

Multivariate Time Series Anomaly Detection with Idempotent Reconstruction

Appendix

Contents

1	Introduction	1
2	Related Work	3
2.1	Multivariate Time Series Anomaly Detection	3
2.2	Idempotent Generative Network	3
3	Method	3
3.1	Overview	3
3.2	Problem Setting	3
3.3	Optimization Objective	4
3.3.1	Reconstruction Objective	4
3.3.2	Idempotent Objective	4
3.3.3	Tightness Objective	5
3.3.4	Final Objective	6
4	Experiment	6
4.1	Experiment Setting	6
4.1.1	Dataset and Baseline	6
4.1.2	Implementation Detail	6
4.1.3	Evaluation Metric	7
4.2	Result	8
4.2.1	Anomaly Detection w/ or w/o IGAD	8
4.2.2	Balance of Robustness and Sensitivity	8
4.2.3	Difference in Distributions of Abnormal Scores	8
4.3	Ablation Study	9
4.3.1	Different Loss Functions	9
4.3.2	The Effectiveness of Each Objective	9
4.4	The Effect of IGAD on Foundational Models	10
4.5	Comparison between Frequency Resampling and Time-Domain PCA	10
5	Conclusion	10
A	Theoretical Analysis of <i>Over Generalization</i>	25
A.1	Contaminated Data for Training and Over Expressive Models	25
A.2	Optimization Dynamics with Regularization Constraint	25
A.3	Mechanistic Interpretation of Over Generalization	26

A.3.1	Density-Driven Error Scaling	26
A.3.2	Latent Manifold Attraction	27
B	Proof of the convergence for IGAD	28
C	Explanations for Basic Models	29
C.1	Models Designed for MTS AD	29
C.2	Time Series Foundation Model	31
D	Hyperparameter Setting	33
E	More Detailed Experimental Results	34
E.1	Distinguishable Distributions of Anomaly Scores and Auxiliary Metrics	34
E.2	Efficiency Evaluation	43
E.3	Hyperparameter Analysis	45
E.4	Visualization of Latent Space	48
E.5	Maintain Data Patterns under Noise	49
E.6	Nyquist Criteria	53
E.7	Comparison with Contrastive-based Models	58
E.8	Codes for IGAD	59
F	Limitation and Future Work	59
G	Impact Statements	59

A Theoretical Analysis of Over Generalization

As we have discussed in Sect.1, we conclude the reasons for over generalization as that this problem may happen for two factors:

- A model incorrectly captures the intrinsic patterns of abnormal series in a contaminated dataset for training. This is **unavoidable** in real-world scenarios because large amounts of time series data are collected under the assumption that all time instances collected are normal points, since a system typically operates correctly under most conditions. However, it is inevitable that the system will have offsets or abnormal states over the long period of the data collection process. Meanwhile, since manually checking each time point for anomalies consumes a lot of human and time resources, these abnormal instances remain in the dataset for training without label scrutiny.
- A model has an excessive decoding power, even for abnormal series. This means that a model not only learns how to reconstruct normal time instances but also gains the ability to learn how to reconstruct abnormal instances.

In this part, we provide mathematical proofs and analysis for over generalization to explore how these two factors have an effect on the training process. Concretely, in Sect.A.1, we show the explanations in ideal and pure models, which have enough model capacity to learn everything. In addition, in Sect.A.2, we provide proofs and analysis in more general cases, and the effect of *regularization* is introduced, which aligns with the traditional principles of model design.

A.1 Contaminated Data for Training and Over Expressive Models

Let \mathcal{P}_x and \mathcal{P}_a denote the distributions of normal and abnormal instances, respectively. When training data contains contaminated anomalies with rate $\eta \in [0, 1)$, the empirical distribution will be transformed to $\mathcal{P}_{\text{train}} = (1 - \eta)\mathcal{P}_x + \eta\mathcal{P}_a$. A reconstruction-based model $f : \mathcal{X} \rightarrow \mathcal{X}$ trained on the contaminated data aims to minimize the composite objective:

$$\mathcal{L}_{\text{recon}}(f(\cdot)) = (1 - \eta)\mathbb{E}_{x \sim \mathcal{P}_x} [\|x - f(x)\|_2^2] + \eta\mathbb{E}_{x \sim \mathcal{P}_a} [\|x - f(x)\|_2^2]. \quad (15)$$

Here, we consider a compact support $K \subset \mathcal{X}$ where $\text{Supp}(\mathcal{P}_x) \cup \text{Supp}(\mathcal{P}_a) \subseteq K$ and $\mathcal{P}_x, \mathcal{P}_a$ are absolutely continuous on K . By the universal approximation theorem [22], in a given assumption space \mathcal{H} , for any compact support $K \subset \mathcal{X}$ containing both normal and anomalous instances, there exists a neural network $f \in \mathcal{H}$, which can satisfy the following objective:

$$\sup_{x \in K} \|x - f(x)\|_2^2 \leq \epsilon, \quad \forall \epsilon > 0. \quad (16)$$

This implies that sufficiently expressive models can achieve arbitrarily small reconstruction errors on both distributions simultaneously. This theoretical capacity relies on the complexity of the unbounded model. Practical architectures with inductive biases, for example, the commonly selected bottleneck constraint, and other implicit regularization introduced during the design of the model alter the solution landscape.

A.2 Optimization Dynamics with Regularization Constraint

For more general cases in practice, when $\eta > 0$, we consider an encoder-decoder-like mapping function $f_\theta(x) = g_\phi(h_\psi(x))$ with a bottleneck dimension sufficiently smaller than the data dimension, that is, $d_{\text{bottleneck}} \ll d_{\text{data}}$, which enforces information compression. This is a demonstrated strategy in this type of structures because it enforces the learning of compressed representations that retain the intrinsic structure of the data while discarding redundant information [21, 38], thus preventing trivial identity mappings [5] and promoting robust feature disentanglement [20]. In terms of Eq.(15), the optimization dynamics is governed by:

$$(1 - \eta)\nabla_\theta \mathbb{E}_x [\|x - f_\theta(x)\|_2^2] + \eta\nabla_\theta \mathbb{E}_{x'} [\|x' - f_\theta(x')\|_2^2] + \gamma\nabla_\theta \mathcal{R}(\theta) = 0, \quad (17)$$

where $\mathcal{R}(\theta)$ captures architectural constraints through implicit regularization and γ quantifies the effective regularization strength from the bottleneck. To elucidate this equilibrium, more generally, we expanding the loss in the parameter space with $\mathcal{R}(\theta)$:

$$\mathcal{L}_{\text{recon}}(f_\theta) = \int_K \|x - f_\theta(x)\|_2^2 [(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)] dx + \gamma\mathcal{R}(\theta). \quad (18)$$

For reconstruction-based methods, $\mathcal{R}(\theta)$ can be introduced naturally to $\mathcal{R}(\theta) = \|f_\theta(x) - \mathbb{E}[x|\theta]\|_2^2$. Here, $\mathbb{E}[x|\theta] \triangleq \mathbb{E}_{z \sim p(z|\theta)}[g_\phi(z)]$ with $p(z|\theta)$ is the empirical distribution induced by the encoder with $z \triangleq h_\psi(x)$. This setting is justified by its alignment with the conservative estimation principle of the traditional design approach in the training process, that is, the model output is constrained to the typical pattern of the data itself, making it possible to reconstruct every time instance well. Crucially, since the training data are contaminated for $\eta > 0$, the empirical distribution $p(z|\theta)$ and $\mathbb{E}[x|\theta]$ are influenced by both normal and abnormal instances. To form the corresponding Euler-Lagrange equation, we take the functional derivative $\frac{\delta \mathcal{L}_{\text{recon}}(f_\theta)}{\delta f_\theta} = 0$ to get the expression:

$$[(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)](x - f_\theta(x)) = \gamma(f_\theta(x) - \mathbb{E}[x|\theta]), \quad (19)$$

We solve this elliptic equation under the strong bottleneck condition ($\gamma \gg \eta\mathcal{P}_a(x)$ and $\gamma \ll [(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)]$):

$$f_\theta^*(x) = \frac{[(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)]x + \gamma\mathbb{E}[x|\theta]}{(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x) + \gamma} \quad (20)$$

$$= \mathbb{E}[x|\theta] + \frac{(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)}{(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x) + \gamma}(x - \mathbb{E}[x|\theta]) \quad (21)$$

$$= \mathbb{E}[x|\theta] + \frac{1}{1 + \frac{\gamma}{(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)}}(x - \mathbb{E}[x|\theta]) \quad (22)$$

$$\approx \underbrace{\mathbb{E}[x|\theta] + \left(1 - \frac{\gamma}{(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)} + \mathcal{O}\left(\frac{\gamma^2}{[(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)]^2}\right)\right)}_{\text{Taylor Expansion}}(x - \mathbb{E}[x|\theta]) \quad (23)$$

$$= \mathbb{E}[x|\theta] + \left(1 - \frac{\gamma}{(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)}\right)(x - \mathbb{E}[x|\theta]) + \mathcal{O}\left(\frac{\gamma^2}{[(1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)]^2}\right) \cdot \frac{x - \mathbb{E}[x|\theta]}{\|x - \mathbb{E}[x|\theta]\|_2}. \quad (24)$$

A.3 Mechanistic Interpretation of Over Generalization

The analytical decomposition from Eq.(20) to Eq.(24) reveals the fundamental mechanisms that govern the reconstruction behavior.

Theorem 1 When $\eta > 0$ and $\gamma > 0$, the optimal reconstruction function $f_\theta^*(x)$ satisfies the following expression, which can be acquired from Eq.(24):

$$\|f_\theta^*(x) - x\|_2 \leq \underbrace{\frac{\gamma}{A(x)}\|x - \mathbb{E}[x|\theta]\|_2}_{\text{Anomaly Suppression}} + \underbrace{\mathcal{O}\left(\frac{\gamma^2}{A(x)^2}\right)}_{\text{Unit Vector}} \cdot \underbrace{\zeta}_{\text{Unit Vector}} \quad (25)$$

$$= \frac{\gamma}{A(x)}\|x - \mathbb{E}[x|\theta]\|_2 + \mathcal{O}\left(\frac{\gamma^2}{A(x)^2}\right) \quad (26)$$

where $A(x) \triangleq (1 - \eta)\mathcal{P}_x(x) + \eta\mathcal{P}_a(x)$ represents the local data density mixture at point x and $\zeta \triangleq \frac{x - \mathbb{E}[x|\theta]}{\|x - \mathbb{E}[x|\theta]\|_2}$ is the term for direction correction.

A.3.1 Density-Driven Error Scaling

The decomposition of the optimal reconstruction function $f_\theta^*(x)$ in Eq.(24) and Eq.(25) reveals a critical mechanism that governs over generalization. With the defined *mixed local density* $A(x)$ and *regularization strength* γ , the primary error term $\frac{\gamma}{A(x)}$ exhibits an inverse proportionality to $A(x)$, leading to the following regimes:

- **High-Density Regions** ($\gamma \ll A(x)$):

$$\frac{\gamma}{A(x)} \rightarrow 0 \implies \|f_\theta^*(x) - x\|_2 \approx \mathcal{O}\left(\frac{\gamma^2}{A(x)^2}\right) \implies f_\theta^*(x) \approx x. \quad (27)$$

Accurate reconstructions dominate as the density of normal instances suppresses abnormal residuals.

- **Low-Density Regions** ($\gamma \sim A(x)$):

$$\frac{\gamma}{A(x)} \approx 1 \implies \|f_\theta^*(x) - x\|_2 \approx \|\mathbb{E}[x|\theta] - x\|_2 + \mathcal{O}\left(\frac{\gamma^2}{A(x)^2}\right) \implies f_\theta^*(x) \approx \mathbb{E}[x|\theta]. \quad (28)$$

The issue of over generalization may emerge as regularization forces reconstructions toward the latent manifold expectation.

A.3.2 Latent Manifold Attraction

The defined $\mathbb{E}[x|\theta]$ encapsulates a dual mathematical role within our framework. *Statistically*, it is rigorously defined through the encoder-decoder architecture as $\mathbb{E}[x|\theta] \triangleq \mathbb{E}_{z \sim p(z|\theta)}[g_\phi(z)]$, where $p(z|\theta)$ represents the empirical latent distribution generated by the encoder $h_\psi(x)$. *Geometrically*, it also serves as the L_2 optimal projection anchor on the learned manifold $\mathcal{M}_{\text{target}}$, which satisfies the following objective:

$$\mathbb{E}[x|\theta] = \arg \min_{\tilde{x} \in \mathcal{M}_{\text{target}}} \mathbb{E}_{x \sim \mathcal{P}_x} \|\tilde{x} - x\|_2^2, \quad (29)$$

where $\mathcal{M}_{\text{target}} = \{g_\phi(z)\} = \{g_\phi(h_\psi(x))\}$ denotes the manifold induced by the decoder. This dual role establishes $\mathbb{E}[x|\theta]$ as an attractor, since both the statistical expectation of the decoder output and the geometric centroid minimize projection errors. These properties explain its ability to govern reconstruction behaviors while remaining sensitive to the underlying data density $\mathcal{P}_x(x)$, thus providing a unified perspective to analyze the expansion of the manifold under regularization constraints.

The learning process establishes a dynamic equilibrium between reconstruction fidelity and regularization forces, governed by the data density landscape. For normal samples $x \sim \mathcal{P}_x$, the model aims to preserve the accurate mapping:

$$f_\theta^*(x) \approx x \implies \|f_\theta(x) - x\|_2^2 \leq \epsilon, \quad (30)$$

which preserves the geometric fidelity of normal instances on the manifold $\mathcal{M}_{\text{target}}$. Conversely, for anomalies $x_a \sim \mathcal{P}_a$, the regularization term enforces alignment with the latent manifold expectation:

$$f_\theta^*(x_a) \approx \mathbb{E}[x|\theta] \implies \|f_\theta(x_a) - \mathbb{E}[x|\theta]\|_2^2 \leq \epsilon. \quad (31)$$

This competition induces a critical contamination threshold η_{crit} defined by:

$$\eta_{\text{crit}} = \sup \left\{ \eta \in (0, 1) \mid \mathbb{E}_{x_a} \|x_a - f_\theta(x_a)\|_2^2 > \mathbb{E}_x \|x - f_\theta(x)\|_2^2 \right\}. \quad (32)$$

When $\eta > \eta_{\text{crit}}$, the expanded manifold $\mathcal{M}_{\text{target}}^* = \mathcal{M}_{\text{target}} \cup \{f_\theta(x_a) | x_a \sim \mathcal{P}_a\}$ exhibits dimensional inflation ($\dim(\mathcal{M}^*) > \dim(\mathcal{M}_{\text{target}})$), causing the anomaly-normal separability to collapse.

From this point, inspired by the manifold theorem, we propose IGAD, which can benefit both the balance of robustness and sensitivity and the tightness of the target manifold $\mathcal{M}_{\text{target}}$ to eliminate potential abnormal instances during training.

B Proof of the convergence for IGAD

Theorem 2 Under ideal conditions, IGAD can converge to the target distribution, which consists only of all normal time instances for a given dataset. For simplification, we select x and z for x^i and z^i , respectively. We define the generated distribution, represented by $\mathcal{P}_\theta(y)$, as the PDF of y when $y = f_\theta(z)$ and $z \sim \mathcal{P}_z$. Here, we only pay attention to the loss items relative, i.e. $\mathcal{L}_{\text{recon}}$, $\mathcal{L}_{\text{idem}}$ and $\mathcal{L}_{\text{tight}}$. The final loss function can be divided into two parts:

$$\mathcal{L}(\theta; \theta') = \underbrace{\lambda_{\text{rec}} \mathcal{L}_{\text{recon}}(\theta) + \lambda_{\text{tight}} \mathcal{L}_{\text{tight}}(\theta; \theta')}_{\mathcal{L}_{\text{rt}}} + \lambda_{\text{idem}} \mathcal{L}_{\text{idem}}(\theta; \theta') \quad (33)$$

We assume a large enough model capacity such that both terms can obtain a global minimum:

$$\theta^* = \arg \min_{\theta} \mathcal{L}_{\text{rt}}(\theta; \theta^*) = \arg \min_{\theta} \mathcal{L}_{\text{idem}}(\theta; \theta^*) \quad (34)$$

Then, $\exists \theta^* : \mathcal{P}_{\theta^*} = \mathcal{P}_x$ and for $\lambda_{\text{idem}} = 1$, this is the only one possible \mathcal{P}_{θ^*} .

We first demonstrate the global minimum \mathcal{L}_{rt} . After that, we further verify the global minimum for $\mathcal{L}_{\text{idem}}$. For a given parameter θ in the parameter space Θ and an input \mathcal{X} , $\Phi_{\theta \in \Theta}(\mathcal{X})$ is used to calculate the differences between $f_\theta(\mathcal{X})$ and \mathcal{X} .

Step 1: Global minimum of \mathcal{L}_{rt} given the current parameters θ^* .

$$\mathcal{L}_{\text{rt}}(\theta; \theta^*) = \mathbb{E}_x [\mathcal{D}(f_\theta(x), x)] - \lambda_{\text{tight}} \mathbb{E}_z [\mathcal{D}(f_\theta(f_{\theta^*}(z)), f_{\theta^*}(z))] \quad (35)$$

$$= \int \Phi_\theta(x) \mathcal{P}_x(x) dx - \lambda_{\text{tight}} \int \Phi_\theta(f_{\theta^*}(z)) \mathcal{P}_{\theta^*}(z) dz \quad (36)$$

Change variables: let $y := x$ for the left integral and $y := f_{\theta^*}(z)$ (a well-learned model should ensure that x and $f_{\theta^*}(z)$ lie on the same $\mathcal{M}_{\text{target}}$) for the right integral. Then Eq.(36) can be transformed into the following formular:

$$\mathcal{L}_{\text{rt}}(\theta; \theta^*) = \int \Phi_\theta(y) \mathcal{P}_x(y) dy - \lambda_{\text{tight}} \int \Phi_\theta(y) \mathcal{P}_{\theta^*}(y) dy \quad (37)$$

$$= \int \Phi_\theta(y) \underbrace{(\mathcal{P}_x(y) - \lambda_{\text{tight}} \mathcal{P}_{\theta^*}(y))}_{\text{Regularization for Tightening the Manifold}} dy \quad (38)$$

Let $\mathbb{M} = \sup_{y_1, y_2} \mathcal{D}(y_1, y_2)$, where the supremum is taken over all possible pairs y_1, y_2 . Since Φ_θ is non-negative, the global minimum is achieved when:

$$\Phi_{\theta^*}(y) = \mathbb{M} \cdot [\mathbf{1}_{\{\mathcal{P}_x(y) < \lambda_{\text{tight}} \mathcal{P}_{\theta^*}(y)\}}], \quad \forall y \quad (39)$$

Step 2: Global minimum of $\mathcal{L}_{\text{idem}}$.

$$\mathcal{L}_{\text{idem}}(\theta, \theta^*) = \mathbb{E}_z [\mathcal{D}(f_{\theta^*}(f_\theta(z)), f_\theta(z))] \quad (40)$$

$$= \mathbb{E}_z [\Phi_{\theta^*}(f_\theta(z))] \quad (41)$$

Substituting Φ_{θ^*} from Eq.(39) and exchange the position of θ and θ^* because we check the minimum of the inner f for $\mathcal{L}_{\text{idem}}$, instead of the outer f in $\mathcal{L}_{\text{tight}}$:

$$\mathcal{L}_{\text{idem}}(\theta; \theta^*) = \mathbb{M} \cdot \mathbb{E}_z [\mathbf{1}_{\{\mathcal{P}_x(y) < \lambda_{\text{idem}} \mathcal{P}_\theta(y)\}}] \quad (42)$$

Taking $\arg \min_{\theta}$ of Eq.(42):

$$\theta^* = \mathbb{M} \cdot \arg \min_{\theta} \mathbb{E}_z [\mathbf{1}_{\{\mathcal{P}_x(y) < \lambda_{\text{idem}} \mathcal{P}_\theta(y)\}}] \quad (43)$$

Given these operations, if $\mathcal{P}_{\theta^*} = \mathcal{P}_x$ and $\lambda_{\text{idem}} \leq 1$, the loss value is 0. Specifically, for $\lambda_{\text{idem}} = 1$, $\theta^* : \mathcal{P}_{\theta^*} = \mathcal{P}_x$ is the only minimizer because the total sum of the probability must be 1. In addition, any deviation where $\mathcal{P}_\theta(y) < \mathcal{P}_x(y)$ implies $\exists y$ with $\mathcal{P}_\theta(y) > \mathcal{P}_x(y)$, increasing the loss.

C Explanations for Basic Models

In our experiments, we select 15 basic reconstruction-based models with different structures to evaluate IGAD more comprehensively. We consider the selective strategy of these basic models from two perspectives. First, we select well-designed models specifically tailored for multivariate time series anomaly detection. In addition, we also take time series foundation models into consideration, which can be competitive candidates in various time series tasks, including reconstruction-based MTS AD. Here, we provide more detailed descriptions of the selected models for a better understanding of their structures.

C.1 Models Designed for MTS AD

We include the following models designed for MTS AD in our experiments: CATCH [60], M2N2 [26], SARAD [10], Anomaly Transformer [62], FGANomaly [13], CAE-M [71], MTAD-GAT [72], MSCRED [70], OnimAnomaly [50] and DAGMM [75].

- CATCH [60]: CATCH framework introduces two key innovations for multivariate time series anomaly detection: (1) A frequency patching mechanism that partitions the frequency domain into fine-grained bands to better capture diverse subsequence anomalies, addressing the limitations of coarse-grained frequency analysis in existing methods; (2) A novel Channel Fusion Module with a dynamic correlation discovery mechanism that employs a bi-level optimization strategy to adaptively learn context-aware channel interactions, clustering relevant channels while mitigating noise from irrelevant ones through masked attention, effectively bridging the gap between channel-independent and channel-dependent approaches. The framework further enhances detection robustness through a dual-domain reconstruction objective based on time and frequency, and a novel point-aligned scoring mechanism that synergizes temporal and spectral anomalies, enabling superior performance in detecting both point and heterogeneous subsequence anomalies across varied real-world and synthetic scenarios.
- M2N2 [26]: M2N2 is a novel test-time adaptation framework for unsupervised time series anomaly detection to address the *new normal problem* caused by distribution shifts between training and test data. First, a trend estimation module using exponential moving averages to dynamically detrend input sequences, enabling adaptation to evolving data patterns while preserving underlying dynamics. Then, a self-supervised model update strategy that selectively updates parameters during inference using predicted normal instances, effectively learning new normal patterns while mitigating contamination from anomalies. The approach bridges test-time adaptation with time series anomaly detection through its dual mechanism of trend-aware normalization and confidence-based parameter adjustment, requiring neither access to training data nor additional supervision. By combining real-time trend adaptation with model fine-tuning on detrended sequences, the method demonstrates superior robustness to distribution shifts across diverse real-world benchmarks while maintaining computational efficiency suitable for streaming applications.
- SARAD [10]: SARAD is a novel approach for time series anomaly detection that integrates Spatial Association Reduction with data reconstruction via Transformer-based models. Its innovation lies in capturing both temporal and spatial dependencies within multivariate time series data, a challenge that previous methods addressed largely only from a temporal perspective. The key feature of SARAD is its dual focus on data reconstruction errors and progression reconstruction errors, where the latter focuses on spatial changes in anomaly propagation. SARAD leverages Multi-Head Self-Attention from Transformer layers to capture spatial relationships between features over time, and uses this information in conjunction with progression-based metrics to robustly detect anomalies. Unlike traditional models that may struggle with short-range anomalies or overlook spatially distributed anomalies, SARAD effectively identifies anomalies by observing how spatial associations evolve, even when the underlying data distribution is shifted.
- Anomaly Transformer [62]: Anomaly Transformer introduces a novel approach to time-series anomaly detection by leveraging a Transformer-based model and focusing on the concept of association discrepancy. This model incorporates a dual-branch mechanism within the Anomaly-Attention module, which enhances its ability to distinguish between nor-

mal and abnormal data points. The key innovation lies in the use of Association Discrepancy, measured through symmetrized Kullback–Leibler divergence, between the learned series association and a prior association. By employing a minimax strategy during optimization, the model minimizes the prior-association in the early phase while maximizing the association discrepancy in the later phase, ensuring a more robust distinction between normal and abnormal time points. This method improves the detection performance by forcing the model to focus more on non-adjacent time series data, thus enhancing its sensitivity to anomalies. The final anomaly score is a combination of reconstruction loss and association discrepancy, ensuring that both components contribute to detection, offering a more accurate and interpretable framework for time-series anomaly detection.

- FGANomaly [13]: The proposed model introduces a novel approach to anomaly detection by leveraging Generative Adversarial Networks in the context of multivariate time series data, with a particular focus on handling polluted or noisy training sets. The core innovation lies in the use of a GAN framework, where the generator learns to reconstruct normal time series, while the discriminator distinguishes between real and reconstructed data. Unlike traditional methods that may struggle with noisy or incomplete training data, this model introduces a specific mechanism to adapt the GAN training process to be robust to polluted data. It utilizes a data preprocessing strategy that filters out or reduces the impact of noisy segments, ensuring the model learns meaningful patterns from the time series. This unique approach enables the model to efficiently detect anomalies by leveraging the powerful generative capabilities of GANs while simultaneously addressing the challenges posed by noisy real-world time series data.
- CAE-M [71]: CAE-M addresses several challenges in multivariate time-series anomaly detection, particularly in the presence of noisy data. This proposed approach integrates a convolutional autoencoder for feature extraction, which captures spatial dependencies in multi-sensor time-series signals, with a memory network that combines both non-linear and linear prediction methods to capture temporal dependencies. The key innovation of CAE-M lies in the joint optimization of these components using a compound objective function, which simultaneously minimizes reconstruction error, prediction error, and a regularization term based on Maximum Mean Discrepancy (MMD). The MMD penalty is particularly crucial as it mitigates the influence of noisy data by encouraging the learned feature distribution to approximate that of a Gaussian distribution, thus reducing over-fitting. This architecture allows the model to effectively differentiate between normal and anomalous data even when the training set is polluted with noise.
- MTAD-GAT [72]: MTAD-GAT introduces a novel framework for anomaly detection in multivariate time series data by explicitly capturing the correlations between different features and timestamps. The unique structure of this model leverages two parallel Graph Attention Network (GAT) layers: one feature-oriented and one time-oriented. The feature-oriented GAT layer models the causal relationships between different time-series features, while the time-oriented GAT layer captures temporal dependencies within each time-series. This dual attention mechanism allows the model to dynamically learn both feature-wise and temporal dependencies. Furthermore, MTAD-GAT integrates both forecasting-based and reconstruction-based models, optimizing them through a joint objective function to enhance the representation of time-series data. The forecasting model focuses on single-timestamp predictions, while the reconstruction model learns a latent representation of the entire time-series, making the model robust against various anomaly types.
- MSCRED [70]: MSCRED introduces an effective approach for unsupervised anomaly detection and diagnosis in multivariate time series. The core innovation lies in its ability to jointly tackle three key tasks: anomaly detection, root cause identification, and anomaly severity interpretation. MSCRED achieves this by constructing multi-scale system signature matrices that represent the inter-correlations between time series at different temporal resolutions. These signature matrices are then processed through a fully convolutional encoder to capture spatial dependencies, while an attention-based Convolutional Long Short-Term Memory Network models the temporal dependencies across time steps. The decoder reconstructs these matrices, and the residuals are used to identify anomalies. This architecture is enhanced by its attention mechanism, which adaptively focuses on the most relevant historical time steps to improve anomaly detection.

- OnimAnomaly [50]: The proposed model introduces an innovative approach to anomaly detection by incorporating a Stochastic Recurrent Neural Network (SRNN) to model the temporal dependencies and capture the inherent uncertainty within multivariate time-series data. The key innovation of this model is the introduction of stochasticity in the recurrent network, where the model learns a distribution over the hidden states instead of a deterministic hidden representation. This probabilistic approach allows the SRNN to better handle incomplete data by explicitly modeling the uncertainty in the data generation process. The network is structured to combine both temporal and spatial dependencies by employing a combination of recurrent layers with stochastic units and a mixture of Gaussian distributions to represent uncertainty. Furthermore, the model includes a robust loss function that incorporates both reconstruction error and a regularization term based on the variance of the learned hidden states.
- DAGMM [75]: DAGMM introduces an architecture for unsupervised anomaly detection that combines the strengths of dimensionality reduction via a deep autoencoder with density estimation through a Gaussian Mixture Model (GMM). The key point of this approach is the joint optimization, where both the dimensionality reduction and the density estimation components are optimized simultaneously in an end-to-end manner, eliminating the need for pre-training and decoupled training. This architecture includes two main components: a compression network that reduces the dimensionality of input data and encodes it alongside the reconstruction error, and an estimation network that evaluates the likelihood of each data point within the GMM framework. This joint training, facilitated by the estimation network’s regularization, allows the autoencoder to avoid suboptimal local minima and better capture the essential features of the data for anomaly detection.

C.2 Time Series Foundation Model

Time series foundation models have shown their powerful potential for downstream time series tasks, including forecasting, imputation, classification, and also reconstruction-based anomaly detection. We select FITS [64], Peri-midFormer [59], ModernTCN [34], OFA [74], and TimesNet [58] in our experiments.

- FITS [64]: FITS introduces an innovative approach to time series analysis by operating within the complex frequency domain. It utilizes complex-valued linear interpolation to capture both amplitude and phase information, enabling the model to effectively learn amplitude scaling and phase shifting. This ability allows FITS to achieve state-of-the-art performance in tasks such as forecasting and anomaly detection. Despite its advanced capabilities, FITS maintains a remarkably compact architecture consisting of approximately 10,000 parameters, making it highly efficient. This compactness ensures that FITS is particularly well-suited for deployment on edge devices with limited computational resources, offering an excellent balance of performance and efficiency. By leveraging these innovative techniques, FITS demonstrates that high accuracy can be achieved in time series analysis without the need for large, resource-intensive models.
- Peri-midFormer [59]: Peri-midFormer presents an innovative transformer-based architecture that decomposes time series data into a periodic pyramid structure. This decomposition captures multi-periodic variations by representing the time series at multiple levels, each corresponding to different periodic components. The model employs self-attention mechanisms to effectively capture complex temporal relationships across these levels, enhancing its performance in tasks such as forecasting, imputation, classification, and anomaly detection.
- ModernTCN [34]: ModernTCN revitalizes convolutional approaches in time series analysis by introducing a pure convolutional structure that efficiently captures both cross-time and cross-variable dependencies. By incorporating large convolutional kernels and multiple convolutional layers, ModernTCN achieves substantial effective receptive fields, enabling it to model complex temporal patterns effectively. This design results in state-of-the-art performance across various time series tasks, including forecasting, imputation, classification, and anomaly detection.
- OFA [74]: One Fits All leverages pre-trained language models (LMs) to enhance time series analysis across multiple tasks. By fine-tuning these LMs on time series data, the model adapts the rich, generalized representations learned from large-scale textual data to the

specific characteristics of time series data. This approach demonstrates that pre-trained models from natural language or image domains can achieve comparable or even superior performance in time series tasks such as classification, forecasting, and anomaly detection, highlighting the versatility and power of pre-trained LMs in this context.

- TimesNet [58]: TimesNet introduces a task-general backbone for time series analysis by transforming 1D time series data into 2D tensors based on multiple periods. This transformation allows the model to capture both intraperiod and interperiod variations effectively. Utilizing a parameter-efficient inception block, TimesNet discovers multi-periodicity adaptively and extracts complex temporal variations from the transformed 2D tensors. This design enables TimesNet to achieve consistent state-of-the-art performance across five common time series analysis tasks, including short- and long-term forecasting, imputation, classification, and anomaly detection.

D Hyperparameter Setting

There are five crucial hyperparameters during our experiments, including λ_{rec} , λ_{idem} , λ_{tight} , λ_{aux} and α . **In the latest study conducted by Liu and Paparrizos [32], the pipeline has made sufficient explorations for the optimal λ_{rec} and λ_{aux} , and these two hyperparameters are fixed when performing experiments.** Meanwhile, we also select certain reconstruction-based models that have not been temporarily imported into this pipeline. **For these models, we use the hyperparameters suggested in their original papers or repositories for λ_{rec} and λ_{aux} .** For the hyperparameters introduced by IGAD, we perform a detailed grid search for λ_{idem} , λ_{tight} , and α . The search intervals for each hyperparameter can be shown as:

- λ_{idem} : [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0],
- λ_{tight} : [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0],
- α : [1.1, 1.2, 1.3, 1.4, 1.5].

Then, we summarize the optimal hyperparameters for each model and each dataset in Tab.5.

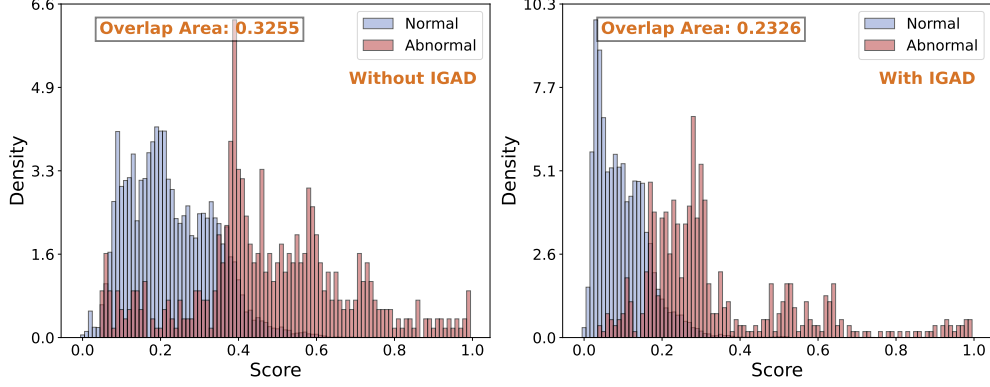
Table 5: Optimal hyperparameters after grid search in our experiments.

Model	Venue	Dataset											
		SMD			MSL			PSM			SMAP		
		λ_{idem}	λ_{tight}	α	λ_{idem}	λ_{tight}	α	λ_{idem}	λ_{tight}	α	λ_{idem}	λ_{tight}	α
CATCH	ICLR, 2025	0.5	0.9	1.1	0.1	0.2	1.5	0.3	0.5	1.4	1.0	0.5	1.5
M2N2	AAAI, 2024	0.5	0.4	1.2	0.1	0.5	1.4	0.1	0.3	1.5	0.1	1.0	1.3
FITS	ICLR, 2024	0.5	0.1	1.1	1.0	1.0	1.5	0.3	1.0	1.2	0.2	0.3	1.5
ModernTCN	ICLR, 2024	0.1	1.0	1.4	0.1	0.8	1.3	0.1	0.1	1.1	0.1	0.1	1.1
Peri-midFormer	NeurIPS, 2024	0.1	1.0	1.5	0.5	0.4	1.4	0.8	0.9	1.4	0.1	0.1	1.3
SARAD	NeurIPS, 2024	0.1	0.1	1.1	0.1	0.9	1.5	0.4	0.2	1.5	0.1	0.1	1.1
TimesNet	ICLR, 2023	0.1	1.0	1.4	0.8	0.1	1.3	0.1	1.0	1.2	1.0	1.0	1.3
OFA	NeurIPS, 2023	0.1	1.0	1.5	0.8	0.9	1.3	0.5	0.4	1.3	0.3	0.1	1.3
A.T.	ICLR, 2022	0.9	1.0	1.4	1.0	0.2	1.1	0.8	0.7	1.4	0.7	0.8	1.3
FGANomaly	TKDE, 2021	0.2	0.1	1.4	0.1	0.1	1.1	0.1	0.4	1.3	0.1	1.0	1.5
CAE-M	TKDE, 2021	0.1	0.1	1.5	0.8	0.1	1.5	0.2	1.0	1.1	0.1	0.1	1.5
MTAD-GAT	ICDM, 2021	0.8	0.8	1.1	0.2	0.3	1.3	0.1	0.6	1.4	0.9	1.0	1.4
OmniAnomaly	KDD, 2019	0.2	0.3	1.5	0.1	0.8	1.3	0.9	0.7	1.5	0.2	0.5	1.5
MSCRED	AAAI, 2019	0.1	0.5	1.4	0.4	0.4	1.5	0.1	0.1	1.1	0.3	0.9	1.5
DAGMM	ICLR, 2018	0.2	1.0	1.1	0.8	0.4	1.4	0.2	0.1	1.1	0.1	0.9	1.5

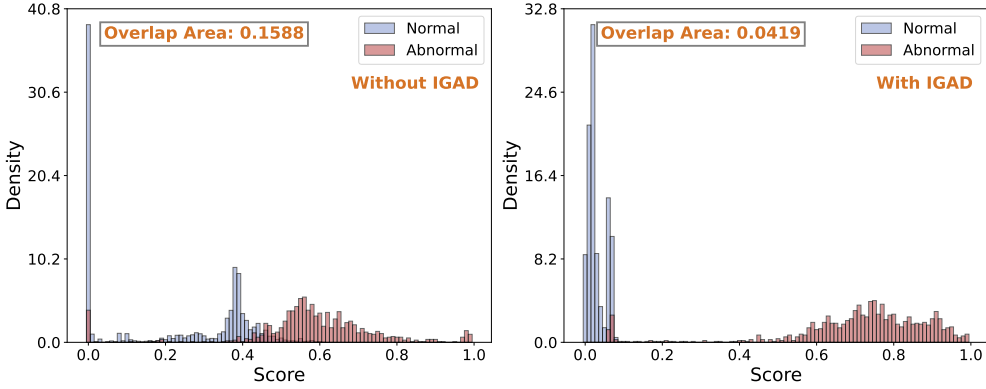
E More Detailed Experimental Results

E.1 Distinguishable Distributions of Anomaly Scores and Auxiliary Metrics

In this section, we will provide more experimental results for other detailed information. First, we show more cases where the application of IGAD effectively generates more distinguishable distributions of anomaly scores for normal and abnormal instances, shown as Fig.7(a) and Fig.7(b). Second, we list other evaluation metrics for classification tasks such as AUC-PR, AUC-ROC, VUS-ROC and different types of F1 from Tab.6 to Tab.13, which can still serve as auxiliary metrics although they exist potential evaluation shortcomings in the field of multivariate time series anomaly detection [32]. Improvements in these metrics can also be observed.



(a) Anomaly scores for MSCRED on dataset SMD.



(b) Anomaly scores for FGANomaly on dataset SMAP.

Figure 7: Anomaly score distributions for different models and datasets.

Table 6: More results on dataset SMD with IGAD.

Model	AUC-PR	AUC-ROC	VUS-ROC	Standard-F1
CATCH	0.2341 \pm 0.0038	0.8952 \pm 0.0019	0.8343 \pm 0.0031	0.1831 \pm 0.0041
M2N2	0.0407 \pm 0.0080	0.6401 \pm 0.0772	0.5537 \pm 0.0630	0.0335 \pm 0.0308
FITS	0.0564 \pm 0.0023	0.7490 \pm 0.0122	0.7300 \pm 0.0175	0.0470 \pm 0.0200
ModernTCN	0.2486 \pm 0.0011	0.8966 \pm 0.0002	0.8383 \pm 0.0004	0.3063 \pm 0.0022
Peri-midFormer	0.2026 \pm 0.0071	0.8797 \pm 0.0037	0.8209 \pm 0.0026	0.1694 \pm 0.0055
SARAD	0.2748 \pm 0.0096	0.9092 \pm 0.0054	0.8259 \pm 0.0089	0.3155 \pm 0.0120
TimesNet	0.0861 \pm 0.0347	0.7553 \pm 0.0272	0.7156 \pm 0.0298	0.1303 \pm 0.0628
OFA	0.1433 \pm 0.0254	0.6935 \pm 0.0121	0.6639 \pm 0.0101	0.2220 \pm 0.0353
A.T.	0.0679 \pm 0.0914	0.5137 \pm 0.0763	0.4960 \pm 0.0536	0.0661 \pm 0.1288
FGANomaly	0.4929 \pm 0.0116	0.9324 \pm 0.0023	0.8705 \pm 0.0065	0.4943 \pm 0.0308
CAE-M	0.1687 \pm 0.1294	0.6635 \pm 0.1191	0.5314 \pm 0.1537	0.1326 \pm 0.1118
MTAD-GAT	0.4562 \pm 0.0379	0.8911 \pm 0.0239	0.8886 \pm 0.0227	0.5304 \pm 0.0193
OmniAnomaly	0.2621 \pm 0.0026	0.9052 \pm 0.0031	0.8274 \pm 0.0024	0.2830 \pm 0.0029
MSCRED	0.4377 \pm 0.0217	0.9216 \pm 0.0331	0.8567 \pm 0.0329	0.3939 \pm 0.0093
DAGMM	0.2285 \pm 0.0684	0.6875 \pm 0.0275	0.4962 \pm 0.0401	0.1449 \pm 0.1425

Model	PA-F1	Event-based-F1	R-based-F1	Affiliation-F1
CATCH	0.4055 \pm 0.1009	0.2638 \pm 0.0186	0.0056 \pm 0.0000	0.6418 \pm 0.0039
M2N2	0.2570 \pm 0.2359	0.0666 \pm 0.0626	0.0027 \pm 0.0024	0.4552 \pm 0.1265
FITS	0.5331 \pm 0.0347	0.0938 \pm 0.0333	0.0046 \pm 0.0000	0.7750 \pm 0.0177
ModernTCN	0.4886 \pm 0.0004	0.3846 \pm 0.0015	0.0056 \pm 0.0000	0.6524 \pm 0.0001
Peri-midFormer	0.4670 \pm 0.1321	0.2529 \pm 0.0294	0.0056 \pm 0.0000	0.6198 \pm 0.0720
SARAD	0.4912 \pm 0.0114	0.3920 \pm 0.0141	0.0056 \pm 0.0000	0.7231 \pm 0.0034
TimesNet	0.6006 \pm 0.0614	0.2222 \pm 0.0866	0.0047 \pm 0.0000	0.8155 \pm 0.0287
OFA	0.7442 \pm 0.0211	0.3747 \pm 0.0296	0.0050 \pm 0.0002	0.8561 \pm 0.0285
A.T.	0.3166 \pm 0.4273	0.2140 \pm 0.3299	0.0033 \pm 0.0031	0.5968 \pm 0.3153
FGANomaly	0.6531 \pm 0.0362	0.6095 \pm 0.0258	0.0056 \pm 0.0000	0.7839 \pm 0.0392
CAE-M	0.3069 \pm 0.1643	0.2520 \pm 0.1600	0.0313 \pm 0.0146	0.2563 \pm 0.1742
MTAD-GAT	0.8913 \pm 0.0249	0.7379 \pm 0.0282	0.0047 \pm 0.0003	0.9257 \pm 0.0210
OmniAnomaly	0.5340 \pm 0.0336	0.3775 \pm 0.0115	0.0056 \pm 0.0000	0.7751 \pm 0.0282
MSCRED	0.6960 \pm 0.0402	0.6109 \pm 0.0312	0.0056 \pm 0.0001	0.7587 \pm 0.0329
DAGMM	0.3319 \pm 0.2084	0.3286 \pm 0.1672	0.0106 \pm 0.0067	0.3648 \pm 0.1950

Table 7: More results on dataset SMD without IGAD.

Model	AUC-PR	AUC-ROC	VUS-ROC	Standard-F1
CATCH	0.2272 \pm 0.0053	0.8891 \pm 0.0014	0.8222 \pm 0.0019	0.1811 \pm 0.0031
M2N2	0.0234 \pm 0.0003	0.4342 \pm 0.0034	0.3623 \pm 0.0068	0.0487 \pm 0.0000
FITS	0.0419 \pm 0.0025	0.6946 \pm 0.0192	0.6769 \pm 0.0220	0.0356 \pm 0.0076
ModernTCN	0.1930 \pm 0.0022	0.8792 \pm 0.0012	0.8188 \pm 0.0020	0.1676 \pm 0.0020
Peri-midFormer	0.2004 \pm 0.0063	0.8782 \pm 0.0036	0.8198 \pm 0.0028	0.1680 \pm 0.0042
SARAD	0.2766 \pm 0.0128	0.9103 \pm 0.0069	0.8264 \pm 0.0113	0.3134 \pm 0.0104
TimesNet	0.0824 \pm 0.0188	0.7583 \pm 0.0170	0.7147 \pm 0.0216	0.1294 \pm 0.0373
OFA	0.0630 \pm 0.0030	0.6887 \pm 0.0068	0.6778 \pm 0.0072	0.1178 \pm 0.0112
A.T.	0.0266 \pm 0.0053	0.5213 \pm 0.0293	0.5175 \pm 0.0240	0.0037 \pm 0.0082
FGANomaly	0.4943 \pm 0.0063	0.9320 \pm 0.0071	0.8790 \pm 0.0143	0.4663 \pm 0.0094
CAE-M	0.1746 \pm 0.1266	0.6676 \pm 0.1213	0.5360 \pm 0.1568	0.1268 \pm 0.1049
MTAD-GAT	0.3949 \pm 0.0031	0.8710 \pm 0.0043	0.8700 \pm 0.0043	0.5150 \pm 0.0048
OmniAnomaly	0.2578 \pm 0.0011	0.9003 \pm 0.0025	0.8231 \pm 0.0020	0.2798 \pm 0.0017
MSCRED	0.4373 \pm 0.0293	0.9228 \pm 0.0326	0.8517 \pm 0.0271	0.3878 \pm 0.0340
DAGMM	0.0983 \pm 0.0067	0.5554 \pm 0.0136	0.4055 \pm 0.0109	0.1041 \pm 0.0080

Model	PA-F1	Event-based-F1	R-based-F1	Affiliation-F1
CATCH	0.3289 \pm 0.0441	0.2380 \pm 0.0248	0.0056 \pm 0.0000	0.6416 \pm 0.0018
M2N2	0.3832 \pm 0.0000	0.0851 \pm 0.0000	0.0044 \pm 0.0000	0.5238 \pm 0.0038
FITS	0.4901 \pm 0.0580	0.0768 \pm 0.0153	0.0046 \pm 0.0000	0.7382 \pm 0.0113
ModernTCN	0.3416 \pm 0.0231	0.2309 \pm 0.0115	0.0056 \pm 0.0000	0.6316 \pm 0.0002
Peri-midFormer	0.5292 \pm 0.0042	0.2522 \pm 0.0027	0.0056 \pm 0.0000	0.6411 \pm 0.0023
SARAD	0.4905 \pm 0.0130	0.3904 \pm 0.0142	0.0056 \pm 0.0000	0.7315 \pm 0.0249
TimesNet	0.6230 \pm 0.0434	0.2284 \pm 0.0559	0.0047 \pm 0.0000	0.8128 \pm 0.0259
OFA	0.6813 \pm 0.0316	0.2243 \pm 0.0241	0.0045 \pm 0.0000	0.8178 \pm 0.0129
A.T.	0.1257 \pm 0.2811	0.0182 \pm 0.0407	0.0009 \pm 0.0020	0.3300 \pm 0.2620
FGANomaly	0.5990 \pm 0.0089	0.5588 \pm 0.0201	0.0056 \pm 0.0000	0.7236 \pm 0.0351
CAE-M	0.3072 \pm 0.1654	0.2522 \pm 0.1607	0.0311 \pm 0.0147	0.2562 \pm 0.1743
MTAD-GAT	0.8556 \pm 0.0044	0.7075 \pm 0.0070	0.0046 \pm 0.0000	0.9264 \pm 0.0007
OmniAnomaly	1.0000 \pm 0.0000	1.0000 \pm 0.0000	0.3893 \pm 0.0526	0.9961 \pm 0.0011
MSCRED	0.6846 \pm 0.0686	0.5927 \pm 0.0695	0.0056 \pm 0.0000	0.7394 \pm 0.0799
DAGMM	0.2317 \pm 0.0012	0.1778 \pm 0.0014	0.0406 \pm 0.0008	0.1789 \pm 0.0003

Table 8: More results on dataset MSL with IGAD.

Model	AUC-PR	AUC-ROC	VUS-ROC	Standard-F1
CATCH	0.0286 \pm 0.0007	0.7808 \pm 0.0045	0.7808 \pm 0.0045	0.0000 \pm 0.0000
M2N2	0.8250 \pm 0.2050	0.9984 \pm 0.0030	0.9984 \pm 0.0030	0.0000 \pm 0.0000
FITS	0.0103 \pm 0.0038	0.5104 \pm 0.0508	0.5105 \pm 0.0508	0.0000 \pm 0.0000
ModernTCN	0.0634 \pm 0.0424	0.8082 \pm 0.0182	0.8085 \pm 0.0181	0.0718 \pm 0.0306
Peri-midFormer	0.0346 \pm 0.0015	0.8160 \pm 0.0252	0.8160 \pm 0.0252	0.0168 \pm 0.0233
SARAD	0.0552 \pm 0.0635	0.6906 \pm 0.0499	0.6908 \pm 0.0497	0.0785 \pm 0.1079
TimesNet	0.0074 \pm 0.0030	0.4316 \pm 0.0824	0.4316 \pm 0.0826	0.0000 \pm 0.0000
OFA	0.0114 \pm 0.0112	0.5077 \pm 0.0696	0.5077 \pm 0.0696	0.0200 \pm 0.0447
A.T.	0.0063 \pm 0.0007	0.5069 \pm 0.0154	0.5068 \pm 0.0152	0.0000 \pm 0.0000
FGANomaly	0.0525 \pm 0.0103	0.8315 \pm 0.0532	0.8314 \pm 0.0531	0.0986 \pm 0.0577
CAE-M	0.0041 \pm 0.0001	0.2066 \pm 0.0074	0.2067 \pm 0.0072	0.0000 \pm 0.0000
MTAD-GAT	0.3401 \pm 0.1288	0.7066 \pm 0.0655	0.7063 \pm 0.0656	0.3037 \pm 0.0928
OmniAnomaly	0.0088 \pm 0.0029	0.5024 \pm 0.0864	0.5025 \pm 0.0862	0.0000 \pm 0.0000
MSCRED	0.0099 \pm 0.0011	0.6883 \pm 0.0361	0.6883 \pm 0.0363	0.0000 \pm 0.0000
DAGMM	0.0072 \pm 0.0029	0.2631 \pm 0.0669	0.2630 \pm 0.0666	0.0000 \pm 0.0000

Model	PA-F1	Event-based-F1	R-based-F1	Affiliation-F1
CATCH	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.9332 \pm 0.0038
M2N2	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	Nan \pm NaN
FITS	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.7619 \pm 0.1013
ModernTCN	0.2940 \pm 0.0347	0.0822 \pm 0.0336	0.0210 \pm 0.0001	0.9703 \pm 0.0013
Peri-midFormer	0.1529 \pm 0.2106	0.0214 \pm 0.0298	0.0083 \pm 0.0114	0.9682 \pm 0.0017
SARAD	0.2991 \pm 0.4104	0.1415 \pm 0.1965	0.0083 \pm 0.0114	0.9389 \pm 0.0360
TimesNet	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.8442 \pm 0.0523
OFA	0.1467 \pm 0.3280	0.0400 \pm 0.0894	0.0032 \pm 0.0072	0.8031 \pm 0.0757
A.T.	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.9499 \pm NaN
FGANomaly	0.2739 \pm 0.1532	0.1135 \pm 0.0660	0.0177 \pm 0.0099	0.9842 \pm 0.0018
CAE-M	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	NaN \pm NaN
MTAD-GAT	1.0000 \pm 0.0000	1.0000 \pm 0.0000	0.0119 \pm 0.0019	0.9990 \pm 0.0003
OmniAnomaly	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.6452 \pm 0.1220
MSCRED	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.3804 \pm NaN
DAGMM	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.4275 \pm 0.0554

Table 9: More results on dataset MSL without IGAD.

Model	AUC-PR	AUC-ROC	VUS-ROC	Standard-F1
CATCH	0.0286 \pm 0.0008	0.7807 \pm 0.0052	0.7806 \pm 0.0052	0.0000 \pm 0.0000
M2N2	0.2024 \pm 0.1532	0.9637 \pm 0.0233	0.9637 \pm 0.0233	0.0000 \pm 0.0000
FITS	0.0419 \pm 0.0025	0.6946 \pm 0.0192	0.6769 \pm 0.0220	0.0356 \pm 0.0076
ModernTCN	0.1930 \pm 0.0022	0.8792 \pm 0.0012	0.8188 \pm 0.0020	0.1676 \pm 0.0020
Peri-midFormer	0.2004 \pm 0.0063	0.8782 \pm 0.0036	0.8198 \pm 0.0028	0.1680 \pm 0.0042
SARAD	0.2766 \pm 0.0128	0.9103 \pm 0.0069	0.8264 \pm 0.0113	0.3134 \pm 0.0104
TimesNet	0.0824 \pm 0.0188	0.7583 \pm 0.0170	0.7147 \pm 0.0216	0.1294 \pm 0.0373
OFA	0.0630 \pm 0.0030	0.6887 \pm 0.0068	0.6778 \pm 0.0072	0.1178 \pm 0.0112
A.T.	0.0266 \pm 0.0053	0.5213 \pm 0.0293	0.5175 \pm 0.0240	0.0037 \pm 0.0082
FGANomaly	0.4943 \pm 0.0063	0.9320 \pm 0.0071	0.8790 \pm 0.0143	0.4663 \pm 0.0094
CAE-M	0.1746 \pm 0.1266	0.6676 \pm 0.1213	0.5360 \pm 0.1568	0.1268 \pm 0.1049
MTAD-GAT	0.3949 \pm 0.0031	0.8710 \pm 0.0043	0.8700 \pm 0.0043	0.5150 \pm 0.0048
OmniAnomaly	0.2578 \pm 0.0011	0.9003 \pm 0.0025	0.8231 \pm 0.0020	0.2798 \pm 0.0017
MSCRED	0.4373 \pm 0.0293	0.9228 \pm 0.0326	0.8517 \pm 0.0271	0.3878 \pm 0.0340
DAGMM	0.0983 \pm 0.0067	0.5554 \pm 0.0136	0.4055 \pm 0.0109	0.1041 \pm 0.0080

Model	PA-F1	Event-based-F1	R-based-F1	Affiliation-F1
CATCH	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.9332 \pm 0.0040
M2N2	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	Nan \pm NaN
FITS	0.4901 \pm 0.0580	0.0768 \pm 0.0153	0.0046 \pm 0.0000	0.7382 \pm 0.0113
ModernTCN	0.3416 \pm 0.0231	0.2309 \pm 0.0115	0.0056 \pm 0.0000	0.6316 \pm 0.0002
Peri-midFormer	0.5292 \pm 0.0042	0.2522 \pm 0.0027	0.0056 \pm 0.0000	0.6411 \pm 0.0023
SARAD	0.4905 \pm 0.0130	0.3904 \pm 0.0142	0.0056 \pm 0.0000	0.7315 \pm 0.0249
TimesNet	0.6230 \pm 0.0434	0.2284 \pm 0.0559	0.0047 \pm 0.0000	0.8128 \pm 0.0259
OFA	0.6813 \pm 0.0316	0.2243 \pm 0.0241	0.0045 \pm 0.0000	0.8178 \pm 0.0129
A.T.	0.1257 \pm 0.2811	0.0182 \pm 0.0407	0.0009 \pm 0.0020	0.3300 \pm 0.2620
FGANomaly	0.5990 \pm 0.0089	0.5588 \pm 0.0201	0.0056 \pm 0.0000	0.7236 \pm 0.0351
CAE-M	0.3072 \pm 0.1654	0.2522 \pm 0.1607	0.0311 \pm 0.0147	0.2562 \pm 0.1743
MTAD-GAT	0.8556 \pm 0.0044	0.7075 \pm 0.0070	0.0046 \pm 0.0000	0.9264 \pm 0.0007
OmniAnomaly	1.0000 \pm 0.0000	1.0000 \pm 0.0000	0.3893 \pm 0.0526	0.9961 \pm 0.0011
MSCRED	0.6846 \pm 0.0686	0.5927 \pm 0.0695	0.0056 \pm 0.0000	0.7394 \pm 0.0799
DAGMM	0.2317 \pm 0.0012	0.1778 \pm 0.0014	0.0406 \pm 0.0008	0.1789 \pm 0.0003

Table 10: More results on dataset PSM with IGAD.

Model	AUC-PR	AUC-ROC	VUS-ROC	Standard-F1
CATCH	0.1380 \pm 0.0030	0.5680 \pm 0.0102	0.4974 \pm 0.0099	0.0257 \pm 0.0027
M2N2	0.4124 \pm 0.0252	0.8527 \pm 0.0024	0.7793 \pm 0.0063	0.0548 \pm 0.0272
FITS	0.1203 \pm 0.0005	0.5137 \pm 0.0020	0.4559 \pm 0.0104	0.0144 \pm 0.0006
ModernTCN	0.1408 \pm 0.0002	0.5659 \pm 0.0004	0.4938 \pm 0.0005	0.0291 \pm 0.0001
Peri-midFormer	0.1378 \pm 0.0003	0.5520 \pm 0.0009	0.4909 \pm 0.0035	0.0338 \pm 0.0005
SARAD	0.1787 \pm 0.0095	0.6627 \pm 0.0186	0.4476 \pm 0.0155	0.0310 \pm 0.0021
TimesNet	0.1355 \pm 0.0076	0.5516 \pm 0.0151	0.4839 \pm 0.0141	0.0259 \pm 0.0070
OFA	0.1490 \pm 0.0108	0.5700 \pm 0.0180	0.5147 \pm 0.0208	0.0145 \pm 0.0057
A.T.	0.1809 \pm 0.0739	0.6199 \pm 0.0833	0.5028 \pm 0.0635	0.1100 \pm 0.1421
FGANomaly	0.2401 \pm 0.0235	0.7214 \pm 0.0110	0.5769 \pm 0.0106	0.0067 \pm 0.0055
CAE-M	0.1773 \pm 0.0376	0.6427 \pm 0.0402	0.4493 \pm 0.0245	0.0275 \pm 0.0117
MTAD-GAT	0.1904 \pm 0.0509	0.6783 \pm 0.0665	0.6099 \pm 0.0623	0.0284 \pm 0.0017
OmniAnomaly	0.1659 \pm 0.0006	0.6233 \pm 0.0037	0.4324 \pm 0.0073	0.0257 \pm 0.0001
MSCRED	0.1928 \pm 0.0246	0.7115 \pm 0.0335	0.5034 \pm 0.0550	0.0220 \pm 0.0124
DAGMM	0.2116 \pm 0.0047	0.6913 \pm 0.0045	0.4736 \pm 0.0070	0.0375 \pm 0.0050

Model	PA-F1	Event-based-F1	R-based-F1	Affiliation-F1
CATCH	0.3629 \pm 0.0010	0.0990 \pm 0.0084	0.0032 \pm 0.0000	0.1300 \pm 0.0100
M2N2	0.3320 \pm 0.0000	0.0274 \pm 0.0000	0.0069 \pm 0.0010	0.0215 \pm 0.0011
FITS	0.7490 \pm 0.0155	0.1197 \pm 0.0086	0.0030 \pm 0.0000	0.3014 \pm 0.0270
ModernTCN	0.3624 \pm 0.0001	0.1077 \pm 0.0002	0.0032 \pm 0.0000	0.1360 \pm 0.0000
Peri-midFormer	0.3640 \pm 0.0001	0.1300 \pm 0.0068	0.0032 \pm 0.0000	0.1686 \pm 0.0116
SARAD	0.3313 \pm 0.0039	0.0710 \pm 0.0018	0.0032 \pm 0.0000	0.1130 \pm 0.0015
TimesNet	0.8278 \pm 0.0434	0.1563 \pm 0.0212	0.0031 \pm 0.0001	0.3117 \pm 0.0466
OFA	0.8248 \pm 0.0810	0.1879 \pm 0.0471	0.0031 \pm 0.0000	0.3150 \pm 0.0950
A.T.	0.3841 \pm 0.2627	0.1285 \pm 0.1321	0.0023 \pm 0.0013	0.3753 \pm 0.3316
FGANomaly	0.2710 \pm 0.1519	0.0360 \pm 0.0244	0.0031 \pm 0.0018	0.0634 \pm 0.0258
CAE-M	0.2783 \pm 0.1384	0.0730 \pm 0.0308	0.0032 \pm 0.0001	0.1079 \pm 0.0242
MTAD-GAT	0.7170 \pm 0.0350	0.1424 \pm 0.0388	0.0032 \pm 0.0000	0.2223 \pm 0.0190
OmniAnomaly	0.0355 \pm 0.0001	0.0409 \pm 0.0002	0.0031 \pm 0.0000	0.0750 \pm 0.0000
MSCRED	0.1744 \pm 0.1606	0.0523 \pm 0.0242	0.0032 \pm 0.0002	0.0945 \pm 0.0226
DAGMM	0.3400 \pm 0.0016	0.0794 \pm 0.0026	0.0032 \pm 0.0000	0.1168 \pm 0.0027

Table 11: More results on dataset PSM without IGAD.

Model	AUC-PR	AUC-ROC	VUS-ROC	Standard-F1
CATCH	0.1323 \pm 0.0036	0.5514 \pm 0.0096	0.5044 \pm 0.0091	0.0194 \pm 0.0116
M2N2	0.3915 \pm 0.0135	0.8464 \pm 0.0033	0.7856 \pm 0.0033	0.0648 \pm 0.0000
FITS	0.1180 \pm 0.0004	0.5083 \pm 0.0009	0.4732 \pm 0.0014	0.0104 \pm 0.0003
ModernTCN	0.1463 \pm 0.0003	0.5742 \pm 0.0005	0.5101 \pm 0.0019	0.0316 \pm 0.0003
Peri-midFormer	0.1378 \pm 0.0004	0.5511 \pm 0.0014	0.4910 \pm 0.0042	0.0349 \pm 0.0005
SARAD	0.1568 \pm 0.0135	0.6524 \pm 0.0268	0.4493 \pm 0.0097	0.0290 \pm 0.0012
TimesNet	0.1211 \pm 0.0022	0.5136 \pm 0.0038	0.4610 \pm 0.0048	0.0173 \pm 0.0028
OFA	0.1310 \pm 0.0004	0.5390 \pm 0.0005	0.4772 \pm 0.0015	0.0110 \pm 0.0006
A.T.	0.1162 \pm 0.0094	0.5161 \pm 0.0360	0.4875 \pm 0.0281	0.0000 \pm 0.0000
FGANomaly	0.2620 \pm 0.0119	0.7480 \pm 0.0082	0.5993 \pm 0.0080	0.0031 \pm 0.0012
CAE-M	0.1608 \pm 0.0009	0.6548 \pm 0.0018	0.4682 \pm 0.0025	0.0248 \pm 0.0004
MTAD-GAT	0.1495 \pm 0.0023	0.6126 \pm 0.0056	0.5373 \pm 0.0069	0.0255 \pm 0.0005
OmniAnomaly	0.1655 \pm 0.0010	0.6218 \pm 0.0031	0.4314 \pm 0.0083	0.0257 \pm 0.0001
MSCRED	0.2165 \pm 0.0100	0.7431 \pm 0.0197	0.5425 \pm 0.0456	0.0220 \pm 0.0095
DAGMM	0.2126 \pm 0.0025	0.6912 \pm 0.0044	0.4765 \pm 0.0040	0.0350 \pm 0.0006

Model	PA-F1	Event-based-F1	R-based-F1	Affiliation-F1
CATCH	0.7125 \pm 0.2074	0.1237 \pm 0.0276	0.0031 \pm 0.0001	0.2804 \pm 0.1142
M2N2	0.3320 \pm 0.0000	0.0274 \pm 0.0000	0.0073 \pm 0.0000	0.0219 \pm 0.0000
FITS	0.8835 \pm 0.0004	0.2184 \pm 0.0080	0.0031 \pm 0.0000	0.5232 \pm 0.0069
ModernTCN	0.3639 \pm 0.0007	0.1138 \pm 0.0073	0.0032 \pm 0.0000	0.1757 \pm 0.0161
Peri-midFormer	0.3642 \pm 0.0002	0.1397 \pm 0.0162	0.0032 \pm 0.0000	0.1911 \pm 0.0249
SARAD	0.1074 \pm 0.1207	0.0569 \pm 0.0049	0.0032 \pm 0.0000	0.1015 \pm 0.0059
TimesNet	0.8550 \pm 0.0021	0.1648 \pm 0.0074	0.0031 \pm 0.0000	0.4767 \pm 0.0394
OFA	0.8683 \pm 0.0048	0.1960 \pm 0.0082	0.0031 \pm 0.0000	0.5743 \pm 0.0147
A.T.	0.0664 \pm 0.1484	0.0050 \pm 0.0112	0.0004 \pm 0.0009	0.0792 \pm NaN
FGANomaly	0.3214 \pm 0.0014	0.0200 \pm 0.0029	0.0028 \pm 0.0000	0.0386 \pm 0.0008
CAE-M	0.3148 \pm 0.0006	0.0547 \pm 0.0001	0.0035 \pm 0.0000	0.0944 \pm 0.0000
MTAD-GAT	0.3319 \pm 0.0016	0.0874 \pm 0.0019	0.0032 \pm 0.0000	0.1823 \pm 0.0002
OmniAnomaly	0.0355 \pm 0.0001	0.0409 \pm 0.0002	0.0031 \pm 0.0000	0.0750 \pm 0.0000
MSCRED	0.3461 \pm 0.0249	0.0633 \pm 0.0130	0.0032 \pm 0.0001	0.1150 \pm 0.0204
DAGMM	0.3397 \pm 0.0005	0.0780 \pm 0.0002	0.0032 \pm 0.0000	0.1140 \pm 0.0002

Table 12: More results on dataset SMAP with IGAD.

Model	AUC-PR	AUC-ROC	VUS-ROC	Standard-F1
CATCH	0.2942 \pm 0.0008	0.6088 \pm 0.0017	0.6088 \pm 0.0017	0.1329 \pm 0.0020
M2N2	0.1971 \pm 0.0427	0.6754 \pm 0.0868	0.6753 \pm 0.0869	0.0000 \pm 0.0000
FITS	0.2858 \pm 0.0126	0.7845 \pm 0.0196	0.7845 \pm 0.0197	0.0194 \pm 0.0045
ModernTCN	0.4165 \pm 0.0079	0.8004 \pm 0.0053	0.8004 \pm 0.0054	0.2067 \pm 0.0025
Peri-midFormer	0.5096 \pm 0.0273	0.8312 \pm 0.0129	0.8298 \pm 0.0130	0.1971 \pm 0.0030
SARAD	0.8437 \pm 0.0213	0.9320 \pm 0.0060	0.9337 \pm 0.0060	0.2425 \pm 0.0064
TimesNet	0.2731 \pm 0.0672	0.7194 \pm 0.0970	0.7194 \pm 0.0970	0.0330 \pm 0.0197
OFA	0.2977 \pm 0.0254	0.7897 \pm 0.0326	0.7897 \pm 0.0326	0.0425 \pm 0.0138
A.T.	0.2557 \pm 0.1120	0.6327 \pm 0.0959	0.6326 \pm 0.0961	0.0439 \pm 0.0884
FGANomaly	0.9840 \pm 0.0012	0.9957 \pm 0.0006	0.9957 \pm 0.0006	0.3473 \pm 0.0099
CAE-M	0.0718 \pm 0.0002	0.0203 \pm 0.0141	0.0203 \pm 0.0141	0.0000 \pm 0.0000
MTAD-GAT	0.5121 \pm 0.2364	0.8976 \pm 0.0501	0.8976 \pm 0.0501	0.0718 \pm 0.0659
OmniAnomaly	0.9064 \pm 0.0272	0.9111 \pm 0.0284	0.9111 \pm 0.0284	0.4186 \pm 0.1677
MSCRED	0.1248 \pm 0.0207	0.3914 \pm 0.1556	0.3914 \pm 0.1556	0.0000 \pm 0.0000
DAGMM	0.1106 \pm 0.0124	0.0597 \pm 0.0144	0.0597 \pm 0.0144	0.0225 \pm 0.0311

Model	PA-F1	Event-based-F1	R-based-F1	Affiliation-F1
CATCH	1.0000 \pm 0.0000	1.0000 \pm 0.0000	0.3063 \pm 0.0027	0.9485 \pm 0.0002
M2N2	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	NaN \pm NaN
FITS	0.9020 \pm 0.0015	0.0988 \pm 0.0209	0.1662 \pm 0.0007	0.5542 \pm 0.0163
ModernTCN	0.9986 \pm 0.0005	0.9879 \pm 0.0040	0.3322 \pm 0.0008	0.9484 \pm 0.0005
Peri-midFormer	0.9998 \pm 0.0002	0.9984 \pm 0.0022	0.3288 \pm 0.0011	0.9498 \pm 0.0002
SARAD	0.9989 \pm 0.0009	0.9922 \pm 0.0063	0.3435 \pm 0.0021	0.9492 \pm 0.0009
TimesNet	0.9183 \pm 0.0165	0.1839 \pm 0.1134	0.1682 \pm 0.0027	0.5879 \pm 0.0432
OFA	0.9151 \pm 0.0135	0.2188 \pm 0.0699	0.1690 \pm 0.0019	0.6027 \pm 0.0358
A.T.	0.7826 \pm 0.4380	0.2615 \pm 0.3943	0.1362 \pm 0.0768	0.6389 \pm 0.1989
FGANomaly	1.0000 \pm 0.0000	1.0000 \pm 0.0000	0.3724 \pm 0.0032	0.9955 \pm 0.0001
CAE-M	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	NaN \pm NaN
MTAD-GAT	0.9480 \pm 0.0293	0.4098 \pm 0.3328	0.1997 \pm 0.0676	0.6742 \pm 0.1815
OmniAnomaly	1.0000 \pm 0.0000	1.0000 \pm 0.0000	0.3893 \pm 0.0526	0.9961 \pm 0.0011
MSCRED	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.3334 \pm 0.0001
DAGMM	0.4000 \pm 0.5477	0.4000 \pm 0.5477	0.1096 \pm 0.1500	0.9490 \pm 0.0000

Table 13: More results on dataset SMAP without IGAD.

Model	AUC-PR	AUC-ROC	VUS-ROC	Standard-F1
CATCH	0.2898 \pm 0.0013	0.5913 \pm 0.0052	0.5912 \pm 0.0052	0.1410 \pm 0.0051
M2N2	0.1954 \pm 0.0056	0.6345 \pm 0.0123	0.6344 \pm 0.0123	0.0909 \pm 0.0081
FITS	0.2713 \pm 0.0115	0.7675 \pm 0.0154	0.7675 \pm 0.0154	0.0203 \pm 0.0048
ModernTCN	0.4594 \pm 0.0098	0.8220 \pm 0.0051	0.8220 \pm 0.0050	0.1746 \pm 0.0053
Peri-midFormer	0.5105 \pm 0.0288	0.8304 \pm 0.0132	0.8304 \pm 0.0133	0.1977 \pm 0.0044
SARAD	0.8490 \pm 0.0184	0.9335 \pm 0.0053	0.9335 \pm 0.0053	0.2462 \pm 0.0076
TimesNet	0.2677 \pm 0.0742	0.7256 \pm 0.0993	0.7256 \pm 0.0993	0.0297 \pm 0.0168
OFA	0.2959 \pm 0.0256	0.7949 \pm 0.0254	0.7949 \pm 0.0254	0.0442 \pm 0.0148
A.T.	0.2397 \pm 0.0828	0.5897 \pm 0.0911	0.5895 \pm 0.0913	0.0810 \pm 0.1053
FGANomaly	0.9208 \pm 0.0270	0.9560 \pm 0.0060	0.9560 \pm 0.0060	0.0606 \pm 0.0012
CAE-M	0.0719 \pm 0.0002	0.0218 \pm 0.0157	0.0218 \pm 0.0157	0.0000 \pm 0.0000
MTAD-GAT	0.2324 \pm 0.0271	0.6605 \pm 0.0518	0.6604 \pm 0.0518	0.0340 \pm 0.0008
OmniAnomaly	0.0764 \pm 0.0002	0.1057 \pm 0.0002	0.1057 \pm 0.0002	0.0000 \pm 0.0000
MSCRED	0.0936 \pm 0.0011	0.1213 \pm 0.0001	0.1213 \pm 0.0001	0.0000 \pm 0.0000
DAGMM	0.0746 \pm 0.0032	0.0242 \pm 0.0042	0.0242 \pm 0.0043	0.0018 \pm 0.0025

Model	PA-F1	Event-based-F1	R-based-F1	Affiliation-F1
CATCH	1.0000 \pm 0.0000	1.0000 \pm 0.0000	0.3093 \pm 0.0042	0.9489 \pm 0.0002
M2N2	0.8854 \pm 0.0087	0.3166 \pm 0.0229	0.1884 \pm 0.0014	0.7227 \pm 0.0062
FITS	0.9022 \pm 0.0017	0.1029 \pm 0.0215	0.1664 \pm 0.0008	0.5547 \pm 0.0177
ModernTCN	1.0000 \pm 0.0000	1.0000 \pm 0.0000	0.3197 \pm 0.0041	0.9500 \pm 0.0003
Peri-midFormer	0.9998 \pm 0.0002	0.9984 \pm 0.0022	0.3291 \pm 0.0016	0.9498 \pm 0.0002
SARAD	0.9992 \pm 0.0006	0.9943 \pm 0.0042	0.3447 \pm 0.0025	0.9495 \pm 0.0006
TimesNet	0.9142 \pm 0.0143	0.1597 \pm 0.0884	0.1677 \pm 0.0024	0.5771 \pm 0.0349
OFA	0.9179 \pm 0.0129	0.2311 \pm 0.0778	0.1692 \pm 0.0021	0.6041 \pm 0.0382
A.T.	0.9865 \pm 0.0131	0.6238 \pm 0.3904	0.1939 \pm 0.0414	0.7903 \pm 0.1868
FGANomaly	1.0000 \pm 0.0000	1.0000 \pm 0.0000	0.2745 \pm 0.0004	0.9487 \pm 0.0000
CAE-M	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	NaN \pm NaN
MTAD-GAT	0.9059 \pm 0.0014	0.1674 \pm 0.0051	0.1685 \pm 0.0001	0.5657 \pm 0.0018
OmniAnomaly	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	NaN \pm NaN
MSCRED	0.0000 \pm 0.0000	0.0000 \pm 0.0000	0.0000 \pm 0.0000	NaN \pm NaN
DAGMM	0.4000 \pm 0.5477	0.4000 \pm 0.5477	0.0991 \pm 0.1358	0.9553 \pm 0.0001

E.2 Efficiency Evaluation

In this section, we compare the Training Time per Epoch ($\mathcal{T}_{w/o}$, \mathcal{T}_w), GPU usage ($\mathcal{G}_{w/o}$, \mathcal{G}_w), and CPU usage ($\mathcal{C}_{w/o}$, \mathcal{C}_w) before and after applying IGAD for each dataset and each model from Tab.14 to Tab.17. From the defined formulas shown as (7) and (8), a model will perform four additional mappings after the application of IGAD. More concretely, before integration with IGAD, a model performs a mapping $f(x)$, while $f(x)$, $f(z)$, $f'(z)$, $f(f'(z))$, and $f'(f(z))$ are carried out sequentially under the effect of IGAD. **While more operations are involved, the application of IGAD does not introduce additional parameters that need to be trained, as IGAD leverages a frozen copy of the training model.** IGAD introduces additional time and resource consumption due to the mapping associated with the manifold constraints, but **these costs occur only in the training phases. During inference, each model can perform only one mapping for reconstruction, calculate anomaly scores, and detect abnormal time points.** This means that a model with IGAD can achieve the same efficiency as the same model without IGAD during inference. For a dataset with larger scale, the time is possible to last longer.

In the evaluation of *Training Time per Epoch*, we find that in most cases, the models with IGAD show an increase in time of less than 10 seconds and in many cases, even less than 3 seconds per training epoch. The increase generally has limited impact on the overall training process, especially when higher performance is desired. Meanwhile, it is also indicated that OFA [74] tends to show a higher increase during one epoch. This phenomenon can be theoretically concluded as that the utilize of a pre-trained language model is more source-sensitive to fine-tune it for time series tasks. An additional fact shows that the PSM dataset needs more training time than SMD, MSL, and SMAP. The reason is that the data scale of the PSM dataset is larger than others, and the cumulative effect of multiple iterations in the same epoch and multiple mappings results in a longer training time. In the evaluation of *Max GPU Allocation for Training* and *Max CPU Allocation for Training*, different degrees of increase are listed. During these, the maximum GPU occupancy is around 8535.80 MB (about 8.34 GB), and the maximum CPU occupancy is around 537.44 MB (about 0.52GB). All experiments are performed with a single NVIDIA RTX 3090 24GB GPU, and 0.52 GB is also acceptable for most hardware conditions for deep learning currently. These additional GPU and CPU memories are selected to save frozen parameters, gradient graphs, and data segments to update the training model.

Meanwhile, we have also envisioned some potential strategies to reduce these additional computational costs in our future implementations: (1) Accelerate calculation by data and model parallelism; (2) For large models such as OFA [74], we can perform parameter-efficient fine-tuning with advanced methods, including [33, 19, 55] based on LoRA [23] to reduce the number of trainable parameters; (3) A memory mechanism can be included to reduce the number of true reconstructions. Concretely, when a piece of reconstructed time series is needed, we can index the memory module to generate this in terms of association.

Table 14: Efficiency evaluation on dataset SMD, comparing training time per epoch ($\mathcal{T}_{w/o}$, \mathcal{T}_w), GPU usage ($\mathcal{G}_{w/o}$, \mathcal{G}_w), and CPU usage ($\mathcal{C}_{w/o}$, \mathcal{C}_w) before and after applying IGAD.

Model	Training Time per Epoch (s)			Max GPU Allocation for Training (MB)			Max CPU Allocation for Training (MB)		
	$\mathcal{T}_{w/o}$	\mathcal{T}_w	$\mathcal{T}_w - \mathcal{T}_{w/o}$	$\mathcal{G}_{w/o}$	\mathcal{G}_w	$\mathcal{G}_w - \mathcal{G}_{w/o}$	$\mathcal{C}_{w/o}$	\mathcal{C}_w	$\mathcal{C}_w - \mathcal{C}_{w/o}$
CATCH	6.0878	17.2277	11.1399	3038.8540	5317.9945	2279.1405	2252.5469	2362.8828	110.3359
M2N2	0.8424	1.2809	0.4385	6.5044	10.9773	4.4729	2190.1797	2603.9336	413.7539
FITS	1.1934	1.9876	0.7942	19.0562	39.9126	20.8564	4719.1914	4756.3789	37.1875
ModernTCN	3.4464	9.9653	6.5188	1679.2700	6333.5298	4654.2598	4776.7734	4793.4648	16.6914
Peri-midFormer	1.1767	5.8734	4.6967	289.3027	8825.1006	8535.7979	4773.2188	4826.9961	53.7773
SARAD	1.4122	2.5430	1.1308	778.5454	2392.6172	1614.0718	4751.1641	4766.2148	15.0508
TimesNet	1.3733	2.6364	1.2632	299.8774	411.8228	111.9453	4790.5352	4832.9365	42.4013
OFA	3.6833	21.4194	17.7361	1918.6016	6223.6816	4305.0801	4863.4492	4910.8242	47.3750
A.T.	2.0369	3.9399	1.9030	1834.0044	4781.5825	2947.5781	4760.8867	4789.8633	28.9766
FGANomaly	1.6585	2.1942	0.5357	58.0522	523.9907	465.9385	4739.8789	4746.5241	6.6452
CAE-M	0.9261	1.2022	0.2761	115.1748	301.7012	186.5264	4732.7109	4749.4766	16.7656
MTAD-GAT	1.2675	2.2031	0.9356	599.8101	2052.2046	1452.3945	4775.2891	4783.0039	7.7148
OmniAnomaly	1.0875	1.8211	0.7336	45.4243	169.6577	124.2334	4748.5391	4760.3894	11.8503
MSCRED	2.8281	10.2687	7.4406	3314.2231	8588.2192	5273.9961	4742.3281	4743.5591	1.2309
DAGMM	0.9995	1.1030	0.1034	10.8193	21.5449	10.7256	4698.9609	4701.4883	2.5273

Table 15: Efficiency evaluation on dataset MSL, comparing training time per epoch ($\mathcal{T}_{w/o}$, \mathcal{T}_w), GPU usage ($\mathcal{G}_{w/o}$, \mathcal{G}_w), and CPU usage ($\mathcal{C}_{w/o}$, \mathcal{C}_w) before and after applying IGAD.

Model	Training Time per Epoch (s)			Max GPU Allocation for Training (MB)			Max CPU Allocation for Training (MB)		
	$\mathcal{T}_{w/o}$	\mathcal{T}_w	$\mathcal{T}_w - \mathcal{T}_{w/o}$	$\mathcal{G}_{w/o}$	\mathcal{G}_w	$\mathcal{G}_w - \mathcal{G}_{w/o}$	$\mathcal{C}_{w/o}$	\mathcal{C}_w	$\mathcal{C}_w - \mathcal{C}_{w/o}$
CATCH	1.5938	3.3659	1.7721	4230.5103	8037.9696	3807.4593	2312.8828	2560.7383	247.8555
M2N2	0.5443	0.9091	0.3647	12.1436	20.9563	8.8127	2193.8672	2581.3945	387.5273
FITS	0.8837	1.0449	0.1612	28.5674	58.7539	30.1865	4731.3164	4768.3633	37.0469
ModernTCN	1.2722	2.3140	1.0418	2438.2866	9111.5781	6673.2915	4776.7539	4792.0039	15.2500
Peri-midFormer	0.8942	1.4531	0.5589	725.4805	4876.8081	4151.3276	4819.9063	4861.1347	41.2284
SARAD	0.9593	1.1728	0.2136	1113.1733	3489.0684	2375.8950	4747.0664	4777.9102	30.8438
TimesNet	0.8971	1.0556	0.1585	303.2749	419.8989	116.6240	4789.4727	4818.4922	29.0195
OFA	1.2976	3.4958	2.1982	1919.3901	6229.9653	4310.5752	4853.4688	5387.6055	534.1367
A.T.	0.9892	1.2458	0.2566	1253.3545	2974.2109	1720.8564	4768.0742	4797.3750	29.3008
FGANomaly	0.9215	1.0265	0.1051	59.7305	165.9316	106.2012	4728.4922	4733.8086	5.3164
CAE-M	0.8479	0.8915	0.0436	165.8027	435.1094	269.3066	4754.1836	4772.0156	17.8320
MTAD-GAT	0.8763	1.0014	0.1251	1006.0601	3351.1172	2345.0571	4759.7734	4762.5593	2.7859
OmniAnomaly	0.8426	0.9404	0.0978	49.0093	183.8770	134.8677	4740.9492	4768.8750	27.9258
MSCRED	1.2203	2.3347	1.1144	4697.5435	12323.8037	7626.2603	4767.2188	4767.6367	0.4180
DAGMM	0.8396	0.8602	0.0206	15.6182	32.5830	16.9648	4727.1992	4728.4023	1.2031

Table 16: Efficiency evaluation on dataset PSM, comparing training time per epoch ($\mathcal{T}_{w/o}$, \mathcal{T}_w), GPU usage ($\mathcal{G}_{w/o}$, \mathcal{G}_w), and CPU usage ($\mathcal{C}_{w/o}$, \mathcal{C}_w) before and after applying IGAD.

Model	Training Time per Epoch (s)			Max GPU Allocation for Training (MB)			Max CPU Allocation for Training (MB)		
	$\mathcal{T}_{w/o}$	\mathcal{T}_w	$\mathcal{T}_w - \mathcal{T}_{w/o}$	$\mathcal{G}_{w/o}$	\mathcal{G}_w	$\mathcal{G}_w - \mathcal{G}_{w/o}$	$\mathcal{C}_{w/o}$	\mathcal{C}_w	$\mathcal{C}_w - \mathcal{C}_{w/o}$
CATCH	44.4446	131.0026	86.5580	2228.2168	5258.5916	3030.3749	2285.8203	2389.8164	103.9961
M2N2	2.5754	4.8112	2.2358	3.7534	8.4933	4.7399	2229.9570	2725.6836	495.7266
FITS	4.0280	8.0651	4.0370	12.5127	26.6787	14.1660	4739.9609	4760.7305	20.7695
ModernTCN	15.9907	54.3645	38.3737	1105.6284	4172.5288	3066.9004	4769.9844	4785.1836	15.1992
Peri-midFormer	4.2939	22.7822	18.4883	146.5879	583.9233	437.3354	4780.7969	4785.6680	4.8711
SARAD	4.9100	14.7800	9.8700	551.7329	1610.5713	1058.8384	4745.7266	4748.1641	2.4375
TimesNet	5.4077	16.3706	10.9629	288.7642	400.0938	111.3296	4781.2070	4791.6211	10.4141
OFA	24.8965	225.4139	200.5174	1917.9985	6221.9976	4303.9990	4872.3086	5395.2734	522.9648
A.T.	13.7400	34.3816	20.6416	1870.1704	4878.5933	3008.4229	4787.3672	4831.0295	43.6623
FGANomaly	9.6031	15.4628	5.8597	56.7183	5440.7212	5384.0029	4759.8750	4766.3008	6.4258
CAE-M	1.7830	3.0684	1.2854	76.6055	201.3789	124.7734	4756.0391	4762.5078	6.4688
MTAD-GAT	3.6808	10.4775	6.7966	374.0786	1268.0796	894.0010	4972.4648	5000.8047	28.3398
OmniAnomaly	4.0156	10.3294	6.3138	43.8564	159.6650	115.8086	4752.7734	4767.4421	14.6687
MSCRED	16.2451	58.4670	42.2220	2259.1724	5729.1802	3470.0078	4765.6328	4765.6641	0.0313
DAGMM	1.8711	3.0836	1.2125	7.1514	14.2622	7.1108	4740.0156	4758.8516	18.8359

Table 17: Efficiency evaluation on dataset SMAP, comparing training time per epoch ($\mathcal{T}_{w/o}$, \mathcal{T}_w), GPU usage ($\mathcal{G}_{w/o}$, \mathcal{G}_w), and CPU usage ($\mathcal{C}_{w/o}$, \mathcal{C}_w) before and after applying IGAD.

Model	Training Time per Epoch (s)			Max GPU Allocation for Training (MB)			Max CPU Allocation for Training (MB)		
	$\mathcal{T}_{w/o}$	\mathcal{T}_w	$\mathcal{T}_w - \mathcal{T}_{w/o}$	$\mathcal{G}_{w/o}$	\mathcal{G}_w	$\mathcal{G}_w - \mathcal{G}_{w/o}$	$\mathcal{C}_{w/o}$	\mathcal{C}_w	$\mathcal{C}_w - \mathcal{C}_{w/o}$
CATCH	2.5922	6.4548	3.8626	2228.2236	4565.1578	2336.9341	2193.1484	2210.7266	17.5781
M2N2	0.6537	0.9807	0.3269	3.7534	7.4113	3.6579	2158.4492	2596.3633	437.9141
FITS	0.9403	1.2141	0.2737	13.1851	26.6787	13.4937	4717.2734	4723.3047	6.0313
ModernTCN	1.6361	3.5830	1.9468	1105.6284	4172.5288	3066.9004	4753.0820	4772.8477	19.7656
Peri-midFormer	1.0926	2.1401	1.0474	1263.2222	1407.9321	144.7100	4773.3984	4784.6780	11.2795
SARAD	1.0455	1.4257	0.3802	551.9321	1610.5713	1058.6392	4736.4648	4750.1250	13.6602
TimesNet	1.0640	1.6658	0.6017	306.8296	408.3438	101.5142	4765.6133	4771.3477	5.7344
OFA	2.2287	10.4260	8.1973	1917.9985	6221.9976	4304.0000	4855.8672	5393.3086	537.4414
A.T.	1.3751	2.2921	0.9170	1936.3228	5066.8916	3130.5688	4766.0039	4792.0977	26.0938
FGANomaly	1.1534	1.3887	0.2354	56.7046	228.3887	171.6841	4727.0039	4740.8906	13.8867
CAE-M	0.8584	0.9544	0.0960	76.6055	201.3789	124.7734	4736.6563	4746.7227	10.0664
MTAD-GAT	0.9592	1.3120	0.3528	374.0786	1268.0796	894.0010	4734.4883	4738.7656	4.2773
OmniAnomaly	0.9489	1.2228	0.2739	43.8564	159.6650	115.8086	4728.8945	4739.2734	10.3789
MSCRED	1.6051	3.7906	2.1856	2259.1724	5729.1802	3470.0078	4743.4219	4744.6059	1.1841
DAGMM	0.8615	0.9203	0.0588	7.1514	14.2622	7.1108	4711.6016	4737.3438	25.7422

E.3 Hyperparameter Analysis

Here, we list the instructions to choose the optimal λ_{idem} , λ_{tight} and α for a given dataset and select the model DAGMM [75] with dataset SMAP to show the analysis of hyperparameters. The results are displayed in Fig.8, Fig.9 and Fig.10.

First, in Fig.8, we calculate the frequency of λ_{idem} , λ_{tight} , and α for all the experiments conducted. The results indicate that, for most cases, the values of λ_{idem} are located in the interval $[0.1, 0.5]$, which means that a relatively smaller λ_{idem} may be better for new datasets. For the values of λ_{tight} , they focus mainly on the two endpoint values and maintain a relatively uniform distribution throughout the other central parts. Finally, a larger α may be considered as a priority.

In the following part, the parameter sensitivity analysis conducted in the following also supports the instructions listed above. We show the results with mean and standard deviation in Fig.9, and 95% confidence intervals in Fig.10. Concretely, we fix two of λ_{idem} , λ_{tight} and α as the optimal values shown in Tab.5. Then, we vary the remaining one from 0.1 to 2.0 with a step of 0.1. For λ_{idem} , the model achieves the best performance when λ_{idem} is 0.1 (a smaller one). For λ_{tight} , with the tightness effect changing from loose to strict (λ_{tight} changing from small to large), the performance of the model changes from up to down. We attribute it to over tightness, which even drops out normal instances. For α , it is clearly shown that the performance improves with larger α and levels off when α is greater than 1.5.

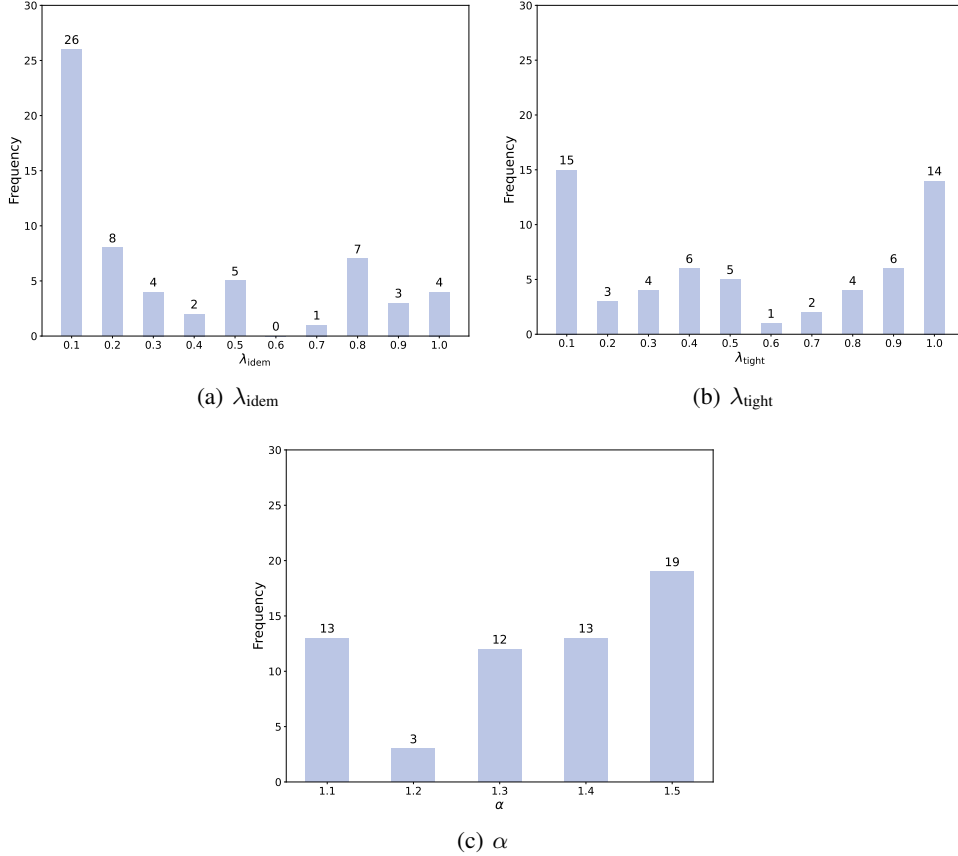
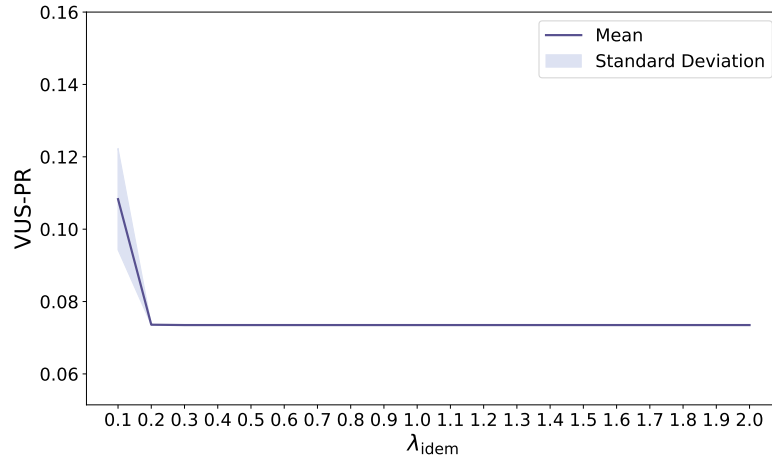
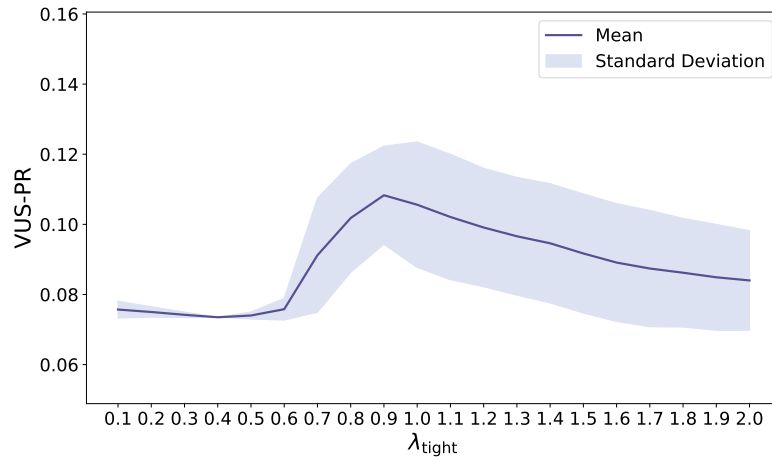


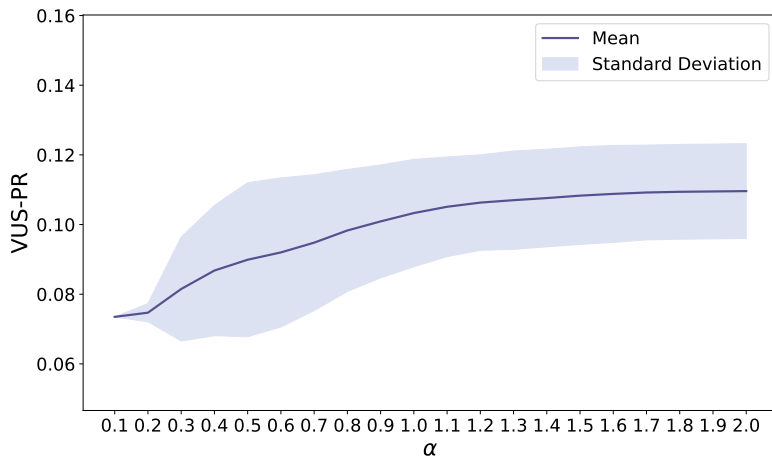
Figure 8: Parameter frequency records for λ_{idem} , λ_{tight} and α .



(a) λ_{idem}

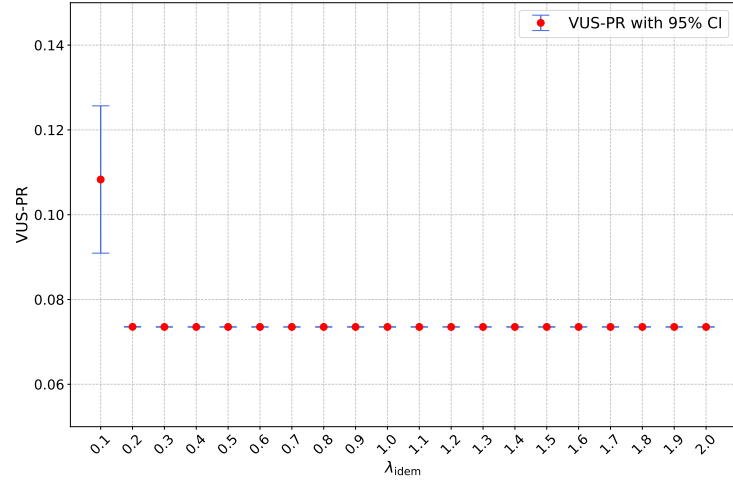


(b) λ_{tight}

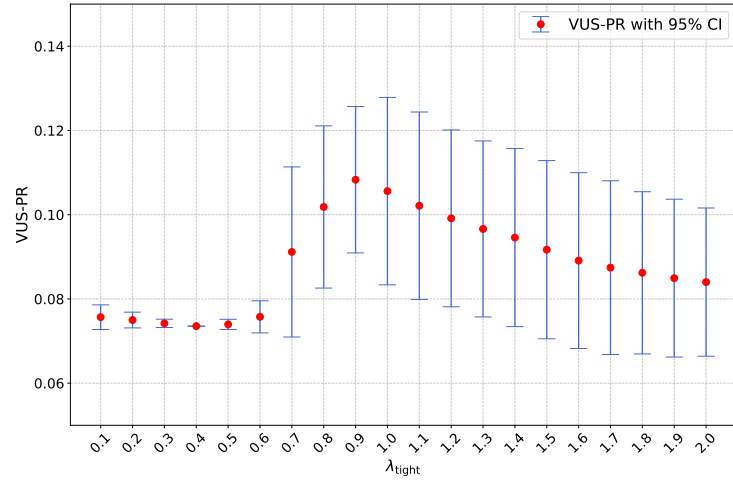


(c) α

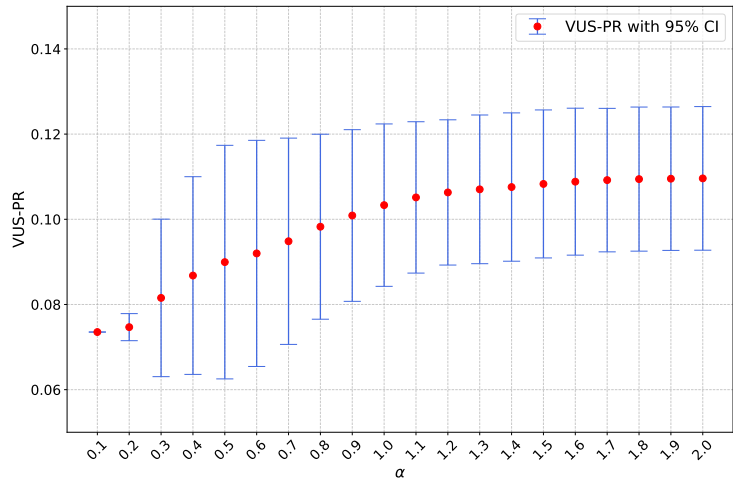
Figure 9: Parameter sensitivity analysis for λ_{idem} , λ_{tight} and α with mean and standard deviation.



(a) λ_{idem}



(b) λ_{tight}

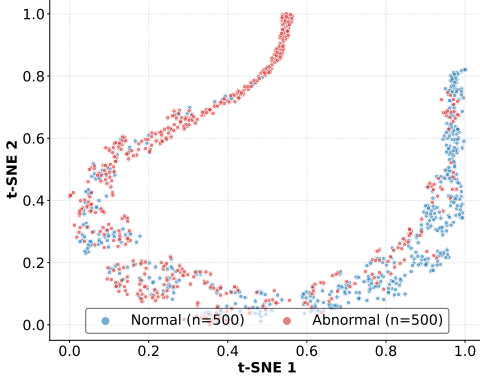


(c) α

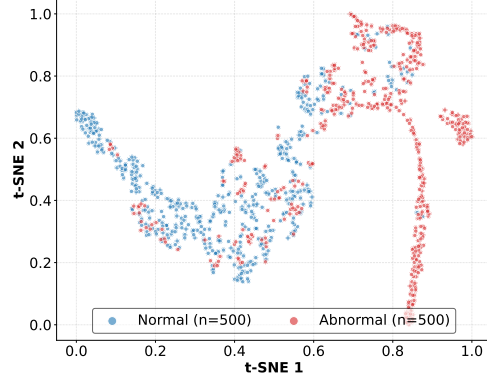
Figure 10: Parameter sensitivity analysis for λ_{idem} , λ_{tight} and α with 95% confidence intervals.

E.4 Visualization of Latent Space

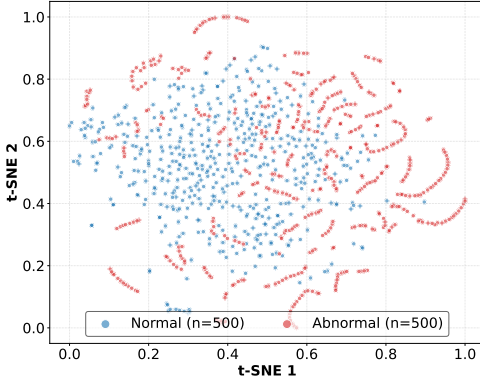
To further verify the effectiveness of IGAD, we visualize the latent space of different models before and after applying IGAD in Fig.11. It can be observed that, under the effect of IGAD, the model gains a clearer boundary to distinguish normal instances from abnormal instances. This aligns with our design principles to modify and tighten the target manifold $\mathcal{M}_{\text{target}}$, with the aim of containing enough normal instances and drop out potential abnormal instances.



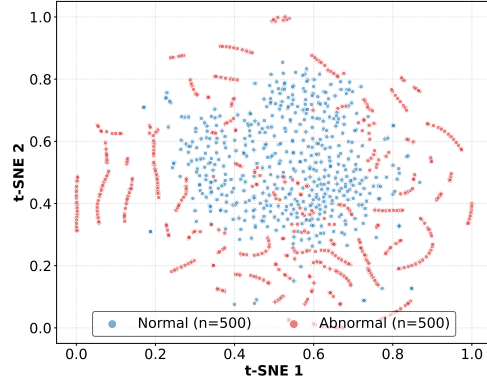
(a) DAGMM on SMD without IGAD.



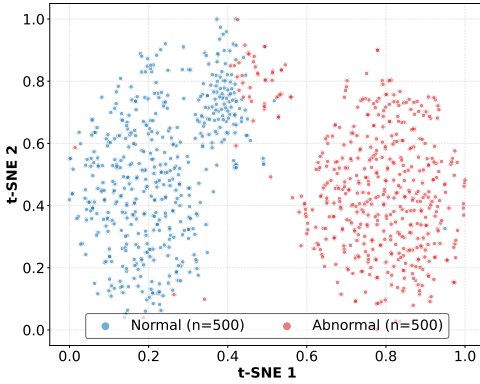
(b) DAGMM on SMD with IGAD.



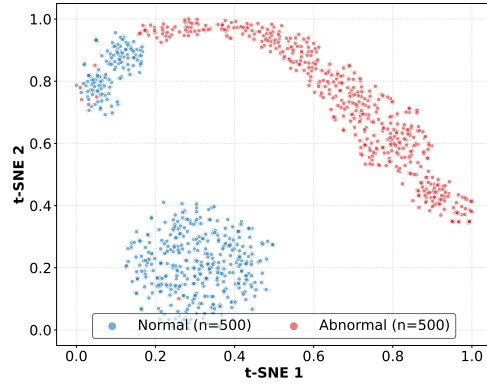
(c) OFA on SMD without IGAD.



(d) OFA on SMD with IGAD.



(e) OmniAnomaly on SMAP without IGAD.

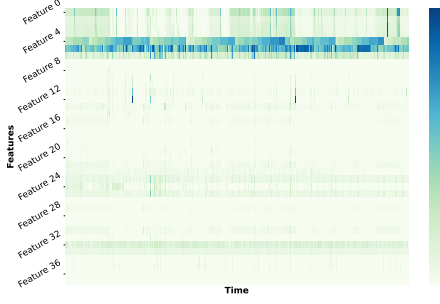


(f) OmniAnomaly on SMAP with IGAD.

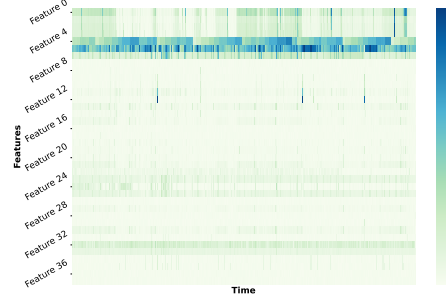
Figure 11: The visualization of latent space using t-SNE before and after applying IGAD.

E.5 Maintain Data Patterns under Noise

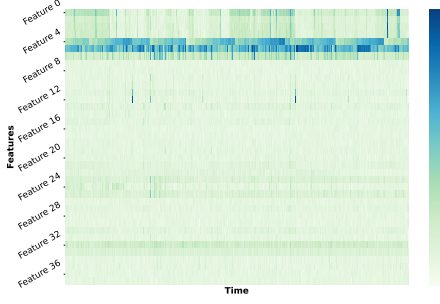
In our experiments to demonstrate that IGAD can help balance robustness and sensitivity in Sect.4.2.3, we have incorporated a noise strategy into the testing data. In this part, we employ heatmaps to show these data with weighted noise from Fig.12 to Fig.15. We have found that noise-effected testing data display similar change patterns with the original data, which means that our noise strategy can verify their abilities to balance the robustness and sensitivity of different models while maintaining the necessary information.



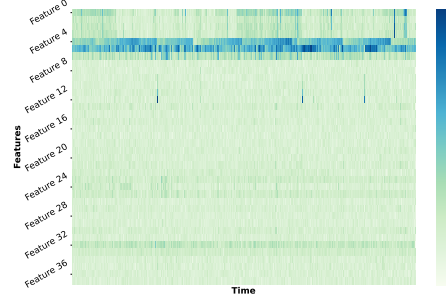
(a) Original time series data.



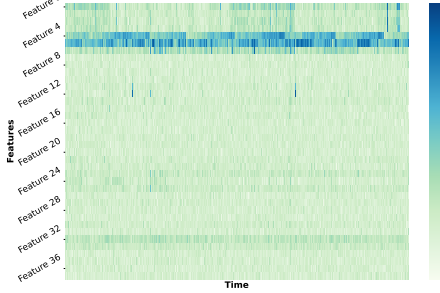
(b) Time series data with noise weight 1%.



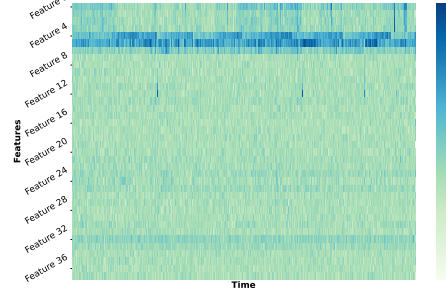
(c) Time series data with noise weight 5%.



(d) Time series data with noise weight 10%.

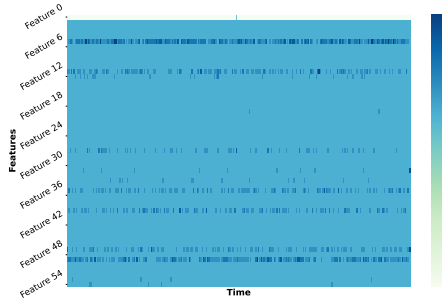


(e) Time series data with noise weight 15%.

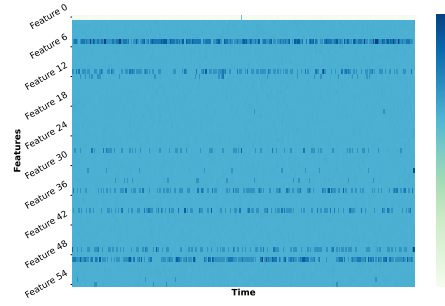


(f) Time series data with noise weight 20%.

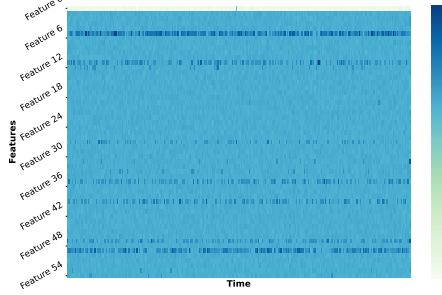
Figure 12: Visualization for original data and noise-effect data on SMD.



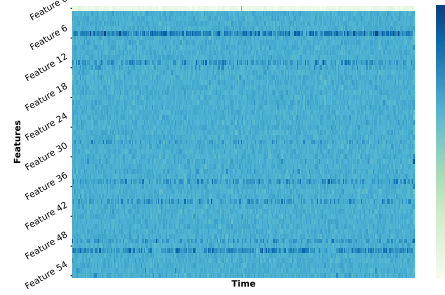
(a) Original time series data.



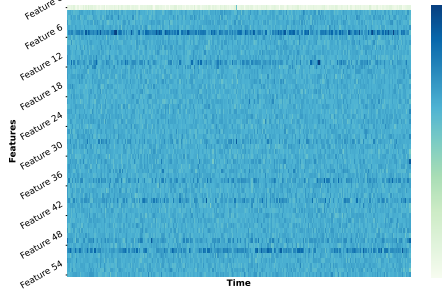
(b) Time series data with noise weight 1%.



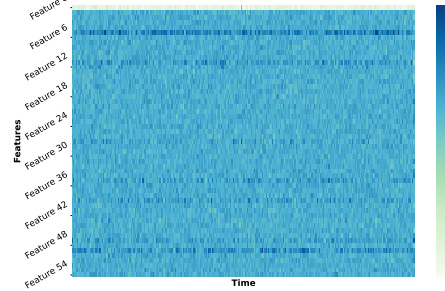
(c) Time series data with noise weight 5%.



(d) Time series data with noise weight 10%.

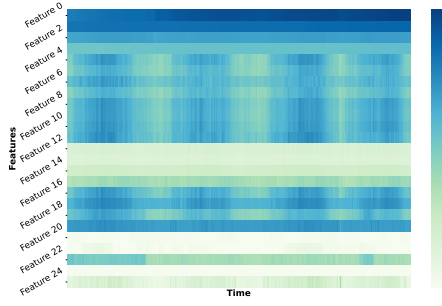


(e) Time series data with noise weight 15%.

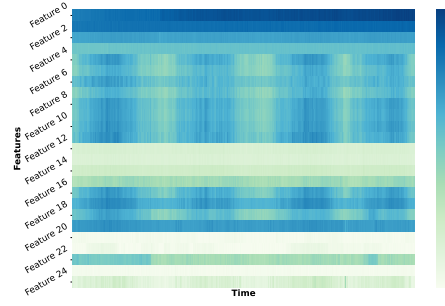


(f) Time series data with noise weight 20%.

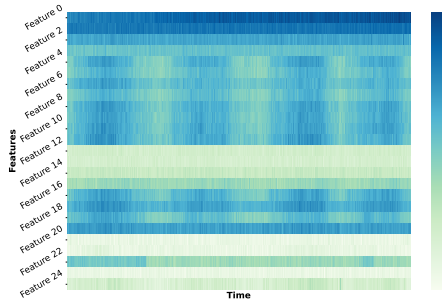
Figure 13: Visualization for original data and noise-effect data on MSL.



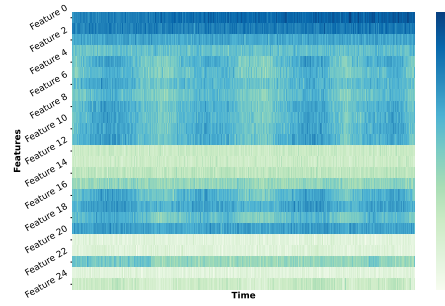
(a) Original time series data.



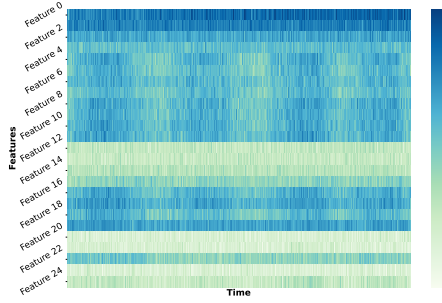
(b) Time series data with noise weight 1%.



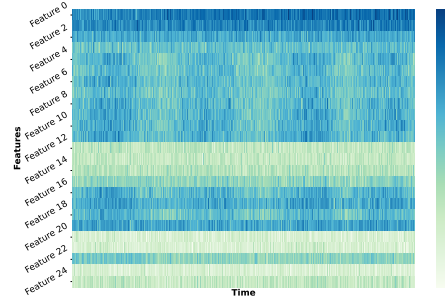
(c) Time series data with noise weight 5%.



(d) Time series data with noise weight 10%.

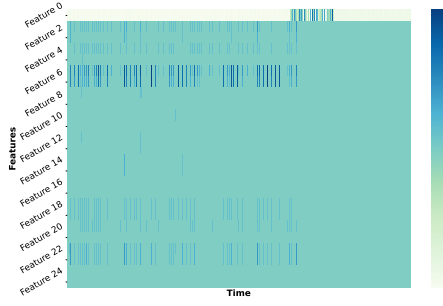


(e) Time series data with noise weight 15%.

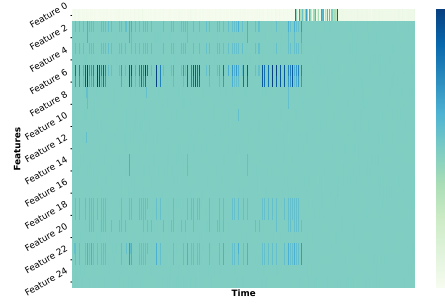


(f) Time series data with noise weight 20%.

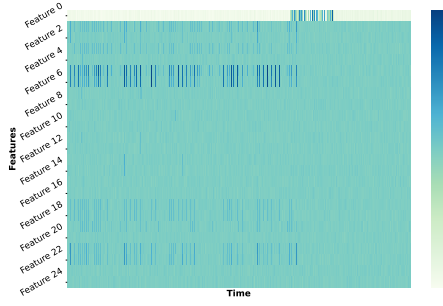
Figure 14: Visualization for original data and noise-effect data on PSM.



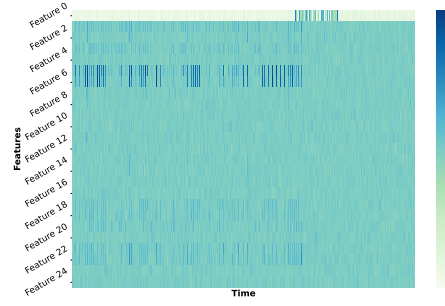
(a) Original time series data.



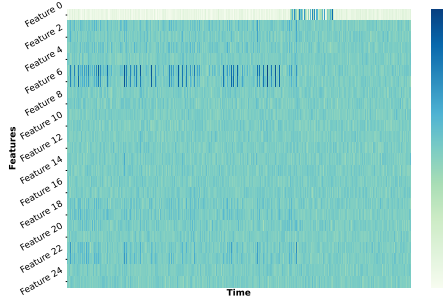
(b) Time series data with noise weight 1%.



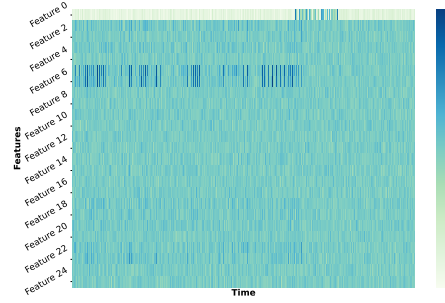
(c) Time series data with noise weight 5%.



(d) Time series data with noise weight 10%.



(e) Time series data with noise weight 15%.



(f) Time series data with noise weight 20%.

Figure 15: Visualization for original data and noise-effect data on SMAP.

E.6 Nyquist Criteria

We perturbed the time series data in the frequency domain to balance robustness and sensitivity in our experiments. This indicates that we should perform more analysis in the frequency domain to support this strategy. The **Nyquist Criterion** establishes a fundamental requirement for faithful reconstruction of a continuous-time signal from its discrete samples. This principle underpins modern digital signal processing systems, analog-to-digital conversion, and telecommunications. The theorem originated from Harry Nyquist's work in [37] and was later rigorously formalized by Claude Shannon in information theory [45]. It states that a band-limited signal with no frequency components exceeding f_{\max} (Hz) must be sampled at a rate $f_s \geq 2f_{\max}$ to avoid aliasing and ensure complete signal recovery. We conclude the key concepts in this theory as follows:

- The minimum sampling rate $2f_{\max}$ is termed the **Nyquist rate**.
- Half of the sampling frequency $\frac{f_s}{2}$ is referred to as the **Nyquist frequency**, f_{Nyq} .
- Spectral overlap, i.e., **aliasing**, may occur if $f_s < 2f_{\max}$ or $f_{\text{Nyq}} < f_{\max}$, causing irreversible distortion.

To verify compliance with the Nyquist-Shannon sampling theorem, the proposed procedure shown in Alg.1 first determines the sampling rate f_s and the corresponding Nyquist frequency f_{Nyq} from a user-provided sampling frequency descriptor for the input multivariate time series dataset \mathbf{D} . For each time series feature within \mathbf{D} , its frequency spectrum is obtained by the Fast Fourier Transform (FFT). The significant frequency components are then identified by comparing their normalized magnitudes against a relative threshold based on the peak magnitude in the spectrum. If the highest significant frequency detected in any feature exceeds f_{Nyq} , the dataset is flagged as non-compliant; otherwise, compliance is affirmed.

Algorithm 1 Nyquist Criterion Compliance Verification

```

1: Input:
    • Multivariate time series dataset  $\mathbf{D} \in \mathbb{R}^{n \times k}$  ( $n$  instances,  $k$  variables)
    • Sampling frequency descriptor  $f_{\text{desc}}$  (e.g., "1 min")
2: Output: Boolean compliance status flagNyq
3: procedure CHECKNYQUIST( $\mathbf{D}, f_{\text{desc}}$ )
4:   Compute sampling rate:  $f_s = 1/\Delta t$  according to  $f_{\text{desc}}$  ▷ Unit: Hz
5:   Calculate Nyquist frequency:  $f_{\text{Nyq}} = f_s/2$ 
6:   for each feature column  $\mathbf{d}_i \in \mathbf{D}$  (where  $i$  is the feature index) do
7:     Let  $N_s = |\mathbf{d}_i|$  be the number of samples in the current feature column.
8:     Compute FFT:  $\mathbf{Y}_i = \mathcal{F}(\mathbf{d}_i)$ 
9:     Generate frequency axis (positive frequencies):  $\mathbf{f}_i = \text{fftfreq}(N_s, \Delta t)[0 : N_s/2]$ 
10:    Compute normalized magnitude:  $\mathbf{A}_i = |\mathbf{Y}_i[0 : N_s/2]|/N_s$ 
11:    Detect significant frequencies:
        
$$\mathcal{F}_{\text{sig}} = \{f \in \mathbf{f}_i \mid A(f) > 0.1 \cdot \max(\mathbf{A}_i)\}$$

12:    if  $\mathcal{F}_{\text{sig}} = \emptyset$  then
13:      Continue ▷ No significant frequencies above threshold for this feature
14:    if  $\max(\mathcal{F}_{\text{sig}}) > f_{\text{Nyq}}$  then
15:      return False ▷ Violation: Max significant frequency exceeds Nyquist frequency
16:  return True ▷ All features comply with Nyquist criterion

```

From the study [50], we can get the information that the selected datasets in our experiments, SMD, MSL, PSM and SMAP, are sampled with a sampling frequency of 1 min, so we set $f_s = 60$ according to the unit in seconds. **After verification, all datasets in our experiments satisfy the Nyquist criterion.** Further, due to the space constraints on each page, we visualized the validation results for the first eight variables of each dataset from Fig.16 to Fig.19. The full validation codes can be found in our anonymized repository and run directly to carry out and check the results of full verification.

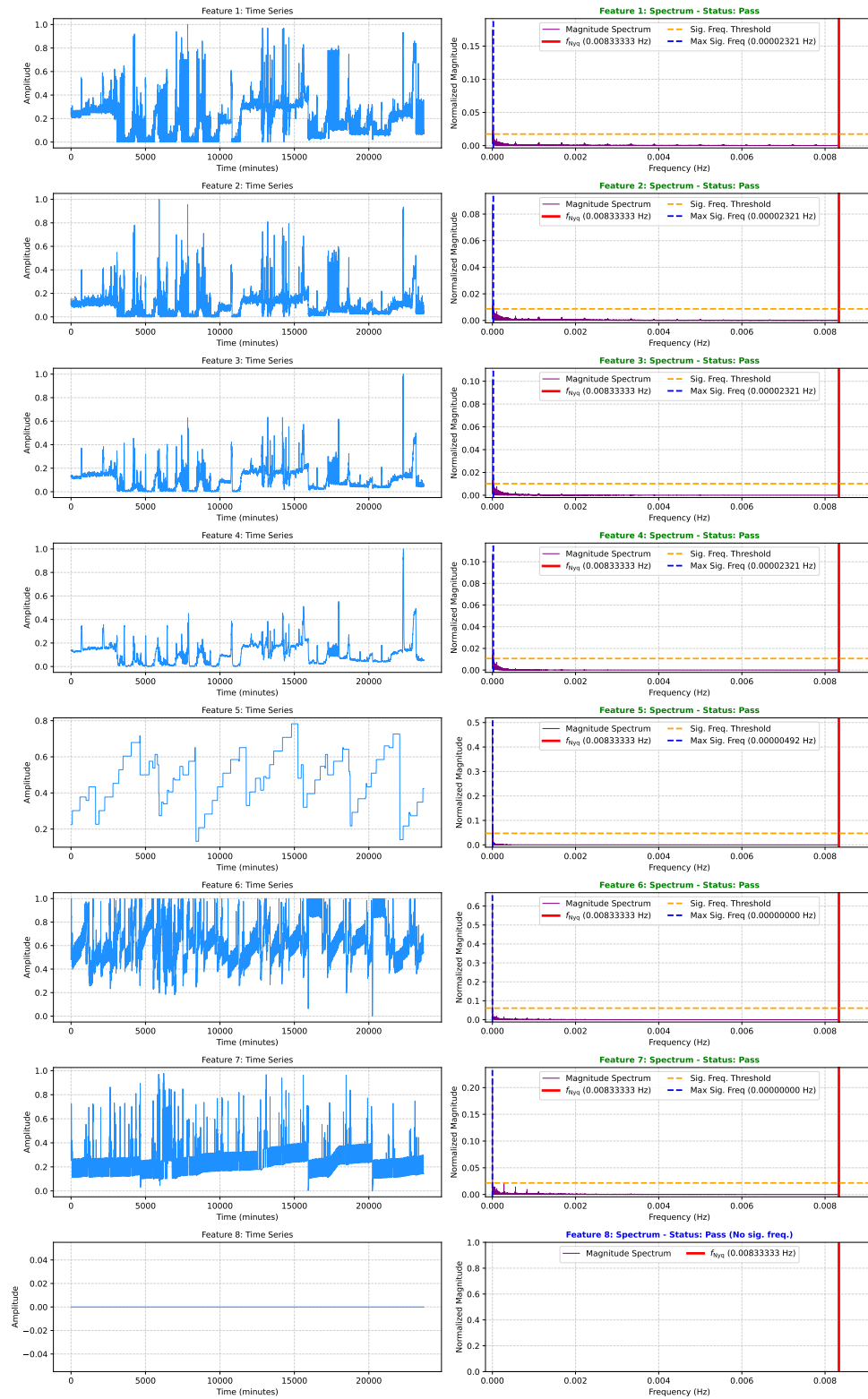


Figure 16: Nyquist criteria verification on SMD.

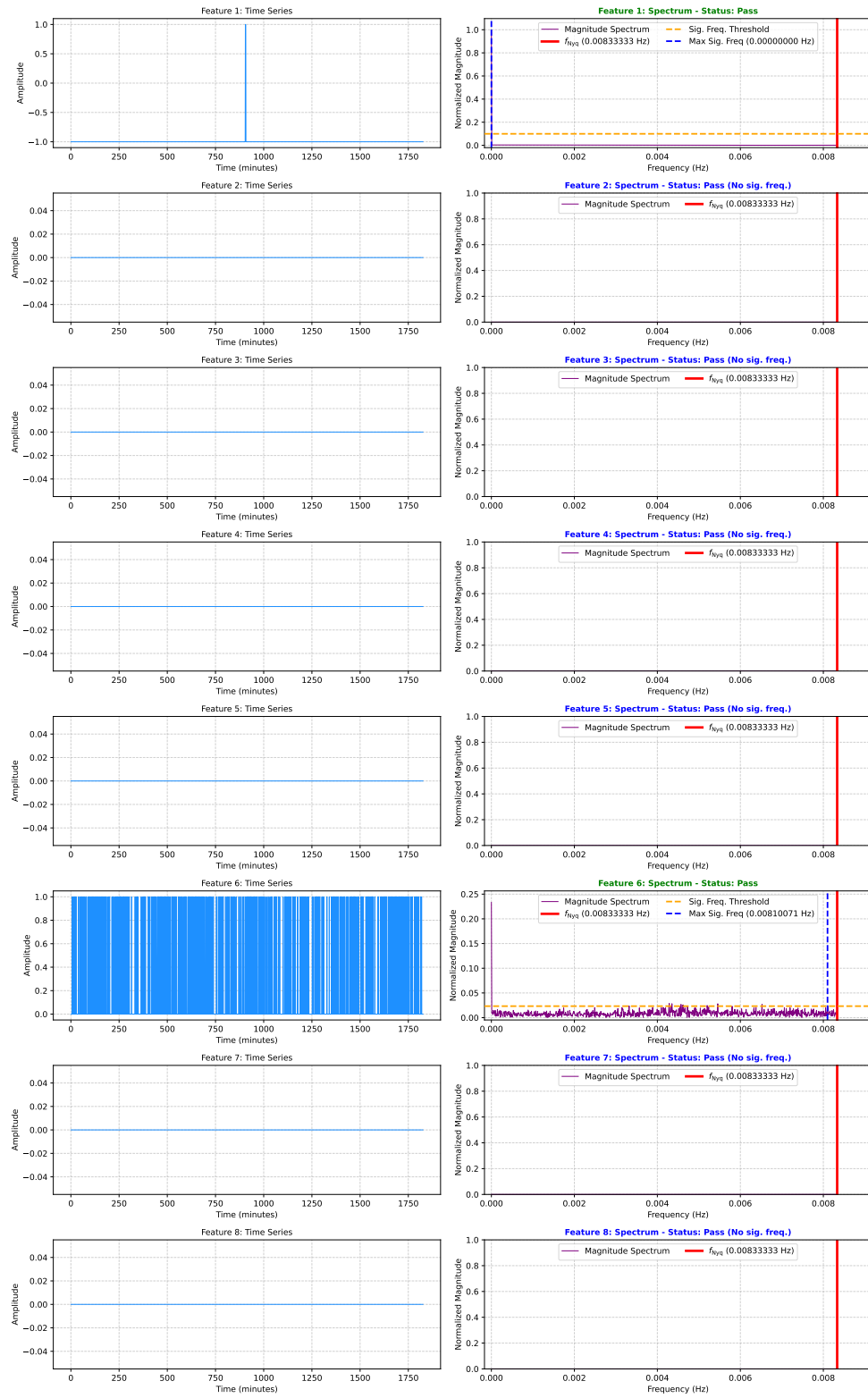


Figure 17: Nyquist criteria verification on MSL.

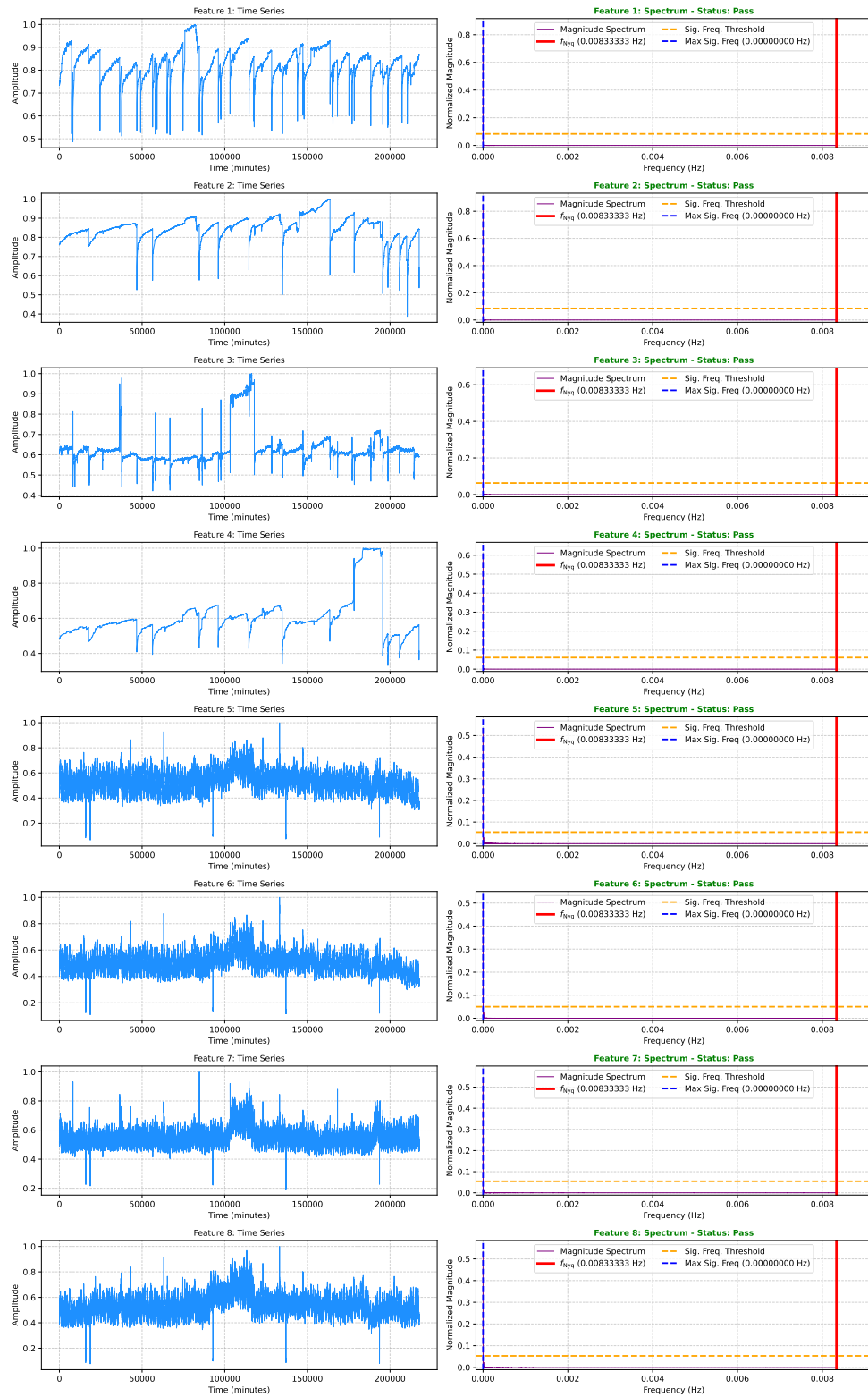


Figure 18: Nyquist criteria verification on PSM.

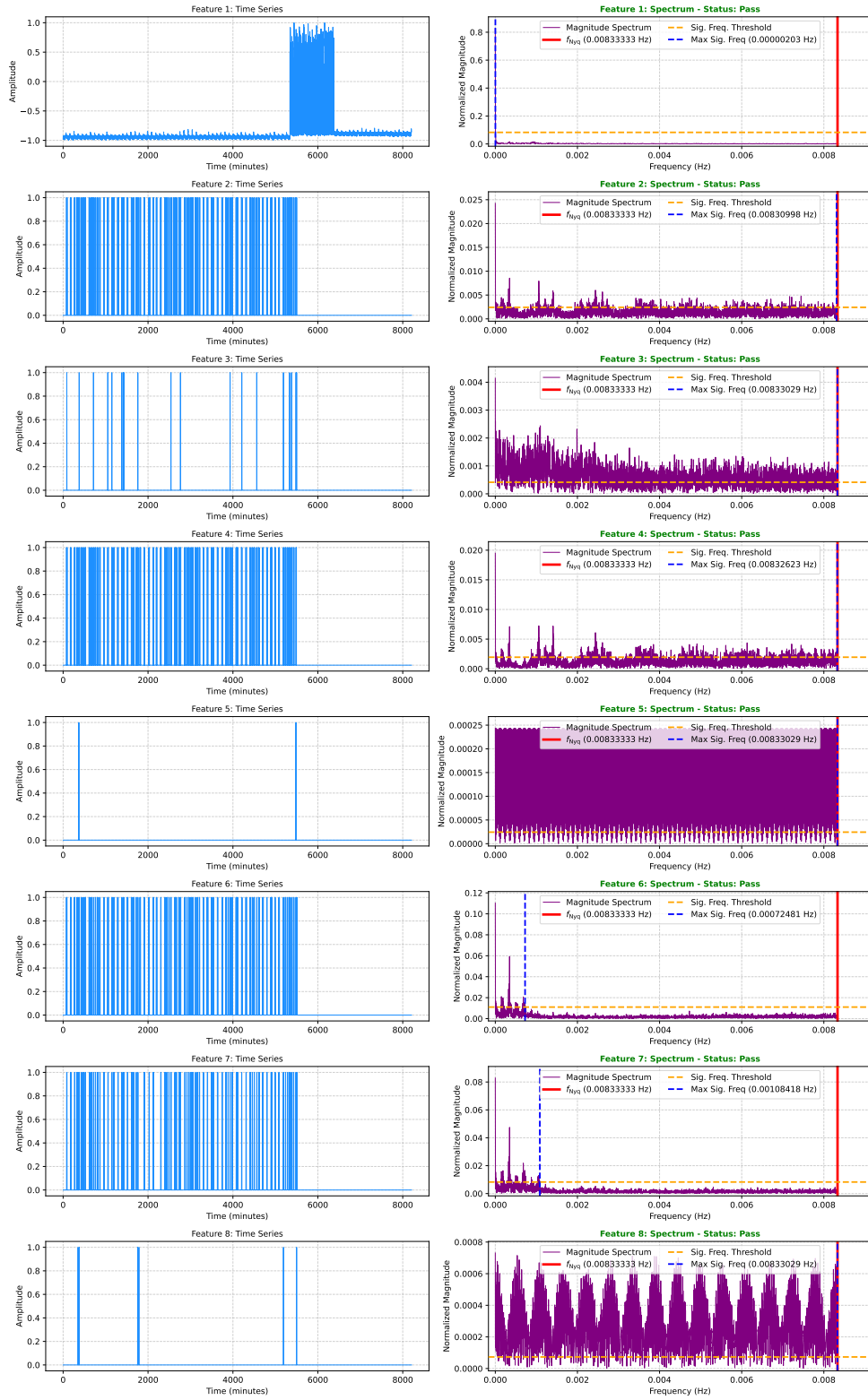


Figure 19: Nyquist criteria verification on SMAP.

E.7 Comparison with Contrastive-based Models

As shown in Tab.1, noticeable improvements can be observed for different models. In this section, our aim is to explain that **reconstruction-based methods are also competitive candidates, so it is necessary to further optimize for better performance when conducting MTS AD**. Given these, we compare selected reconstruction-based methods with contrastive-based models, including DCdetector [66], TS-TCC [15] and CoST [57], which are also powerful tools in recent years. The results are listed in Tab.18. The results indicate that many reconstruction-based methods also outperform contrastive-based methods and become more competitive under the effect of IGAD.

Table 18: $\overline{\text{VUS-PR}}$ under five random seeds for contrastive learning model. \dagger denotes the average value of all the mean values reported in Tab.1.

Model	Dataset			
	SMD	MSL	PSM	SMAP
DCdetector	0.0210	0.0096	0.1117	0.1508
TS-TCC	0.0221	0.0088	0.1064	0.1491
CoST	0.0217	0.0228	0.1118	0.1459

Model	Dataset			
	SMD	MSL	PSM	SMAP
Mean of Contrastive-based Methods	0.0216	0.0137	0.1010	0.1486
Mean of Reconstruction-based Methods w/o IGAD	0.1554 \dagger	0.0416 \dagger	0.1542 \dagger	0.3223 \dagger
Mean of Reconstruction-based Methods w/ IGAD	0.1752\dagger	0.0920\dagger	0.1592\dagger	0.4056\dagger

E.8 Codes for IGAD

For better understanding of IGAD, we also provide a pseudo-code block to help better describe the IGAD working flow during the training process, as shown in Code.1.

```
1  # First, we define:
2  # f: The training model initialized by f =
   ↪ Model(parameters...).to(device)
3  # f_copy (the defined f'): The frozen model initialized by
4  # f_copy = Model(parameters...).requires_grad(False).to(device)
5  def train_in_a_single_iteration(f, f_copy, data):
6
7      # f_copy is the frozen of current training model
8      f_copy.load_state_dict(f.state_dict())
9
10     recon_data = f(data)
11
12     z = get_augumented_data(data) # Get z^i for x^i
13     fz = f(z) # f(z^i)
14     f_z = fz.detach() # f'(z^i)
15     ff_z = f(f_z) # f(f'(z^i))
16     f_fz = f_copy(fz) # f'(f(z^i))
17
18     # Calculate losses
19     loss_rec = (recon_data - data).pow(2) # Reconstruction
20     loss_idem = (f_fz - fz).pow(2) # Idempotent
21     loss_tight = -(ff_z - f_z).pow(2) # Tightness
22     # loss_auxiliary if exists
23
24     # Optimize for losses
25     loss = lambda_rec * loss_rec + lambda_idem * loss_idem +
   ↪ lambda_tight * loss_tight # loss_auxiliary if exists
26     opt.zero_grad()
27     loss.backward()
28     opt.step()
```

Listing 1: Python implementation for IGAD.

F Limitation and Future Work

Although significant improvements have been observed, there remain unexplained performance drops in a limited number of experiments. This warrants further investigation to identify the underlying causes. Meanwhile, the slightly larger standard deviation observed in certain cases suggests the need to optimize the training process to achieve more stable convergence in our future work. The workflow of IGAD also inspires us to explore potential strategies to reduce computational complexity for large models, such as OFA [74].

G Impact Statements

This paper presents work focused on advancing the field of multivariate time series anomaly detection, with applications in healthcare, finance, and industrial monitoring. Although the ethical implications of anomaly detection are generally well-established, the misuse of such methods in sensitive areas could lead to privacy concerns and unintended biases in decision-making. We believe that this research contributes to improving anomaly detection techniques, improving system reliability, and early warning capabilities. Specific ethical issues are not identified beyond these general considerations.