JELI: JOINT MODELING OF LATENT AND INPUT-SPACE EVIDENCE FOR TIME SERIES ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Time-series anomaly detection is a crucial task, commonly formulated as learning characteristic temporal patterns from normal observations and identifying abnormal deviations. Prior work has broadly explored time-series anomaly detection in two spaces, with input-space approaches detecting anomalies through discrepancies between inputs and their reconstructions, while latent-space approaches identify deviations from normal patterns in learned representations. However, the two approaches have largely evolved as separate research directions, even though they can serve complementary roles in anomaly detection. In this paper, we propose **JELI** (**J**oint **E**vidence from **L**atent and **I**nput Spaces), a unified framework that integrates contextual anomaly evidence from both spaces for time-series anomaly detection. Specifically, we first train a JEPA-based backbone to learn contextual representations from surrounding input context and predict target representations at arbitrary temporal positions, forming a normal representation space. Second, we finetune the pretrained backbone for contextual reconstruction with an attention-masking bottleneck that prevents each target from attending to its own representation while preserving alignment with the pretraining objective. Lastly, to bridge the two sources of anomaly evidence, we design a joint-distribution-based scoring method that explicitly accounts for their dependence. Through extensive experiments, we demonstrate that our framework substantially outperforms prior state-of-the-art baselines on the time-series anomaly detection benchmarks.

1 INTRODUCTION

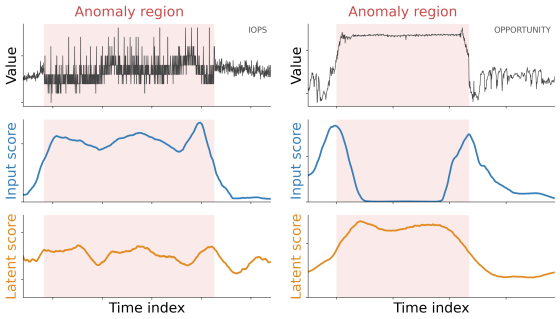
Time-Series Anomaly Detection (*TSAD*) aims to identify abnormal observations that deviate from normal temporal behavior. It is essential for various real-world applications, including industrial monitoring (Wen et al., 2022), healthcare (Wang et al., 2023), and online service (Ren et al., 2019). However, detecting anomalies is challenging due to complex temporal dependencies and non-stationarity in time-series data. Moreover, TSAD is commonly studied in an unsupervised setting where anomaly labels are unavailable, requiring models to learn normal temporal behavior from normal observations and determine what constitutes abnormality. Therefore, recent deep learning approaches have focused on extracting reliable anomaly evidence from learned normality (Zamanzadeh Darban et al., 2024; Sánchez-Ferrera et al., 2025).

Among prevalent directions, prior work can be broadly grouped according to whether anomalies are assessed in the input space or the latent space. In the input space, a common strategy is to learn normal temporal patterns through reconstruction and assess anomalies based on discrepancies between observations and their reconstructions. Recent methods enrich the reconstruction process by incorporating frequency-domain modeling (Xu et al., 2024; Wu et al., 2025) and multi-scale temporal modeling (Li et al., 2025). By directly comparing observed and reconstructed values within each window, reconstruction-based scoring can reveal fine-grained discrepancies in local temporal patterns. In contrast, latent-space approaches assess anomalies through learned representations, often obtained through self-supervised learning (Yue et al., 2022; Yang et al., 2023). For example, a recent direction is to learn window-level[1] representations in a self-supervised manner and evaluate them against normal reference representations obtained from training data (Park & Kang, 2026). Unlike methods that rely on comparisons within a single window, this approach compares temporal patterns

[1]Here, a window denotes a fixed-length contiguous segment of the time series.

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.

summarized over each window with these normal references, providing a coarse-grained perspective on deviations from learned normality.



(a) Fine-grained changes. (b) Window-level shift.

Figure 1: Complementary responses of the input- and latent-space anomaly scores. The input-space score is sensitive to fine-grained local variations, whereas the latent-space score captures sustained deviations at the window level.

Accordingly, input-space and latent-space anomaly scores provide distinct perspectives on anomalous behavior. As shown in Figure 1a, when fine-grained changes occur within an anomalous segment, the input-space score responds strongly because it directly measures discrepancies between observed and reconstructed values within each window. In contrast, the latent-space score is less sensitive to these fine-grained variations and remains at similar levels across the anomalous and normal segments. Figure 1b shows a different case, where the anomaly appears as a sustained shift over the window. In this case, the input-space score is less responsive because it reflects discrepancies in local temporal patterns within each window rather than directly assessing deviations at the window level. In contrast, the latent-space score remains elevated throughout the anomalous segment by evaluating how its window-level representation differs from normal references, thereby capturing the sustained deviation from a coarse-grained perspective. Despite this complementarity, input-space and latent-space approaches have largely been studied as separate research directions (Zamanzadeh Darban et al., 2024).

Motivated by these complementary roles, we propose **JELI** (**J**oint **E**vidence from **L**atent and **I**nput Spaces), a unified framework that integrates contextual anomaly evidence from both spaces for time-series anomaly detection. To capture anomaly evidence at different granularities, we construct coarse-grained signals in the latent space and fine-grained signals in the input space, and jointly model the two signals to obtain a unified anomaly score. (*i*) For latent-space signals, we pretrain the model with a JEPA-style objective (Assran et al., 2023) that predicts target representations from surrounding context, while incorporating statistical information through separate statistical tokens. We then average the token embeddings within each window to obtain a coarse-grained window representation. We model the distribution of these representations over normal windows and compute the Mahalanobis distance (Mahalanobis, 1936) of each test window to derive its latent-space anomaly score. (*ii*) For input-space signals, we finetune the pretrained model for contextual reconstruction with an attention-masking bottleneck, maintaining the contextual learning scheme established during pretraining. Subsequently, we compute the point-wise difference between the reconstructed and original inputs to obtain fine-grained anomaly evidence, which is averaged within each window to derive the input-space anomaly score. (*iii*) To integrate the two anomaly scores, we model their joint distribution to account for their dependence. The resulting joint distribution provides a unified anomaly score that reflects how the two signals vary together.

Through extensive experiments, we demonstrate the effectiveness of our framework against recent baselines on the TSB-AD benchmark (Liu & Paparrizos, 2024). Notably, our method ranks first on all six evaluation metrics on both TSB-AD-U and TSB-AD-M, which correspond to the univariate and multivariate settings, respectively. Compared with the recent work, PaAno (Park & Kang, 2026), our method achieves relative VUS-PR improvements of 23.5% on TSB-AD-U and 11.8% on TSB-AD-M. We further provide an ablation study in Section 3.3 to assess the contribution of our design choices, and present case studies in Section 3.4 to provide insights into how our framework operates.

Overall, our contributions are summarized as follows:

- We propose a unified framework that jointly couples complementary anomaly evidence from the latent and input spaces.
- We introduce an anomaly scoring method that models the joint distribution of latent- and input-space scores to capture their dependence and combine them without weighting hyperparameters.
- Our method achieves state-of-the-art performance on the TSB-AD benchmark, ranking first across all six evaluation metrics on both TSB-AD-U and TSB-AD-M settings.

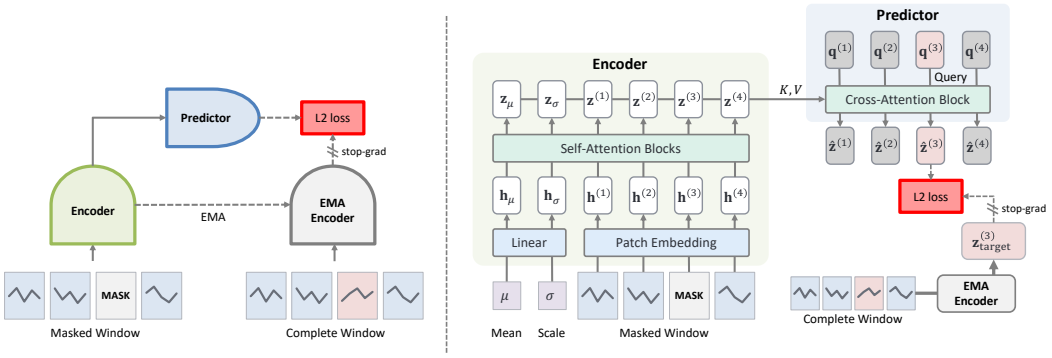


Figure 2: **Left**: the overall pretraining process for predicting target representations from masked inputs. **Right**: the detailed encoder-predictor architecture.

2 METHODOLOGY

In this section, we present **JELI**, a unified framework that derives complementary contextual anomaly evidence from the latent and input spaces and integrates this evidence through joint distribution modeling. We first learn latent-space representations through contextual prediction, while explicitly retaining window-level statistics that are not preserved in normalized patch representations (Section 2.1). Building on the pretrained model, we then derive complementary input-space evidence through contextual reconstruction with a bottleneck that blocks direct access to each target patch's own representation (Section 2.2). Finally, we jointly model the resulting latent- and input-space anomaly scores to account for their dependence (Section 2.3).

Problem Setup. In this paper, we consider time-series anomaly detection, which aims to identify anomalous observations in time-series data. Formally, given an input window $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_L] \in \mathbb{R}^{C \times L}$, where C and L denote the number of channels and the window length, respectively, an anomaly score a_t is computed for each time point $t \in \{1, \ldots, L\}$. Then, the resulting score is used to determine the anomaly label $\hat{y}_t \in \{0, 1\}$, with larger values of a_t indicating a stronger degree of abnormality. Since our method models each channel independently as a univariate time series, we omit the channel index in what follows. For multivariate time series, the final score is obtained by averaging across channels.

2.1 LEARNING REPRESENTATIONS VIA CONTEXTUAL PREDICTION

To derive latent-space anomaly evidence, we aim to learn representations that capture how each patch relates to its surrounding context. An observation can be anomalous not only because of its individual characteristics, but also because of its inconsistency with the surrounding context. To this end, we pretrain our model with a JEPA-style contextual prediction objective (Assran et al., 2023), predicting the representation of a target patch from the remaining context (see Figure 2). This objective encourages the model to capture context-consistent patterns, allowing contextual deviations to serve as latent-space anomaly evidence.

To further retain statistical information that can be important for anomaly detection, we construct the input sequence from normalized patch embeddings and separate statistical tokens. Following recent patch-based time-series models (Nie et al., 2023; Wu et al., 2025), given an input window $\mathbf{x} \in \mathbb{R}^L$, we partition it into $N = L/P$ non-overlapping patches $\mathbf{p}^{(i)} \in \mathbb{R}^P$, for $i = 1, \ldots, N$, and denote the resulting patch sequence $\mathcal{X}$ as follows:

$$\mathcal{X} = \left[\mathbf{p}^{(1)}; \ldots; \mathbf{p}^{(N)}\right] \in \mathbb{R}^{N \times P}. \tag{1}$$

We then compute the mean μ and standard deviation σ of the input $\mathbf{x}$ and use them to normalize each patch. Each normalized patch is linearly projected with $\mathbf{W}_p \in \mathbb{R}^{P \times D}$, and a positional embedding $\mathbf{e}^{(i)} \in \mathbb{R}^D$ is added to obtain the patch token $\mathbf{h}^{(i)} \in \mathbb{R}^D$. In parallel, μ and σ are mapped to statistical tokens $\mathbf{h}_\mu, \mathbf{h}_\sigma \in \mathbb{R}^D$ using learnable embeddings $\mathbf{w}_\mu, \mathbf{w}_\sigma \in \mathbb{R}^D$, respectively. A Transformer

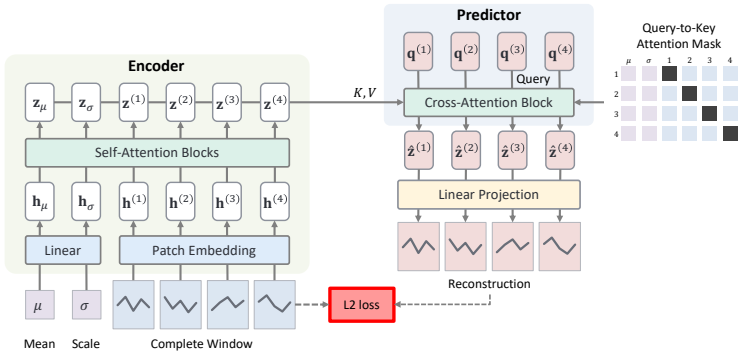


Figure 3: An overview of the **JELI** fine-tuning process with contextual reconstruction.

encoder f jointly processes the patch and statistical tokens constructed from $\mathcal{X}$ to form contextualized representations:

$$\mathbf{Z} = [\mathbf{z}_\mu; \mathbf{z}_\sigma; \mathbf{z}^{(1)}; \ldots; \mathbf{z}^{(N)}] = f(\mathcal{X}) \in \mathbb{R}^{(N+2)\times D}. \tag{2}$$

For contextual prediction, we use an online encoder f_{online} and a target encoder f_{target}. To encourage the model to infer each patch representation from its surrounding context, we randomly select a target patch $\mathbf{p}^{(k)}$ and mask it from the online input. The resulting masked patch sequence is constructed as follows:

$$\mathcal{X}^{(\backslash k)} = \left[\mathbf{p}^{(1)}; \ldots; \mathbf{p}^{(k-1)}; [\texttt{MASK}]; \mathbf{p}^{(k+1)}; \ldots; \mathbf{p}^{(N)}\right] \in \mathbb{R}^{N\times P}. \tag{3}$$

The online encoder then processes $\mathcal{X}^{(\backslash k)}$ to construct the contextualized representations:

$$\mathbf{Z}_{\text{online}} = f_{\text{online}}\left(\mathcal{X}^{(\backslash k)}\right) \in \mathbb{R}^{(N+2)\times D}. \tag{4}$$

Note that the mean $\mu^{(\backslash k)}$ and standard deviation $\sigma^{(\backslash k)}$ used for token construction are computed only from the observed patches, excluding the target patch $\mathbf{p}^{(k)}$. Accordingly, the masked position contains no patch-value information and retains only its positional embedding $\mathbf{e}^{(k)}$.

Leveraging these contextualized representations, the query-based predictor g, consisting of cross-attention followed by a feedforward network (FFN), predicts the representation of the masked target patch. We use a set of learnable patch queries $\mathbf{Q} = [\mathbf{q}^{(1)}; \ldots; \mathbf{q}^{(N)}] \in \mathbb{R}^{N\times D}$, where each $\mathbf{q}^{(i)}$ corresponds to the i-th patch position. For the target patch $\mathbf{p}^{(k)}$, the predictor performs cross-attention using the corresponding query $\mathbf{q}^{(k)}$ and the contextualized representations $\mathbf{Z}_{\text{online}}$ as follows:

$$\hat{\mathbf{z}}^{(k)} = g\left(\mathbf{q}^{(k)}, \mathbf{Z}_{\text{online}}\right) \in \mathbb{R}^{D}. \tag{5}$$

To provide the target representations for prediction, we use a target encoder f_{target}, whose parameters are maintained as an exponential moving average (EMA) of the online encoder f_{online}. Unlike the online encoder, f_{target} processes the complete patch sequence $\mathcal{X}$ without masking, using the full-window mean and standard deviation to incorporate all available context:

$$\mathbf{Z}_{\text{target}} = f_{\text{target}}\left(\mathcal{X}\right) \in \mathbb{R}^{(N+2)\times D}. \tag{6}$$

Finally, we minimize the representation prediction loss $\mathcal{L}_{\text{rep}}$ between $\hat{\mathbf{z}}^{(k)}$ and the corresponding target representation $\mathbf{z}^{(k)}_{\text{target}}$ obtained from $\mathbf{Z}_{\text{target}}$:

$$\mathcal{L}_{\text{rep}} = \left\| \hat{\mathbf{z}}^{(k)} - \text{sg}\left(\mathbf{z}^{(k)}_{\text{target}}\right) \right\|_2^2, \tag{7}$$

where $\text{sg}(\cdot)$ denotes the stop-gradient operation.

This contextual prediction objective encourages the model to learn representations that reflect each patch in relation to its surrounding context. The pretrained representations are used for latent-space anomaly scoring in Section 2.3.

2.2 Reconstruction via a Contextual Bottleneck

While contextual representations characterize anomalous deviations in the latent space, reconstruction provides a complementary, fine-grained view in the input space by measuring how well each input can be recovered from its surrounding context. Building on the contextual prediction model pretrained in Section 2.1, we therefore finetune the online encoder f_{online} and predictor g for contextual reconstruction. For notational simplicity, we omit the subscript online and denote f_{online} simply as f throughout this section. To preserve the contextual structure, we introduce a contextual bottleneck that prevents each patch query from directly accessing its own encoded representation (see Figure 3).

During finetuning, we encode the complete patch sequence as $\mathbf{Z} = f(\mathcal{X})$ and apply the pretrained predictor g to all patch queries $\mathbf{Q}$. To enforce the contextual bottleneck, we mask the direct attention from each query to its corresponding patch representation, while preserving access to the statistical and remaining patch representations. Specifically, we add an attention mask $\mathbf{M} \in \mathbb{R}^{N\times(N+2)}$ to the cross-attention logits, where the entry corresponding to each query's own patch representation is set to $-\infty$ and all other entries are set to 0. The masked cross-attention within g is computed as follows:

$$\text{Attn}(\mathbf{Q}, \mathbf{Z}; \mathbf{M}) = \text{Softmax}\left(\frac{(\mathbf{Q}\mathbf{W}_Q)(\mathbf{Z}\mathbf{W}_K)^\top}{\sqrt{D}} + \mathbf{M}\right)\mathbf{Z}\mathbf{W}_V. \tag{8}$$

Under this attention constraint, the predictor refines the patch queries as follows:

$$\hat{\mathbf{Z}} = [\hat{\mathbf{z}}^{(1)}; \ldots; \hat{\mathbf{z}}^{(N)}] = g(\mathbf{Q}, \mathbf{Z}; \mathbf{M}) \in \mathbb{R}^{N\times D}. \tag{9}$$

Leveraging these representations, we reconstruct each patch by projecting $\hat{\mathbf{z}}^{(i)}$ back to the patch space with a learnable linear projection $\mathbf{W}_{\text{rec}} \in \mathbb{R}^{P\times D}$ and restoring its original scale using μ and σ:

$$\hat{\mathbf{p}}^{(i)} = \sigma \cdot \mathbf{W}_{\text{rec}}\hat{\mathbf{z}}^{(i)} + \mu, \qquad i = 1, \ldots, N. \tag{10}$$

The reconstructed patches are then concatenated in temporal order to form $\hat{\mathbf{x}} = [\hat{\mathbf{p}}^{(1)}; \ldots; \hat{\mathbf{p}}^{(N)}] \in \mathbb{R}^L$. We jointly finetune the encoder f, predictor g, and reconstruction projection $\mathbf{W}_{\text{rec}}$ by minimizing the reconstruction loss:

$$\mathcal{L}_{\text{rec}} = \frac{1}{L}\left\|\hat{\mathbf{x}} - \mathbf{x}\right\|_2^2. \tag{11}$$

This reconstruction objective adapts the pretrained model to recover each patch from contextual information, enabling anomaly scoring in the input space through reconstruction error. We further evaluate the contributions of the design choices in Section 3.3 .

2.3 Anomaly Scoring via a Joint Distribution

With anomaly signals obtained from both the latent and input spaces, a key remaining question is how to effectively integrate them into a unified anomaly score. We aim to jointly characterize these two scores under the normal data distribution and derive a unified anomaly score from their relationship. Our scoring procedure first derives latent- and input-space anomaly scores, $s_{\text{latent}}(\mathbf{x})$ and $s_{\text{input}}(\mathbf{x})$, for each window $\mathbf{x} \in \mathbb{R}^L$, integrates them into a joint window-level score $s_{\text{joint}}(\mathbf{x})$, and finally converts the window-level scores into pointwise anomaly scores a_t by aggregating overlapping windows.

Specifically, we first derive a latent-space anomaly score $s_{\text{latent}}(\mathbf{x})$ from the representations learned in Section 2.1. Given the representation sequence $\mathbf{Z} = [\mathbf{z}_\mu; \mathbf{z}_\sigma; \mathbf{z}^{(1)}; \ldots; \mathbf{z}^{(N)}]$ obtained from the pretrained encoder, we summarize both the statistical and patch representations into a single latent representation:

$$\bar{\mathbf{z}} = \frac{1}{N+2}\left(\mathbf{z}_\mu + \mathbf{z}_\sigma + \sum_{i=1}^{N}\mathbf{z}^{(i)}\right) \in \mathbb{R}^D. \tag{12}$$

To quantify how far $\bar{\mathbf{z}}$ deviates from normal representations, we measure its distance with respect to the normal latent distribution. Since the latent dimensions may exhibit different scales and correlations, we use the Mahalanobis distance (Mahalanobis, 1936) for $s_{\text{latent}}(\mathbf{x})$ as follows:

$$s_{\text{latent}}(\mathbf{x}) = \sqrt{(\bar{\mathbf{z}} - \mathbf{m})^\top \mathbf{\Sigma}^{-1}(\bar{\mathbf{z}} - \mathbf{m})}, \tag{13}$$

where $\mathbf{m}$ and $\mathbf{\Sigma}$ are the mean and covariance of the normal reference representations, respectively.

Second, we derive an input-space anomaly score $s_{\text{input}}(\mathbf{x})$ from the reconstruction model introduced in Section 2.2. Rather than evaluating individual time points separately, we measure how well the input is reconstructed over the entire window by averaging the reconstruction errors:

$$s_{\text{input}}(\mathbf{x}) = \frac{1}{L} \|\hat{\mathbf{x}} - \mathbf{x}\|_2^2 . \tag{14}$$

Having obtained the latent- and input-space anomaly scores, we model their joint distribution using normal windows from the training period. Let S_{latent} and S_{input} denote the corresponding score variables over these normal windows. To reduce skewness, we model the log-transformed scores as $(\ln S_{\text{latent}}, \ln S_{\text{input}}) \sim \mathcal{N}(\boldsymbol{\nu}, \mathbf{\Lambda})$, where $\boldsymbol{\nu}$ and $\mathbf{\Lambda}$ are the mean and covariance estimated from normal training windows.

For a window $\mathbf{x}$ with scores $s_{\text{latent}}(\mathbf{x})$ and $s_{\text{input}}(\mathbf{x})$, the joint anomaly score is defined as follows:

$$s_{\text{joint}}(\mathbf{x}) = -\ln \Pr\left(\ln S_{\text{latent}} \geq \ln s_{\text{latent}}(\mathbf{x}),\ \ln S_{\text{input}} \geq \ln s_{\text{input}}(\mathbf{x})\right). \tag{15}$$

As shown in Figure 4, this probability corresponds to the upper-right region beyond the observed score pair. A smaller probability mass indicates that such a combination of latent- and input-space scores is less likely to occur under normal conditions, resulting in a larger anomaly score.

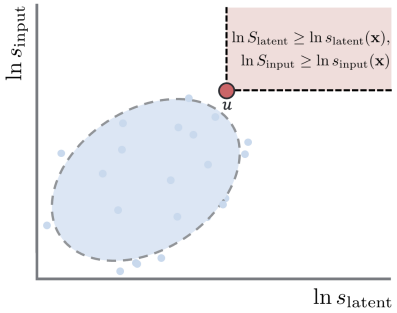


Figure 4: Joint upper-tail region for an observed score pair.

Finally, to obtain pointwise anomaly scores, we average the scores of all overlapping windows that contain each time point. Let $\mathcal{W}(t)$ denote the set of windows containing time point t, where $\mathbf{x}_w$ denotes the w-th input window. The final anomaly score is defined as follows:

$$a_t = \frac{1}{|\mathcal{W}(t)|} \sum_{w \in \mathcal{W}(t)} s_{\text{joint}}(\mathbf{x}_w). \tag{16}$$

Overall, this formulation integrates complementary latent- and input-space signals into a unified anomaly score while accounting for their dependence. We provide ablation studies on this joint scoring design in Section 3.3.

3 Experiments

3.1 Experimental Setup

Dataset. We conduct experiments on TSB-AD (Liu & Paparrizos, 2024), a recent comprehensive benchmark consisting of a broad range of time-series anomaly detection datasets. It is categorized into TSB-AD-U and TSB-AD-M, with 350 univariate and 180 multivariate evaluation time series, respectively, collected from 40 source datasets.

Evaluation protocol. For evaluation, we use three range-wise metrics, VUS-PR, VUS-ROC, and Range-F1, and three point-wise metrics, AUC-PR, AUC-ROC, and Point-F1 (Paparrizos et al., 2022; Liu & Paparrizos, 2024). Recent studies have highlighted potential biases in conventional TSAD evaluation practices, including point adjustment, which treats an entire anomalous segment as correctly detected when only a single point is identified, and method-specific threshold tuning (Kim et al., 2022). Accordingly, we avoid these practices in our evaluation. For the threshold-dependent Range-F1 and Point-F1, we uniformly report the maximum F1 score over the evaluated thresholds.

Baselines. We compare our method with recent TSAD approaches that detect anomalies mainly through either reconstruction discrepancies in the input space or deviations in learned representations. Reconstruction-based baselines include AnomalyTransformer (Xu et al., 2022), PatchTST (Nie et al., 2023), FITS (Xu et al., 2024), DADA (Shentu et al., 2025), CATCH (Wu et al., 2025), and CrossAD (Li et al., 2025), while representation-based approaches include PaAno (Park & Kang, 2026) and TimeRCD (Lan et al., 2026). We further compare with hybrid approach, MEMTO (Song et al., 2023), which incorporates additional representation-level discrepancies.

Further details on the experimental setup are provided in Appendix A.1.

Table 1: Experimental results for **univariate time-series anomaly detection** on TSB-AD-U. The best and second-best scores are highlighted in **bold** and underlined, respectively. Avg Rank denotes the average rank across all six evaluation measures.

Method	VUS-PR	VUS-ROC	Range-F1	AUC-PR	AUC-ROC	Point-F1	Avg Rank
AnomalyTransformer (2022)	0.1038	0.5407	0.1626	0.0662	0.4742	0.1095	10.00
MEMTO (2023)	0.1060	0.5614	0.1727	0.0688	0.5209	0.1178	9.00
PatchTST (2023)	0.4784	0.8382	0.4546	0.3880	0.7918	0.4107	4.33
FITS (2024)	0.3903	0.7638	0.3549	0.3142	0.6831	0.3627	7.17
DADA (2025)	0.3235	0.7704	0.3221	0.3092	0.7100	0.3504	7.67
CATCH (2025)	0.3699	0.8207	0.3645	0.3520	0.7644	0.3915	6.17
CrossAD (2025)	0.4264	0.8234	0.3956	0.4114	0.7695	0.4410	4.67
TimeRCD (2026)	0.5224	0.8711	0.5540	0.4953	0.8433	0.5274	2.50
PaAno (2026)	0.5257	0.8833	0.4903	0.4701	0.8667	0.5175	2.50
Ours	**0.6493**	**0.9202**	**0.6313**	**0.6158**	**0.9042**	**0.6398**	**1.00**

Table 2: Experimental results for **multivariate time-series anomaly detection** on TSB-AD-M. The best and second-best scores are highlighted in **bold** and underlined, respectively. Avg Rank denotes the average rank across all six evaluation measures.

Method	VUS-PR	VUS-ROC	Range-F1	AUC-PR	AUC-ROC	Point-F1	Avg Rank
AnomalyTransformer (2022)	0.1154	0.5054	0.1561	0.0744	0.4692	0.1209	9.50
MEMTO (2023)	0.1021	0.5226	0.1704	0.0686	0.5039	0.1137	9.50
PatchTST (2023)	0.3905	0.7168	0.4064	0.3373	0.6750	0.3779	4.83
FITS (2024)	0.3651	0.6995	0.3236	0.3053	0.6510	0.3708	6.50
DADA (2025)	0.3581	0.7240	0.2719	0.3509	0.6975	0.3903	4.67
CATCH (2025)	0.3264	0.7347	0.2575	0.3149	0.7045	0.3648	6.00
CrossAD (2025)	0.3193	0.7247	0.2849	0.3180	0.6944	0.3683	6.00
TimeRCD (2026)	0.3168	0.7066	0.4299	0.3400	0.7216	0.3842	4.67
PaAno (2026)	0.4188	0.7859	0.4018	0.3825	0.7598	0.4300	2.33
Ours	**0.4684**	**0.8215**	**0.4687**	**0.4421**	**0.8141**	**0.4984**	**1.00**

3.2 Time-Series Anomaly Detection Results

The experimental results on TSB-AD-U and TSB-AD-M are summarized in Tables 1 and 2, respectively. Our method consistently outperforms all baselines on both benchmarks, achieving the best performance across all six evaluation metrics and an average rank of 1.00 in both settings. Compared with PaAno, the strongest representation-based baseline in terms of VUS-PR, our method achieves relative improvements of 23.5% and 11.8% on TSB-AD-U and TSB-AD-M, respectively. Compared with PatchTST, the strongest reconstruction-based baseline under the same metric, the corresponding improvements are 35.7% and 19.9%. Notably, MEMTO, which jointly considers input- and latent-space deviations, records average ranks of 9.00 and 9.50 on TSB-AD-U and TSB-AD-M, respectively, whereas our method achieves an average rank of 1.00 on both benchmarks. This substantial gap highlights the effectiveness of explicitly learning and integrating anomaly-relevant information across the input and latent spaces. We further provide a stability analysis across five random seeds in Appendix A.2, showing consistently small variations across all evaluation metrics.

3.3 Design Ablation Studies

We analyze the key design choices of our framework from three perspectives: latent-space representations, input-space reconstruction, and joint anomaly scoring. We first examine how latent representations should be constructed and compared to capture anomalous patterns, and then study the design of input-space reconstruction for complementary anomaly evidence. Finally, we investigate how the two sources of anomaly information can be integrated into a unified score. All ablation experiments in this section are conducted on TSB-AD-U.

Latent-space representations. For the latent representations, we aim to preserve instance-specific statistical information while maintaining stable patch representations. As shown in Table 3, without instance normalization, statistical information is preserved but the representations are less stable, whereas instance normalization yields more consistent representations but removes potentially useful statistics. By retaining both benefits, our design achieves the best overall performance. We also study the distance function used to compute s_{latent} in Section 2.3. We use the Mahalanobis distance (Ma-

Table 3: Effect of instance normalization and statistical tokens on latent-space scoring.

Norm.	Statistical Tokens	VUS-PR	VUS-ROC
✗	✗	0.4881	0.8288
✓	✗	0.4997	0.8532
✓	✓	**0.5588**	**0.8917**

Table 4: Comparison of distance functions for latent-space scoring.

Distance Function	VUS-PR	VUS-ROC
Euclidean (L_2)	0.3251	0.6513
Diagonal Mahalanobis	0.3531	0.6872
Mahalanobis	**0.5588**	**0.8917**

Table 5: Effect of JEPA pretraining and diagonal masking augmentation on input-space scoring.

Pretraining	Diagonal Mask	VUS-PR	VUS-ROC
✗	✗	0.5377	0.8510
	✓	0.5805	0.8613
✓	✗	0.5427	0.8551
	✓	**0.5949**	**0.8761**

Table 6: Comparison of anomaly score combination strategies.

Scoring Method	VUS-PR	VUS-ROC
Input-space only	0.5949	0.8761
Latent-space only	0.5588	0.8917
z-score sum	0.6244	0.9161
Product	0.6149	0.8941
Independent-tail	0.6428	0.9153
Joint-tail	**0.6493**	**0.9202**

halanobis, 1936) to account for both dimension-wise variance and cross-dimensional correlations. As shown in Table 4, it outperforms diagonal Mahalanobis distance, which ignores correlations, and Euclidean distance, which ignores both variance and correlations, demonstrating the importance of considering both factors in latent-space anomaly scoring.

Input-space reconstruction. For the input-space anomaly score s_{input}, we aim to obtain reliable reconstruction errors by reconstructing each target from its surrounding context. To verify the design choices for this contextual reconstruction, we examine the effect of JEPA pretraining together with the diagonal masking augmentation described in Section 2.2. As shown in Table 5, both JEPA pretraining and diagonal masking improve anomaly detection performance based on reconstruction errors. Notably, the gain from JEPA pretraining is larger when diagonal masking is applied than when it is not. This indicates that the two design choices complement each other, and that combining JEPA pretraining with diagonal masking is effective for contextual reconstruction.

Joint anomaly scoring. To obtain an effective joint anomaly score s_{joint}, how the input- and latent-space anomaly scores are combined is critical. As shown in Table 6, our joint tail probability-based scoring achieves the best performance on both metrics, outperforming simple sum- and product-based approaches. It also outperforms independent tail probability estimation, which models the tail behavior of the two scores separately under an independence assumption. These results demonstrate the benefit of jointly modeling the tail behavior of the two anomaly scores while accounting for their correlation.

3.4 Case Study on Complementary Anomaly Scores

In this section, we visualize representative cases to analyze how our joint distribution-based scoring operates by combining input- and latent-space anomaly information. As shown in Figure 5, our joint score consistently provides clearer separation between normal and anomalous samples than either individual score. For example, for WSD in the first row, the anomaly distribution in (*a*) partially overlaps with the normal distributions under the input-space score, whereas this overlap is reduced under the latent-space score in (*b*). By combining the two signals, our joint score in (*c*) further separates anomalous samples from the normal distributions. Similar improvements in separation are also observed in the other cases. The joint distributions in (*d*) further support our scoring mechanism. As illustrated in Figure 4, anomalous samples are mostly concentrated in the upper-right region, where both anomaly scores are high. Interestingly, we also observe that the score distribution can be multi-modal, as illustrated by NAB in the third row. For this case, fitting a Gaussian mixture model (McLachlan et al., 2019) better captures the score distribution and further improves performance (see Appendix A.3). Although we use a uni-modal Gaussian model throughout our main experiments for a consistent comparison, this observation suggests that more flexible modeling of the score distribution could further improve anomaly scoring, which we leave for future work.

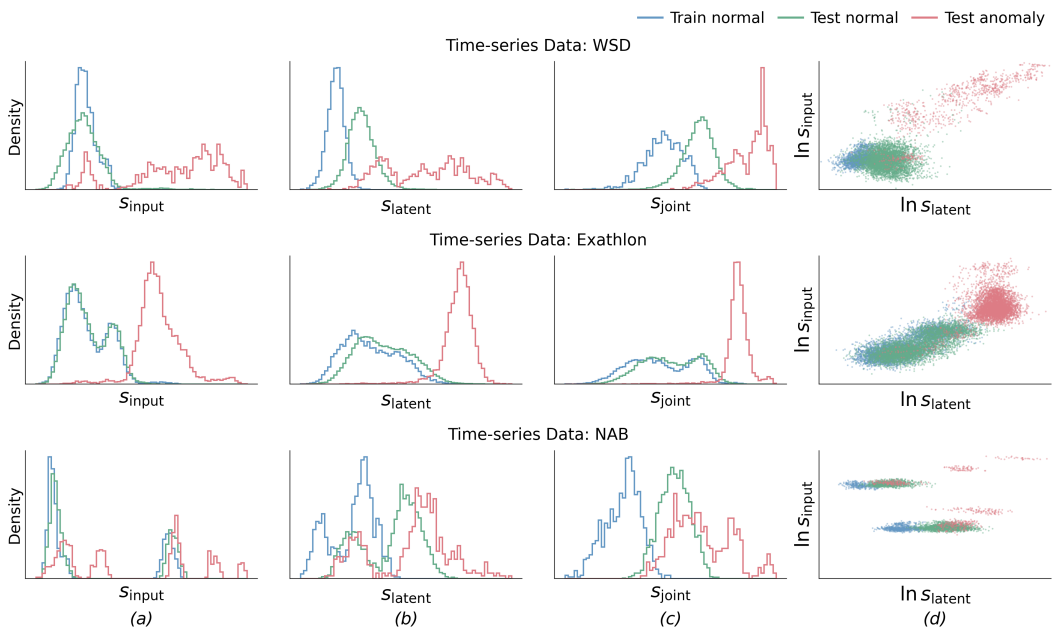


Figure 5: Case study of anomaly score distributions. (*a*) Input-space score s_{input}, (*b*) latent-space score s_{latent}, (*c*) our joint score s_{joint}, and (*d*) the joint distribution of s_{input} and s_{latent}.

4 Related Works

Recent time-series anomaly detection methods broadly model anomaly evidence in two spaces: the input space and the latent space. Input-space methods reconstruct a window toward normal behavior and derive anomaly evidence from discrepancies with the observed values. Prior works have improved this process through frequency-domain modeling (Xu et al., 2024; Wu et al., 2025), multi-scale temporal modeling (Li et al., 2025), and adaptive bottleneck modeling (Shentu et al., 2025). Even with richer dependency modeling, anomaly scores are often still computed point-wise, motivating approaches that aggregate reconstruction discrepancies via subsequence-level scoring (Wu et al., 2025). Another line of work focuses on latent-space approaches that learn representations through self-supervised learning. For example, contrastive methods encourage consistency between representations from different contextual views of the same window (Yue et al., 2022; Yang et al., 2023). Within this space, recent work has also explored retrieval-based scoring, which compares window-level representations with normal reference representations obtained from training data (Park & Kang, 2026). Although MEMTO (Song et al., 2023) leverages both spaces, its latent space is learned within an input-space reconstruction framework. In contrast, we derive latent- and input-space anomaly scores from distinct modeling objectives and explicitly model their joint distribution, allowing their statistical dependence to determine how the two sources of evidence are integrated.

5 Conclusion

In this paper, we propose **JELI**, a unified framework that models anomaly evidence from both the latent and input spaces and integrates them by modeling the joint distribution of their anomaly scores. Our method achieves state-of-the-art performance on the TSB-AD benchmark, consistently outperforming recent baselines across both univariate and multivariate settings. Our ablation studies demonstrate the contribution of the proposed design choices, while the case studies further illustrate how the framework responds to different anomaly patterns. We hope that our study encourages further exploration of integrating anomaly evidence from multiple perspectives for time-series anomaly detection.

AI USE STATEMENT

We use AI tools mainly to check consistency across baseline settings, identify possible errors in implementations, and support fair experimental comparisons. We also use them to find typographical errors and improve wording in the paper.

ETHICS STATEMENT

Our work focuses on a methodological contribution to time-series anomaly detection and does not involve ethically sensitive components or procedures. We use publicly available benchmark datasets and do not introduce additional ethical risks through our experimental setup. Therefore, we do not identify any direct ethical concerns arising from this work.

REPRODUCIBILITY STATEMENT

We provide our code together with the supplementary material, including a `README.md` file and scripts for reproducing the experiments. Detailed experimental settings, including hyperparameter configurations and tuning procedures, are provided in Appendix A.1. All experiments are implemented in PyTorch (Paszke et al., 2019) and are primarily conducted on NVIDIA RTX 4090 GPUs.

REFERENCES

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *The IEEE / CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

Siwon Kim, Kukjin Choi, Hyun-Soo Choi, Byunghan Lee, and Sungroh Yoon. Towards a rigorous evaluation of time-series anomaly detection. In *The Association for the Advancement of Artificial Intelligence (AAAI)*, 2022.

Tian Lan, Hao Duong Le, Jinbo Li, Wenjun He, Meng Wang, Chenghao Liu, and Chen Zhang. Towards foundation models for zero-shot time series anomaly detection: Leveraging synthetic data and relative context discrepancy. In *International Conference on Machine Learning (ICML)*, 2026.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2), 2004.

Beibu Li, Qichao Shentu, Yang Shu, Hui Zhang, Ming Li, Ning Jin, Bin Yang, and Chenjuan Guo. CrossAD: Time series anomaly detection with cross-scale associations and cross-window modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Qinghua Liu and John Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track (NeurIPS)*, 2024.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Prasanta Chandra Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences of India*, 2(1), 1936.

Geoffrey J McLachlan, Sharon X Lee, and Suren I Rathnayake. Finite mixture models. *Annual review of statistics and its application*, 6(1), 2019.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations (ICLR)*, 2023.

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

Jinju Park and Seokho Kang. PaAno: Patch-based representation learning for time-series anomaly detection. In *International Conference on Learning Representations (ICLR)*, 2026.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Hansheng Ren, Bixiong Xu, Yujing Wang, Chao Yi, Congrui Huang, Xiaoyu Kou, Tony Xing, Mao Yang, Jie Tong, and Qi Zhang. Time-series anomaly detection service at microsoft. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2019.

Aitor Sánchez-Ferrera, Borja Calvo, and Jose A. Lozano. A review on self-supervised learning in time series anomaly detection: Recent advances and open challenges. *ACM Computing Surveys*, 58(5), 2025.

Qichao Shentu, Beibu Li, Kai Zhao, Yang Shu, Zhongwen Rao, Lujia Pan, Bin Yang, and Chenjuan Guo. Towards a general time series anomaly detector with adaptive bottlenecks and dual adversarial decoders. In *International Conference on Learning Representations (ICLR)*, 2025.

Junho Song, Keonwoo Kim, Jeonglyul Oh, and Sungzoon Cho. MEMTO: Memory-guided transformer for multivariate time series anomaly detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Huazhang Wang, Zhaojing Luo, James W.L. Yip, Chuyang Ye, and Meihui Zhang. Ecggan: A framework for effective and interpretable electrocardiogram anomaly detection. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2023.

Qingsong Wen, Linxiao Yang, Tian Zhou, and Liang Sun. Robust time series analysis and applications: An industrial perspective. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2022.

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 *International Conference on Learning Representations (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 (ICLR)*, 2022.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling time series with 10k parameters. In *International Conference on Learning Representations (ICLR)*, 2024.

Yiyuan Yang, Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. DCdetector: Dual attention contrastive representation learning for time series anomaly detection. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2023.

Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series. In *The Association for the Advancement of Artificial Intelligence (AAAI)*, 2022.

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

A Appendix

A.1 Implementation Details

Table 7: Statistics of the TSB-AD benchmark used in our experiments.

	Tuning			Evaluation		
Benchmark	Count	Avg. Length	Anomaly Ratio	Count	Avg. Length	Anomaly Ratio
TSB-AD-U	48	47,143	3.5%	350	51,866	4.5%
TSB-AD-M	20	98,164	5.7%	180	108,826	5.0%

Dataset. We conduct experiments on TSB-AD (Liu & Paparrizos, 2024), which consists of diverse time-series anomaly detection datasets collected from 40 source datasets. The benchmark is divided into univariate (TSB-AD-U) and multivariate (TSB-AD-M) settings, each with separate tuning and evaluation splits. TSB-AD-U contains 48 tuning and 350 evaluation time series, while TSB-AD-M contains 20 tuning and 180 evaluation time series. We use the tuning split exclusively for hyperparameter selection and report the final results on the evaluation split. Each time series also provides a predefined split point, with the preceding segment used for model training. Detailed dataset statistics are summarized in Table 7.

Hyperparameters. For experiments on TSB-AD-U and TSB-AD-M, we use the following hyperparameter search space to configure our model. Following the TSB-AD evaluation protocol (Liu & Paparrizos, 2024), hyperparameters are selected using the dedicated tuning datasets, which are kept separate from the evaluation sets. We consider window lengths $L \in \{48, 96\}$, patch sizes $P \in \{12, 24\}$, model dimensions $d_{\text{model}} \in \{256, 512\}$, encoder depths in $\{2, 3\}$, and the number of attention heads $n_{\text{heads}} \in \{4, 8\}$. For optimization, we use AdamW (Loshchilov & Hutter, 2019) with learning rates in $\{1 \times 10^{-4}, 3 \times 10^{-4}\}$ and batch sizes in $\{128, 256\}$. Both the pretraining and reconstruction stages are trained for 50 epochs with a cosine decay learning-rate schedule (Loshchilov & Hutter, 2017). For covariance estimation, we use the shrinkage estimator (Ledoit & Wolf, 2004) to improve numerical stability. The reported results are averaged over five runs with different random seeds.

Evaluation metrics. Following the evaluation protocol of TSB-AD (Liu & Paparrizos, 2024), we evaluate our method using both range-wise and point-wise metrics. We follow the notation in Section 2, where a_t denotes the anomaly score at time point t, and additionally denote the ground-truth label as $y_t \in \{0, 1\}$. For a threshold τ, the predicted label is given by $\hat{y}_t(\tau) = \mathbb{I}(a_t \geq \tau)$. We do not apply point adjustment in any metric.

For range-wise evaluation, we report VUS-PR, VUS-ROC (Paparrizos et al., 2022), and Range-F1. VUS-PR and VUS-ROC extend conventional PR- and ROC-based evaluation by considering a temporal tolerance around each anomalous range, and measure the volume obtained by varying both the detection threshold and the tolerance. For Range-F1, we use the range-based precision and recall implemented in the TSB-AD evaluation toolkit (Liu & Paparrizos, 2024), which account for the extent of overlap and the number of overlapping ranges. We report the maximum harmonic mean of range precision and recall over the threshold candidates evaluated by the implementation.

For point-wise evaluation, we report AUC-PR, AUC-ROC, and Point-F1. At each threshold, precision and recall are defined as $P = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP})$ and $R = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$, while the false positive rate is $\mathrm{FPR} = \mathrm{FP}/(\mathrm{FP} + \mathrm{TN})$. AUC-PR and AUC-ROC measure the areas under the corresponding precision–recall and ROC curves as the threshold varies. Point-F1 is computed as $2PR/(P + R)$, and we again report its maximum value over all possible thresholds.

A.2 Error Bar

Table 8: Performance averaged over all evaluation time series, reported as mean ± standard deviation over five random seeds.

Benchmark	VUS-PR	VUS-ROC	Range-F1	AUC-PR	AUC-ROC	Point-F1
TSB-AD-U	0.6493 ± 0.0009	0.9202 ± 0.0005	0.6313 ± 0.0008	0.6158 ± 0.0011	0.9042 ± 0.0004	0.6398 ± 0.0005
TSB-AD-M	0.4684 ± 0.0006	0.8215 ± 0.0011	0.4687 ± 0.0012	0.4421 ± 0.0008	0.8141 ± 0.0011	0.4984 ± 0.0008

Stability across random seeds. We further evaluate the stability of JELI across five different random seeds. Table 8 reports the mean and standard deviation of each metric over the five runs, where each run is averaged over all evaluation time series. JELI shows consistently small variations across both TSB-AD-U and TSB-AD-M, demonstrating stable performance across random seeds. Notably, JELI maintains its performance advantage over the baselines even when accounting for the observed variation across random seeds.

A.3 Score Distribution Analysis and Additional Case Studies

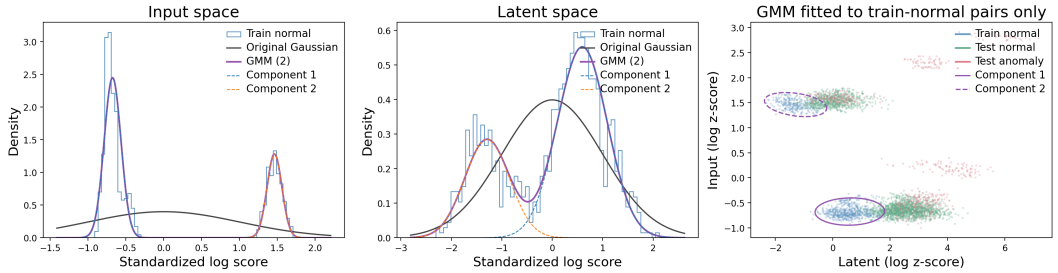


Figure 6: Multi-modal score distribution analysis on NAB. We compare the uni-modal Gaussian used in our main experiments with a two-component Gaussian mixture model in the input, latent, and joint score spaces.

Multi-modal score distribution. We further analyze the multi-modal score distribution observed in NAB, which represents an uncommon case. As shown in Figure 6, the train-normal scores form multiple modes in both the input and latent spaces. A single Gaussian tends to smooth over these modes, whereas a two-component Gaussian mixture model (GMM) (McLachlan et al., 2019) captures the individual normal modes more closely. In the joint score space, the corresponding normal samples form distinct clusters, while anomalous samples tend to lie outside these high-density regions. We further verify this by fitting a GMM, which improves VUS-PR on the NAB series shown in the Figure 6 from 0.9594 to 0.9758. Although we use a uni-modal Gaussian model throughout our main experiments for a consistent comparison, this result suggests that more flexible modeling of the score distribution can better capture heterogeneous normal patterns and may further improve anomaly scoring. We leave this extension for future work.

Additional case studies. Beyond the representative cases discussed in Section 3.4, we provide additional examples to further examine our joint anomaly scoring. As shown in Figure 7, combining the input- and latent-space scores generally improves the separation between normal and anomalous samples, indicating that our joint scoring operates as intended. Further improvements may be possible by refining how the individual anomaly scores are defined and how they are combined, including more flexible treatments of multi-modal cases as discussed above.

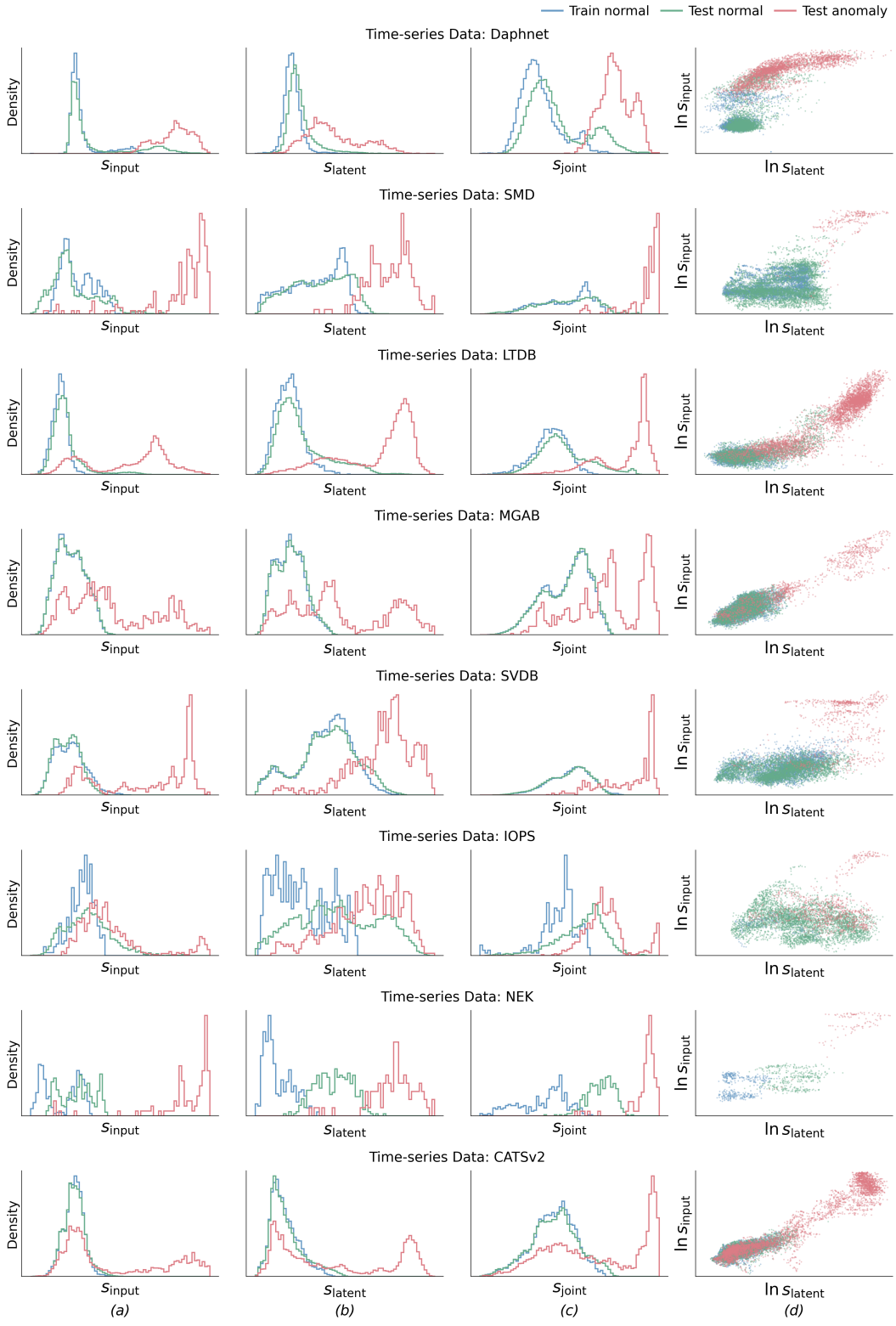


Figure 7: Additional case study of anomaly score distributions. (*a*) Input-space score s_{input}, (*b*) latent-space score s_{latent}, (*c*) our joint score s_{joint}, and (*d*) the joint distribution of s_{input} and s_{latent}.