard deviation where applicable.

All ZERO-AD experiments use the same model architecture, calibration procedure, and scoring rule. No dataset-specific anomaly labels are used for model fitting, checkpoint selection, calibration, or hyperparameter selection.

Absence of anomaly-label leakage. The fit, validation, and calibration subsets are constructed entirely from the available normal training prefix. The fitting subset is used to optimize the network parameters, the validation subset is used for checkpoint selection, and the calibration subset is used to estimate robust constraint statistics and reliability weights. Test anomaly labels are therefore never used for model fitting, checkpoint selection, score calibration, hyperparameter selection, or construction of the anomaly score.

Ground-truth labels from the test sequence are accessed only by the benchmark evaluator after inference when computing the final evaluation metrics. This includes the oracle-threshold Point-F1 and Range-F1 metrics described above.

D Complete Experimental Results

The main paper reports the central aggregate results and a compact dataset-wise summary. Here we provide the complete baseline comparison, the full 17-dataset comparison against PaAno, and an additional control separating representation learning from score aggregation.

D.1 Full TSB-AD-M Method Comparison

Table 5 gives the complete comparison against all baseline methods reported in the corresponding PaAno benchmark table, together with ranks recomputed after adding ZERO-AD.

D.2 Complete Dataset-Wise Comparison with PaAno

Table 6 reports the complete six-metric comparison between ZERO-AD and PaAno across all 17 TSB-AD-M dataset families. This table complements the compact dataset-wise discussion in the main paper and makes both favorable and unfavorable cases explicit.

Table 5: **Performance comparison on TSB-AD-M.** Baseline results are taken from Table 17 of PaAno. Ranks are recomputed after adding ZERO-AD. The best result is shown in **bold** and the second-best is underlined.

Type	Method	VUS-PR		VUS-ROC		Range-F1		AUC-PR		AUC-ROC		Point-F1	
		Score	Rank	Score	Rank	Score	Rank	Score	Rank	Score	Rank	Score	Rank
Statistical / ML	PCA	0.31	4	0.74	5	0.29	12	0.31	5	0.70	5	0.37	4
	NLinear	0.30	9	0.70	14	0.21	20	0.26	14	0.65	15	0.31	14
	DLinear	0.29	15	0.70	14	0.21	20	0.27	11	0.66	13	0.32	11
	KMeansAD	0.29	15	0.73	7	0.33	8	0.25	16	0.69	7	0.31	14
	CBLOF	0.27	17	0.70	14	0.31	10	0.28	9	0.67	9	0.32	11
	MCD	0.27	17	0.69	17	0.20	25	0.27	11	0.65	15	0.33	9
	OCSVM	0.26	19	0.67	23	0.30	11	0.23	19	0.61	24	0.28	20
	RobustPCA	0.24	21	0.61	29	0.33	8	0.24	17	0.58	26	0.29	17
	EIF	0.21	22	0.71	10	0.26	14	0.19	23	0.67	9	0.26	23
	COPOD	0.20	25	0.69	17	0.24	16	0.20	21	0.65	15	0.27	22
	IForest	0.20	25	0.69	17	0.24	16	0.19	23	0.66	13	0.26	23
	HBOS	0.19	27	0.67	23	0.24	16	0.16	25	0.63	23	0.24	25
	KNN	0.18	29	0.59	31	0.21	20	0.14	28	0.51	32	0.19	30
	LOF	0.14	31	0.60	30	0.14	31	0.10	31	0.53	30	0.15	31
Neural Network	KAN-AD	0.40	3	0.76	3	0.41	2	0.35	3	0.73	3	0.40	3
	DADA	0.31	4	0.73	7	0.25	15	0.31	5	0.69	7	0.35	7
	DeepAnT	0.31	4	0.76	3	0.37	5	0.32	4	0.73	3	0.37	4
	LSTMAD	0.31	4	0.74	5	0.38	4	0.31	5	0.70	5	0.36	6
	OmniAnomaly	0.31	4	0.69	17	0.37	5	0.27	11	0.65	15	0.32	11
	AutoEncoder	0.30	9	0.69	17	0.28	13	0.30	8	0.67	9	0.34	8
	USAD	0.30	9	0.68	22	0.37	5	0.26	14	0.64	20	0.31	14
	Donut	0.26	19	0.71	10	0.21	20	0.20	21	0.64	20	0.28	20
	FITS	0.21	22	0.66	25	0.16	30	0.15	26	0.58	26	0.22	26
	TimesNet	0.19	27	0.64	27	0.17	28	0.13	30	0.56	28	0.20	29
	TranAD	0.18	29	0.65	26	0.21	20	0.14	28	0.59	25	0.21	27
Transformer	CATCH	0.30	9	0.72	9	0.18	26	0.23	19	0.65	15	0.29	17
	iTransformer	0.30	9	0.71	10	0.18	26	0.24	17	0.64	20	0.29	17
	PatchTST	0.30	9	0.71	10	0.22	19	0.28	9	0.67	9	0.33	9
	OFA	0.21	22	0.63	28	0.17	28	0.15	26	0.55	29	0.21	27
	AnomalyTransformer	0.12	32	0.57	32	0.14	31	0.07	32	0.52	31	0.12	32
	DCdetector	0.10	33	0.54	33	0.09	33	0.05	33	0.48	33	0.10	33
Patch-based	PaAno	0.43	2	**0.79**	1	0.41	2	0.38	2	0.76	2	0.43	2
Geometry-based	**ZERO-AD (Ours)**	**0.437**	**1**	0.780	2	**0.435**	**1**	**0.396**	**1**	**0.782**	**1**	**0.470**	**1**

D.3 Separating Geometry from Score Aggregation

Representation versus scoring. Table 7 helps disentangle the contribution of the learned zero-set geometry from that of the calibrated scoring pipeline. Applying the full scoring procedure to the zero-only model improves its performance, indicating that robust calibration and score aggregation are themselves important for detection. At the same time, keeping the learned geometry while replacing the full scoring pipeline with a simple mean constraint-energy score still yields a clear improvement over the zero-only representation. The complete ZERO-AD model performs best, suggesting that geometry-aware constraint learning and calibrated dense–sparse scoring provide complementary gains, rather than the overall improvement being explained only by post-processing.

E Failure-Case Analysis

This section complements the successful qualitative examples in the main paper with representative cases where ZERO-AD is less effective. We consider two different types of failure: one mainly caused by calibration mismatch and another caused by limited separation between anomalous and normal behavior in the learned constraint space.

Table 6: **Dataset-wise comparison with PaAno on all 17 TSB-AD-M datasets.** For each metric, PaAno and ZERO-AD are shown in separate columns and the better value is highlighted in **bold**. The final column reports the number of metrics on which ZERO-AD outperforms PaAno.

Dataset	VUS-PR		VUS-ROC		Range-F1		AUC-PR		AUC-ROC		Point-F1		Wins
	PaAno	Ours	PaAno	Ours	PaAno	Ours	PaAno	Ours	PaAno	Ours	PaAno	Ours	
CATSv2	0.0690	**0.1892**	**0.6881**	0.6396	0.1381	**0.3469**	0.0744	**0.2482**	**0.6813**	0.6467	0.1285	**0.3261**	4/6
CreditCard	0.0343	**0.0577**	0.5558	**0.7142**	**0.0351**	0.0237	**0.0068**	0.0058	0.5652	**0.6802**	**0.0416**	0.0253	3/6
Daphnet	0.2706	**0.2969**	**0.8925**	0.8874	**0.3117**	0.3018	0.2564	**0.2852**	**0.9052**	0.8876	**0.4189**	0.4152	2/6
Exathlon	0.7769	**0.9324**	0.9545	**0.9725**	0.6045	**0.8111**	0.7482	**0.9186**	0.9518	**0.9705**	0.8357	**0.9220**	**6/6**
GECCO	0.1601	**0.3155**	0.8727	**0.9278**	**0.2268**	0.0996	0.2192	**0.3174**	0.9050	**0.9105**	0.2959	**0.3955**	5/6
GHL	0.0085	**0.0296**	0.3391	**0.4284**	0.0240	**0.0346**	0.0073	**0.0297**	0.3090	**0.4231**	0.0241	**0.0587**	**6/6**
Genesis	0.2964	**0.6780**	0.9902	**0.9985**	**0.2930**	0.2740	0.1945	**0.2098**	0.9867	**0.9939**	0.3023	**0.3412**	5/6
LTDB	**0.6066**	0.4218	**0.8640**	0.7508	**0.5965**	0.4175	**0.5856**	0.3924	**0.8327**	0.7120	**0.5861**	0.4875	0/6
MITDB	**0.3841**	0.1981	**0.8803**	0.7352	**0.4647**	0.3494	**0.4264**	0.2681	**0.8567**	0.6996	**0.4864**	0.3307	0/6
MSL	0.2416	**0.3842**	0.7768	**0.8296**	0.3630	**0.3671**	0.1969	**0.3106**	0.7498	**0.7781**	0.2937	**0.4045**	**6/6**
OPPORTUNITY	**0.1701**	0.1245	**0.6069**	0.4853	**0.3463**	0.2212	**0.1646**	0.1107	**0.5810**	0.4840	**0.2134**	0.2126	0/6
PSM	0.2033	**0.2613**	0.6597	**0.7115**	**0.2749**	0.1985	0.1839	**0.2743**	0.6571	**0.7435**	0.2621	**0.3460**	5/6
SMAP	**0.5363**	0.4141	**0.9009**	0.8365	**0.5860**	0.4583	**0.5220**	0.3477	**0.9055**	0.8344	**0.5428**	0.4934	0/6
SMD	**0.3545**	0.3045	**0.8479**	0.8227	**0.4044**	0.3712	**0.3850**	0.2215	**0.8728**	0.8000	**0.4445**	0.3658	0/6
SVDB	**0.5574**	0.3152	**0.9050**	0.7744	**0.5436**	0.3107	**0.5259**	0.3316	**0.8814**	0.7221	**0.5496**	0.3675	0/6
SWaT	0.2428	**0.4246**	0.6864	**0.7085**	**0.3421**	0.2642	0.2174	**0.5357**	0.6679	**0.8111**	0.3213	**0.6626**	5/6
TAO	0.7273	**0.7351**	0.8535	**0.8648**	**0.1848**	0.1487	**0.0849**	0.0819	**0.4966**	0.4831	**0.1591**	0.1583	2/6

Table 7: Separating the contributions of zero-set geometry and anomaly-score aggregation.

Variant	VUS-PR	Range-F1
Zero-only + Simple Score	0.355	0.330
Zero-only + Full Score	0.394	0.385
Full Geometry + Simple Score	0.397	0.378
Full ZERO-AD	**0.437**	**0.435**

Failure-case analysis. Figure 5 shows two representative failure modes of ZERO-AD. These examples complement the successful qualitative result in Figure 3 and help clarify the situations in which the learned zero-set representation remains challenging.

CreditCard: calibration-induced false positives. As shown in Figure 5, the labeled anomaly in CreditCard does produce a clear geometric departure. Several constraints become strongly activated around the anomalous interval, the trajectory moves outside the normal calibration region in the learned constraint space, and the final anomaly score rises well above the normal-only threshold. Thus, in this case, ZERO-AD successfully reacts to the anomalous event itself.

The main difficulty is that elevated scores also appear outside the annotated anomalous interval. The constraint heatmap shows noticeable activity during some unlabeled regions, and the local trajectory occasionally moves outside the compact calibration boundary even when no anomaly is annotated. This suggests that the variability observed later in the sequence is not fully represented by the normal calibration distribution. As a result, legitimate but previously unseen normal behavior may also be interpreted as a zero-set violation, leading to false positives. This issue is especially important for highly imbalanced datasets such as CreditCard, where even a small number of false alarms can substantially reduce precision-oriented metrics.

GHL: insufficient anomaly–normal separation. GHL exhibits a different type of failure. The normal constraint responses already show considerable variation, with several constraints remaining active over a broad portion of the sequence. In the learned constraint-space projection, the anomalous trajectory does not form a clearly separated region; instead, much of it remains within or close to the broader normal distribution.

Consequently, the ZERO-AD score stays mostly below the normal-only threshold during the labeled anomalous interval, while a later unlabeled region produces a stronger score increase. This suggests

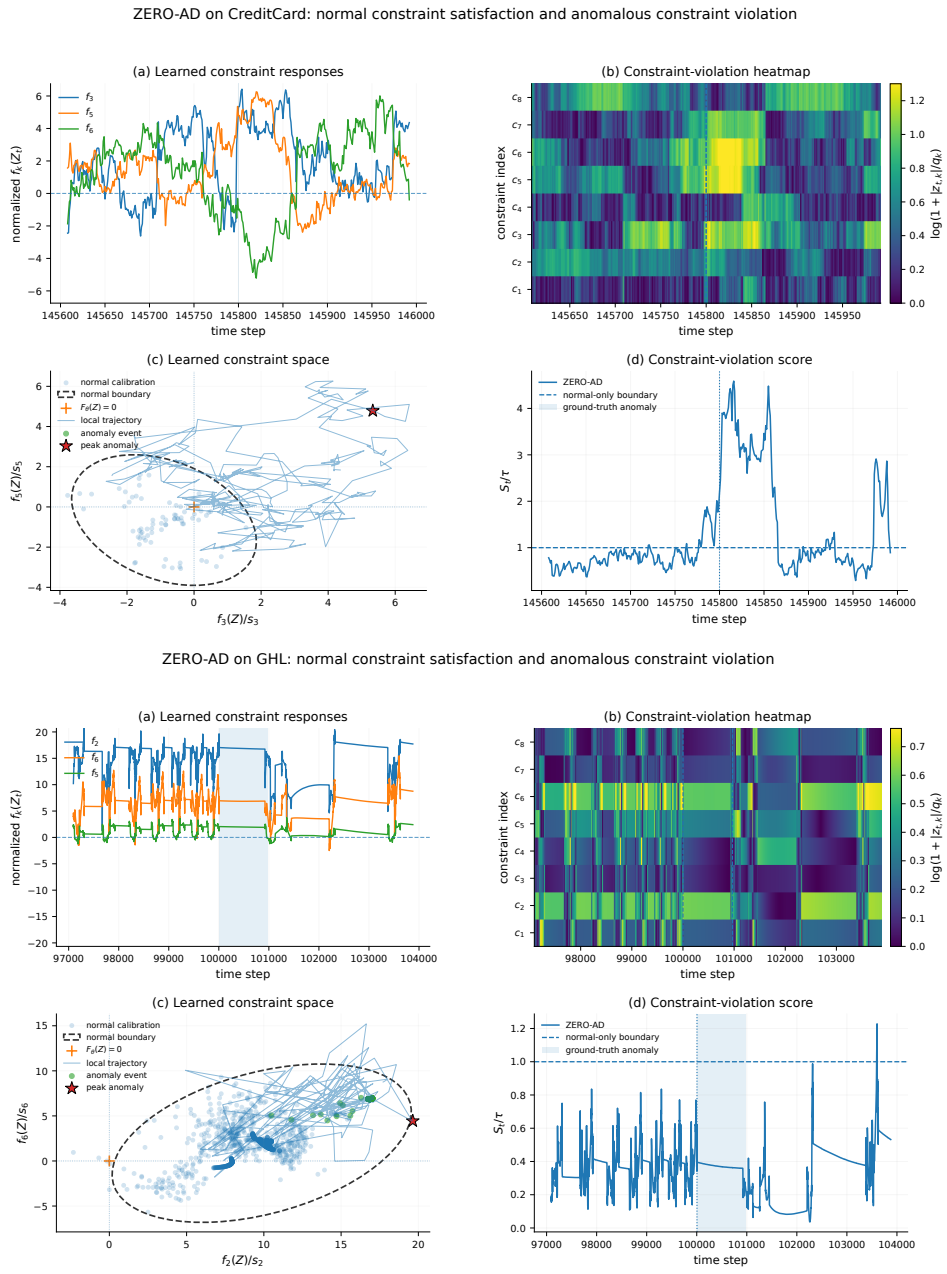


Figure 5: **Failure-case analysis of ZERO-AD on CreditCard (top) and GHL (bottom).** CreditCard shows a calibration-related failure, where the true anomaly produces a clear constraint violation but some normal regions also receive high anomaly scores. GHL shows a separability-related failure, where the labeled anomalous trajectory remains close to the learned normal region and therefore produces only a weak anomaly response.

that the learned constraints do not provide sufficient geometric separation between the anomalous behavior and the heterogeneous normal patterns in this example. The difficulty therefore cannot be explained by threshold selection alone: the anomaly itself lies in a region that is relatively difficult to distinguish from normal variation in the learned constraint space.

Implications. These two examples highlight complementary limitations of ZERO-AD. CreditCard mainly reveals a *calibration* issue: the representation responds strongly to the true anomaly, but

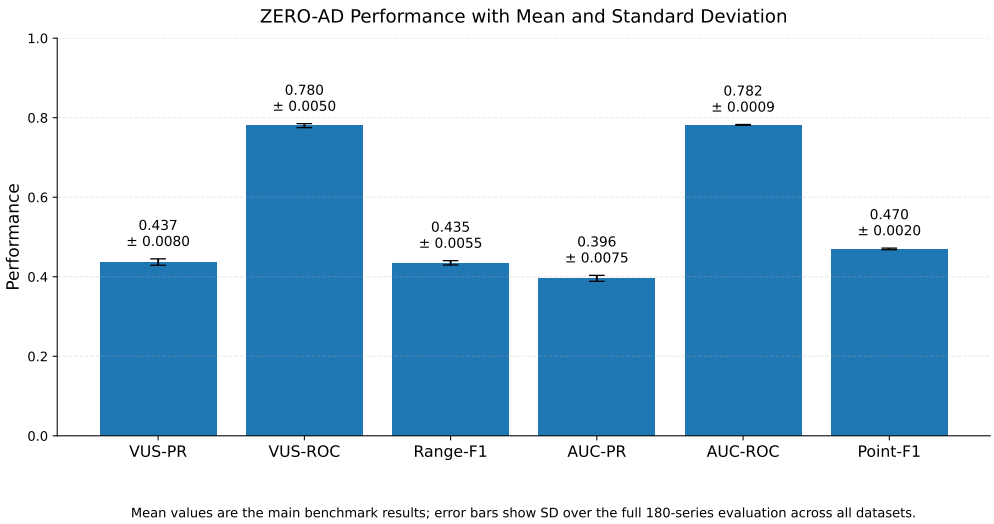


Figure 6: Stability of ZERO-AD across three random seeds on the full TSB-AD-M evaluation setup. Bar heights show the mean performance, and error bars indicate the standard deviation across seeds. The consistently small error bars across all six metrics show that ZERO-AD is stable under random initialization.

the learned normal calibration does not fully capture later variations in nominal behavior. GHL instead reveals a *separability* issue: the anomalous trajectory overlaps with the learned normal region, resulting in relatively weak constraint violations. These observations suggest two useful directions for future improvement: adaptive or regime-aware calibration to reduce false positives under changing normal behavior, and richer local constraint geometry to improve separation when anomalies closely resemble legitimate normal variation.

F Stability Across Random Seeds

To assess the robustness of ZERO-AD to random initialization, we repeat the full evaluation under three random seeds while keeping all model, optimization, and scoring hyperparameters fixed. The experiment is conducted on the complete TSB-AD-M setup, covering all 180 evaluation time series across all dataset families. Figure 6 summarizes the mean and standard deviation of the resulting performance.

The standard deviations are small across all metrics: 0.0080 for VUS-PR, 0.0050 for VUS-ROC, 0.0055 for Range-F1, 0.0075 for AUC-PR, 0.0009 for AUC-ROC, and 0.0020 for Point-F1. This indicates that the performance of ZERO-AD is highly consistent across random seeds. In particular, the very small deviations in VUS-ROC and AUC-ROC suggest stable ranking behavior, while the similarly small deviations in VUS-PR, AUC-PR, Range-F1, and Point-F1 show that threshold-sensitive and range-sensitive detection performance is also robust. Overall, these results demonstrate that the improvements of ZERO-AD are not due to favorable random initialization, but reflect stable behavior under repeated runs.