Summary of Our Responses and Long-term Plan

We sincerely thank all the reviewers for their insightful comments and valuable suggestions. We have carefully read through them and provided corresponding responses individually. The primary changes are summarized below, and we will update the revised paper as soon as possible.

Related Work

We enrich the related work in Section 2 to discuss relevant AD methods in other data modalities like Computer Vision (CV) and Natural Language Processing (NLP), including E3Outlier [1], GOAD [2], CSI [3] and SLA2P [4].

Dataset Description

We update Table A1 in the Appendix, which additionally shows the details about the training sets for each datasets, including the number of unlabeled samples and that of labeled anomalies. The number of ground-truth normal and abnormal samples in the testing set is also provided for each datasets.

Experiment

Following the valuable suggestions of all reviewers, we conduct the experiments illustrated below, as to further explore the proposed Overlap loss in various data/model settings. If not otherwise specified, we reported the average AUC-PR results on 25 real-world datasets due to the space limit.

- We perform additional ablation studies on two data scenarios: 1. Normal data is generated from a Gaussian (mixture) distribution, where the anomalies follow the original distribution; 2. Anomalies are generated from a Gaussian (mixture) distribution, where the normal ones follow the original distribution. We evaluate different basic methods mentioned in Section 3 under diverse network architectures, where the conclusion obtained is **basically consistent with the original paper**.
- We investigate the detection performance of proposed Overlap loss when the score distribution estimator KDE is supported by different kernels and bandwidths, and **acquire consistent results** on different network architectures like ResNet and FTTransformer. These results can be regarded as an important supplement to Section 4.2.
- We include stronger supervised baselines like tree-based ensemble methods XGBoost [5] and CatBoost [6] in Section 4.2.1, indicating that the proposed **Overlap loss is still competitive compared to these ensemble methods,** especially when only very limited labeled data is available during the training stage.
- We provide additional experiments in Appendix Section .3, evaluating whether the ensembling strategy in Overlap Area Calculation is effective when the situation of multiple intersection points becomes more often. We observe that this situation is correlated to the selection of bandwidth in the KDE estimator, where the **ensembling strategy can further improve performance when more situations of multiple intersection points occur**.

Future Direction

We extend the future directions of the proposed Overlap loss in Section 5. As suggested by the reviewers, we will consider the extension of Overlap loss to unsupervised AD scenarios and OOD problems across multiple data modalities, as promising research directions. Besides, we commit to providing more theoretical proof of our proposed method.

Long-term plan

We promise to maintain and enrich the open-source project of our proposed methods, and providing user-friendly API for both researchers and practitioners to evaluate the Overlap loss in various AD frameworks or different data scenarios. Also, we consider integrating our method into large-scale Python packages or benchmarks of anomaly detection like PyOD [7] and ADBench [8].

[1] Wang, Siqi, et al. "Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network." NeurIPS 2019.

[2] Bergman, Liron, and Yedid Hoshen. "Classification-Based Anomaly Detection for General Data." ICLR 2020.

[3] Tack, Jihoon, et al. "Csi: Novelty detection via contrastive learning on distributionally shifted instances." NeurIPS 2020.

[4] Wang, Yizhou, et al. "Self-supervision Meets Adversarial Perturbation: A Novel Framework for Anomaly Detection." CIKM 2022.

[5] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." KDD 2016.

[6] Prokhorenkova, Liudmila, et al. "CatBoost: unbiased boosting with categorical features." NeurIPS 2018.

[7] Zhao, Yue, Zain Nasrullah, and Zheng Li. "PyOD: A Python Toolbox for Scalable Outlier Detection." JMLR 2019.

[8] Han, Songqiao, et al. "Adbench: Anomaly detection benchmark." NeurIPS 2022.

Reviewer J7gT

Q1. Ensuring the correct gradient update direction in Overlap Area Calculation is not well-motivated. Is it necessary to ensure the order in anomaly scores (i.e., anomalies are assigned lower anomaly scores)? As far as I am concerned, encouraging a good discrimination/separation is sufficient for anomaly detection with no need to keep a stricter restriction. Please clarify necessity of ensuring a correct score order clearly.

R1. Thanks for mentioning this. Indeed, encouraging a good discrimination (or separation) in either feature representation or anomaly scores is the first and foremost step in anomaly detection, which helps to distinguish normal and abnormal samples in the downstream tasks.

One of the reasons that Overlap loss further ensures a correct order in anomaly score (i.e., the anomaly score of abnormal samples should be greater than the normal ones) is to keep consistent with current AD research. Among the existing AD methods, like *distance-based* or *reconstruction-based* methods, the definition of anomaly scores is often natural. For example, anomalies are often referred to the data objects that deviate significantly from the majority of data objects. Thus the distance to the nearest neighbor can be regarded as an effective measurement of abnormality——**The greater the distance (higher anomaly score), the more abnormal**. This distance-based definition of anomaly scores is applied to many AD methods, from the simple KNN [1] to the deep learning method REPEN [2]. For reconstruction-based AD methods [3, 4], they typically train AutoEncoder or GAN model solely on normal data to learn regular patterns. As a result, the trained model fails to reconstruct anomalies, leading to——**higher reconstruction errors (higher anomaly score), the more abnormal**. Although there is no natural definition of anomaly score in the end-to-end learning AD methods, we indicate that previous literature [5, 6] often trains models to map the anomaly score of the abnormal samples higher and that of the normal samples lower.

Therefore, we follow the research papers mentioned above to minimize the overlap area of arbitrary score distributions while ensuring correct order in anomaly scores, i.e., the anomaly score of abnormal data should be generally higher than that of normal data. The consistent definitions of anomaly score allow us to better compare Overlap loss with other AD methods and well generalize to diverse data scenarios. Moreover, for a newcoming testing dataset, we do not need to first determine the relative magnitude of anomaly scores, i.e., whether a higher anomaly score indicates more anomalous or vice versa, before making final decisions or evaluating model performance. **Our experimental results shown in Table 2 also indicate that the Overlap-Proposed method significantly outperforms the Overlap-Arbitrary method.** This is because the Overlap-Proposed realizes a suitable score distribution discrimination while ensuring the order in anomaly score, whereas Overlap-Arbitrary only achieves differentiation in score distribution, which leads to the disorder in anomaly scores.

[1] Angiulli, Fabrizio, and Clara Pizzuti. "Fast outlier detection in high dimensional spaces." PKDD 2002.

[2] Pang, Guansong, et al. "Learning representations of ultrahigh-dimensional data for random distance-based outlier detection." KDD 2018.

[3] Akcay, Samet, Amir Atapour-Abarghouei, and Toby P. Breckon. "Ganomaly: Semi-supervised anomaly detection via adversarial training." ACCV 2019.

[4] Zenati, Houssam, et al. "Adversarially learned anomaly detection." ICDM 2018.

[5] Pang, Guansong, Chunhua Shen, and Anton van den Hengel. "Deep anomaly detection with deviation networks." KDD 2019.

[6] Zhou, Yingjie, et al. "Feature encoding with autoencoders for weakly supervised anomaly detection." TNNLS 2021.

Q2. Although anomalies do not always follow a Gaussian distribution, it is reasonable to assume that normal samples often do [1]. Therefore, we suggest adding an additional experiment in Table 2 to compare the performance of different models in situations where normal data are sampled from a Gaussian (mixture) distribution and abnormal data is modeled by KDE, and vice versa.

R2. Thanks for mentioning this interesting point! Following your suggestions and the related work [1] mentioned, we compare several basic methods mentioned in subsection 4.2.3, as well as our proposed Overlap loss in two data settings: 1. Normal samples are generated from a Gaussian mixture model (GMM), while anomalies are the original input data; 2. Anomalies are generated from GMM, while normal samples are the original input data. The GMM model first fits on the corresponding input data, and then is used to sample synthetic data that follows a Gaussian (mixture) distribution, where the number of mixture components is selected based on the Bayesian information criterion (BIC) criterion.

We show the experimental results in the following Tables w.r.t. the ratio of labeled anomalies γ_l =5%. We find that whether normal or abnormal samples follow a Gaussian (mixture) distribution generates similar conclusions, which are also **consistent with the experimental results of the original paper**. Compared to the other methods, the Overlap-Proposed achieves better AUC-PR performance in different network architectures.

Table: AUC-PR results of ablation studies where normal samples follow a Gaussian (mixture) distribution.

γ_l =5%	VAE	MLP	AE	ResNet	FTT
Overlap-Gaussian	0.141±0.127	/	/	/	/
Overlap-Arbitrary	/	0.393±0.244	0.380±0.187	0.360±0.205	0.316±0.215
Overlap-Ranking	/	0.555±0.252	0.568±0.253	0.580±0.259	0.678±0.250
Overlap-Combined	/	0.661±0.260	0.676±0.265	0.664±0.262	0.494±0.267
Overlap-Proposed	/	0.660±0.266	0.678±0.268	0.686±0.268	0.667±0.273

Table: AUC-PR results of ablation studies where abnormal samples follow a Gaussian (mixture) distribution.

γ_l =5%	VAE	MLP	AE	ResNet	FTT
Overlap-Gaussian	0.139±0.125	/	/	/	/
Overlap-Arbitrary	/	0.353±0.196	0.392±0.150	0.379±0.186	0.294±0.165
Overlap-Ranking	/	0.553±0.270	0.581±0.269	0.588±0.274	0.673±0.241
Overlap-Combined	/	0.646±0.263	0.676±0.257	0.670±0.259	0.574±0.255
Overlap-Proposed	/	0.643±0.273	0.674±0.263	0.687±0.262	0.665±0.260

Besides, we indicate that compared to the other basic methods and the Overlap-proposed, Overlap-Gaussian is relatively difficult to train, leading to an inferior performance with the default hyper-parameter settings (learning rate=0.001, epochs=20). Only when we increase the learning rate (from 0.001 to 0.01) and training epochs (from 20 to 100), Overlap-Gaussian improves AUC-PR from 0.141 to 0.545 w.r.t. normal data are sampled from a Gaussian (mixture) distribution, and improves AUC-PR from 0.139 to 0.551 w.r.t. anomalies are sampled from that distribution. However, there is still a gap between its AUC-PR performance and other methods, since Gaussian mixture distributions sometimes deviate from its unimodal Gaussian assumptions.

Table: AUC-PR results of Overlap-Gaussian and its further tuned version

	Normal Gaussian	Anomaly Gaussian
Overlap-Gaussian	0.141±0.127	0.139±0.125
Overlap-Gaussian (tuned)	0.545±0.297	0.551±0.268

[1] Lee, K., Lee, K., Lee, H., & Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. NeurIPS 2018.

Q3. As one of the core contributions, the hyper-parameters of KDE (bandwidth and kernel functions beyond Gaussian kernel) should be discussed.

R3. Thanks for pointing this out! Based on your suggestion, we conduct corresponding experiments to investigate the performance of our proposed Overlap loss with respect to different bandwidths (from 0.1 to 10.0) and various kernel choices (including Gaussian, Epanechnikov, Quartic, and Triangular), as shown in the following Table.

Table: AUC-PR results of MLP-Overlap with different kernels and bandwidths

γ_l =5%	bw=0.01	bw=0.1	bw=1.0	bw=10.0
Gaussian	0.577	0.555	0.623	0.341
Epanechnikov	0.389	0.598	0.604	0.398
Quartic	0.541	0.507	0.594	0.523
Triangular	0.571	0.529	0.606	0.533

γ_l =10%	bw=0.01	bw=0.1	bw=1.0	bw=10.0
Gaussian	0.619	0.629	0.674	0.356
Epanechnikov	0.399	0.624	0.646	0.417
Quartic	0.616	0.597	0.643	0.535
Triangular	0.608	0.612	0.651	0.552

γ_l =20%	bw=0.01	bw=0.1	bw=1.0	bw=10.0
Gaussian	0.631	0.661	0.696	0.362
Epanechnikov	0.415	0.636	0.667	0.439
Quartic	0.662	0.669	0.663	0.551
Triangular	0.667	0.679	0.671	0.559

From the above experimental results, we find that the **Gaussian kernel is still a better choice** for estimating the distribution of anomaly scores, achieving higher AUC-PR than the other kernels in general. In fact, the Gaussian kernel is the most commonly used method in kernel density estimation [1, 2, 3]. This may be due to the fact that compared to the other kernel like Triangular kernel, Gaussian kernel usually obtains a smoother probability density function (PDF) [4] and satisfies the central limit theorem, thus is closer to the ground-truth score distribution.

For the choice of bandwidth, we indicate that **too small or too large bandwidth is harmful to estimating score distribution**, where the former leads to undersmoothing (i.e., the PDF will look like a combination of individual peeks) and the latter leads to oversmoothing (i.e., the PDF will look like a unimodal distribution). Moreover, We show that the detection performance of the proposed methods is similar when the bandwidth is around 1.0. This conclusion is generally consistent for different network architectures, as shown in the Table below. In future work, we will consider some techniques like automated model selection [5] or adaptive bandwidth estimation [6] to determine the best kernel and bandwidth of KDE in our Score Distribution Estimator illustrated in subsection 3.3.1.

Table: AUC-PR results of Overlap loss (with Gaussian kernel and different bandwidths) on various network architectures

γ_l =**5%**	**bw=0.5**	**bw=0.75**	**bw=1**	**bw=1.25**	**bw=1.5**
MLP-Overlap	0.637	0.648	0.647	0.647	0.642
AE-Overlap	0.667	0.674	0.676	0.672	0.659
ResNet-Overlap	0.628	0.653	0.651	0.669	0.668
FTTransformer-Overlap	0.645	0.645	0.651	0.645	0.645

γ_l =**10%**	**bw=0.5**	**bw=0.75**	**bw=1**	**bw=1.25**	**bw=1.5**
MLP-Overlap	0.694	0.700	0.699	0.697	0.691
AE-Overlap	0.713	0.712	0.721	0.717	0.703
ResNet-Overlap	0.730	0.733	0.725	0.725	0.719
FTTransformer-Overlap	0.690	0.690	0.712	0.690	0.690

γ_l =**20%**	**bw=0.5**	**bw=0.75**	**bw=1**	**bw=1.25**	**bw=1.5**
MLP-Overlap	0.722	0.724	0.721	0.717	0.708
AE-Overlap	0.742	0.738	0.740	0.736	0.719
ResNet-Overlap	0.768	0.764	0.768	0.757	0.749
FTTransformer-Overlap	0.721	0.721	0.756	0.721	0.721

[1] Kim, JooSeuk, and Clayton D. Scott. "Robust kernel density estimation." JMLR 2012.

[2] Vandermeulen, Robert A., and Clayton Scott. "Robust kernel density estimation by scaling and projection in hilbert space." NeurIPS 2014.

[3] Humbert, Pierre, Batiste Le Bars, and Ludovic Minvielle. "Robust kernel density estimation with median-of-means principle." ICML 2022.

[4] Zhang, Liangwei, Jing Lin, and Ramin Karim. "Adaptive kernel density-based anomaly detection for nonlinear systems." KBS 2018.

[5] Zhao, Yue, Ryan Rossi, and Leman Akoglu. "Automatic unsupervised outlier model selection." NeurIPS 2021.

[6] Zhang, Jia-Dong, and Chi-Yin Chow. "Geosoca: Exploiting geographical, social and categorical correlations for point-of-interest recommendations." SIGIR 2015.

Q4. If I understand correctly, the unlabeled training set contains all classes of anomalies, drawn from stratified sampling. However, it may be interesting to investigate a more practical scenario where some classes of anomalies are never observed during training, either as labeled anomalies or unlabeled ones

R4. We appreciate the advice of considering the scenario when unknown (i.e., out-of-distribution, OOD) anomalies [1, 2] occur in the testing phase. The OOD problem is another important branch of AD study, which focuses more on ensuring the robustness of AD models to maintain high performance on OOD samples with domain shift [3], or emphasizes the model reliability by requiring the identification of samples with semantic shift [2]. A plethora of OOD methods (say more than 30) [4] has been developed in the past five years, ranging from classification-based to density-based to distance-based methods. Different from our paper, the OOD problem has its unique pipeline, including different preprocessing, network backbones, postprocessing, evaluator, and so on. Therefore, directly transferring our methods to the OOD problem or comparing the existing OOD methods in our setting is non-trivial work. Due to the scope of our work and page limits, we will leave a broader coverage of the OOD topics for future work on the Overlap loss, but will enrich the related works on this topic in the updated paper version.

[1] Hendrycks, Dan, and Kevin Gimpel. "A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks." ICLR 2017.

[2] Yang, Jingkang, et al. "Generalized out-of-distribution detection: A survey." arXiv 2021.

[3] Zhou, Kaiyang, et al. "Domain generalization: A survey." TPAMI 2022.

[4] Yang, Jingkang, et al. "OpenOOD: Benchmarking Generalized Out-of-Distribution Detection." NeurIPS 2022.

Reviewer DzEi

Q1. There lack some strong semi-supervised/unsupervised anomaly detection baseline methods. Just to name a few: E3Outlier [1], GOAD [2], CSI [3], SLA2P [4].

R1. Thanks for mentioning these unsupervised/semi-supervised AD methods [1, 2, 3, 4]. In the original paper, we include few unsupervised or semi-supervised AD baselines for comparison, mainly considering the following reasons:

1. Unsupervised or semi-supervised (trained on only normal data) AD methods are proven to be inferior to those anomaly-informed AD models [5, 6], since they lack prior knowledge about abnormal behaviors. In our experimental results, we also observe that even for DL-based unsupervised methods like DeepSVDD (with AUC-PR 0.147 w.r.t. the ratio of labeled anomalies γ_l =5%) or GAN-based methods like GANomaly (with AUC-PR 0.297) , their detection performance is still far below that of anomaly-informed methods like DeepSAD (with 0.506 AUC-PR) and our Overlap loss based models (with 0.623 AUC-PR for MLP-Overlap). Besides, [5] indicates that “none of the unsupervised methods is statistically better than the others”, which means for tabular data, unsupervised methods are likely to perform similarly overall.
2. Transferring AD methods across different data modalities (i.e., tabular to CV/NLP or vice versa) for comparison is non-trivial and may lead to unfairness if the experiment settings is not well standardized. This is because that standard experimental pipeline like anomaly definition, data preprocessing, and network architecture often differ in various data modalities. Therefore it is difficult to directly compare these CV or NLP based AD methods to those tabular based methods, and vice versa.

We would like to thank you again for pointing out the need to discuss the AD problem beyond the scope of anomaly-informed AD scenarios and tabular data modality. In future work, we will extend the proposed Overlap loss to unsupervised or semi-supervised data scenarios, and evaluate the effectiveness of Overlap loss in other data modalities, by integrating it with other network architectures like ViT [7] in CV domain and RoBerta [8] in NLP domain.

[1] Wang, Siqi, et al. "Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network." NeurIPS 2019.

[2] Bergman, Liron, and Yedid Hoshen. "Classification-Based Anomaly Detection for General Data." ICLR 2020.

[3] Tack, Jihoon, et al. "Csi: Novelty detection via contrastive learning on distributionally shifted instances." NeurIPS 2020.

[4] Wang, Yizhou, et al. "Self-supervision Meets Adversarial Perturbation: A Novel Framework for Anomaly Detection." CIKM 2022.

[5] Han, Songqiao, et al. "Adbench: Anomaly detection benchmark." NeurIPS 2022.

[6] Pang, Guansong, Chunhua Shen, and Anton van den Hengel. "Deep anomaly detection with deviation networks." KDD 2019.

[7] Dosovitskiy, Alexey, et al. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale." ICLR 2021.

[8] Liu, Yinhan, et al. "Roberta: A robustly optimized bert pretraining approach." arXiv 2019.

Q2. The theoretical analysis of why the proposed overlap loss can work is in lack.

R2. We appreciate the advice on analyzing the theoretical basis of our proposed Overlap loss. Currently, we are working towards more theoretic proof from the perspective of information theory, analyzing the connection between minimizing overlap area in anomaly score distribution and entropy minimization. Since the entropy reflects the uncertainty when the model distinguishes between normal and abnormal samples, minimizing the overlap area to realize score distribution discrimination can be regarded as minimizing entropy with minimum efforts, as we do not further widen the distance between the score distributions of normal and abnormal samples.

The idea of the above theoretical proof can also be supported by several studies. [1] proves that reducing class distribution overlap in the feature space is equivalent to minimizing the model entropy, which is essentially motivated by the entropy minimization principle. [2] also explain the connection between DeepSVDD [3] and entropy minimization. However, we need to say that proving the theoretical validity of the proposed method is still very challenging, since both score distribution estimation and overlap area calculation steps are complicated during the gradient update process, while little AD literature provides detailed proof from this perspective. We provide the empirical evidence as a **first step** to prove the effectiveness of the proposed Overlap loss, which shows superiority over the existing AD loss functions. In future work, we will continue to explore the relationship between minimizing score distribution overlap and entropy minimization, and the reasons why the model inherits minor parameter changes and retains diversity in feature space during the training stage through the optimization theory.

[1] Chen, Yanbei, Xiatian Zhu, and Shaogang Gong. "Semi-supervised deep learning with memory." ECCV 2018.

[2] Ruff, Lukas, et al. "Deep Semi-Supervised Anomaly Detection." ICLR 2020.

[3] Ruff, Lukas, et al. "Deep one-class classification." ICML 2018.

Reviewer UVN9

Q1. The baseline includes several types of supervision methods. Training data includes labelled data and unlabelled data. The number of training data (labelled and unlabelled separately) used for each baseline method is not shown. The number of training data (labelled and unlabelled) used for each baseline method do be stated to make sure the comparison is fair.

R1. Thanks for pointing this out! We provide detailed information about the training set, where the total number of training samples N_train, the number of unlabeled samples (0) and labeled anomalies (1) corresponding to the ratio of labeled anomalies γ_l =5%, 10% and 20% are shown in parentheses, respectively. For the testing set, we show the total number of testing samples N_test, together with the number of ground-truth normal samples (0) and anomalies (1). Considering the expensive computational cost when comparing with a great number of baseline methods, we downsample those datasets with more than 10, 000 samples to 10, 000, while we find similar conclusions can be drawn on the full dataset. We will add this Table in the appendix of the updated paper versions.

For the experiments conducted in the main paper, we **ensured the same training and testing settings for all compared baselines**. Since unsupervised AD methods cannot directly leverage label information, we combined the labeled anomalies with unlabeled data to construct the validation set. This allowed us to tune the hyperparameters of these unsupervised methods, as tuning their hyperparameters on a small validation set often yields better performance than using the default settings [1].

Table: Dataset description (updated version)

	N	D	#anomalies	#anomaly ratio (%)	N_train (0/1)$\gamma_l = 5\%$	N_train (0/1)$\gamma_l = 10\%$	N_train (0/1)$\gamma_l = 20\%$	N_test (0/1)
ALOI	10000	27	315	3.15	7000 (6989/11)	7000 (6978/22)	7000 (6956/44)	3000 (2906/94)
annthyroid	7200	6	534	7.42	5040 (5022/18)	5040 (5003/37)	5040 (4966/74)	2160 (2000/160)
Cardiotocography	2114	21	466	22.04	1479 (1463/16)	1479 (1447/32)	1479 (1414/65)	635 (495/140)
fault	1941	27	673	34.67	1358 (1335/23)	1358 (1311/47)	1358 (1264/94)	583 (381/202)
http	10000	3	46	0.46	7000 (6999/1)	7000 (6997/3)	7000 (6994/6)	3000 (2986/14)
landsat	6435	36	1333	20.71	4504 (4458/46)	4504 (4411/93)	4504 (4318/186)	1931 (1531/400)
letter	1600	32	100	6.25	1120 (1117/3)	1120 (1113/7)	1120 (1106/14)	480 (450/30)
magic.gamma	10000	10	3548	35.48	7000 (6876/124)	7000 (6752/248)	7000 (6504/496)	3000 (1936/1064)
mammography	10000	6	230	2.3	7000 (6992/8)	7000 (6984/16)	7000 (6968/32)	3000 (2931/69)
mnist	7603	100	700	9.21	5322 (5298/24)	5322 (5273/49)	5322 (5224/98)	2281 (2071/210)
musk	3062	166	97	3.17	2143 (2140/3)	2143 (2137/6)	2143 (2130/13)	919 (890/29)
optdigits	5216	64	150	2.88	3651 (3646/5)	3651 (3641/10)	3651 (3630/21)	1565 (1520/45)
PageBlocks	5393	10	510	9.46	3775 (3758/17)	3775 (3740/35)	3775 (3704/71)	1618 (1465/153)
pendigits	6870	16	156	2.27	4809 (4804/5)	4809 (4799/10)	4809 (4788/21)	2061 (2014/47)
satellite	6435	36	2036	31.64	4504 (4433/71)	4504 (4362/142)	4504 (4219/285)	1931 (1320/611)
satimage-2	5803	36	71	1.22	4062 (4060/2)	4062 (4057/5)	4062 (4052/10)	1741 (1720/21)
shuttle	10000	9	669	6.69	7000 (6977/23)	7000 (6954/46)	7000 (6907/93)	3000 (2799/201)
skin	10000	3	2081	20.81	7000 (6928/72)	7000 (6855/145)	7000 (6709/291)	3000 (2376/624)
SpamBase	4207	57	1679	39.91	2944 (2886/58)	2944 (2827/117)	2944 (2709/235)	1263 (759/504)
speech	3686	400	61	1.65	2580 (2578/2)	2580 (2576/4)	2580 (2572/8)	1106 (1088/18)
thyroid	3772	6	93	2.47	2640 (2637/3)	2640 (2634/6)	2640 (2627/13)	1132 (1104/28)
vowels	1456	12	50	3.43	1019 (1018/1)	1019 (1016/3)	1019 (1012/7)	437 (422/15)
Waveform	3443	21	100	2.9	2410 (2407/3)	2410 (2403/7)	2410 (2396/14)	1033 (1003/30)
Wilt	4819	5	257	5.33	3373 (3364/9)	3373 (3355/18)	3373 (3337/36)	1446 (1369/77)
yeast	1484	8	507	34.16	1038 (1021/17)	1038 (1003/35)	1038 (967/71)	446 (294/152)

[1] Soenen, J., Van Wolputte, E., Perini, L., Vercruyssen, V., Meert, W., Davis, J., & Blockeel, H. (2021). The effect of hyperparameter tuning on the comparative evaluation of unsupervised anomaly detection methods. KDD 2021 Workshop.

Q2. The baseline of supervised learning method is not strong, which does not include tree-based method like xgboost or catboost for tabular data. It is better to evaluate the performance tree based method like xgboost or catboost with limited labelled data as a supervised baseline.

R2. We appreciate the thought of including stronger supervised baselines. Following your suggestion, we compared the proposed Overlap loss based AD methods with three tree-based ensemble methods, including Random Forest [1], XGBoost [2], and CatBoost [3], as shown in the Table below. We use the default hyperparameters in their corresponding Python packages.

Table: Comparison between the proposed Overlap loss based AD methods and tree-based ensemble models.

	γ_l =**5%**	γ_l =**10%**	γ_l =**20%**
MLP-Overlap	0.623±0.291	0.674±0.286	0.696±0.288
AE-Overlap	0.652±0.290	0.695±0.294	0.713±0.296
ResNet-Overlap	0.627±0.297	0.699±0.289	0.742±0.283
FTTransformer-Overlap	0.627±0.277	0.686±0.282	0.730±0.285
Random Forest	0.430±0.202	0.574±0.257	0.687±0.274
XGBoost	0.574±0.283	0.631±0.280	0.700±0.277
CatBoost	0.629±0.279	0.690±0.281	0.745±0.267

The results show that Overlap loss based AD methods are better than these tree-based supervised models, when only a handful of labeled anomalies are available during training stage. For example, when the ratio of labeled anomalies γ_l =5%, MLP-, ResNet- and FTTransformer-Overlap significantly outperform Random Forest and XGBoost, and achieve comparable performance to CatBoost. Besides, AE-Overlap achieves better AUC-PR performance than CatBoost w.r.t. γ_l =5%. When more labeled samples are provided, our proposed method remains competitive compared to the ensemble methods. Specifically, most of the Overlap loss based AD methods are still better than Random Forest and XGBoost when training data becomes more balance (say from γ_l =5% to γ_l =20%), where we observe that the AUC-PR of ResNet-Overlap (0.742) is close to that of CatBoost (0.745) w.r.t. γ_l =20%, due to that CatBoost often performs better with more labeled or balanced samples [5].

We need to point out that these results are generally satisfactory, since tree-based methods have been verified to prevail on the (AD) tabular data [4, 5, 6], especially more labeled samples are available in the training stage [5]. Besides, after conducting a detailed investigation on recent AD articles, we find few studies compare DL-based methods with tree-based ensemble methods, probably because it may be considered unfair to compare a single model to an ensembled model. We also want to thank you again. Inspired by your valuable idea, we plan to integrate our Overlap loss based AD methods with ensembling techniques like bagging or boosting in future work, as deep architecture will benefit more from ensembling and perform better than single models [4, 7].

[1] Breiman, Leo. "Random forests." Machine learning 2001.

[2] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." KDD 2016.

[3] Prokhorenkova, Liudmila, et al. "CatBoost: unbiased boosting with categorical features." NeurIPS 2018.

[4] Gorishniy, Yury, et al. "Revisiting deep learning models for tabular data." NeurIPS 2021.

[5] Han, Songqiao, et al. "Adbench: Anomaly detection benchmark." NeurIPS 2022.

[6] Grinsztajn, Leo, Edouard Oyallon, and Gael Varoquaux. "Why do tree-based models still outperform deep learning on typical tabular data?." NeurIPS 2022.

[7] Fort, Stanislav, Huiyi Hu, and Balaji Lakshminarayanan. "Deep ensembles: A loss landscape perspective." arXiv 2019.

Q3. The existing ablation study to show the performance of a randomly selected intersection point is conducted on the whole dataset, which including scenarios of one intersection point and multiple intersection points. However, the ratio of data samples with multiple intersection points is not shown, which weakens the persuasion. The ratio of data samples with multiple intersection points should be shown in Appendix 3. And it is better to evaluate the performance of randomly selected intersection point on dataset with multiple intersection points only.

R3. Thank you for pointing out this important question. As you suggested, we first record the occurrence of multiple intersection points during the training stage. Since the overlap area of score distribution is calculated in each training batch, we calculate the proportion of multiple intersections occurrence as the **count of multiple intersection points / training steps**, where training steps=epochs * (sample size / batch size). We report the results when bandwidth is equal to 0.1 and 1, respectively, as shown in the following Table.

Table: The proportion of multiple intersection points on different datasets.

	bw=0.1	bw=1.0
ALOI	99.98	18.26
annthyroid	100.00	9.58
Cardiotocography	100.00	2.60
fault	100.00	6.44
http	80.28	0.11
landsat	100.00	13.82
letter	98.00	4.86
magic.gamma	100.00	8.06
mammography	92.98	12.17
mnist	99.83	0.70
musk	44.47	1.07
optdigits	67.48	0.48
PageBlocks	99.86	14.11
pendigits	71.47	0.39
satellite	100.00	5.18
satimage-2	26.63	0.30
shuttle	34.87	0.43
skin	93.55	9.96
SpamBase	100.00	12.43
speech	21.47	0.58
thyroid	91.89	10.11
vowels	49.00	3.83
Waveform	52.06	0.88
Wilt	95.88	12.48
yeast	100.00	49.00

We observe that a smaller bandwidth (say bw=0.1) significantly increases the number of situations where multiple intersection points occur, since the KDE estimator with a smaller bandwidth would generate less smooth score distribution between normal and abnormal samples, thus generating more intersection points in these two distributions. **A corresponding question is: Is the ensemble strategy of multiple intersection points better when this situation becomes more often?**

In order to clarify this question, we perform experiments on the datasets whose proportion of multiple intersections occurrence is **greater than the average**, as shown in the Tables below. The results indicate that the ensemble strategy shows effectiveness when multiple intersection points occur more often, where MLP-Overlap-E outperforms MLP-Overlap w.r.t. γ_l =10% and γ_l = 20%. When bandwidth is equal to 1 (which is the default setting in the main paper), we still observe that the AUC-PR values between MLP-Overlap and MLP-Overlap-E are close to each other, since a larger bandwidth generates smoother score distribution and fewer intersection points.

Table: AUC-PR results of different intersection point selection strategies.

bw=0.1	γ_l =**5%**	γ_l =**10%**	γ_l =**20%**
MLP-Overlap	0.560±0.214	0.601±0.277	0.605±0.297
MLP-Overlap-E	0.559±0.211	0.610±0.280	0.621±0.305

bw=1.0	γ_l =**5%**	γ_l =**10%**	γ_l =**20%**
MLP-Overlap	0.555±0.208	0.586±0.228	0.570±0.249
MLP-Overlap-E	0.563±0.205	0.578±0.226	0.566±0.239

end