LOCAL-GLOBAL ALIGNMENT FOR ONE-CLASS TABULAR ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

In this work, we address anomaly detection in tabular data under a one-class classification setting, where only normal records are available for training. We introduce local-global Representation Alignment for Tabular Anomaly Detection (Lo-GRAD), a method that detects anomalies by measuring how poorly a record's local feature representations align with its global representation. Our method assumes that anomalous records present a weaker alignment between their local and global representations. A Transformer Encoder produces both types of representations, which are optimized by a contrastive objective that increases the dot-product alignment between local-global pairs from the same record and decreases it for pairs from different records. Two auxiliary objectives strengthen this alignment: masked-feature modeling for the local representations and a reconstruction objective for the global representation. At inference, records with elevated local-global misalignment receive higher anomaly scores. We find that multiple scoring functions can effectively measure this misalignment, suggesting that the learned representations capture an anomaly signal that is not tied to a specific scoring function. Lo-GRAD also naturally provides feature-level anomaly attributions, since each local-global pair score is associated with a single feature. Across 25 ADBench datasets and without dataset-specific tuning, Lo-GRAD achieves the best macro-averaged AUROC and AUPRC compared with classical and state-of-the-art baselines, ranking first on 14 datasets for each metric.

1 INTRODUCTION

Anomaly detection (AD), also known as outlier detection, is a fundamental machine learning task with a variety of applications, ranging from medical diagnosis to industrial fault monitoring (Han et al., 2022). In many of these applications, data observations are represented as tabular records composed of different types of features. Unlike images or time series, tabular data lacks a spatial or sequential structure, and their features are highly diverse in type, scale, and semantics (Grinsztajn et al., 2022). Moreover, representative anomaly in these settings are often difficult to obtain, since abnormal events are rare and captured anomalies may not cover the full range of deviations occurring in real-world environments. In this work, we study the one-class classification setting, where the training set is assumed to contain only normal records (Ruff et al., 2018). Therefore, the goal of this methodology is to capture the structure of normal data and identify records that depart from them.

Self-supervised learning offers a natural way to extract such structure from normal data by defining pretext tasks whose supervision is obtained from the data itself. For tabular records, these tasks commonly rely on recovering or predicting a set of corrupted or masked features to expose dependencies among attributes (Yoon et al., 2020; Bahri et al., 2022; Arık & Pfister, 2021; Du et al., 2024; Yin et al., 2024; Thimonier et al., 2025). However, learning useful representations alone does not ensure effective AD. The self-supervised objective must learn dependencies shared by normal records, while its output must provide a meaningful anomaly score at inference (Xu et al., 2023).

In this work, we propose local-global Representation Alignment for Tabular Anomaly Detection (Lo-GRAD), a self-supervised local-global contrastive method for AD in tabular data. In this method, each record is mapped to a sequence of feature embeddings, where an additional embedding, the global embedding, is prepended to that sequence (Gorishniy et al., 2023; Devlin et al., 2019). A Transformer Encoder (Vaswani et al., 2017) maps this sequence to a local representation

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.

for each feature and a global representation for the complete record. During training, the contrastive objective treats local-global pairs of the same record as positives and pairs of different records as negatives, guiding the model to capture the dependencies that define normal records (Hjelm et al., 2019; Veličković et al., 2019; Sun et al., 2020). To preserve relevant information in both views, we introduce two auxiliary objectives. Masked-feature modeling (MFM) requires the local representations to recover masked features from the unmasked ones, while a reconstruction objective uses the global embedding alone to reconstruct the complete record. The model is trained jointly on the local-global contrastive loss and the two auxiliary objectives.

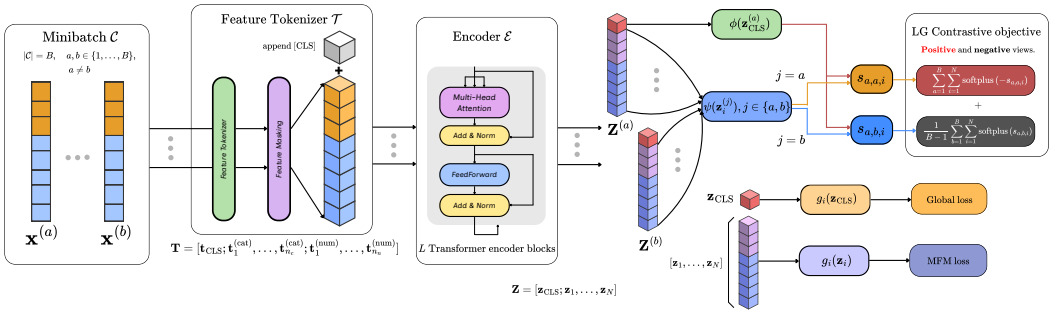

Figure 1: Training pipeline of Lo-GRAD. A minibatch $\mathcal{C}$ is sampled from the training set. Each record is tokenized into N feature embeddings ($\mathbb{R}^{N \times D}$), where a random subset of features is replaced by a mask token, and a learnable global token $\mathbf{t}_{\text{CLS}}$ is prepended. The encoder maps the resulting record sequence to a global representation $\mathbf{z}_{\text{CLS}}$ and local feature representations $[\mathbf{z}_1, \ldots, \mathbf{z}_N]$. These are enriched via auxiliary local and global training objectives. The local-global objective increases the alignment of each pair from the record a $(\mathbf{z}_{\text{CLS}}^{(a)}, \mathbf{z}_i^{(a)})$ and decreases it for mismatched pairs $(\mathbf{z}_{\text{CLS}}^{(a)}, \mathbf{z}_i^{(b)})$, where $b \neq a$.

At inference, the anomaly score is computed from the positive-pair representations of the local-global objective by measuring the alignment between the global representation of a record and its own feature representations. Normal records are expected to preserve the local-global alignment learned during training, while anomalous records are expected to break some of the learned alignment relationships. The main contributions of this work are summarized as follows:

- **Local-global modeling for tabular AD.** We introduce a contrastive objective that models the alignment between a record-level representation (global) and each of its feature representations (local) using normal records for AD, supported by two auxiliary objectives. To the best of our knowledge, this is the first tabular AD method in which the anomaly score is defined by the alignment between a global row representation and the representations of each feature.
- **Representation learning for tabular AD.** We show that the discrepancies between learned local and global representation pairs can provide a robust signal for AD across multiple scoring functions, indicating that detection relies on the local-global alignment learned during training.
- **Feature attribution.** Since the anomaly score decomposes over local-global pairs, our method provides an explicit contribution for each feature. These contributions identify the feature representations that are least compatible with the record's global representation.

2 Related Work

One-class AD methods usually capture the structure of the training data and flag records that deviate from the learned structure. In this section, we analyze existing methods by how they learn this structure from tabular data. We also distinguish whether the anomaly score is provided by the learning objective itself or computed from a separate scoring function.

Record-level relationships. One-class AD methods commonly learn a relationship between a record and a reference, such as its reconstruction, a center, or a set of transformations. For example, autoencoders detect anomalies using reconstruction error and have achieved strong results on tabular AD (Japkowicz et al., 1995; Ortega-Fernandez et al., 2024), while Deep SVDD (Ruff et al., 2018) learns a mapping that encloses normal data in a hypersphere around a fixed center. Transformation-based methods instead learn normality by predicting or contrasting transformed views. GOAD uses random affine transformations and trains an open-set classifier to associate a record with its transformation identity (Bergman & Hoshen, 2020). NeuTraL-AD replaces the transformations with learnable mappings and uses the resulting contrastive loss for anomaly scoring (Qiu et al., 2021). CSI instead uses transformations shiftings as negatives and derives an anomaly score from the resulting representation (Tack et al., 2020). SSD (Sehwag et al., 2021) separates representation learning from AD and applies a Mahalanobis-distance score to self-supervised representations after training.

Intra-record relationships. Unlike images and sequences, tabular records do not provide a natural spatial or temporal structure from which to construct self-supervised tasks. Therefore, tabular representation learning usually derives supervision from corruption methods, such as reconstructing corrupted attributes, replacing them with marginal samples, or predicting latent representations of feature subsets (Yoon et al., 2020; Bahri et al., 2022; Du et al., 2024; Thimonier et al., 2025). Methods designed for this setting adapt this mechanism so that relations induced by corruption can become detection signals. ICL constructs supervision within each record by partitioning it into a contiguous feature subset and its complement. Two mappings are learned such that the representation of a complement agrees more strongly with its corresponding subset than with other subsets from the same record (Shenkar & Wolf, 2022). MCM learns multiple masks that capture correlations among attributes, and uses an encoder-decoder network to reconstruct the original record from each masked version, while the reconstruction error across masks defines the score (Yin et al., 2024). SLAD constructs sub-vectors at different scales, learns to align their predicted distributions with scale-derived targets, and uses the alignment error as the anomaly score (Xu et al., 2023). DRL represents normal records with combinations of fixed orthogonal bases and constrains the latent space to increase the discrepancy between normal and anomalous records (Ye et al., 2025a). Disent-AD uses a self-attention module to split the features into two correlated latent subsets, each of which must reconstruct the complete record, and the reconstruction error serves as the anomaly score. (Ye et al., 2025b).

Local-global record relations. local-global objectives learn a compatibility score between a global view of an input and the local views of its attributes, so that the global view retains information shared across the local views. For example, Deep InfoMax maximizes mutual information between a global representation and local features (Hjelm et al., 2019), while InfoGraph extends this principle to graph-level representations and their substructures (Sun et al., 2020). For tabular data, PCLAD captures both intra-record regularities among transformed sub-vectors and inter-record structure by constraining the spatial centers of those sub-vectors to retain the similarities of the original records (Li et al., 2025).

Motivated by these lines of work, our method applies local-global contrastive learning to tabular AD. Each record is represented by a global view and a set of local feature views. Two auxiliary objectives make both views more informative, while the local-global objective learns their alignment. Unlike approaches that score anomalies through reconstruction error or latent-space deviations alone, our method defines anomaly scores as the discrepancy between a record's global representation and its individual feature representations.

3 METHOD

3.1 AN OVERVIEW OF THE METHODOLOGY

Let $\mathcal{X}$ denote the heterogeneous input space of tabular records and $\mathcal{P}$ the unknown distribution over $\mathcal{X}$. The training set is

$$\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^{I}, \qquad \mathbf{x} \sim \mathcal{P}, \qquad \mathbf{x} \in \mathcal{X}.$$

Each record is modeled as a heterogeneous row $\mathbf{x} = (\mathbf{x}^{\text{cat}}, \mathbf{x}^{\text{num}})$ with $N = n_c + n_n$ input features, where n_c features are categorical $\mathbf{x}^{\text{cat}} = (c_1, \ldots, c_{n_c})$, and n_n continuous features

$\mathbf{x}^{\text{num}} = (n_1, \ldots, n_{n_n})$. Our goal is to learn an anomaly score $A : \mathcal{X} \to \mathbb{R}$ that assigns low values to records consistent with $\mathcal{P}$ and high values otherwise.

For each record, Lo-GRAD produces two complementary views: a global representation that summarizes the entire row and a set of local representations corresponding to the individual features. The pairs derived from the same record are treated as matched pairs (positives), and the pairs formed from different records are treated as mismatched (negatives). The proposed model is optimized using a local-global contrastive objective and two auxiliary objectives. Figure 1 presents the complete procedure.

3.2 Local and global representations

We first map the features of a record to a common embedding space of dimension D. Let

$$\mathcal{T}(\mathbf{x}) = [\mathbf{t}_1, \ldots, \mathbf{t}_N] \in \mathbb{R}^{N \times D}$$

denote the output of a feature tokenizer $\mathcal{T}$ (Gorishniy et al., 2023), where categorical and continuous features are embedded by feature-specific mappings. During training, we sample a subset of features $M \subseteq \{1, \ldots, N\}$ and replace their embeddings with a learnable mask. A learnable token $\mathbf{t}_{\text{CLS}}$ is prepended to the resulting sequence. Given the random masking operator $\mathcal{M}$, a Transformer Encoder $\mathcal{E}$ (Vaswani et al., 2017) computes

$$\mathcal{E}([\mathbf{t}_{\text{CLS}}; \mathcal{M}(\mathcal{T}(\mathbf{x}))]) = [\mathbf{z}_{\text{CLS}}; \mathbf{z}_1, \ldots, \mathbf{z}_N], \quad \mathbf{z}_{\text{CLS}} \in \mathbb{R}^D, \quad [\mathbf{z}_1, \ldots, \mathbf{z}_N] \in \mathbb{R}^{N \times D}.$$

Although $\mathbf{z}_i$ is associated with feature i, attention allows it to incorporate information from the remaining features. Similarly, the global token $\mathbf{t}_{\text{CLS}}$ attends to the complete sequence and its output $\mathbf{z}_{\text{CLS}}$ serves the global representation.

3.3 Auxiliary representation-learning objectives

The auxiliary objectives help the model learn meaningful global and local representations used by the contrastive objective. They are used only during training and do not contribute directly to the anomaly score.

Masked-feature modeling. During training, $\mathcal{M}$ replaces a subset of features with a learnable mask embedding $\mathbf{t}_{\text{mask}} \in \mathbb{R}^D$ at a masking rate of r. Each masked feature is reconstructed from its corresponding local representation using feature-specific prediction heads $g_i^{(\text{cat})}$ and $g_i^{(\text{num})}$.

$$g_i^{(\text{cat})}(\mathbf{z}_i) = \hat{c}_i, \qquad g_i^{(\text{cat})} : \mathbb{R}^D \to \mathbb{R}^{V_i},$$
$$g_j^{(\text{num})}(\mathbf{z}_j) = \hat{n}_j, \qquad g_j^{(\text{num})} : \mathbb{R}^D \to \mathbb{R}.$$

For categorical features, the prediction head $g_i^{(\text{cat})}$ produces logits over the feature vocabulary of size V_i, while continuous heads $g_i^{(\text{num})}$ predict the original scalar value. The MFM loss is defined as:

$$\mathcal{L}_{\text{MFM}} = \frac{1}{|m_c|} \sum_{i \in m_c} \text{CE}(\hat{c}_i, c_i) + \lambda \frac{1}{|m_n|} \sum_{j \in m_n} (n_j - \hat{n}_j)^2,$$

where m_c and m_n denote the sets of masked categorical and continuous feature indices, respectively. The first term reconstructs masked categorical features using cross-entropy (CE), while the second reconstructs masked continuous features using the mean squared error between the original and reconstructed continuous feature. The term λ balances the contribution of the two reconstruction function. At inference, masking is disabled.

Global row reconstruction. A set of per-feature reconstruction heads, $g_i^{(cat)} : \mathbb{R}^D \to \mathbb{R}^{V_i}$ and $g_j^{(num)} : \mathbb{R}^D \to \mathbb{R}$, are applied to $\mathbf{z}_{\text{CLS}}$ to reconstruct the complete uncorrupted row from this representation. The loss function is defined as:

$$\mathcal{L}_{\text{CLS}} = \frac{1}{n_c} \sum_{i=1}^{n_c} \text{CE}(\hat{c}_i^{\text{CLS}}, c_i) + \lambda \frac{1}{n_n} \sum_{j=1}^{n_n} (n_j - \hat{n}_j^{\text{CLS}})^2,$$

where $\hat{\mathbf{c}}_i^{\mathrm{CLS}}$ denotes the categorical prediction for feature i, and $\hat{x}_j^{\mathrm{CLS}}$ denotes the reconstructed value of continuous feature j, both obtained from $\mathbf{z}_{\mathrm{CLS}}$.

In this way, the two auxiliary objectives provide complementary learning signals that strengthen the local and global representations on which the contrastive objective relies. Specifically, each masked feature x_i, for $i \in M$, is predicted from the unmasked context $\mathbf{x}_{\bar{M}}$, encouraging $\mathbf{z}_i$ to encode the conditional structure of $x_i \mid \mathbf{x}_{\bar{M}}$. In contrast, the reconstruction objective enriches the global representation by requiring $\mathbf{z}_{\mathrm{CLS}}$ to preserve sufficient information to reconstruct the complete record.

3.4 LOCAL-GLOBAL CONTRASTIVE LEARNING

The local-global objective is the central learning signal. Let $\mathcal{C} = \{\mathbf{x}^{(i)}\}_{i=1}^{B} \subset \mathcal{X}$ denote a comparison set, which during training is implemented as a minibatch. For each record a, $\mathcal{E}$ produces a global representation $\mathbf{z}_{\mathrm{CLS}}^{(a)}$ and N local representations $[\mathbf{z}_1^{(a)}, \ldots, \mathbf{z}_N^{(a)}]$ in a single forward pass. Two projection heads embed each view into a shared space $\phi, \psi : \mathbb{R}^D \to \mathbb{R}^p$. The alignment score between the global representation of record a and the i-th local representation of record b is defined as

$$s_{a,b,i} = \phi\left(\mathbf{z}_{\mathrm{CLS}}^{(a)}\right) \cdot \psi\left(\mathbf{z}_i^{(b)}\right), \quad a, b \in \{1, \ldots, B\}, \quad i = 1, \ldots, N.$$

Positive pairs are formed within the same record ($a = b$), while negative pairs use local tokens from other records in the comparison set ($a \neq b$). This objective encourages the global representation of each record to be closer to its own feature-token representations than to those of other records:

$$s_{a,a,i} > s_{a,b,i}, \qquad b \neq a, \qquad i = 1, \ldots, N.$$

We optimize this relationship using a Jensen-Shannon local-global objective (Sun et al., 2020; Hjelm et al., 2019). The loss is defined as:

$$\mathcal{L}_{\mathrm{LG}} = \frac{1}{BN} \underbrace{\sum_{a=1}^{B} \sum_{i=1}^{N} \mathrm{sp}(-s_{a,a,i})}_{\text{positive pairs}} + \frac{1}{B(B-1)N} \underbrace{\sum_{a=1}^{B} \sum_{\substack{b=1 \\ b \neq a}}^{B} \sum_{i=1}^{N} \mathrm{sp}(s_{a,b,i})}_{\text{negative pairs}}, \quad \mathrm{sp}(x) = \log(1+\exp(x)).$$

Minimizing $\mathcal{L}_{\mathrm{LG}}$ increases the scores of matched pairs and decreases the scores of mismatched pairs. The complete training objective is

$$\mathcal{L} = \gamma \mathcal{L}_{\mathrm{MFM}} + \beta \mathcal{L}_{\mathrm{CLS}} + \mathcal{L}_{\mathrm{LG}}, \qquad \beta, \gamma \geq 0,$$

where the two reconstruction losses shape the local and global views, and the contrastive objective encourages internal coherence between positive pairs. Coefficients γ and β control the contributions of the local and global objectives, respectively.

3.5 ANOMALY SCORING

At inference, the local-global pairs $(\mathbf{z}_{\mathrm{CLS}}, \mathbf{z}_i)$ are scored with a discrepancy function d, and the N resulting values are combined by an aggregation function ρ into an anomaly score:

$$A(\mathbf{x}) = \rho(d(\mathbf{z}_{\mathrm{CLS}}, \mathbf{z}_1), \ldots, d(\mathbf{z}_{\mathrm{CLS}}, \mathbf{z}_N)), \qquad A : \mathcal{X} \to \mathbb{R},$$

where higher values indicate greater misalignment with the relationships learned during training.

Scoring from the training signal. The training objective already measures the alignment between global and local views, and its terms can be evaluated at inference. This defines three anomaly functions: the positive and negative pair terms, and their combination $\mathcal{L}_{\mathrm{LG}}$. The positive term involves only the current record and corresponds to $d(\mathbf{z}_{\mathrm{CLS}}, \mathbf{z}_i) = -\phi(\mathbf{z}_{\mathrm{CLS}}) \cdot \psi(\mathbf{z}_i)$:

$$A(\mathbf{x}) = \rho(-\phi(\mathbf{z}_{\mathrm{CLS}}) \cdot \psi(\mathbf{z}_1), \ldots, -\phi(\mathbf{z}_{\mathrm{CLS}}) \cdot \psi(\mathbf{z}_N)).$$

This score is large when the global representation of the record fails to agree with its feature representations. The remaining two terms require a comparison set $\mathcal{C}$, because they involve pairs from other records. As these terms make the score dependent on the composition of $\mathcal{C}$, we compute A only from a record's own local-global pairs. All scoring functions considered below follow this restriction, so each score depends only on the record being evaluated. Scoring with negative pairs is discussed in Appendix D.1.

Scoring from the representations. A training signal is not the only viable choice of d, any discrepancy function between $\mathbf{z}_{\mathrm{CLS}}$ and $\mathbf{z}_i$ can define an anomaly function under the same formulation. We evaluate a family of pairwise functions: dot-product, cosine distance, root-mean-square (RMS) deviation, and the Chebyshev and Canberra distances Each function is computed in the representation space $\mathbb{R}^D$ and in the projected space $\mathbb{R}^p$.

Aggregation over features. The operator ρ combines per-feature discrepancies into a record-level score. In this work, we evaluate mean and max aggregation functions.

4 RESULTS

Datasets. We selected 25 representative tabular AD datasets from ADBench (Han et al., 2022). This selection covers both small and large datasets across domains such as healthcare, science, and finance. For the ablation studies, we use three additional ADBench datasets which are not part of the main evaluation set. Dataset-level statistics are provided in Appendix C.

Evaluation protocol and metrics. For each dataset, normal records are randomly divided into training, validation and test subsets with a $70/10/20\%$ split. While the training and validation sets contain exclusively normal data, the test set includes both normal and anomalous records. Continuous features are standardized using statistics computed from the training set, while categorical features are mapped to integer indices. We evaluate performance with two threshold-independent metrics, AUROC and AUPRC. Each experiment is repeated with five random seeds, and we report the mean and standard deviation.

Baselines. We compare Lo-GRAD with six baselines: correlation or contrastive methods, including ICL (Shenkar & Wolf, 2022) and Disent-AD (Ye et al., 2025b), decomposition methods such as DRL (Ye et al., 2025a), masking or transformation methods, including MCM (Yin et al., 2024), NeuTraL-AD (Qiu et al., 2021), and deep methods such as an autoencoder. We also included three classical AD baselines: Isolation Forest (Liu et al., 2008), OCSVM (Schölkopf et al., 1999) and ECOD (Li et al., 2023). We use the official implementations of all baselines with the hyperparameters recommended by their authors.

Implementation. The encoder $\mathcal{E}$ consists of six Transformer Encoder blocks with model dimension $D = 32$ and 8 attention heads of dimension 16, and projection heads with $p = 16$. We use a masking rate of $r = 0.5$, set $\lambda = 5$ and fix $\gamma = \beta = 0.5$. This configuration is shared across all datasets, with no dataset-specific hyperparameter tuning. The scorer used for the main results is the mean Chebyshev distance in $\mathbb{R}^D$, as it reported the strongest performance (Table 2). All model and scoring choices were selected exclusively on three ablation datasets (Yeast, Fault, and Cardio), which are disjoint from the 25 datasets used for the final evaluation.

4.1 MAIN RESULTS

Table 1 summarizes the macro-averaged results across the 25 ADBench datasets. The proposed method achieves the highest score on both metrics, with an AUROC of 0.8781 and an AUPRC of 0.8240. MCM is the strongest baseline, reaching 0.8336 AUROC and 0.7820 AUPRC. Lo-GRAD improves these results by 4.45 and 4.20 percentage points, respectively. The next-strongest baselines, DRL for AUROC and ICL for AUPRC, trail Lo-GRAD by 4.84 and 8.65 points, respectively.

At the dataset level, the proposed method obtains the highest AUROC and AUPRC on 14 of the 25 benchmarks, as shown in Tables 6 and 7. Complete, per dataset results are reported in Appendix A.

Table 1: Macro AUROC and AUPRC on the selected ADBench datasets. Values are macro averages over 25 datasets, where each dataset score is averaged over five seeds.

Metric	Lo-GRAD	ICL	MCM	NeuTraL	Disent	DRL	Autoencoder	IForest	OCSVM	ECOD
AUROC	**0.8781**	0.8116	0.8336	0.7666	0.7677	0.8297	0.7965	0.8070	0.7668	0.7581
AUPRC	**0.8240**	0.7375	0.7820	0.6801	0.6783	0.7194	0.7182	0.6719	0.6463	0.6208

4.2 Ablation Experiments

To assess the contribution of the different components of the proposed method, we compared several variants on three ADBench datasets: Yeast, Fault, and Cardio. For every variant, anomaly scores were computed in both $\mathbb{R}^D$ and $\mathbb{R}^p$, and performance was reported using average AUROC and AUPRC over five seeds.

Hyperparameters. We tuned the method hyperparameters, batch size, projection dimension, masking rate and encoder configuration through a three stage search. This process selected an encoder of six Transformer Encoder blocks with model dimension $D = 32$ and 8 attention heads with dimension 16, projection heads to $p = 16$, a masking rate of $r = 0.5$, $\lambda = 5$ and $\gamma = \beta = 0.5$ for the encoder space. For evaluations in $\mathbb{R}^p$, the following parameters were used instead: $\gamma = 1$, $B = 1024$, $p = 8$ and $r = 0.25$. Details are provided in Appendix E.1.

Anomaly scoring. We compare five discrepancy functions in the encoder and projected spaces. Results in Table 2 and Appendix E.2 show that the encoder space generally performs better for most functions, with mean Chebyshev distance achieving the strongest results. In $\mathbb{R}^p$ most scorers perform substantially worse, with the exception of the dot-product.

Table 2: Anomaly scorer ablation results for each space and aggregation function (mean or max) for Yeast, Fault, and Cardio. Scores are presented as average AUROC/AUPRC over five seeds.

	$\mathbb{R}^D$		$\mathbb{R}^p$	
Aggregation	Mean	Max	Mean	Max
`dot`	0.7323 / 0.9079	0.7199 / 0.8992	0.7063 / 0.8818	**0.7196 / 0.8957**
`cosine`	0.7462 / 0.9113	0.7210 / 0.8992	0.6433 / 0.8220	0.7132 / 0.8788
`rms`	0.7534 / 0.9148	0.7203 / 0.8982	0.5684 / 0.7605	0.6062 / 0.7934
`chebyshev`	**0.7648 / 0.9155**	0.7266 / 0.9012	0.5934 / 0.7877	0.5841 / 0.7873
`canberra`	0.7441 / 0.9123	0.7168 / 0.8990	0.6272 / 0.7958	0.6742 / 0.8481

Training objectives. We evaluated the contribution of the auxiliary objectives using dot-product in $\mathbb{R}^p$ and Chebyshev in $\mathbb{R}^D$. As shown in Table 3, the full training objective achieves the strongest performance in $\mathbb{R}^D$, although the projected dot-product does not improve correspondingly. While pretext objectives alone achieve competitive results, their best performance is obtained when paired with local-global alignment. Per dataset results are in Appendix E.3.

Table 3: Macro results for the objective ablation for Yeast, Fault, and Cardio. For each seed, we average the scores of the three datasets to obtain one macro value per seed and report the mean and standard deviation across seeds.

	`p_dot`		`cheby`		MFM-R loss		
Objective	AUROC	AUPRC	AUROC	AUPRC	Scorer	AUROC	AUPRC
LG	$0.7005^{\pm 0.0143}$	$0.8772^{\pm 0.0145}$	$0.7057^{\pm 0.0237}$	$0.8992^{\pm 0.0115}$	`mfm`	$0.6984^{\pm 0.0124}$	$0.8916^{\pm 0.0061}$
LG-MFM	$0.7105^{\pm 0.0108}$	$0.8826^{\pm 0.0141}$	$0.7279^{\pm 0.0214}$	$0.9055^{\pm 0.0094}$	`cls`	$0.7016^{\pm 0.0156}$	$0.8899^{\pm 0.0067}$
LG-R	$0.7057^{\pm 0.0202}$	$0.8719^{\pm 0.0225}$	$0.7285^{\pm 0.0191}$	$0.9013^{\pm 0.0116}$	-	-	-
LG-MFM-R	$0.7063^{\pm 0.0197}$	$0.8818^{\pm 0.0170}$	$\mathbf{0.7648}^{\pm 0.0170}$	$\mathbf{0.9155}^{\pm 0.0109}$	-	-	-
MFM-R	-	-	$0.6772^{\pm 0.0164}$	$0.8686^{\pm 0.0134}$	-	-	-

Attention. To measure the contribution of attention to the detection performance, we replaced the Transformer with a Feed-Forward Neural Network (FFNN). The Transformer outperformed the FFNN consistently on the benchmarks. Full results are reported in Appendix E.4.

Overall, the ablations show that the local-global representations trained with both auxiliary objectives provide the strongest anomaly signal in $\mathbb{R}^D$ when paired with mean Chebyshev distance. At the same time, their robust performance across multiple discrepancy functions indicates that detection is not restricted to the objective used during training.

4.3 ANOMALY SIGNAL

The anomaly score in this work is formulated as the discrepancy between a record's global representation and its local representations. We evaluate five discrepancy functions with mean and max aggregation, separately in the encoder representation space and in the projected space learned by the local-global objective. Complete results and function definitions are provided in Appendix D.

Table 4 shows a clear separation between the encoder representation space and the projected space. In the encoder space $\mathbb{R}^D$, the performance is consistently strong across all discrepancy functions, indicating that the learned representations preserve a robust anomaly signal that is not tied to a single scorer. The best results for this space are obtained with mean aggregation using Chebyshev distance. Scores computed in the projected space are substantially weaker for most functions and more inconsistent, and the stronger scorer is the dot-product derived from training. A possible explanation is that the projection heads retain only the information necessary for the contrastive objective, making space $\mathbb{R}^p$ less consistent for AD. Aggregation functions also behave differently across spaces. In $\mathbb{R}^D$, mean aggregation is generally stronger, and in the projected space, max aggregation consistently improves over mean aggregation.

Table 4: Macro AUROC and AUPRC for each scorer across ADBench datasets. Each dataset score is averaged over five seeds and values report the mean of these 25 scores $\pm$ their standard deviation.

	$\mathbb{R}^D$				$\mathbb{R}^p$			
	AUROC		AUPRC		AUROC		AUPRC	
Scorer	Mean	Max	Mean	Max	Mean	Max	Mean	Max
`dot`	$0.8437^{\pm 0.1336}$	$0.8549^{\pm 0.1229}$	$0.7725^{\pm 0.2084}$	$0.7729^{\pm 0.1957}$	$0.5934^{\pm 0.1940}$	$\mathbf{0.7269}^{\pm 0.1383}$	$0.5433^{\pm 0.2436}$	$\mathbf{0.6638}^{\pm 0.2148}$
`cosine`	$0.8758^{\pm 0.1158}$	$0.8615^{\pm 0.1214}$	$0.8167^{\pm 0.1764}$	$0.7801^{\pm 0.1886}$	$0.5030^{\pm 0.2213}$	$0.6586^{\pm 0.1891}$	$0.4838^{\pm 0.2698}$	$0.6199^{\pm 0.2399}$
`rms`	$\mathbf{0.8781}^{\pm 0.1141}$	$0.8606^{\pm 0.1215}$	$0.8225^{\pm 0.1747}$	$0.7803^{\pm 0.1876}$	$0.4439^{\pm 0.2519}$	$0.4900^{\pm 0.2531}$	$0.4203^{\pm 0.2776}$	$0.4484^{\pm 0.2786}$
`chebyshev`	$\mathbf{0.8781}^{\pm 0.1140}$	$0.8633^{\pm 0.1193}$	$\mathbf{0.8240}^{\pm 0.1734}$	$0.7856^{\pm 0.1830}$	$0.4515^{\pm 0.2386}$	$0.5081^{\pm 0.2344}$	$0.4211^{\pm 0.2778}$	$0.4542^{\pm 0.2832}$
`canberra`	$0.8547^{\pm 0.1199}$	$0.8535^{\pm 0.1193}$	$0.7904^{\pm 0.1842}$	$0.7701^{\pm 0.1929}$	$0.5714^{\pm 0.1592}$	$0.6552^{\pm 0.1545}$	$0.4679^{\pm 0.2407}$	$0.5504^{\pm 0.2431}$

4.4 FEATURE ATTRIBUTION

We evaluate whether the feature attributions of local-global pairs produced by the discrepancy functions can provide faithful explanations of the model's anomaly scores. Attribution is evaluated through synthetic localization and normalized repair of features. Synthetic localization tests whether the attribution ranking recovers features deliberately modified using donor swaps (subtle inconsistencies) or tail shifts (rare values). P@b measures the fraction of the b modified features found among the b highest-attributed features. Normalized repair tests whether the attribution identifies the sources of anomalous evidence by replacing an anomaly's features with values from a normal row. Features are replaced in descending attribution order, ascending order, or random order. A larger normalized repair area indicates that the attribution ranking is more faithful. Full details and results are included in Appendix B.

Table 5 shows that the feature attributions provide useful anomaly attributions. As expected, tail shifts are localized consistently well, while the donor swap results reflect the difficulty of detecting subtle changes. In the repair experiment, descending attribution order achieves the largest normalized repair area on every dataset. These values substantially exceed both random and ascending repair, showing that replacing the highest-attributed features removes anomalous evidence earlier, supporting the faithfulness of the attribution ranking. Overall, local-global discrepancies can offer meaningful and generally faithful indicators of which features drive the anomaly score, however, the nonlinear interactions in $\mathcal{E}$ prevent these attributions from being fully interpretable.

Table 5: Feature-attribution results using datasets Yeast, Fault, and Cardio. Synthetic localization values are ranges across corruption budgets.

	Synthetic localization			Normalized repair area		
Dataset	Donor AP	Tail AP	Tail P@b	Descending	Random	Ascending
Yeast	0.378-0.464	0.810-0.841	0.730-0.747	0.674	0.307	0.125
Fault	0.167-0.350	0.746-0.781	0.676-0.704	0.600	-0.013	-0.303
Cardio	0.291-0.445	0.783-0.814	0.689-0.727	0.821	0.405	0.071

5 DISCUSSION

An important question in self-supervised AD is whether detection performance is driven by the learned representations or by a particular scoring function used at inference, especially when that function is inherited directly from the training objective. Reconstruction-based methods exemplify this, since the reconstruction error serves both as training signal and the anomaly score. Because the detector reuses the training objective, models such as autoencoders may learn to reconstruct anomalous inputs accurately as well, which weakens the detection signal (Gong et al., 2019).

Lo-GRAD separates representation learning from AD by allowing any discrepancy function to compare each local representation with the global representation. In our experiments, the detection performance in the encoder space is very consistent across the discrepancy functions evaluated. Among these, the dot-product score derived from training is one of the weakest scorers, showing that its role during training does not translate into superior performance. This finding reinforces the observation that the function used for representation learning is not necessarily the best choice for AD (Gong et al., 2019; Sohn et al., 2021). In the projected space this consistency is lost, and only the dot-product remains reasonably effective, although it still performs substantially worse than every scorer in $\mathbb{R}^D$. A possible explanation for this behavior is that the projection heads might discard information that is useful for AD but not required for local-global alignment. However, the experiments do not establish whether this information is discarded or whether the projection changes how it can be extracted. Overall, our results indicate that representations in $\mathbb{R}^D$ produce very consistent scoring results, while the projected representations appear more closely tied to the training score.

The feature attribution experiments show that individual local-global discrepancies can represent meaningful and generally faithful indicators of which features drive the anomaly score. However, because the Transformer Encoder learns nonlinear, contextualized representations, attributions may reflect feature interactions and are not guaranteed to be faithful. Therefore, they should be interpreted as empirically supported indicators of anomalous evidence rather than as causal explanations.

6 CONCLUSION

In this work, we introduce Lo-GRAD, a self-supervised method for one-class tabular AD based on the alignment between a record's feature representations and its global representation. To learn this alignment, a Transformer Encoder is trained on normal records with a contrastive objective that maximizes the alignment between the local and global representations. At inference, this alignment is used to assign higher anomaly scores to records with weaker local-global alignment.

Across 25 ADBench datasets, without dataset-specific hyperparameter tuning, Lo-GRAD achieved the highest macro AUROC and AUPRC among the evaluated methods, and ranked first on 14 datasets by AUROC and on 14 datasets by AUPRC. In the encoder space, we show that several discrepancy functions remain effective, indicating that the model learns a useful representation for AD rather than a specific scoring function derived from the training objective. These findings support Lo-GRAD as an effective method for tabular anomaly detection.

REPRODUCIBILITY STATEMENT

To ensure the full reproducibility of the results of this work, we have documented our evaluation protocol, model architecture, and training procedures in the main manuscript.

The benchmarks, data splits, evaluation protocol, and implementation details, including hyperparameter selection, are described in Section 4 of the manuscript, which also points to the appendices for further documentation. The experiments can be reproduced on a single commodity GPU with approximately 4 GB of VRAM.

The complete source code, including data processing, model implementation and training, is provided in the supplementary material.

AI USE STATEMENT

We acknowledge the use of AI assistants in the preparation of this manuscript. Specifically, Large Language Models were utilized to assist with grammar, style refinement and code debugging. All AI-generated outputs were thoroughly reviewed, evaluated, and edited by the human authors to ensure scientific accuracy and proper academic tone. The human authors take full responsibility for the content, the integrity of the experimental results, and the originality of the scientific contributions presented in this work. No AI tools were used to generate the core ideas, the methodological design, or the empirical data contained in this paper.

ACKNOWLEDGMENTS

Omitted for anonymous review.

REFERENCES

Sercan Ö. Arık and Tomas Pfister. TabNet: Attentive interpretable tabular learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 6679–6687, 2021. doi: 10.1609/aaai.v35i8.16826. URL https://ojs.aaai.org/index.php/AAAI/article/view/16826.

Dara Bahri, Heinrich Jiang, Yi Tay, and Donald Metzler. Scarf: Self-supervised contrastive learning using random feature corruption. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=CuV_qYkmKb3.

Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=H1lK_lBtvS.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*, 2019. URL https://arxiv.org/abs/1810.04805.

Tianyu Du, Luca Melis, and Ting Wang. Remasker: Imputing tabular data with masked autoencoding. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=KI9NqjLVDT.

Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton Van Den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1705–1714, 2019. doi: 10.1109/ICCV.2019.00179.

Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko. Revisiting Deep Learning Models for Tabular Data. *arXiv preprint arXiv:2106.11959*, 2023. URL https://arxiv.org/abs/2106.11959.

Leo Grinsztajn, Edouard Oyallon, and Gael Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 507–520. Curran Associates, Inc., 2022. doi: 10.52202/068431-0037. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/0378c7692da36807bdec87ab043cdadc-Paper-Datasets_and_Benchmarks.pdf.

Songqiao Han, Xiyang Hu, Hailiang Huang, Minqi Jiang, and Yue Zhao. Adbench: Anomaly detection benchmark. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 32142–32159. Curran Associates, Inc., 2022. doi: 10.52202/068431-2329. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/cf93972b116ca5268827d575f2cc226b-Paper-Datasets_and_Benchmarks.pdf.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bklr3j0cKX.

Nathalie Japkowicz, Catherine Myers, and Mark Gluck. A novelty detection approach to classification. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*, pp. 518–523. Morgan Kaufmann, 1995. URL https://www.ijcai.org/Proceedings/95-1/Papers/068.pdf.

Yizhou Li, Yijie Wang, Hongzuo Xu, Bin Li, and Xiaohui Zhou. Deep anomaly detection with partition contrastive learning for tabular data. *Data Mining and Knowledge Discovery*, 39 (4):30, 2025. doi: 10.1007/s10618-025-01102-w. URL https://doi.org/10.1007/s10618-025-01102-w.

Zheng Li, Yue Zhao, Xiyang Hu, Nicola Botta, Cezar Ionescu, and George H. Chen. Ecod: Unsupervised outlier detection using empirical cumulative distribution functions. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12181–12193, 2023. doi: 10.1109/TKDE.2022.3159580.

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

Ines Ortega-Fernandez, Marta Sestelo, Juan C. Burguillo, and Camilo Piñón-Blanco. Network intrusion detection system for ddos attacks in ics using deep autoencoders. *Wireless Networks*, 30(6):5059–5075, aug 2024. ISSN 1572-8196. doi: 10.1007/s11276-022-03214-3. URL https://doi.org/10.1007/s11276-022-03214-3.

Chen Qiu, Timo Pfrommer, Marius Kloft, Stephan Mandt, and Maja Rudolph. Neural transformation learning for deep anomaly detection beyond images. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8703–8714. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/qiu21a.html.

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

Bernhard Schölkopf, Robert C Williamson, Alex Smola, John Shawe-Taylor, and John Platt. Support vector method for novelty detection. In S. Solla, T. Leen, and K. Müller (eds.), *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 1999. URL https://proceedings.neurips.cc/paper_files/paper/1999/file/8725fb777f25776ffa9076e44fcfd776-Paper.pdf.

Vikash Sehwag, Mung Chiang, and Prateek Mittal. {SSD}: A unified framework for self-supervised outlier detection. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=v5gjXpmR8J.

Tom Shenkar and Lior Wolf. Anomaly detection for tabular data with internal contrastive learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=_hszZbt46bT.

Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minho Jin, and Tomas Pfister. Learning and evaluating representations for deep one-class classification. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=HCSgyPUfeDj.

Fan-Yun Sun, Jordan Hoffman, Vikas Verma, and Jian Tang. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1lfF2NYvH.

Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 11839–11852. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/8965f76632d7672e7d3cf29c87ecaa0c-Paper.pdf.

Hugo Thimonier, José Lucas De Melo Costa, Fabrice Popineau, Arpad Rimmel, and Bich-Liên DOAN. T-JEPA: Augmentation-free self-supervised learning for tabular data. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=gx3LMRB15C.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rklz9iAcKQ.

Hongzuo Xu, Yijie Wang, Juhui Wei, Songlei Jian, Yizhou Li, and Ning Liu. Fascinating supervisory signals and where to find them: Deep anomaly detection with scale learning. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 38655–38673. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/xu23p.html.

Hangting Ye, He Zhao, Wei Fan, Mingyuan Zhou, Dan dan Guo, and Yi Chang. DRL: Decomposed representation learning for tabular anomaly detection. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=CJnceDksRd.

Jianan Ye, Zhaorui Tan, Yijie Hu, Xi Yang, Guangliang Cheng, and Kaizhu Huang. Disentangling tabular data towards better one-class anomaly detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(12):13061–13068, 2025b. doi: 10.1609/aaai.v39i12.33425. URL https://doi.org/10.1609/aaai.v39i12.33425.

Jiaxin Yin, Yuanyuan Qiao, Zitang Zhou, Xiangchao Wang, and Jie Yang. MCM: Masked cell modeling for anomaly detection in tabular data. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=lNZJyEDxy4.

Jinsung Yoon, Yao Zhang, James Jordon, and Mihaela van der Schaar. Vime: Extending the success of self- and semi-supervised learning to tabular domain. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 11033–11043. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/7d97667a3e056acab9aaf653807b4a03-Paper.pdf.

A ADBENCH BREAKDOWN

Tables 6 and 7 report the per-dataset results with macro averages discussed in subsection 4.1 for the 25 ADBench benchmarks. All methods share the same one-class protocol, splits, preprocessing, and every result is the mean and standard deviation over five seeds.

Lo-GRAD achieves the best AUROC and AUPRC on 14 of the 25 datasets, more than any competing method in both cases. It also obtains the best mean rank across datasets, with average ranks of 2.02 for AUROC and 1.80 for AUPRC, compared with 3.90 and 3.10 for the strongest baseline, MCM. In terms of macro-average performance, Lo-GRAD improves over MCM by 4.45 AUROC points and 4.20 AUPRC points.

Table 6: AUROC (%) by dataset and method on the 25 selected ADBench datasets. Dataset results are mean ± standard deviation over five seeds. The macro row averages the 25 dataset means and reports ± the standard error of that macro mean. Bold marks the best method in each row.

Dataset	Lo-GRAD	ICL	MCM	NeuTraL-AD	Disent-AD	DRL	Autoencoder	IForest	OCSVM	ECOD
Annthyroid	**91.60** ± 3.10	80.51 ± 1.15	80.35 ± 4.60	85.93 ± 5.27	76.71 ± 1.68	83.18 ± 4.64	85.26 ± 4.44	87.19 ± 1.33	66.30 ± 1.21	78.30 ± 0.37
BreastW	**99.50** ± 0.30	85.76 ± 2.64	98.94 ± 0.45	77.93 ± 4.44	98.38 ± 0.56	98.61 ± 0.70	96.02 ± 0.37	99.46 ± 0.31	99.43 ± 0.17	98.90 ± 0.46
Campaign	**82.21** ± 4.81	80.75 ± 0.57	75.04 ± 2.15	73.39 ± 0.13	70.76 ± 2.35	74.84 ± 2.14	74.57 ± 0.64	73.04 ± 1.32	68.89 ± 0.28	70.47 ± 0.19
Cardiotocography	**84.49** ± 3.56	58.19 ± 1.51	68.75 ± 3.71	47.24 ± 2.08	76.30 ± 0.98	74.04 ± 2.95	80.77 ± 3.85	80.27 ± 2.85	83.51 ± 0.92	75.08 ± 0.97
CIFAR10	74.19 ± 0.87	63.29 ± 1.48	**74.73** ± 0.67	68.77 ± 1.81	64.71 ± 6.43	70.17 ± 1.43	72.79 ± 0.71	70.95 ± 0.92	72.28 ± 0.43	71.00 ± 0.50
FashionMNIST	87.27 ± 0.77	88.31 ± 2.06	**90.62** ± 0.73	87.82 ± 0.54	79.99 ± 5.18	85.33 ± 1.05	84.67 ± 0.71	84.35 ± 0.38	86.49 ± 0.58	82.44 ± 0.69
Glass	90.79 ± 5.56	92.91 ± 6.46	80.63 ± 7.25	**94.50** ± 2.95	67.35 ± 6.64	91.00 ± 0.78	76.61 ± 5.61	79.84 ± 8.34	45.53 ± 2.70	69.54 ± 6.14
Hepatitis	**88.21** ± 9.08	71.03 ± 7.63	80.77 ± 10.13	57.18 ± 10.87	78.21 ± 12.54	75.11 ± 10.54	75.38 ± 7.63	74.44 ± 9.60	76.21 ± 5.74	72.19 ± 9.69
Letter	88.19 ± 2.57	**95.09** ± 0.47	85.20 ± 2.42	92.06 ± 0.90	54.48 ± 1.74	90.10 ± 1.19	52.86 ± 1.67	65.51 ± 1.97	49.21 ± 0.94	57.04 ± 1.62
MagicGamma	78.02 ± 2.76	80.38 ± 1.19	**83.04** ± 2.39	69.53 ± 1.27	67.30 ± 0.93	75.47 ± 1.76	61.53 ± 1.15	74.36 ± 1.10	70.51 ± 0.31	63.65 ± 0.29
Mammography	**91.09** ± 1.26	83.13 ± 2.11	82.38 ± 1.44	60.91 ± 8.52	79.36 ± 1.81	79.54 ± 2.91	80.28 ± 0.41	87.77 ± 0.41	89.56 ± 0.22	90.54 ± 0.10
MNIST	91.78 ± 2.46	90.65 ± 0.90	90.16 ± 0.92	87.79 ± 0.96	**92.43** ± 1.24	85.61 ± 2.78	85.06 ± 0.91	86.47 ± 1.09	88.27 ± 0.58	74.11 ± 0.79
Musk	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	98.46 ± 1.84	**100.00** ± 0.00	94.33 ± 0.28
Optdigits	96.46 ± 1.34	**98.69** ± 0.39	90.73 ± 1.88	95.80 ± 1.60	89.66 ± 2.04	65.36 ± 5.27	71.93 ± 5.17	85.37 ± 0.80	60.69 ± 0.86	60.08 ± 0.50
PageBlocks	95.12 ± 0.45	91.66 ± 1.15	**95.56** ± 1.78	90.61 ± 0.68	88.11 ± 0.93	95.38 ± 0.31	93.51 ± 0.98	92.70 ± 0.72	92.05 ± 0.32	91.19 ± 0.39
Pendigits	**99.69** ± 0.17	98.52 ± 0.33	99.38 ± 0.27	92.64 ± 9.83	79.36 ± 0.40	97.47 ± 1.70	94.71 ± 1.95	96.65 ± 0.80	94.75 ± 0.28	92.61 ± 0.35
Pima	**71.55** ± 3.07	55.67 ± 2.40	61.15 ± 1.17	52.66 ± 1.88	64.66 ± 2.72	69.68 ± 1.27	69.37 ± 2.79	71.47 ± 2.05	68.28 ± 1.92	58.54 ± 1.97
Shuttle	99.80 ± 0.03	99.83 ± 0.05	**99.93** ± 0.05	99.48 ± 0.93	99.09 ± 0.20	99.89 ± 0.05	99.64 ± 0.06	99.61 ± 0.11	98.43 ± 0.18	99.22 ± 0.03
SpamBase	82.83 ± 3.42	81.46 ± 1.32	**85.55** ± 1.62	76.88 ± 1.03	62.31 ± 4.02	83.35 ± 1.72	79.61 ± 0.18	82.15 ± 1.24	73.62 ± 1.16	64.24 ± 0.90
Stamps	**96.46** ± 2.59	78.19 ± 11.43	90.65 ± 3.23	74.32 ± 14.72	91.56 ± 2.21	90.01 ± 3.92	92.53 ± 2.45	91.30 ± 2.20	91.40 ± 1.59	88.42 ± 2.02
SVHN	**68.32** ± 2.36	61.91 ± 1.81	61.99 ± 1.18	57.28 ± 3.68	62.01 ± 5.07	59.57 ± 1.06	63.86 ± 1.01	55.90 ± 1.61	61.78 ± 0.77	54.71 ± 0.79
Thyroid	**98.96** ± 0.58	77.06 ± 6.41	95.24 ± 0.72	87.99 ± 4.22	94.20 ± 1.18	98.35 ± 0.45	98.34 ± 0.25	87.04 ± 0.15	83.86 ± 0.43	87.66 ± 0.28
Vertebral	**55.97** ± 11.46	53.94 ± 7.09	43.71 ± 9.18	55.71 ± 5.73	33.75 ± 4.29	53.40 ± 5.42	39.43 ± 9.30	40.16 ± 6.84	41.25 ± 3.98	40.49 ± 5.86
Waveform	**83.10** ± 2.45	65.58 ± 1.06	71.19 ± 4.51	71.53 ± 1.31	52.11 ± 1.36	80.25 ± 1.89	63.80 ± 1.86	53.21 ± 3.00	55.18 ± 0.81	60.77 ± 1.00
WBC	99.77 ± 0.16	96.42 ± 1.69	98.37 ± 1.52	58.65 ± 17.93	96.56 ± 2.39	98.65 ± 0.86	98.60 ± 1.65	**99.81** ± 0.09	99.49 ± 0.34	99.72 ± 0.17
Macro average	**87.81** ± 0.34	81.16 ± 0.34	83.36 ± 0.33	76.66 ± 0.56	76.77 ± 0.35	82.97 ± 0.29	79.65 ± 0.29	80.70 ± 0.29	76.68 ± 0.15	75.81 ± 0.24

Table 7: AUPRC percentage by dataset and method on the 25 selected ADBench datasets. Dataset results are mean ± standard deviation over five seeds. The macro row averages the 25 dataset means and reports ± the standard error of that macro mean. Bold marks the best method in each row.

Dataset	Lo-GRAD	ICL	MCM	NeuTraL-AD	Disent-AD	DRL	Autoencoder	IForest	OCSVM	ECOD
Annthyroid	**81.87** ± 6.26	59.22 ± 2.52	77.29 ± 9.32	68.20 ± 7.19	60.07 ± 3.91	73.94 ± 7.04	72.58 ± 5.50	75.52 ± 3.03	51.00 ± 1.66	59.14 ± 1.17
BreastW	**99.79** ± 0.15	88.77 ± 2.71	99.47 ± 0.32	85.31 ± 2.98	99.28 ± 0.31	99.31 ± 0.44	99.53 ± 0.27	99.77 ± 0.14	98.77 ± 0.08	99.56 ± 0.21
Campaign	**73.75** ± 4.54	70.78 ± 0.49	69.94 ± 1.84	64.03 ± 0.79	63.82 ± 2.60	66.06 ± 1.22	67.99 ± 0.51	64.50 ± 0.98	62.83 ± 0.44	65.44 ± 0.52
Cardiotocography	**87.10** ± 3.25	71.94 ± 0.97	78.34 ± 2.43	60.99 ± 2.05	82.88 ± 1.23	81.48 ± 2.05	86.02 ± 2.70	83.55 ± 2.43	85.88 ± 0.90	81.47 ± 1.33
CIFAR10	42.69 ± 3.33	30.89 ± 1.48	**43.89** ± 0.97	36.67 ± 1.70	35.91 ± 6.61	40.52 ± 1.71	41.56 ± 0.98	38.37 ± 1.60	41.04 ± 0.89	38.73 ± 0.76
FashionMNIST	65.41 ± 0.98	66.21 ± 4.74	**67.81** ± 3.79	64.14 ± 1.66	55.61 ± 4.38	56.93 ± 2.88	61.15 ± 3.33	53.38 ± 2.17	60.28 ± 2.91	50.50 ± 2.12
Glass	80.36 ± 8.16	84.00 ± 15.14	73.95 ± 12.10	**85.17** ± 9.04	58.19 ± 8.10	62.73 ± 1.24	58.09 ± 7.53	42.23 ± 8.10	28.81 ± 4.61	40.04 ± 4.88
Hepatitis	**94.00** ± 5.00	84.91 ± 5.68	90.02 ± 5.51	73.11 ± 7.14	89.59 ± 6.67	79.83 ± 8.47	89.92 ± 4.03	71.18 ± 9.13	72.96 ± 7.35	68.49 ± 10.83
Letter	84.55 ± 4.96	**87.09** ± 1.36	80.76 ± 6.96	80.76 ± 1.18	30.84 ± 1.18	83.11 ± 3.33	29.25 ± 1.49	35.42 ± 1.78	29.01 ± 0.77	29.85 ± 1.12
MagicGamma	92.85 ± 1.30	92.62 ± 0.46	**93.37** ± 0.97	84.26 ± 0.74	86.07 ± 0.53	91.26 ± 0.80	87.97 ± 0.52	89.90 ± 0.46	87.38 ± 0.20	83.32 ± 0.15
Mammography	**75.17** ± 3.89	53.20 ± 5.20	73.85 ± 2.88	26.01 ± 3.83	46.13 ± 3.89	34.94 ± 4.28	71.69 ± 1.11	53.62 ± 1.52	59.68 ± 1.76	68.70 ± 0.74
MNIST	**90.32** ± 3.25	78.54 ± 1.26	85.56 ± 3.57	78.52 ± 3.09	76.31 ± 1.65	74.88 ± 3.27	82.68 ± 0.94	74.03 ± 1.99	78.83 ± 0.86	51.22 ± 1.10
Musk	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	**100.00** ± 0.00	89.20 ± 13.54	**100.00** ± 0.00	72.35 ± 1.89
Optdigits	83.83 ± 5.48	**90.58** ± 3.07	58.72 ± 6.16	80.71 ± 8.15	71.70 ± 4.81	17.48 ± 3.58	32.54 ± 4.49	35.06 ± 1.13	14.89 ± 0.42	14.69 ± 0.25
PageBlocks	90.05 ± 0.96	89.16 ± 0.87	**92.62** ± 2.50	82.32 ± 2.01	80.75 ± 1.26	91.53 ± 1.01	87.22 ± 1.39	84.46 ± 1.00	84.35 ± 0.98	81.49 ± 1.05
Pendigits	**97.04** ± 1.80	85.54 ± 2.74	92.12 ± 3.19	61.87 ± 26.21	34.81 ± 0.94	79.43 ± 11.35	63.01 ± 8.96	70.99 ± 5.79	60.85 ± 2.91	58.26 ± 2.58
Pima	**85.98** ± 2.03	78.32 ± 2.27	85.49 ± 0.94	75.57 ± 2.51	81.47 ± 1.62	83.51 ± 1.32	83.84 ± 0.84	85.31 ± 1.17	83.22 ± 1.24	79.60 ± 1.00
Shuttle	99.50 ± 0.15	**99.66** ± 0.10	99.50 ± 0.24	98.66 ± 2.29	97.86 ± 0.53	99.03 ± 0.52	98.19 ± 0.20	99.20 ± 0.28	98.05 ± 0.05	97.28 ± 0.10
SpamBase	93.08 ± 1.66	92.58 ± 0.89	**94.63** ± 0.57	90.24 ± 0.80	84.65 ± 2.47	93.93 ± 0.74	92.17 ± 0.21	91.14 ± 0.45	88.91 ± 0.64	82.89 ± 0.84
Stamps	**94.58** ± 5.04	74.46 ± 10.69	89.83 ± 6.03	70.19 ± 16.10	86.71 ± 6.41	76.72 ± 9.38	87.95 ± 5.64	80.70 ± 7.67	75.64 ± 4.44	71.49 ± 6.32
SVHN	**50.11** ± 2.76	46.49 ± 1.92	47.56 ± 1.29	42.68 ± 2.30	45.61 ± 4.36	27.83 ± 0.77	47.37 ± 1.26	25.02 ± 1.02	27.64 ± 0.54	23.67 ± 0.38
Thyroid	**92.13** ± 3.77	26.70 ± 6.46	84.58 ± 2.36	37.62 ± 9.36	74.07 ± 5.13	88.22 ± 3.96	89.30 ± 1.93	82.60 ± 1.14	74.48 ± 1.35	80.23 ± 2.65
Vertebral	**66.37** ± 6.89	63.29 ± 5.91	54.44 ± 4.87	65.03 ± 5.95	49.76 ± 2.42	47.38 ± 5.76	52.77 ± 5.71	36.63 ± 3.94	36.70 ± 2.31	37.12 ± 3.69
Waveform	40.46 ± 7.76	39.27 ± 1.64	28.29 ± 6.20	**54.85** ± 1.76	14.72 ± 1.01	53.96 ± 2.65	19.60 ± 1.29	18.66 ± 2.18	16.26 ± 0.37	17.50 ± 0.55
WBC	99.12 ± 0.59	89.59 ± 6.59	92.98 ± 6.69	33.37 ± 18.91	88.87 ± 10.94	94.41 ± 3.50	93.16 ± 8.74	**99.27** ± 0.36	98.20 ± 1.08	98.87 ± 0.72
Macro average	**82.40** ± 0.37	73.75 ± 0.44	78.20 ± 0.43	68.01 ± 0.75	67.83 ± 0.39	71.94 ± 0.39	71.82 ± 0.35	67.19 ± 0.39	64.63 ± 0.21	62.08 ± 0.27

B FEATURE-ATTRIBUTION RESULTS

For an input row $\mathbf{x}$ containing N features, the encoder produces a global representation $\mathbf{z}_{\mathrm{CLS}} \in \mathbb{R}^D$ and a contextualized local representation $\mathbf{z}_i \in \mathbb{R}^D$ for every feature i. The contribution of feature i is defined as the value that a discrepancy function assigns to the pair $(\mathbf{z}_{\mathrm{CLS}}, \mathbf{z}_i)$ of record $\mathbf{x}$,

$$a_i(\mathbf{x}) = d(\mathbf{z}_{\mathrm{CLS}}, \mathbf{z}_i),$$

and the anomaly score for a record $\mathbf{x}$ is defined as the aggregation of individual scores, in this case, using mean aggregation:

$$A(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} a_i(\mathbf{x}).$$

The feature contributions therefore form a complete additive decomposition of the scalar anomaly score. A large $a_i(\mathbf{x})$ indicates strong misalignment between the contextualized representation of feature i and the global representation of the row. Because attention makes each local representation depend on the complete input, a contribution can identify either an individually unusual feature value or a value that is inconsistent with the other features.

We evaluate the faithfulness of the feature attributions produced by local-global pair explanations of anomaly scores on Yeast, Fault, and Cardio datasets. To this end, we use two evaluation protocols: synthetic localization and normalized repair.

Synthetic localization. This protocol provides feature-level ground truth that is not available from real anomaly labels. We modify a known subset of features for each benchmark, using $\{1, 2\}$ for Yeast, $\{1, 3, 6\}$ for Fault, and $\{1, 3, 5\}$ for Cardio. We considered two feature corruption strategies:

- **Donor swaps**: replace selected features with different values from normal training rows to test whether the attribution identifies plausible but contextually inconsistent values.
- **Tail shifts**: move continuous features toward extremes or categorical features toward rare alternative categories to test whether the attribution identifies individually unusual values.

Localization is evaluated using average precision (AP) over the complete feature ranking and $\mathrm{P@}b$, the fraction of the b corrupted features recovered among the b features with the highest attribution scores. Because exactly b features are modified, $\mathrm{P@}b$ equals recall at b. We also report AP lift relative to a random ranking, the score increase over the corresponding clean row, and the detection rate at the normal-training 95th-percentile threshold.

Across all evaluations, the attribution-based ranking achieves higher AP than the random baseline (AP lift > 1), as shown in Figure 2 and Table 8. Localization is substantially stronger for tail shifts, which achieve AP values of 0.746-0.841 and $\mathrm{P@}b$ values of 0.676-0.747. Donor swaps obtain lower AP (0.167-0.464) and $\mathrm{P@}b$ (0.057-0.333). The anomaly scores reflect the same gap: only 11.7-24.5% of single-feature donor swaps are detected. Among interventions that increase the anomaly score, single-feature AP reaches 0.401, 0.193, and 0.339 for donor swaps and 0.838, 0.781, and 0.827 for tail shifts on Yeast, Fault, and Cardio, respectively. These results show that feature attributions can localize pronounced anomalous changes and capture some subtle inconsistencies.

Anomaly repair tests. This method evaluates whether feature attributions are connected to the model's score on real anomalies rather than only on synthetic corruptions. Each anomaly is paired with its nearest normal training row under an equally weighted Gower-style distance. The anomaly's features are then replaced sequentially with the corresponding values in descending attribution order, ascending order as a negative control, or random order. All orders use the same anomaly and donor and reach the same fully repaired row, and their difference lies in how rapidly they reduce the score. For anomalies whose full repair reduces the score, we divide the score-reduction trajectory by the final reduction and compute the area under it as a function of the fraction of repaired features. A larger area indicates that an ordering removes the model's anomalous evidence earlier.

Figure 3 and Table 9 show that descending repair achieves the largest normalized area and the fastest early score reduction on every dataset. Its normalized areas are 0.674, 0.600, and 0.821 on Yeast, Fault, and Cardio, compared with 0.307, -0.013, and 0.405 for random repair. The same pattern

Figure 2: Average precision for corrupted features.

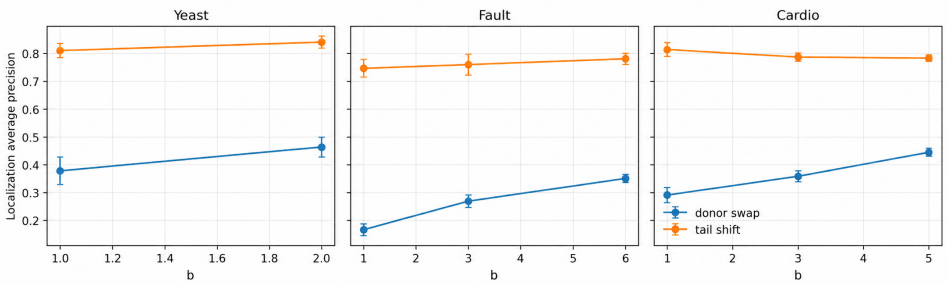

Table 8: Synthetic localization over all corruptions. Values are mean ± standard deviation over five seeds. Detection is the fraction above the normal-training 95th-percentile threshold.

Dataset	Corruption	b	AP	P@b	AP lift	Score increase / detection
Yeast	Donor swap	1	$0.378^{\pm 0.049}$	$0.163^{\pm 0.064}$	$1.688^{\pm 0.245}$	0.524 / 0.245
Yeast	Donor swap	2	$0.464^{\pm 0.035}$	$0.279^{\pm 0.047}$	$1.367^{\pm 0.150}$	1.089 / 0.336
Yeast	Tail shift	1	$0.810^{\pm 0.025}$	$0.730^{\pm 0.038}$	$3.515^{\pm 0.206}$	2.071 / 0.493
Yeast	Tail shift	2	$0.841^{\pm 0.021}$	$0.747^{\pm 0.019}$	$2.366^{\pm 0.089}$	3.790 / 0.819
Fault	Donor swap	1	$0.167^{\pm 0.021}$	$0.057^{\pm 0.017}$	$2.148^{\pm 0.268}$	0.235 / 0.117
Fault	Donor swap	3	$0.269^{\pm 0.023}$	$0.170^{\pm 0.021}$	$1.683^{\pm 0.110}$	1.006 / 0.258
Fault	Donor swap	6	$0.350^{\pm 0.015}$	$0.261^{\pm 0.012}$	$1.306^{\pm 0.038}$	2.220 / 0.468
Fault	Tail shift	1	$0.746^{\pm 0.031}$	$0.676^{\pm 0.034}$	$10.740^{\pm 0.775}$	1.647 / 0.369
Fault	Tail shift	3	$0.760^{\pm 0.038}$	$0.683^{\pm 0.044}$	$4.833^{\pm 0.346}$	4.098 / 0.800
Fault	Tail shift	6	$0.781^{\pm 0.020}$	$0.704^{\pm 0.024}$	$2.861^{\pm 0.084}$	6.915 / 0.978
Cardio	Donor swap	1	$0.291^{\pm 0.027}$	$0.145^{\pm 0.023}$	$3.238^{\pm 0.421}$	0.807 / 0.190
Cardio	Donor swap	3	$0.358^{\pm 0.020}$	$0.239^{\pm 0.014}$	$1.764^{\pm 0.143}$	2.214 / 0.365
Cardio	Donor swap	5	$0.445^{\pm 0.015}$	$0.333^{\pm 0.008}$	$1.474^{\pm 0.051}$	3.612 / 0.555
Cardio	Tail shift	1	$0.814^{\pm 0.025}$	$0.727^{\pm 0.031}$	$8.949^{\pm 0.334}$	2.489 / 0.439
Cardio	Tail shift	3	$0.787^{\pm 0.015}$	$0.689^{\pm 0.016}$	$3.945^{\pm 0.221}$	5.750 / 0.874
Cardio	Tail shift	5	$0.783^{\pm 0.012}$	$0.691^{\pm 0.016}$	$2.592^{\pm 0.147}$	7.875 / 0.990

holds among detected anomalies, for which descending repair obtains normalized areas of 0.695, 0.787, and 0.791, exceeding the random baselines of 0.558, 0.495, and 0.392. Moreover, ascending repair initially increases the anomaly score on Fault and Cardio, while descending repair consistently reduces it. These results indicate that highly attributed features contain more of the model's anomalous evidence than low-attribution or randomly selected features, supporting the faithfulness of the ranking although this does not imply that each attribution reflects an isolated causal effect of a single feature.

Figure 3: Normalized repair curves on detected anomalies with favorable full repairs. A curve that rises earlier indicates that the ordering reaches the features responsible for the anomaly score sooner.

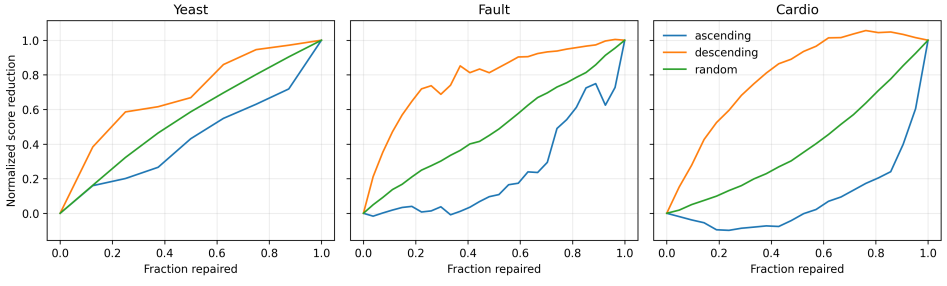

Table 9: Repair faithfulness over all labeled anomalies. Coverage is the fraction for which full donor repair reduces the score. Norm. area is the area under normalized score reduction; raw area remains in score units.

Dataset	Order	Coverage	Norm. area	Raw area	Red. @1	@10%	@20%	@50%	@100%
Yeast	Ascending	0.728	0.125	0.340	0.017	0.017	0.095	0.264	1.226
Yeast	Descending	0.728	0.674	0.803	0.409	0.409	0.603	0.852	1.226
Yeast	Random	0.728	0.307	0.568	0.121	0.121	0.258	0.537	1.226
Fault	Ascending	0.795	−0.303	0.245	−0.014	−0.008	−0.006	0.107	1.484
Fault	Descending	0.795	0.600	1.035	0.245	0.522	0.735	1.152	1.484
Fault	Random	0.795	−0.013	0.602	0.026	0.086	0.193	0.560	1.484
Cardio	Ascending	0.917	0.071	0.514	−0.033	−0.095	−0.088	0.240	3.416
Cardio	Descending	0.917	0.821	2.436	0.449	1.067	1.641	2.865	3.416
Cardio	Random	0.917	0.405	1.347	0.079	0.262	0.471	1.253	3.416

C DATASET STATISTICS

Table 10 summarizes the ADBench datasets considered in our experiments, reporting their number of records, features, number of features treated as categorical, and the distribution of normal and anomalous records. This selection covers datasets with widely varying dimensionality, record size, and anomalies, providing a diverse benchmark for evaluating AD methods.

Table 10: Statistics of the selected ADBench tabular datasets. Categorical represents the number of features tokenized as categorical.

Dataset	Records	Features	Categorical	Normal	Anomalies	Anomaly (%)
Annthyroid	7,200	6	0	6,666	534	7.42
BreastW	683	9	0	444	239	34.99
Campaign	41,188	62	52	36,548	4,640	11.27
Cardio	1,831	21	0	1,655	176	9.61
Cardiotocography	2,114	21	1	1,648	466	22.04
FashionMNIST	6,315	512	0	6,000	315	4.99
Fault	1,941	27	3	1,268	673	34.67
Glass	214	7	0	205	9	4.21
Hepatitis	80	19	13	67	13	16.25
Letter	1,600	32	0	1,500	100	6.25
MagicGamma	19,020	10	0	12,332	6,688	35.16
Mammography	11,183	6	0	10,923	260	2.32
MNIST	7,603	100	0	6,903	700	9.21
Musk	3,062	166	0	2,965	97	3.17
Optdigits	5,216	64	0	5,066	150	2.88
PageBlocks	5,393	10	0	4,883	510	9.46
Pendigits	6,870	16	0	6,714	156	2.27
Pima	768	8	0	500	268	34.90
Shuttle	49,097	9	0	45,586	3,511	7.15
SpamBase	4,207	57	0	2,528	1,679	39.91
Stamps	340	9	0	309	31	9.12
Thyroid	3,772	6	0	3,679	93	2.47
Vertebral	240	6	0	210	30	12.50
Waveform	3,443	21	0	3,343	100	2.90
WBC	223	9	0	213	10	4.48
Yeast	1,484	8	2	977	507	34.16
CIFAR10	5,263	512	0	5,000	263	5.00
SVHN	5,208	512	0	4,948	260	5.00

D ANOMALY SCORING ANALYSIS

The anomaly scorers considered in this work compare the global representation of a record with each of its local feature-token representations. We let $\mathbf{z}_{\mathrm{CLS}}$ denote the global representation of a record $\mathbf{x}$ and let $\mathbf{z}_1, \ldots, \mathbf{z}_N$ denote the contextualized representations of its N feature tokens. A scorer operates either in the encoder representation space $\mathbb{R}^D$ or in the projected space $\mathbb{R}^p$. We can set

$$\mathbf{g} = \mathbf{z}_{\mathrm{CLS}} \text{ or } \phi(\mathbf{z}_{\mathrm{CLS}}), \qquad \mathbf{l}_i = \mathbf{z}_i \text{ or } \psi(\mathbf{z}_i), \qquad \epsilon = D \text{ or } p,$$

to denote both global and local representations in $\mathbb{R}^D$ and the projected representations in $\mathbb{R}^p$, where ϕ and ψ are the global and local projection heads defined in section 3.4.

The five discrepancy functions d are described below, together with the names of the corresponding scorers.

Discrepancy	$d(\mathbf{g}, \mathbf{l}_i)$
dot-product	$-\mathbf{g} \cdot \mathbf{l}_i$
Cosine	$1 - \dfrac{\mathbf{g}\mathbf{l}_i}{\lVert\mathbf{g}\rVert_2 \lVert\mathbf{l}_i\rVert_2}$
RMS	$\sqrt{\frac{\lVert\mathbf{g}-\mathbf{l}_i\rVert_2^2}{\epsilon}}$
Chebyshev	$\max_{1\le k\le\epsilon} \lvert\mathbf{g}_k - \mathbf{l}_{i,k}\rvert$
Canberra	$\sum_{k=1}^{\epsilon} \dfrac{\lvert\mathbf{g}_k - \mathbf{l}_{i,k}\rvert}{\lvert\mathbf{g}_k\rvert + \lvert\mathbf{l}_{i,k}\rvert}$

The aggregation function ρ reduces the N per-feature discrepancies to a single record-level score. In this work, we utilized mean and max aggregation functions:

$$A_{\mathrm{mean}}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} d(\mathbf{g}, \mathbf{l}_i), \qquad A_{\max}(\mathbf{x}) = \max(d(\mathbf{g}, \mathbf{l}_1), \ldots, d(\mathbf{g}, \mathbf{l}_N)).$$

Each of the ten scorers is evaluated in both spaces, so the complete grid contains 2 × 5 × 2 = 20 scorers.

Table 11: Scorer results for Annthyroid. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	$0.8701^{\pm 0.0255}$	$0.7686^{\pm 0.0301}$	$0.4568^{\pm 0.1361}$	$0.3428^{\pm 0.0996}$
dot_max	$0.8743^{\pm 0.0236}$	$0.7715^{\pm 0.0211}$	$0.5292^{\pm 0.1247}$	$\mathbf{0.4420}^{\pm 0.1078}$
cosine_mean	$0.9069^{\pm 0.0240}$	$0.8001^{\pm 0.0520}$	$0.2445^{\pm 0.0493}$	$0.2048^{\pm 0.0273}$
cosine_max	$0.8920^{\pm 0.0233}$	$0.7839^{\pm 0.0275}$	$0.3451^{\pm 0.0681}$	$0.3027^{\pm 0.0617}$
rms_mean	$0.9095^{\pm 0.0275}$	$0.8058^{\pm 0.0571}$	$0.2611^{\pm 0.0961}$	$0.2010^{\pm 0.0274}$
rms_max	$0.8918^{\pm 0.0246}$	$0.7844^{\pm 0.0291}$	$0.2742^{\pm 0.1020}$	$0.2060^{\pm 0.0322}$
chebyshev_mean	$\mathbf{0.9160}^{\pm 0.0310}$	$\mathbf{0.8187}^{\pm 0.0626}$	$0.2668^{\pm 0.1287}$	$0.1998^{\pm 0.0342}$
chebyshev_max	$0.8961^{\pm 0.0109}$	$0.7904^{\pm 0.0278}$	$0.2928^{\pm 0.1385}$	$0.2077^{\pm 0.0403}$
canberra_mean	$0.8777^{\pm 0.0336}$	$0.7685^{\pm 0.0643}$	$0.4994^{\pm 0.1368}$	$0.2802^{\pm 0.0634}$
canberra_max	$0.8702^{\pm 0.0288}$	$0.7537^{\pm 0.0515}$	$\mathbf{0.5806}^{\pm 0.1378}$	$0.3411^{\pm 0.0832}$

Table 12: Scorer results for BreastW. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	$\mathbf{0.9953} \pm 0.0027$	$\mathbf{0.9981} \pm 0.0012$	0.1279 ± 0.1394	0.5542 ± 0.0541
dot_max	0.9862 ± 0.0060	0.9925 ± 0.0036	$\mathbf{0.6366} \pm 0.2494$	$\mathbf{0.8152} \pm 0.1240$
cosine_mean	0.9952 ± 0.0027	0.9980 ± 0.0012	0.1404 ± 0.2493	0.5628 ± 0.0932
cosine_max	0.9862 ± 0.0060	0.9925 ± 0.0036	0.4964 ± 0.3305	0.7501 ± 0.1582
rms_mean	0.9949 ± 0.0035	0.9978 ± 0.0017	0.2571 ± 0.4050	0.6338 ± 0.1939
rms_max	0.9858 ± 0.0062	0.9922 ± 0.0037	0.4407 ± 0.3500	0.7110 ± 0.1771
chebyshev_mean	0.9950 ± 0.0030	0.9979 ± 0.0015	0.2706 ± 0.3971	0.6237 ± 0.1784
chebyshev_max	0.9859 ± 0.0099	0.9918 ± 0.0088	0.4993 ± 0.3669	0.7407 ± 0.1738
canberra_mean	0.9934 ± 0.0037	0.9972 ± 0.0019	0.3754 ± 0.3223	0.6697 ± 0.1605
canberra_max	0.9825 ± 0.0079	0.9876 ± 0.0088	0.5541 ± 0.3546	0.7381 ± 0.1783

Table 13: Scorer results for Campaign. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.8124 ± 0.0498	0.7423 ± 0.0478	0.4549 ± 0.0638	0.3808 ± 0.0454
dot_max	0.8587 ± 0.0014	0.7729 ± 0.0089	$\mathbf{0.5855} \pm 0.0386$	$\mathbf{0.5039} \pm 0.0343$
cosine_mean	0.8463 ± 0.0385	0.7623 ± 0.0408	0.2708 ± 0.0416	0.2852 ± 0.0168
cosine_max	$\mathbf{0.8748} \pm 0.0047$	$\mathbf{0.7819} \pm 0.0105$	0.4017 ± 0.0475	0.3913 ± 0.0506
rms_mean	0.8241 ± 0.0550	0.7389 ± 0.0525	0.2471 ± 0.0181	0.2686 ± 0.0080
rms_max	0.8736 ± 0.0045	0.7788 ± 0.0088	0.2667 ± 0.0196	0.2741 ± 0.0074
chebyshev_mean	0.8221 ± 0.0481	0.7375 ± 0.0454	0.2696 ± 0.0234	0.2766 ± 0.0107
chebyshev_max	0.8649 ± 0.0072	0.7663 ± 0.0112	0.2991 ± 0.0401	0.2857 ± 0.0164
canberra_mean	0.7778 ± 0.0864	0.6939 ± 0.0931	0.4827 ± 0.0638	0.3987 ± 0.0505
canberra_max	0.8647 ± 0.0125	0.7761 ± 0.0081	0.5586 ± 0.1036	0.4787 ± 0.1133

Table 14: Scorer results for Cardiotocography. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	$\mathbf{0.8526} \pm 0.0285$	$\mathbf{0.8777} \pm 0.0258$	0.5351 ± 0.0946	0.6181 ± 0.0804
dot_max	0.8048 ± 0.0568	0.8531 ± 0.0424	0.6445 ± 0.1008	0.7296 ± 0.0811
cosine_mean	0.8336 ± 0.0341	0.8723 ± 0.0288	0.5367 ± 0.0667	0.6228 ± 0.0591
cosine_max	0.8028 ± 0.0576	0.8521 ± 0.0427	$\mathbf{0.6469} \pm 0.0992$	$\mathbf{0.7358} \pm 0.0835$
rms_mean	0.8484 ± 0.0301	0.8723 ± 0.0273	0.5056 ± 0.0958	0.5742 ± 0.0527
rms_max	0.8022 ± 0.0580	0.8516 ± 0.0429	0.5667 ± 0.1134	0.6423 ± 0.0858
chebyshev_mean	0.8449 ± 0.0356	0.8710 ± 0.0325	0.5312 ± 0.0642	0.5845 ± 0.0614
chebyshev_max	0.7977 ± 0.0487	0.8494 ± 0.0481	0.5873 ± 0.0407	0.6352 ± 0.0436
canberra_mean	0.8044 ± 0.0409	0.8337 ± 0.0428	0.4926 ± 0.1336	0.5708 ± 0.1040
canberra_max	0.7764 ± 0.0666	0.8223 ± 0.0611	0.5845 ± 0.1207	0.6542 ± 0.1193

Table 15: Scorer results for CIFAR10. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.6542 ± 0.0314	0.3517 ± 0.0219	0.4796 ± 0.0582	0.2214 ± 0.0298
dot_max	0.7000 ± 0.0133	0.3910 ± 0.0161	$\mathbf{0.6230} \pm 0.0500$	0.3266 ± 0.0302
cosine_mean	0.7428 ± 0.0136	0.4246 ± 0.0229	0.4164 ± 0.0794	0.1888 ± 0.0283
cosine_max	0.7173 ± 0.0096	0.3994 ± 0.0163	0.6044 ± 0.0480	$\mathbf{0.3278} \pm 0.0320$
rms_mean	$\mathbf{0.7462} \pm 0.0169$	$\mathbf{0.4345} \pm 0.0349$	0.4060 ± 0.1117	0.1905 ± 0.0406
rms_max	0.7182 ± 0.0099	0.4004 ± 0.0182	0.4271 ± 0.1236	0.2002 ± 0.0482
chebyshev_mean	0.7419 ± 0.0087	0.4269 ± 0.0333	0.4187 ± 0.1079	0.1875 ± 0.0353
chebyshev_max	0.7193 ± 0.0114	0.3967 ± 0.0076	0.4585 ± 0.1229	0.2031 ± 0.0431
canberra_mean	0.7094 ± 0.0161	0.3992 ± 0.0269	0.4889 ± 0.0681	0.2092 ± 0.0292
canberra_max	0.7143 ± 0.0151	0.3926 ± 0.0193	0.5526 ± 0.0421	0.2486 ± 0.0202

Table 16: Scorer results for FashionMNIST. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.7869 ± 0.0288	0.5231 ± 0.0477	0.4484 ± 0.0444	0.2390 ± 0.0126
dot_max	0.8615 ± 0.0073	0.6318 ± 0.0263	$\mathbf{0.7670} \pm 0.0261$	$\mathbf{0.5693} \pm 0.0386$
cosine_mean	0.8705 ± 0.0072	0.6390 ± 0.0105	0.3446 ± 0.0402	0.1905 ± 0.0083
cosine_max	0.8762 ± 0.0088	0.6466 ± 0.0289	0.7349 ± 0.0289	0.5392 ± 0.0370
rms_mean	0.8726 ± 0.0071	0.6512 ± 0.0060	0.2881 ± 0.0288	0.1414 ± 0.0069
rms_max	$\mathbf{0.8780} \pm 0.0090$	0.6525 ± 0.0298	0.3254 ± 0.0406	0.1504 ± 0.0111
chebyshev_mean	0.8727 ± 0.0077	0.6541 ± 0.0098	0.2949 ± 0.0182	0.1434 ± 0.0055
chebyshev_max	0.8773 ± 0.0099	$\mathbf{0.6631} \pm 0.0334$	0.3724 ± 0.0276	0.1670 ± 0.0132
canberra_mean	0.8140 ± 0.0131	0.5770 ± 0.0243	0.5650 ± 0.0501	0.2563 ± 0.0366
canberra_max	0.8659 ± 0.0118	0.6330 ± 0.0264	0.6877 ± 0.0359	0.3614 ± 0.0511

Table 17: Scorer results for Glass. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.8804 ± 0.0467	0.7399 ± 0.0898	0.8095 ± 0.0777	0.7427 ± 0.0500
dot_max	0.8804 ± 0.0598	0.7335 ± 0.1002	$\mathbf{0.9005} \pm 0.0380$	$\mathbf{0.8086} \pm 0.1117$
cosine_mean	0.8974 ± 0.0544	0.7733 ± 0.0948	0.7661 ± 0.0826	0.7178 ± 0.0567
cosine_max	0.8825 ± 0.0704	0.7467 ± 0.1187	0.8772 ± 0.0776	0.7959 ± 0.1162
rms_mean	0.9090 ± 0.0483	0.7911 ± 0.0839	0.7037 ± 0.1698	0.5655 ± 0.1702
rms_max	0.8847 ± 0.0691	0.7458 ± 0.1248	0.7312 ± 0.1534	0.5996 ± 0.2002
chebyshev_mean	0.9079 ± 0.0556	0.8036 ± 0.0816	0.7206 ± 0.2379	0.6099 ± 0.2581
chebyshev_max	0.8878 ± 0.0667	0.7428 ± 0.1396	0.7439 ± 0.2447	0.6236 ± 0.2459
canberra_mean	$\mathbf{0.9217} \pm 0.0515$	$\mathbf{0.8196} \pm 0.0752$	0.8360 ± 0.1012	0.6579 ± 0.1941
canberra_max	0.8878 ± 0.0658	0.7527 ± 0.1299	0.8720 ± 0.1121	0.7789 ± 0.1814

Table 18: Scorer results for Hepatitis. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.8744 ± 0.1055	0.9364 ± 0.0547	0.8718 ± 0.0920	0.9376 ± 0.0549
dot_max	0.8667 ± 0.1090	0.9356 ± 0.0534	0.8667 ± 0.0983	0.9390 ± 0.0445
cosine_mean	0.8795 ± 0.1028	0.9456 ± 0.0525	0.8718 ± 0.0395	0.9404 ± 0.0369
cosine_max	0.8667 ± 0.1134	0.9349 ± 0.0558	0.8615 ± 0.0926	0.9413 ± 0.0408
rms_mean	0.8821 ± 0.0935	0.9380 ± 0.0537	0.9026 ± 0.0729	0.9486 ± 0.0475
rms_max	0.8641 ± 0.1152	0.9348 ± 0.0571	$\mathbf{0.9256} \pm 0.0681$	$\mathbf{0.9687} \pm 0.0279$
chebyshev_mean	0.8821 ± 0.0908	0.9400 ± 0.0500	0.8538 ± 0.1120	0.9439 ± 0.0454
chebyshev_max	0.8821 ± 0.0974	$\mathbf{0.9527} \pm 0.0387$	0.8897 ± 0.0936	0.9477 ± 0.0429
canberra_mean	$\mathbf{0.8872} \pm 0.1101$	0.9519 ± 0.0496	0.9051 ± 0.0548	0.9533 ± 0.0285
canberra_max	0.8769 ± 0.1090	0.9433 ± 0.0510	0.9128 ± 0.0618	0.9572 ± 0.0326

Table 19: Scorer results for Letter. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.8079 ± 0.0555	0.6699 ± 0.0489	$\mathbf{0.6853} \pm 0.0494$	$\mathbf{0.5351} \pm 0.0472$
dot_max	0.8252 ± 0.0353	0.6718 ± 0.0387	0.6311 ± 0.0129	0.4929 ± 0.0354
cosine_mean	0.8538 ± 0.0250	0.8007 ± 0.0401	0.5165 ± 0.0311	0.3363 ± 0.0498
cosine_max	0.8387 ± 0.0312	0.6814 ± 0.0414	0.6311 ± 0.0157	0.4901 ± 0.0413
rms_mean	0.8716 ± 0.0269	0.8454 ± 0.0532	0.5350 ± 0.0346	0.2975 ± 0.0404
rms_max	0.8208 ± 0.0313	0.6854 ± 0.0433	0.5595 ± 0.0314	0.3251 ± 0.0497
chebyshev_mean	$\mathbf{0.8819} \pm 0.0257$	$\mathbf{0.8455} \pm 0.0496$	0.5150 ± 0.0649	0.2771 ± 0.0412
chebyshev_max	0.8295 ± 0.0454	0.6942 ± 0.0429	0.5475 ± 0.0654	0.3064 ± 0.0564
canberra_mean	0.8773 ± 0.0119	0.7788 ± 0.0570	0.5090 ± 0.0776	0.2667 ± 0.0596
canberra_max	0.8208 ± 0.0120	0.6666 ± 0.0329	0.5465 ± 0.0752	0.3019 ± 0.0554

Table 20: Scorer results for MagicGamma. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.8092 ± 0.0191	0.9398 ± 0.0064	0.4756 ± 0.0646	0.7544 ± 0.0460
dot_max	0.8174 ± 0.0111	$\mathbf{0.9429} \pm 0.0038$	$\mathbf{0.5423} \pm 0.0639$	$\mathbf{0.7988} \pm 0.0372$
cosine_mean	0.7970 ± 0.0222	0.9361 ± 0.0108	0.2885 ± 0.0356	0.6386 ± 0.0272
cosine_max	$\mathbf{0.8221} \pm 0.0146$	0.9416 ± 0.0073	0.3743 ± 0.0385	0.7090 ± 0.0260
rms_mean	0.7941 ± 0.0302	0.9307 ± 0.0145	0.2556 ± 0.0350	0.6069 ± 0.0217
rms_max	0.8185 ± 0.0168	0.9403 ± 0.0080	0.2628 ± 0.0364	0.6098 ± 0.0222
chebyshev_mean	0.7802 ± 0.0276	0.9285 ± 0.0130	0.2781 ± 0.0339	0.6195 ± 0.0243
chebyshev_max	0.7990 ± 0.0209	0.9356 ± 0.0099	0.2903 ± 0.0363	0.6245 ± 0.0254
canberra_mean	0.7472 ± 0.0362	0.9129 ± 0.0163	0.4719 ± 0.1144	0.7137 ± 0.0621
canberra_max	0.7913 ± 0.0277	0.9397 ± 0.0120	0.5377 ± 0.1118	0.7584 ± 0.0718

Table 21: Scorer results for Mammography. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.9029 ± 0.0087	0.7319 ± 0.0368	0.7072 ± 0.0568	0.4194 ± 0.0955
dot_max	0.9060 ± 0.0114	0.7418 ± 0.0342	0.7840 ± 0.0461	$\mathbf{0.5361} \pm 0.0574$
cosine_mean	0.9090 ± 0.0114	$\mathbf{0.7673} \pm 0.0237$	0.5132 ± 0.1207	0.2237 ± 0.0576
cosine_max	0.9081 ± 0.0110	0.7511 ± 0.0460	0.6928 ± 0.0648	0.4330 ± 0.0857
rms_mean	0.9074 ± 0.0138	0.7408 ± 0.0525	0.4428 ± 0.2131	0.2004 ± 0.1250
rms_max	0.9067 ± 0.0119	0.7468 ± 0.0473	0.5064 ± 0.2186	0.2418 ± 0.1530
chebyshev_mean	0.9109 ± 0.0126	0.7517 ± 0.0389	0.4101 ± 0.2464	0.2163 ± 0.1838
chebyshev_max	$\mathbf{0.9121} \pm 0.0142$	0.7667 ± 0.0320	0.4549 ± 0.2289	0.2301 ± 0.1865
canberra_mean	0.8620 ± 0.0377	0.6256 ± 0.0990	0.7629 ± 0.0502	0.3840 ± 0.0937
canberra_max	0.8996 ± 0.0152	0.7298 ± 0.0529	$\mathbf{0.8171} \pm 0.0108$	0.5093 ± 0.0525

Table 22: Scorer results for MNIST. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.9066 ± 0.0236	0.8936 ± 0.0353	0.6802 ± 0.0445	0.6642 ± 0.0146
dot_max	0.9231 ± 0.0094	$\mathbf{0.9159} \pm 0.0118$	$\mathbf{0.8554} \pm 0.0162$	$\mathbf{0.8652} \pm 0.0098$
cosine_mean	$\mathbf{0.9292} \pm 0.0201$	0.9144 ± 0.0283	0.6207 ± 0.0989	0.5948 ± 0.0775
cosine_max	0.9217 ± 0.0101	0.9111 ± 0.0162	0.8217 ± 0.0210	0.8367 ± 0.0158
rms_mean	0.9251 ± 0.0232	0.9085 ± 0.0312	0.6077 ± 0.1825	0.5796 ± 0.1504
rms_max	0.9180 ± 0.0114	0.9066 ± 0.0182	0.6409 ± 0.1947	0.6310 ± 0.1700
chebyshev_mean	0.9178 ± 0.0246	0.9032 ± 0.0325	0.5809 ± 0.1859	0.5493 ± 0.1471
chebyshev_max	0.9153 ± 0.0113	0.9032 ± 0.0174	0.6135 ± 0.1851	0.6177 ± 0.1732
canberra_mean	0.8973 ± 0.0304	0.8837 ± 0.0232	0.6203 ± 0.0420	0.5789 ± 0.0350
canberra_max	0.9094 ± 0.0157	0.8994 ± 0.0220	0.7475 ± 0.0506	0.7324 ± 0.0584

Table 23: Scorer results for Musk. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	$\mathbf{1.0000} \pm 0.0000$	$\mathbf{1.0000} \pm 0.0000$	0.7591 ± 0.2703	0.5743 ± 0.3652
dot_max	0.9935 ± 0.0036	0.9480 ± 0.0296	0.9930 ± 0.0092	0.9542 ± 0.0619
cosine_mean	$\mathbf{1.0000} \pm 0.0000$	$\mathbf{1.0000} \pm 0.0000$	0.6776 ± 0.2713	0.5052 ± 0.3406
cosine_max	0.9936 ± 0.0035	0.9486 ± 0.0292	$\mathbf{0.9944} \pm 0.0076$	$\mathbf{0.9618} \pm 0.0498$
rms_mean	$\mathbf{1.0000} \pm 0.0000$	$\mathbf{1.0000} \pm 0.0000$	0.3601 ± 0.1872	0.1377 ± 0.0683
rms_max	0.9936 ± 0.0035	0.9492 ± 0.0287	0.6343 ± 0.1873	0.3602 ± 0.3187
chebyshev_mean	$\mathbf{1.0000} \pm 0.0000$	$\mathbf{1.0000} \pm 0.0000$	0.3220 ± 0.1422	0.1059 ± 0.0181
chebyshev_max	0.9935 ± 0.0032	0.9459 ± 0.0315	0.6233 ± 0.1813	0.2664 ± 0.1587
canberra_mean	0.9757 ± 0.0436	0.9203 ± 0.1208	0.6818 ± 0.2126	0.4553 ± 0.3702
canberra_max	0.9945 ± 0.0031	0.9507 ± 0.0354	0.8701 ± 0.1674	0.6447 ± 0.3216

Table 24: Scorer results for Optdigits. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.7955 ± 0.0486	0.4577 ± 0.0765	0.8076 ± 0.0896	0.5715 ± 0.2008
dot_max	0.7945 ± 0.0519	0.4114 ± 0.0844	0.8573 ± 0.0956	$\mathbf{0.6645} \pm 0.1814$
cosine_mean	0.9439 ± 0.0224	0.7509 ± 0.0966	0.8109 ± 0.0700	0.6019 ± 0.1666
cosine_max	0.8508 ± 0.0400	0.4795 ± 0.0741	$\mathbf{0.8647} \pm 0.0633$	0.6226 ± 0.1566
rms_mean	$\mathbf{0.9653} \pm 0.0141$	0.8322 ± 0.0720	0.6456 ± 0.2098	0.3737 ± 0.2030
rms_max	0.8564 ± 0.0366	0.4899 ± 0.0704	0.6566 ± 0.2108	0.3786 ± 0.2041
chebyshev_mean	0.9646 ± 0.0134	$\mathbf{0.8383} \pm 0.0548$	0.6399 ± 0.1277	0.3246 ± 0.0970
chebyshev_max	0.8829 ± 0.0267	0.5585 ± 0.0645	0.6719 ± 0.1049	0.3357 ± 0.0793
canberra_mean	0.9401 ± 0.0210	0.7693 ± 0.0723	0.7120 ± 0.0371	0.4012 ± 0.0533
canberra_max	0.8224 ± 0.0572	0.4577 ± 0.0902	0.7603 ± 0.0327	0.4713 ± 0.1093

Table 25: Scorer results for PageBlocks. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.9313 ± 0.0522	0.8919 ± 0.0473	0.6022 ± 0.1257	0.6072 ± 0.1117
dot_max	0.9452 ± 0.0352	0.9078 ± 0.0272	$\mathbf{0.7150} \pm 0.0891$	$\mathbf{0.7241} \pm 0.0711$
cosine_mean	0.9553 ± 0.0030	0.9096 ± 0.0034	0.2549 ± 0.1309	0.3406 ± 0.1064
cosine_max	$\mathbf{0.9583} \pm 0.0045$	$\mathbf{0.9154} \pm 0.0050$	0.5084 ± 0.0803	0.5772 ± 0.0674
rms_mean	0.9531 ± 0.0040	0.9058 ± 0.0082	0.0780 ± 0.0534	0.2021 ± 0.0066
rms_max	0.9579 ± 0.0040	0.9151 ± 0.0045	0.0939 ± 0.0779	0.2049 ± 0.0117
chebyshev_mean	0.9512 ± 0.0045	0.9005 ± 0.0096	0.0950 ± 0.0570	0.2039 ± 0.0075
chebyshev_max	0.9543 ± 0.0040	0.9065 ± 0.0056	0.1181 ± 0.0891	0.2083 ± 0.0150
canberra_mean	0.9312 ± 0.0112	0.8788 ± 0.0182	0.5367 ± 0.1716	0.3884 ± 0.1254
canberra_max	0.9479 ± 0.0077	0.9024 ± 0.0107	0.6839 ± 0.1348	0.5560 ± 0.1322

Table 26: Scorer results for Pendigits. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.9816 ± 0.0123	0.8728 ± 0.0744	0.5325 ± 0.1132	0.1637 ± 0.0703
dot_max	0.9821 ± 0.0076	0.8494 ± 0.0382	$\mathbf{0.7660} \pm 0.1330$	$\mathbf{0.4532} \pm 0.1782$
cosine_mean	0.9958 ± 0.0024	0.9546 ± 0.0374	0.3600 ± 0.1156	0.1069 ± 0.0509
cosine_max	0.9847 ± 0.0052	0.8638 ± 0.0270	0.6517 ± 0.1665	0.3692 ± 0.1922
rms_mean	$\mathbf{0.9970} \pm 0.0021$	0.9695 ± 0.0255	0.3066 ± 0.1053	0.0755 ± 0.0146
rms_max	0.9852 ± 0.0049	0.8691 ± 0.0236	0.4134 ± 0.1530	0.0958 ± 0.0249
chebyshev_mean	0.9969 ± 0.0017	$\mathbf{0.9704} \pm 0.0180$	0.3716 ± 0.0824	0.1021 ± 0.0451
chebyshev_max	0.9854 ± 0.0041	0.8748 ± 0.0202	0.4862 ± 0.1335	0.1589 ± 0.0994
canberra_mean	0.9861 ± 0.0121	0.8757 ± 0.0914	0.4683 ± 0.1884	0.1025 ± 0.0515
canberra_max	0.9736 ± 0.0117	0.8032 ± 0.0604	0.5847 ± 0.1663	0.1408 ± 0.0645

Table 27: Scorer results for Pima. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	$\mathbf{0.7278} \pm 0.0406$	$\mathbf{0.8633} \pm 0.0265$	0.5897 ± 0.0717	0.8005 ± 0.0327
dot_max	0.7211 ± 0.0289	0.8562 ± 0.0163	$\mathbf{0.6250} \pm 0.0540$	$\mathbf{0.8174} \pm 0.0236$
cosine_mean	0.7233 ± 0.0304	0.8624 ± 0.0197	0.5367 ± 0.0751	0.7764 ± 0.0408
cosine_max	0.7172 ± 0.0258	0.8549 ± 0.0148	0.5828 ± 0.0565	0.8000 ± 0.0251
rms_mean	0.7134 ± 0.0317	0.8606 ± 0.0185	0.4278 ± 0.0964	0.6859 ± 0.0728
rms_max	0.7170 ± 0.0256	0.8547 ± 0.0147	0.4510 ± 0.0987	0.6958 ± 0.0724
chebyshev_mean	0.7155 ± 0.0307	0.8598 ± 0.0203	0.4553 ± 0.0930	0.6991 ± 0.0553
chebyshev_max	0.7149 ± 0.0257	0.8511 ± 0.0179	0.4754 ± 0.0887	0.7160 ± 0.0522
canberra_mean	0.6852 ± 0.0491	0.8510 ± 0.0293	0.5506 ± 0.0953	0.7590 ± 0.0529
canberra_max	0.7189 ± 0.0353	0.8547 ± 0.0263	0.5856 ± 0.0835	0.7787 ± 0.0370

Table 28: Scorer results for Shuttle. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.9907 ± 0.0026	0.9880 ± 0.0018	0.9241 ± 0.0942	0.9328 ± 0.0794
dot_max	0.9917 ± 0.0026	0.9876 ± 0.0033	$\mathbf{0.9804} \pm 0.0095$	$\mathbf{0.9794} \pm 0.0083$
cosine_mean	0.9973 ± 0.0003	0.9939 ± 0.0014	0.8637 ± 0.1354	0.8908 ± 0.1069
cosine_max	0.9968 ± 0.0007	0.9923 ± 0.0035	0.9571 ± 0.0452	0.9638 ± 0.0343
rms_mean	0.9979 ± 0.0003	0.9946 ± 0.0012	0.0082 ± 0.0021	0.1542 ± 0.0001
rms_max	0.9960 ± 0.0019	0.9853 ± 0.0115	0.0111 ± 0.0058	0.1543 ± 0.0002
chebyshev_mean	$\mathbf{0.9980} \pm 0.0003$	$\mathbf{0.9950} \pm 0.0015$	0.0101 ± 0.0020	0.1542 ± 0.0001
chebyshev_max	0.9968 ± 0.0006	0.9918 ± 0.0014	0.0232 ± 0.0148	0.1548 ± 0.0009
canberra_mean	0.9969 ± 0.0007	0.9941 ± 0.0012	0.6305 ± 0.2959	0.4515 ± 0.2635
canberra_max	0.9959 ± 0.0004	0.9912 ± 0.0019	0.8561 ± 0.1678	0.7478 ± 0.2846

Table 29: Scorer results for SpamBase. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.6000 ± 0.1276	0.8120 ± 0.0759	0.4296 ± 0.0973	0.7528 ± 0.0551
dot_max	0.7672 ± 0.0132	0.9005 ± 0.0105	$\mathbf{0.6702} \pm 0.0864$	$\mathbf{0.8740} \pm 0.0438$
cosine_mean	0.8196 ± 0.0110	0.9218 ± 0.0054	0.4535 ± 0.1236	0.7631 ± 0.0634
cosine_max	0.7816 ± 0.0134	0.9045 ± 0.0112	0.6575 ± 0.0849	0.8698 ± 0.0403
rms_mean	0.8279 ± 0.0334	$\mathbf{0.9313} \pm 0.0163$	0.4841 ± 0.1445	0.7557 ± 0.0777
rms_max	0.7830 ± 0.0149	0.9059 ± 0.0123	0.5223 ± 0.1593	0.7744 ± 0.0884
chebyshev_mean	$\mathbf{0.8283} \pm 0.0342$	0.9308 ± 0.0166	0.4954 ± 0.1409	0.7653 ± 0.0730
chebyshev_max	0.7794 ± 0.0121	0.9033 ± 0.0126	0.5544 ± 0.1593	0.7973 ± 0.0889
canberra_mean	0.7395 ± 0.0715	0.8941 ± 0.0319	0.5074 ± 0.1280	0.7526 ± 0.0595
canberra_max	0.7721 ± 0.0259	0.9005 ± 0.0139	0.5984 ± 0.1225	0.8203 ± 0.0566

Table 30: Scorer results for Stamps. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.9732 ± 0.0181	0.9613 ± 0.0389	0.8412 ± 0.1239	0.8449 ± 0.1044
dot_max	0.9806 ± 0.0238	0.9669 ± 0.0540	$\mathbf{0.9220} \pm 0.0475$	$\mathbf{0.8944} \pm 0.0670$
cosine_mean	0.9721 ± 0.0211	0.9620 ± 0.0384	0.7823 ± 0.1648	0.8021 ± 0.1325
cosine_max	0.9806 ± 0.0238	0.9669 ± 0.0541	0.8997 ± 0.0807	0.8790 ± 0.0827
rms_mean	0.9650 ± 0.0257	0.9447 ± 0.0525	0.7022 ± 0.0953	0.6784 ± 0.1154
rms_max	$\mathbf{0.9815} \pm 0.0226$	0.9683 ± 0.0521	0.7938 ± 0.0425	0.7209 ± 0.1138
chebyshev_mean	0.9646 ± 0.0259	0.9458 ± 0.0504	0.6576 ± 0.1057	0.6448 ± 0.1292
chebyshev_max	0.9802 ± 0.0209	$\mathbf{0.9740} \pm 0.0315$	0.7584 ± 0.1186	0.7312 ± 0.1699
canberra_mean	0.9613 ± 0.0205	0.9519 ± 0.0338	0.8518 ± 0.0612	0.8125 ± 0.0882
canberra_max	0.9806 ± 0.0239	0.9681 ± 0.0520	0.8905 ± 0.0347	0.8742 ± 0.0557

Table 31: Scorer results for SVHN. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.6594 ± 0.0290	0.4802 ± 0.0253	0.5796 ± 0.0063	0.4324 ± 0.0291
dot_max	0.6344 ± 0.0072	0.4576 ± 0.0138	$\mathbf{0.6713} \pm 0.0186$	$\mathbf{0.5186} \pm 0.0273$
cosine_mean	0.6665 ± 0.0270	0.4880 ± 0.0376	0.5678 ± 0.0362	0.4118 ± 0.0186
cosine_max	0.6314 ± 0.0069	0.4585 ± 0.0123	0.6645 ± 0.0185	0.5129 ± 0.0197
rms_mean	0.6771 ± 0.0286	0.4915 ± 0.0310	0.5729 ± 0.0161	0.3925 ± 0.0281
rms_max	0.6347 ± 0.0106	0.4602 ± 0.0087	0.6004 ± 0.0241	0.4236 ± 0.0185
chebyshev_mean	$\mathbf{0.6832} \pm 0.0236$	$\mathbf{0.5011} \pm 0.0276$	0.5977 ± 0.0584	0.4227 ± 0.0592
chebyshev_max	0.6410 ± 0.0192	0.4521 ± 0.0172	0.6234 ± 0.0423	0.4432 ± 0.0314
canberra_mean	0.6549 ± 0.0169	0.4783 ± 0.0404	0.5608 ± 0.0136	0.3933 ± 0.0137
canberra_max	0.6351 ± 0.0084	0.4570 ± 0.0200	0.5973 ± 0.0170	0.4313 ± 0.0178

Table 32: Scorer results for Thyroid. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.9884 ± 0.0064	0.9037 ± 0.0594	0.5830 ± 0.1792	0.2845 ± 0.2259
dot_max	0.9824 ± 0.0090	0.8743 ± 0.0649	$\mathbf{0.7177} \pm 0.1058$	$\mathbf{0.4413} \pm 0.1970$
cosine_mean	0.9879 ± 0.0056	0.9053 ± 0.0485	0.2481 ± 0.2029	0.1475 ± 0.1611
cosine_max	0.9811 ± 0.0084	0.8715 ± 0.0628	0.4724 ± 0.2228	0.3187 ± 0.2273
rms_mean	$\mathbf{0.9901} \pm 0.0038$	$\mathbf{0.9235} \pm 0.0266$	0.0382 ± 0.0267	0.0598 ± 0.0007
rms_max	0.9809 ± 0.0083	0.8719 ± 0.0612	0.0614 ± 0.0446	0.0607 ± 0.0016
chebyshev_mean	0.9896 ± 0.0058	0.9213 ± 0.0377	0.0998 ± 0.1616	0.0645 ± 0.0114
chebyshev_max	0.9786 ± 0.0143	0.8363 ± 0.1032	0.1562 ± 0.2082	0.0704 ± 0.0212
canberra_mean	0.9880 ± 0.0047	0.9044 ± 0.0532	0.3852 ± 0.3058	0.1242 ± 0.1007
canberra_max	0.9804 ± 0.0066	0.8446 ± 0.0841	0.5264 ± 0.3131	0.2410 ± 0.2933

Table 33: Scorer results for Vertebral. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.5200 ± 0.1121	0.6202 ± 0.0625	0.7489 ± 0.0276	0.8034 ± 0.0268
dot_max	0.5111 ± 0.1031	0.6122 ± 0.0570	0.7181 ± 0.0512	0.7624 ± 0.0604
cosine_mean	0.5429 ± 0.1183	0.6497 ± 0.0666	$\mathbf{0.7962} \pm 0.0469$	$\mathbf{0.8516} \pm 0.0354$
cosine_max	0.5203 ± 0.1050	0.6149 ± 0.0548	0.7654 ± 0.0437	0.8161 ± 0.0486
rms_mean	0.5590 ± 0.1170	0.6702 ± 0.0786	0.7625 ± 0.0189	0.7935 ± 0.0496
rms_max	0.5190 ± 0.1044	0.6115 ± 0.0536	0.7727 ± 0.0355	0.7887 ± 0.0436
chebyshev_mean	0.5597 ± 0.1146	0.6637 ± 0.0689	0.7619 ± 0.0533	0.7893 ± 0.0621
chebyshev_max	0.5390 ± 0.1061	0.6164 ± 0.0663	0.7616 ± 0.0542	0.7961 ± 0.0803
canberra_mean	$\mathbf{0.5952} \pm 0.0874$	$\mathbf{0.6905} \pm 0.0547$	0.6940 ± 0.0856	0.7492 ± 0.0857
canberra_max	0.5517 ± 0.0917	0.6437 ± 0.0577	0.7038 ± 0.0858	0.7634 ± 0.0719

Table 34: Scorer results for Waveform. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.7746 ± 0.0570	0.3043 ± 0.0638	0.4728 ± 0.1179	0.1575 ± 0.0464
dot_max	0.7927 ± 0.0597	0.3519 ± 0.0913	$\mathbf{0.5110} \pm 0.1193$	$\mathbf{0.1974} \pm 0.0551$
cosine_mean	0.8236 ± 0.0267	0.4001 ± 0.0792	0.3296 ± 0.0884	0.1105 ± 0.0199
cosine_max	0.7813 ± 0.0305	0.3634 ± 0.0771	0.3815 ± 0.0703	0.1619 ± 0.0261
rms_mean	0.8238 ± 0.0239	0.3951 ± 0.0769	0.3240 ± 0.1484	0.1057 ± 0.0328
rms_max	0.7768 ± 0.0298	0.3611 ± 0.0746	0.3332 ± 0.1511	0.1031 ± 0.0312
chebyshev_mean	$\mathbf{0.8310} \pm 0.0245$	$\mathbf{0.4046} \pm 0.0776$	0.3924 ± 0.1166	0.1212 ± 0.0307
chebyshev_max	0.7863 ± 0.0340	0.3794 ± 0.0849	0.4117 ± 0.1114	0.1279 ± 0.0337
canberra_mean	0.7479 ± 0.0449	0.3292 ± 0.0550	0.4693 ± 0.0618	0.1459 ± 0.0413
canberra_max	0.7362 ± 0.0448	0.3182 ± 0.0678	0.4920 ± 0.0608	0.1574 ± 0.0447

Table 35: Scorer results for WBC. Each entry is the mean AUROC or AUPRC over five seeds on this dataset, followed by the standard deviation across those seeds.

	Representation ($\mathbb{R}^D$)		Projected ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.9963 ± 0.0031	0.9852 ± 0.0128	0.2321 ± 0.4158	0.2481 ± 0.2996
dot_max	0.9716 ± 0.0198	0.8447 ± 0.1248	0.6600 ± 0.3429	0.4864 ± 0.3365
cosine_mean	0.9963 ± 0.0031	0.9852 ± 0.0128	0.3637 ± 0.4348	0.2811 ± 0.2962
cosine_max	0.9716 ± 0.0198	0.8447 ± 0.1248	0.5777 ± 0.3592	0.3907 ± 0.3354
rms_mean	0.9972 ± 0.0010	0.9894 ± 0.0034	0.9753 ± 0.0249	0.8859 ± 0.1044
rms_max	0.9716 ± 0.0198	0.8447 ± 0.1248	0.9781 ± 0.0116	0.8897 ± 0.0686
chebyshev_mean	$\mathbf{0.9977} \pm 0.0016$	$\mathbf{0.9912} \pm 0.0059$	0.9781 ± 0.0299	0.8991 ± 0.1342
chebyshev_max	0.9823 ± 0.0110	0.8959 ± 0.0890	$\mathbf{0.9902} \pm 0.0038$	$\mathbf{0.9593} \pm 0.0157$
canberra_mean	0.9949 ± 0.0042	0.9807 ± 0.0161	0.2265 ± 0.3994	0.2237 ± 0.2449
canberra_max	0.9679 ± 0.0207	0.8643 ± 0.0836	0.2791 ± 0.3946	0.2720 ± 0.3372

D.1 SCORING WITH NEGATIVE PAIRS

As a separate analysis, we evaluate the training signal (JSD) as in-batch scorer. Let $\mathcal{C}$ denote the comparison set. As mentioned in the manuscript, during training, $\mathcal{C}$ is implemented as a minibatch. At inference, for a batch of B records, let $s_{\mathbf{x},b,i}$ be the local-global alignment between the global representation of record $\mathbf{x}$ and feature i from record b, and let $s(u) = \mathrm{softplus}(u)$. The score for record a is defined in the same manner we define the training loss:

$$A(\mathrm{x};\mathcal{C}) = \frac{1}{N}\underbrace{\sum_{i=1}^{N}\mathrm{sp}(-s_{\mathbf{x},\mathbf{x},i})}_{\text{positive pairs}} + \frac{1}{BN}\underbrace{\sum_{b=1}^{B}\sum_{i=1}^{N}\mathrm{sp}(s_{\mathbf{x},b,i})}_{\text{negative pairs}},$$

Although in the manuscript we only use the positive representations as the primary anomaly signals to ensure that each prediction depends only on the record being evaluated, the full JSD score may be advantageous when a suitable set $\mathcal{C}$ is available. In particular, its negative term measures the alignment between the query's global representation and the local representations of other records. If $\mathcal{C}$ covers some anomalies likely to occur in practice, through domain-informed perturbations or noise injection, this signal may amplify anomalous patterns that are weakly expressed by a record's own local-global pairs alone. Using a fixed $\mathcal{C}$ would also make the score invariant to the composition of the incoming query batch, although it would remain conditional on the quality and coverage of the set. By contrast, constructing $\mathcal{C}$ from the evaluation minibatch can provide a similar benefit when that batch contains representative anomalies, but makes the score dependent on other test records. Therefore, our main results use batch-independent scores, and we treat full JSD scoring as a promising context-dependent alternative.

To assess these alternatives, Table 36 reports the performance of the full JSD score and its negative term across 25 datasets, using either the evaluation minibatch or a validation reference set of normal records. When $\mathcal{C}$ is part of the evaluation batch (which may contain anomalies), both scorers achieve high detection performance, suggesting that the JSD effectiveness is attributable to its negative term. In contrast, using a validation set that excludes potential anomalies substantially degrades performance, highlighting the sensitivity of this relational signal to the composition and coverage of the reference set.

Table 36: Results of the non-positive scorers, macro-averaged over the 25 datasets and five seeds.

Scorer	AUROC	AUPRC
`jsd_in_batch`	$0.9031^{\pm 0.0036}$	$0.8910^{\pm 0.0046}$
`jsd_validation_c`	$0.8173^{\pm 0.0051}$	$0.7993^{\pm 0.0059}$
`negative_in_batch`	$0.8874^{\pm 0.0037}$	$0.8578^{\pm 0.0044}$
`negative_validation_c`	$0.7448^{\pm 0.0055}$	$0.7202^{\pm 0.0062}$

E ABLATION STUDIES

E.1 HYPERPARAMETER SELECTION

We tuned the loss weights, batch size, projection size, masking rate, and encoder configuration.

Protocol. The hyperparameter selection uses Yeast, Fault, and Cardio with five seeds. Datasets utilize the feature typing defined in Table 10, the features 4 and 5, and 11,12,20 from Yeast and Fault are categorical, while every Cardio feature is continuous. The same one-class AD strategy is applied in every dataset. To select the best configurations, the dot-product scorer derived from training was used in both the projected and representation spaces. This scorer was used in this experiments to avoid introducing any bias in this process by preselecting a specific AD function. We perform two different hyperparameter searches, one per each space, and in each space we select the configuration that maximizes AUROC.

Search space. Table 37 summarizes the hyperparameter search. We divided the search into three stages, each inheriting the best settings from the previous ones:

- Stage 1 evaluates the grid for γ, β and λ at batch 256, projection $p = 16$, $\mathcal{E} = (32, 6, 8, 16)$, and mask rate 0.25.
- Stage 2 uses each branch's winner with batch $\{64, 256, 1024\}$ and $p \in \{8, 16, 32\}$.
- Stage 3 retains weights, batch, and projection size, and crosses masking rates $\{0.25, 0.50\}$ with the following encoder tuples configurations: $(32, 3, 4, 8)$, $(32, 3, 8, 16)$, $(32, 6, 4, 8)$, $(32, 6, 8, 16)$, $(32, 6, 16, 16)$, $(64, 6, 8, 8)$, $(64, 6, 8, 16)$, and $(64, 6, 16, 32)$. Where a tuple (D, L, H, d_h) represents embedding dimension D, L sequential encoder blocks, H attention heads and head dimension d_h.

Table 37: Sequential hyperparameter search.

Stage	Variables	Settings
1	$\gamma, \beta \in \{0.1, 0.25, 0.5, 1\}, \lambda \in \{1, 2, 5\}$	$B = 256, p = 16, (32, 6, 8, 16), m = 0.25$
2	$B \in \{64, 256, 1024\}, p \in \{8, 16, 32\}$	Stage-1
3	$r \in \{0.25, 0.50\}, 8 \times \mathcal{E}$	Stage-1 and Stage-2

Optimization and loss-based stopping. Candidates are trained for at most 2,000 epochs with AdamW and an initial learning rate of 5×10^{-4}.

Selected configurations. Table 38 reports the selected configuration for each space. Each configuration is evaluated with the mean dot-product scorer, since it is the most similar to the positive part of the training objective.

Table 38: Selected scorer configurations and averaged results on the selected datasets.

Space	γ	β	λ	B	p	Architecture	Mask	AUROC	AUPRC
$\mathbb{R}^D$	0.50	0.50	5	64	16	$(32, 6, 8, 16)$	0.5	0.7323	0.9079
$\mathbb{R}^p$	1.00	0.50	5	1024	8	$(32, 6, 8, 16)$	0.25	0.7063	0.8818

Interpretation. The representation space achieved better results than the projected space, possibly because its representations are also shaped by the auxiliary objectives. We find that both spaces selected the same encoder architecture and β values. As expected, the space optimized directly by the contrastive objective selected the largest batch size, whereas the representation space selected the smallest.

E.2 Anomaly Scoring Ablation

In this section, we evaluate the scoring functions on the three development datasets: Cardio, Fault, and Yeast. At inference, the anomaly score is defined by the discrepancy (measured by a discrepancy function d) between each of the local-global pairs of the same record,

$$A(\mathbf{x}) = \rho(d(\mathbf{z}_{\text{CLS}}, \mathbf{z}_1), \ldots, d(\mathbf{z}_{\text{CLS}}, \mathbf{z}_N)) \qquad A : \mathcal{X} \to \mathbb{R}.$$

We evaluate five discrepancy functions: dot-product, cosine, RMS, Chebyshev, and Canberra, with two aggregation functions (Mean and Max). We computed these scores both in the encoder representation space ($\mathbb{R}^D$) and in the space produced by the projection heads ($\mathbb{R}^p$).

The results in Tables 39, 41 and 40 reveal that the $\mathbb{R}^D$ achieves better detection performance across most discrepancy metrics. In contrast, the projected space favors the dot-product scoring derived from the contrastive training phase, indicating that the projection heads may discard some information to optimize the training objective.

Table 39: Scorer results for Cardio. Values show mean ± standard deviation over five seeds.

	Representation Space ($\mathbb{R}^D$)		Projected Space ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	**0.9450** ± 0.0111	0.9398 ± 0.0131	0.8624 ± 0.0479	0.8675 ± 0.0737
dot_max	0.9437 ± 0.0181	0.9269 ± 0.0192	**0.9018** ± 0.0373	**0.9038** ± 0.0379
cosine_mean	0.9447 ± 0.0151	0.9395 ± 0.0198	0.6927 ± 0.0306	0.6956 ± 0.0668
cosine_max	0.9441 ± 0.0175	0.9274 ± 0.0189	0.8672 ± 0.0662	0.8517 ± 0.0806
rms_mean	0.9439 ± 0.0130	0.9389 ± 0.0176	0.5541 ± 0.1018	0.5522 ± 0.0847
rms_max	0.9434 ± 0.0185	0.9246 ± 0.0196	0.6614 ± 0.0685	0.6410 ± 0.0790
chebyshev_mean	0.9443 ± 0.0155	**0.9415** ± 0.0190	0.6176 ± 0.1883	0.6184 ± 0.0847
chebyshev_max	0.9442 ± 0.0147	0.9301 ± 0.0287	0.5981 ± 0.1075	0.6121 ± 0.0772
canberra_mean	0.9403 ± 0.0117	0.9386 ± 0.0120	0.6722 ± 0.0450	0.6254 ± 0.0458
canberra_max	0.9367 ± 0.0166	0.9230 ± 0.0210	0.7889 ± 0.0499	0.7698 ± 0.0649

Table 40: Scorer results for Fault. Values show mean ± standard deviation over five seeds.

	Representation Space ($\mathbb{R}^D$)		Projected Space ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.7179 ± 0.0244	0.9231 ± 0.0081	0.6505 ± 0.0292	0.8971 ± 0.0151
dot_max	0.7121 ± 0.0244	0.9218 ± 0.0068	0.6604 ± 0.0285	**0.9093** ± 0.0066
cosine_mean	0.7467 ± 0.0280	0.9327 ± 0.0108	0.6334 ± 0.0432	0.8923 ± 0.0136
cosine_max	0.7100 ± 0.0160	0.9211 ± 0.0055	**0.6690** ± 0.0287	0.9085 ± 0.0046
rms_mean	**0.7564** ± 0.0302	**0.9375** ± 0.0103	0.5685 ± 0.0562	0.8655 ± 0.0208
rms_max	0.7078 ± 0.0144	0.9208 ± 0.0052	0.5713 ± 0.0553	0.8726 ± 0.0156
chebyshev_mean	0.7518 ± 0.0303	0.9362 ± 0.0101	0.5670 ± 0.0534	0.8667 ± 0.0255
chebyshev_max	0.7127 ± 0.0062	0.9220 ± 0.0051	0.5585 ± 0.0756	0.8688 ± 0.0253
canberra_mean	0.7044 ± 0.0164	0.9244 ± 0.0058	0.6221 ± 0.0760	0.8849 ± 0.0313
canberra_max	0.6802 ± 0.0242	0.9137 ± 0.0044	0.6351 ± 0.0510	0.8936 ± 0.0137

Table 41: Scorer results for Yeast. Values show mean ± standard deviation over five seeds.

	Representation Space ($\mathbb{R}^D$)		Projected Space ($\mathbb{R}^p$)	
Scorer	AUROC	AUPRC	AUROC	AUPRC
dot_mean	0.5341 ± 0.0379	0.8609 ± 0.0173	**0.6059** ± 0.0381	0.8808 ± 0.0133
dot_max	0.5038 ± 0.0366	0.8489 ± 0.0221	0.5966 ± 0.0316	0.8741 ± 0.0133
cosine_mean	0.5473 ± 0.0322	0.8617 ± 0.0171	0.6037 ± 0.0316	0.8782 ± 0.0179
cosine_max	0.5089 ± 0.0351	0.8491 ± 0.0206	0.6033 ± 0.0364	0.8762 ± 0.0186
rms_mean	0.5598 ± 0.0294	0.8680 ± 0.0145	0.5827 ± 0.0272	0.8638 ± 0.0178
rms_max	0.5098 ± 0.0350	0.8491 ± 0.0208	0.5859 ± 0.0309	0.8667 ± 0.0197
chebyshev_mean	**0.5982** ± 0.0219	0.8688 ± 0.0126	0.5955 ± 0.0315	0.8780 ± 0.0119
chebyshev_max	0.5229 ± 0.0364	0.8515 ± 0.0185	0.5958 ± 0.0268	**0.8811** ± 0.0096
canberra_mean	0.5876 ± 0.0399	**0.8739** ± 0.0164	0.5872 ± 0.0560	0.8772 ± 0.0275
canberra_max	0.5335 ± 0.0360	0.8602 ± 0.0189	0.5986 ± 0.0567	0.8810 ± 0.0222

E.3 LOCAL-GLOBAL TRAINING OBJECTIVE

For every variant trained with the local-global objective, anomaly scores are derived from the alignment between the global representation $\mathbf{z}_{\mathrm{CLS}}$ and the contextualized representation $\mathbf{z}_i$ of each feature i. Given a record $\mathbf{x}$ with N features, the projected dot-product score is

$$A_{dot}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} -\phi(\mathbf{z}_{\mathrm{CLS}}) \cdot \psi(\mathbf{z}_i),$$

where ϕ and ψ are the learned global and local projection heads from $\mathcal{E}$, as introduced in the main article. A high score indicates weak alignment between the global representation and the record's local feature representations. The Chebyshev score does not use projection heads and directly measures in the encoder embedding dimension D:

$$A_{cheb}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} \max_{1\le k\le D} |\mathbf{z}_{\mathrm{CLS},k} - \mathbf{z}_{i,k}|.$$

We used the Chebyshev distance to measure local-global misalignment, providing a basis for comparing the detection performance of the proposed variants of the model.

We also include a variant, MFM-R, which does not include local-global alignment during training, and only utilizes the pretext objectives. This allows us to assess AD performance within the same encoder architecture, using only the masking and reconstruction objectives. Therefore, this variant is trained without the local-global loss and without the projection heads. However, its Chebyshev score is computed with the same expression above, to evaluate whether MFM and CLS reconstruction alone can induce comparable local-global alignment. Also, we evaluate AD scores by using its two training objectives. For the MFM score, one feature is masked at a time while all remaining features are kept visible.

$$A_{\mathrm{mfm}}(\mathbf{x}) = \sum_{i=1}^{N} \mathcal{L}_{\mathrm{MFM}}(\mathbf{x}_M), \qquad M = \{i\}.$$

For CLS reconstruction, the complete unmasked record $\mathbf{x}$ used as input for $\mathcal{E}$, and every feature is reconstructed from $\mathbf{z}_{\mathrm{CLS}}$:

$$A_{\mathrm{cls}}(\mathbf{x}) = \mathcal{L}_{\mathrm{CLS}}(\mathbf{x}).$$

In these settings, A_{mfm} measures whether an individual feature can be inferred from its context, whereas A_{cls} measures if the global representation retains enough information to reconstruct the complete record. The two reconstruction scores are reported separately.

To assess the contribution of each auxiliary training objective to local-global alignment, we compared five variants on three ADBench datasets: Yeast, Fault, and Cardio. Table 42 shows that the projected dot-product score barely improves as objectives are added, whereas the combination of LG, MFM, and reconstruction achieves the strongest aggregate performance with Chebyshev distance, reaching a macro AUROC and AUPRC of 0.7648 and 0.9155, respectively. The observed degradation in the projected dot-product score is attributable to loss balancing, since as more objectives are added, the relative weight of the contrastive term decreases. Additionally, comparing LG-MFM-R against MFM-R confirms the contribution of local-global alignment to the learned representations, increasing the Chebyshev metrics on all three datasets. However, the reconstruction scores in MFM-R demonstrate that the auxiliary objectives are also effective anomaly detectors. Overall, the results suggest that auxiliary objectives provide useful anomaly signals on their own, but local-global is important for reshaping their learning signals into useful representations for local-global alignment, achieving the best performance overall.

E.4 ENCODER ABLATION

To assess the contribution of attention, we conduct an encoder ablation in which the Transformer Encoder is replaced with an Feed-Forward Neural Network (FFNN). We evaluate both variants on three datasets from ADBench while keeping the masking procedure, pretext objectives, training protocol, and anomaly-scoring function unchanged. This comparison isolates the effect of the encoder on the learned local and global representations. As shown in Table 43, the Transformer-based model

Table 42: Ablation of the training objectives on Yeast, Fault and Cardio. The objectives using local-global alignment are evaluated with the projected dot mean (`p_dot`) and mean Chebyshev distance (`cheby`). The MFM-R objective is trained without LG and evaluated using `cheby`, MFM reconstruction loss (`mfm`), and CLS reconstruction (`cls`). Dataset columns report mean ± standard deviation over five seeds. The Macro column averages the three datasets within each seed and reports mean ± standard deviation over those five per-seed averages.

Objective	Scorer	Yeast	Fault	Cardio	Macro
AUROC					
LG	p_dot	$0.6052^{\pm 0.0113}$	$0.6628^{\pm 0.0095}$	$0.8336^{\pm 0.0384}$	$0.7005^{\pm 0.0143}$
	cheby	$0.5851^{\pm 0.0412}$	$0.6085^{\pm 0.0260}$	$0.9236^{\pm 0.0142}$	$0.7057^{\pm 0.0237}$
LG-MFM	p_dot	$0.5955^{\pm 0.0181}$	$0.6659^{\pm 0.0317}$	$0.8702^{\pm 0.0350}$	$0.7105^{\pm 0.0108}$
	cheby	$0.5889^{\pm 0.0427}$	$0.6550^{\pm 0.0338}$	$0.9398^{\pm 0.0145}$	$0.7279^{\pm 0.0214}$
LG-R	p_dot	$0.6032^{\pm 0.0222}$	$0.6640^{\pm 0.0339}$	$0.8501^{\pm 0.0542}$	$0.7058^{\pm 0.0202}$
	cheby	$0.5906^{\pm 0.0188}$	$0.6590^{\pm 0.0254}$	$0.9359^{\pm 0.0190}$	$0.7285^{\pm 0.0191}$
MFM-R	cheby	$0.4802^{\pm 0.0154}$	$0.6435^{\pm 0.0144}$	$0.9078^{\pm 0.0410}$	$0.6772^{\pm 0.0164}$
	mfm	$0.4650^{\pm 0.0318}$	$0.6926^{\pm 0.0103}$	$0.9375^{\pm 0.0107}$	$0.6984^{\pm 0.0124}$
	cls	$0.4909^{\pm 0.0444}$	$0.6812^{\pm 0.0116}$	$0.9328^{\pm 0.0085}$	$0.7016^{\pm 0.0156}$
LG-MFM-R	p_dot	$\mathbf{0.6059}^{\pm 0.0381}$	$0.6505^{\pm 0.0292}$	$0.8624^{\pm 0.0479}$	$0.7063^{\pm 0.0197}$
	cheby	$0.5982^{\pm 0.0219}$	$\mathbf{0.7518}^{\pm 0.0303}$	$\mathbf{0.9443}^{\pm 0.0155}$	$\mathbf{0.7648}^{\pm 0.0170}$
AUPRC					
LG	p_dot	$0.8772^{\pm 0.0043}$	$0.9068^{\pm 0.0056}$	$0.8475^{\pm 0.0422}$	$0.8772^{\pm 0.0145}$
	cheby	$0.8725^{\pm 0.0155}$	$0.8860^{\pm 0.0157}$	$0.9390^{\pm 0.0134}$	$0.8992^{\pm 0.0115}$
LG-MFM	p_dot	$0.8709^{\pm 0.0107}$	$0.9005^{\pm 0.0084}$	$0.8766^{\pm 0.0382}$	$0.8826^{\pm 0.0141}$
	cheby	$0.8786^{\pm 0.0122}$	$0.9033^{\pm 0.0101}$	$0.9347^{\pm 0.0198}$	$0.9055^{\pm 0.0094}$
LG-R	p_dot	$0.8771^{\pm 0.0157}$	$0.8925^{\pm 0.0066}$	$0.8460^{\pm 0.0629}$	$0.8719^{\pm 0.0225}$
	cheby	$0.8766^{\pm 0.0088}$	$0.8956^{\pm 0.0136}$	$0.9319^{\pm 0.0197}$	$0.9013^{\pm 0.0116}$
MFM-R	cheby	$0.8420^{\pm 0.0106}$	$0.8782^{\pm 0.0040}$	$0.8856^{\pm 0.0360}$	$0.8686^{\pm 0.0134}$
	mfm	$0.8316^{\pm 0.0098}$	$0.9047^{\pm 0.0046}$	$0.9384^{\pm 0.0064}$	$0.8916^{\pm 0.0061}$
	cls	$0.8371^{\pm 0.0150}$	$0.9026^{\pm 0.0043}$	$0.9302^{\pm 0.0070}$	$0.8899^{\pm 0.0067}$
LG-MFM-R	p_dot	$\mathbf{0.8808}^{\pm 0.0133}$	$0.8971^{\pm 0.0151}$	$0.8675^{\pm 0.0737}$	$0.8818^{\pm 0.0170}$
	cheby	$0.8688^{\pm 0.0126}$	$\mathbf{0.9362}^{\pm 0.0101}$	$\mathbf{0.9415}^{\pm 0.0190}$	$\mathbf{0.9155}^{\pm 0.0109}$

consistently achieves better results compared with the FFNN variant. In encoder space, the Transformer outperforms the FFNN in all comparisons, with especially large gains on Cardio and Fault. In projected space, the Transformer also consistently improves detection performance. Overall, these results indicate that attention provides more effective local and global feature interactions than an FFNN encoder.

The three objectives in Lo-GRAD impose two architectural requirements. First, each representation must be contextualized by the other features in the record. Second, the model must expose both row-level and feature-level representations in a single forward pass. The attention mechanism in a Transformer Encoder provides these properties directly, supports multiple training objectives, and produces informative representations (Devlin et al., 2019), whereas an FFNN does not. Also, in the MFM objective, the local representation $\mathbf{z}_i$ must capture the conditional structure $x_i \mid \mathbf{x}_{\bar{M}}$. Attention supports this by forming $\mathbf{z}_i$ through an input-dependent aggregation of the feature embeddings, allowing the model to emphasize attributes that are informative for the masked position. An FFNN instead produces a single record-level vector, leaving such dependencies implicit and providing no natural feature-specific representations. For global row reconstruction, the prepended token $\mathbf{t}_{\mathrm{CLS}}$ attends to all feature tokens and summarizes the complete record in $\mathbf{z}_{\mathrm{CLS}}$, matching the role required by a global representation. An FFNN has no intrinsic global token or principled mechanism for separating record-level and feature-level views. Finally, the local-global objective requires views that are distinct yet mutually informed. The attention mechanism allows $\mathbf{z}_{\mathrm{CLS}}$ and $\mathbf{z}_1, \ldots, \mathbf{z}_N$ to interact, so $s_{a,a,i}$ measures alignment between a contextualized record and its corre-

Table 43: Mean-aggregation results for the FFNN and Transformer Encoder. Entries report mean ± standard deviation over five seeds.

			Yeast		Fault		Cardio	
Space	Discrepancy	Model	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
$\mathbb{R}^D$	Dot	FFNN	$0.4840^{\pm 0.0531}$	$0.8346^{\pm 0.0196}$	$0.5154^{\pm 0.0557}$	$0.8307^{\pm 0.0435}$	$0.7389^{\pm 0.0385}$	$0.7802^{\pm 0.0409}$
		Transformer	$0.5341^{\pm 0.0379}$	$0.8609^{\pm 0.0173}$	$0.7179^{\pm 0.0244}$	$0.9231^{\pm 0.0081}$	$0.9450^{\pm 0.0111}$	$0.9398^{\pm 0.0131}$
	Cosine	FFNN	$0.4813^{\pm 0.0557}$	$0.8351^{\pm 0.0229}$	$0.5110^{\pm 0.0578}$	$0.8305^{\pm 0.0425}$	$0.7224^{\pm 0.0428}$	$0.7701^{\pm 0.0410}$
		Transformer	$0.5473^{\pm 0.0322}$	$0.8617^{\pm 0.0171}$	$0.7467^{\pm 0.0280}$	$0.9327^{\pm 0.0108}$	$0.9447^{\pm 0.0151}$	$0.9395^{\pm 0.0198}$
	RMS	FFNN	$0.4919^{\pm 0.0848}$	$0.8417^{\pm 0.0404}$	$0.4276^{\pm 0.1082}$	$0.8020^{\pm 0.0520}$	$0.7504^{\pm 0.0714}$	$0.7672^{\pm 0.0284}$
		Transformer	$0.5598^{\pm 0.0294}$	$0.8680^{\pm 0.0145}$	$0.7564^{\pm 0.0302}$	$0.9375^{\pm 0.0103}$	$0.9439^{\pm 0.0130}$	$0.9389^{\pm 0.0176}$
	Chebyshev	FFNN	$0.4850^{\pm 0.0703}$	$0.8381^{\pm 0.0382}$	$0.4719^{\pm 0.1076}$	$0.8234^{\pm 0.0426}$	$0.7614^{\pm 0.1440}$	$0.7823^{\pm 0.0816}$
		Transformer	$0.5982^{\pm 0.0219}$	$0.8688^{\pm 0.0126}$	$0.7518^{\pm 0.0303}$	$0.9362^{\pm 0.0101}$	$0.9443^{\pm 0.0155}$	$0.9415^{\pm 0.0190}$
	Canberra	FFNN	$0.4555^{\pm 0.0545}$	$0.8208^{\pm 0.0327}$	$0.5154^{\pm 0.0628}$	$0.8453^{\pm 0.0302}$	$0.7646^{\pm 0.1256}$	$0.7899^{\pm 0.0749}$
		Transformer	$0.5876^{\pm 0.0399}$	$0.8739^{\pm 0.0164}$	$0.7044^{\pm 0.0164}$	$0.9244^{\pm 0.0058}$	$0.9403^{\pm 0.0117}$	$0.9386^{\pm 0.0120}$
$\mathbb{R}^P$	Dot	FFNN	$0.5594^{\pm 0.0512}$	$0.8588^{\pm 0.0245}$	$0.6365^{\pm 0.0296}$	$0.8877^{\pm 0.0090}$	$0.4659^{\pm 0.1369}$	$0.4938^{\pm 0.0696}$
		Transformer	$0.6059^{\pm 0.0381}$	$0.8808^{\pm 0.0133}$	$0.6505^{\pm 0.0292}$	$0.8971^{\pm 0.0151}$	$0.8624^{\pm 0.0479}$	$0.8675^{\pm 0.0737}$
	Cosine	FFNN	$0.5505^{\pm 0.0421}$	$0.8571^{\pm 0.0189}$	$0.6145^{\pm 0.0621}$	$0.8889^{\pm 0.0225}$	$0.4811^{\pm 0.1766}$	$0.5196^{\pm 0.1159}$
		Transformer	$0.6037^{\pm 0.0316}$	$0.8782^{\pm 0.0179}$	$0.6334^{\pm 0.0432}$	$0.8923^{\pm 0.0136}$	$0.6927^{\pm 0.0306}$	$0.6956^{\pm 0.0668}$
	RMS	FFNN	$0.4648^{\pm 0.0590}$	$0.8164^{\pm 0.0254}$	$0.4171^{\pm 0.0918}$	$0.8132^{\pm 0.0439}$	$0.5527^{\pm 0.2031}$	$0.5837^{\pm 0.1684}$
		Transformer	$0.5827^{\pm 0.0272}$	$0.8638^{\pm 0.0178}$	$0.5685^{\pm 0.0562}$	$0.8655^{\pm 0.0208}$	$0.5541^{\pm 0.1018}$	$0.5522^{\pm 0.0847}$
	Chebyshev	FFNN	$0.4577^{\pm 0.0792}$	$0.8188^{\pm 0.0407}$	$0.4351^{\pm 0.0759}$	$0.8207^{\pm 0.0246}$	$0.5293^{\pm 0.1234}$	$0.5522^{\pm 0.1095}$
		Transformer	$0.5955^{\pm 0.0315}$	$0.8780^{\pm 0.0119}$	$0.5670^{\pm 0.0534}$	$0.8667^{\pm 0.0255}$	$0.6176^{\pm 0.1883}$	$0.6184^{\pm 0.0847}$
	Canberra	FFNN	$0.4983^{\pm 0.0728}$	$0.8332^{\pm 0.0267}$	$0.4723^{\pm 0.0731}$	$0.8346^{\pm 0.0302}$	$0.6297^{\pm 0.1689}$	$0.6241^{\pm 0.1516}$
		Transformer	$0.5872^{\pm 0.0560}$	$0.8772^{\pm 0.0275}$	$0.6221^{\pm 0.0760}$	$0.8849^{\pm 0.0313}$	$0.6722^{\pm 0.0450}$	$0.6254^{\pm 0.0458}$

sponding contextualized features rather than between two projections of the same undifferentiated vector. Without this interaction, the local and global views risk becoming redundant, weakening the positive-pair misalignment signal used for AD.