Neurocomputing

Normality-Calibrated Autoencoder for Unsupervised Anomaly Detection on Data Contamination

--Manuscript Draft--

Manuscript Number:	NEUCOM-D-23-06700R1
Article Type:	Regular article
Keywords:	Unsupervised anomaly detection; normality calibration; autoencoder; data contamination; data pollution
Corresponding Author:	Jongmin Yu University of Cambridge UNITED KINGDOM
First Author:	Jongmin Yu
Order of Authors:	Jongmin Yu
	Minkyung Kim
	Junsik Kim
	Hyeontaek Oh
Abstract:	Anomaly detection, or outlier detection, refers to identifying rare or abnormal instances or patterns within a dataset that deviate significantly from the expected or normal behaviour. Various methods have been proposed, but most assume that their training datasets take full, complete integrity. However, the innocent integrity of data is not easy to maintain in reality. Existing anomaly detection methods generally see given data as a single class and learn features that can represent it well, but this approach is very vulnerable to data contamination. This paper proposes a Normality-Calibrated Autoencoder (NCAE), which can boost anomaly detection performance on the contaminated datasets without any prior information or explicit abnormal samples in the training phase. The NCAE adversarially generates highly confident normal samples from a latent space with low entropy and leverages them to predict abnormal samples in a training dataset. NCAE is trained to minimise reconstruction errors in uncontaminated samples and maximise reconstruction errors in contaminated samples. The experimental results demonstrate that our method outperforms shallow, hybrid, and deep methods for unsupervised anomaly detection and achieves comparable performance compared with semi-supervised methods using labelled anomaly samples in the training phase.

Cover Letter

UNIVERSITY OF CAMBRIDGE

Departments of Applied Mathematics and Theoretical Physics
Centre for Mathematical Sciences, • Wilberforce Rd, Cambridge, CB3 0WA • United Kingdom

---

Dear reviewers and associated editors,

We resubmit the manuscript entitled "Normality-Calibrated Autoencoder for Unsupervised Anomaly Detection on Data Contamination" (NEUCOM-D-23-06700) for reconsideration by Pattern Recognition Letters.

We would like to thank you for your time and effort in reviewing our manuscript and for the opportunity to revise our manuscript. The constructive suggestions and thoughtful review comments have helped us significantly enhance the manuscript's quality.

We have revised our manuscript according to all of the reviewers' comments. Our responses to all reviewers' comments are attached as a separate file. The revisions in the manuscript are highlighted in blue, and the revised manuscript without highlights is also attached.

All the authors have read and approved the revised manuscript. We hope that our resubmission is now suitable for inclusion in Pattern Recognition Letters, and we look forward to hearing from you.

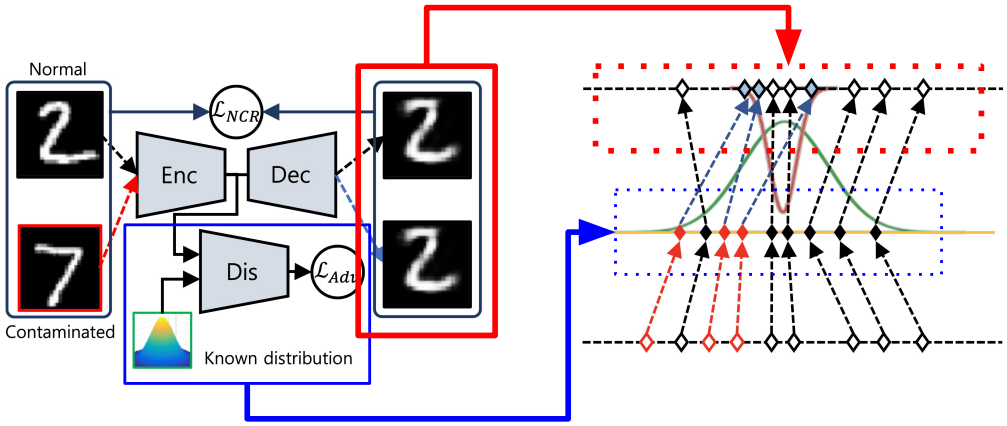
Warmest Regards,
Jongmin Yu

Graphical Abstract

Normality-Calibrated Autoencoder for Unsupervised Anomaly Detection on Data Contamination

Jongmin Yu, Minkyung Kim, Junsik Kim, Hyeontaek Oh

Using the generated data using a noise sampled from centre of gaussian distribution (Red distribution), we compute the normality-calibrated reconstruction (NCR) loss.



Using adversarial learning, we align the latent feature distribution into a known probabilistic model such as Gaussian distribution

Highlights

Normality-Calibrated Autoencoder for Unsupervised Anomaly Detection on Data Contamination

Jongmin Yu, Minkyung Kim, Junsik Kim, Hyeontaek Oh

- This paper proposes a Normality-calibrated autoencoder (NCAE). A new AD method is robust to data contamination without any prior knowledge about the data contamination.
- This paper presents generative adversarial learning for identifying contaminated data and a joint learning scheme for the NCAE. We propose adversarial learning to generate a high-confidence normal data sample and apply it to find contaminated data during the model training. Moreover, we propose an algorithm that efficiently optimises the proposed joint learning models for the NCAE.
- This paper provides comprehensive experimental results of AD on data contamination. We provide various ablation studies and performance comparisons with existing SOTA AD methods on data contamination.

Dear Editor-in-Chief,

We would like to thank all the editor and reviewers for their efforts in reviewing this manuscript. The manuscript has been revised according to all of the reviewers' comments. The reviewers' comments are shown in the boxes below, followed by their responses.

Response to Editor

E1: Reviewers' report has been received on the paper. These reports and the manuscript have been inspected, and we concur with the reported issues. While the reviewers find this paper interesting, severe concerns about the presentation clarity, lit review, and experiments have also been raised. All the review questions should be well addressed if a resubmission is planned. Please carefully revise it throughout and pay attention to the posed concerns. If these concerns and comments are not addressed thoroughly and carefully, it may halt the review process for your paper in this journal and result in its dismissal. Two review comments have been received for this manuscript.

Response: Thanks for the review comments and all the effort that the two reviewers and editor spent on this paper. We reviewed all comments very carefully and revised our manuscript. We tried to answer all of the review comments. Our answers to the reviewer's comments are described on the next pages. We tried to improve the language quality of our paper. Also, we conducted additional experiments using new datasets and new evaluation metrics. We believe that those efforts improve the quality of our paper significantly.

E2: This document will benefit significantly if the authors share some demo code in a public repository or on the web to help readers adopt the proposed method.

Response: We are currently in the process of patenting this work. Since there are some information security issues, we can not share the full source codes of this work. However, we are considering releasing the source code after it is accepted. The address of the source code repository will be added to the final version of this paper if it is accepted for publication.

E3: The authors must improve the linguistic quality of their manuscript; upon screening the manuscript, there are still a few language issues. This is highly important to ensure a proper understanding of your manuscript.

Response: Thanks for the review comments. The second reviewer also pointed out this linguistic issue. We acknowledge that our paper has multiple typos and grammatical errors. Also, we found out that our paper contains several duplicated and redundant contexts, which can cause misleading problems for the readers. During the revision process, we tried to revise the typos and grammatical errors and improve the readability of our manuscript. We believe that the current version of the paper is much more improved compared with the previous paper.

E4: Please ensure that there is a section (CreDiT) outlining the specific contributions made by each author.

Response: We do apologise that we did not attach the document to explain the contribution of each author. We submitted the document this time.

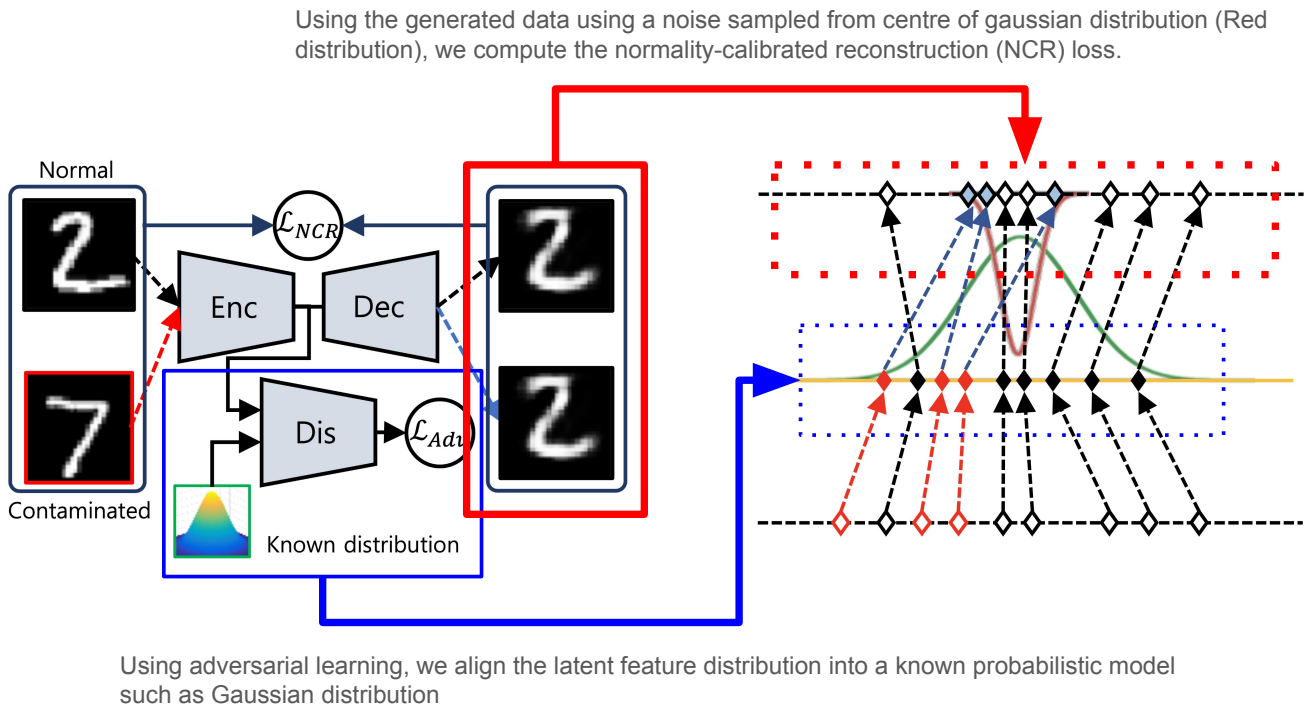
Response to Reviewer 1

R1-1: This manuscript presents a method to conduct unsupervised anomaly detection with contaminated training data. The topic is currently attracting much attention in the area of anomaly detection. The proposed method is generally reasonable.

Response: Thanks for the comments. Anomaly detection with contaminated training data is a very practical topic, so it can not be denied that developing robust data contamination anomaly detection methods is essential for various industries. We appreciate that the reviewer acknowledges your effort on this topic.

R1-2: The Graphical Abstract should be improved to highlight the main idea of the proposed method.

Response: Thanks for the review comments about the graphical abstract. We overlooked the importance of the graphical abstract, so we just copied and pasted an illustration on our manuscript. To provide clarified but easy-to-understand graphical information, we have revised the graphical abstract. We also included some explanations about the illustration. The newly updated graphical abstract is as follows:



R1-3: (para 4 of the introduction section) the motivation could be enhanced by extending the analysis and discussion of related works, such as training data augmentation (not limited to those with self-supervised)

Response: Thanks for the comments. We acknowledge that the current description of our motivation is not clear enough. In particular, when we state the motivation, we only analyse the self-supervised methods (finding some contaminated data using some geometric or probabilistic assumption). We enhance the description of our motivation by using analytical methodologies and other AD methods. We have revised some paragraphs in the introduction and the related work sections. You can find our revision from the 3rd paragraph of the introduction section. The revised paragraphs are as follows:

However, it is a more reasonable and practical hypothesis that the contamination ratio cannot be usually estimated. Also, for the semi-supervised approaches [1, 2, 3, 4], we cannot guarantee that the prepared abnormal data will cover all other unobserved data anomalies. Moreover, even if we successfully find out the contamination ratio of a particular domain or we finally find some specific abnormal data which can cover all other data anomalies for the domain, they can only be applied to derive AD models in that particular domain, not others. Consequently, developing more generalised solutions for AD robust to contaminated datasets is still very challenging but important.

To address this issue, AD methods based on contamination sample prediction using geometric distance measurements [5, 6, 7] have been proposed. These methods assume that contaminated data is always distributed far away from the data distribution's centre or in the highest entropy space. Then, the methods sorted the contaminated data in ascending or descending order by distance from the centre of the data distribution or by entropy, and they filtered out data that was contaminated by a specific percentage [5, 6, 7]. However, as shown in Fig. 1, if a training dataset is highly contaminated (like over 10% of data samples on a dataset), the contaminated samples can also form a low entropy space by themselves. As a result, the development of an AD method which does not require prior information about data anomalies and also does not take strong geometric assumptions in finding contaminated samples is essential.

Not only the above paragraphs, but we have also added additional descriptions to clarify our motivation for the related work section (Section 2).

R1-4: How is the entropy of each sample calculated in fig.1?

Response: Thanks for the comments. Previously, we did not explain how to compute entropy. We acknowledge that it can mislead the readers and cause misunderstanding about our methods. We estimate the entropy of each sample using the probability computed based on kernel density estimation (KDE). We have added an explanation about how we compute the entropy to the caption in Fig. 1. The newly added description is as follows:

The entropy of each sample is computed using the probability estimated by the kernel density estimation (KDE).

R1-5: For the architectural of the proposed NCAE, how is it different with current variants of AE(such as denoising AE, e.t.c.)

Response: Thanks for the comments. The architectural details of the proposed NCAE, particularly the autoencoder parts (the encoder and decoder), are the same as the normal AE. We add an additional discriminator to apply adversarial learning in training the normality-calibrated autoencoder. This structural difference is represented in Fig. 2. However, we acknowledge that the contents of the current paper may not be enough to provide a clarified description of the architectural details of the NCAE. We have added an explanation of the architectural details of the proposed NCAE to the 6th paragraph of the introduction section. The newly added explanations are as follows:

*Fig.2 shows the architectural difference between an autoencoder (AE) and the proposed NCAE. The NCAE has a structure that combines an AE and a discriminator for applying adversarial learning. Data is compressed into low-dimensional latent features through the encoder, and the distribution of these latent features is induced into a specific data distribution through adversarial learning (See $\mathcal{L}_{Adv}$ in Fig. 2(b)). This process reduces the uncertainty of the data distribution and minimises the blind spot [8] on the latent feature space. Based on the decoder part of the AE and the specific data distribution used for adversarial learning, samples of the region with the highest probability (*a.k.a. *the lowest entropy) are generated. The generated samples are used to distinguish normal samples from contaminated samples. After identifying the contaminated samples, NCAE calibrates the normality of an AD model by maximising the reconstruction error of the found samples.*

R1-6: for formula 8, there should be a couple of hyper-paras in it (might be 4 hyper-paras). What is their influence to the detection performance?

Response: Thanks for the comments. We only have two hyper-parameters σ and τ involved in learning the NCAE. The other parameter is the threshold β for identifying data anomalies (Eq. 9). We provide ablation studies to monitor the performance trends depending on the values of those hyper-parameters in Section 5. However, we acknowledge that the current description of equation 8 may be misleading, so we have provided a more detailed explanation of equation 8 under the equation. The newly added contents are as follows:

The objective function for joint learning of the entire components of our method is as follows:

$$\begin{aligned}\min_{f,g}\max_{D_l,D_s} \quad & \underbrace{\mathbb{E}_{x\sim p_{\mathcal{X}^{\mathrm{N}}}}||x-g\circ f(x)||^2+\mathbb{E}_{x\sim p_{\mathcal{X}^{\mathrm{C}}}}||x-\bar{x}'||^2}_{\text{(a)}} \\ & +\underbrace{\mathbb{E}_{\omega\sim N(\mu_{\mathcal{Z}},I_d)}[\log D_l(\omega)]+\mathbb{E}_{x\sim P_{\mathcal{X}}}[\log\left(1-D_l(f(x))\right)]}_{\text{(b)}} \\ & +\underbrace{\mathbb{E}_{x\sim P_{\mathcal{X}}}[\log D_s(x)]+\mathbb{E}_{\omega'\sim N(\mu_{\mathcal{Z}},\sigma I_d)}[\log(1-D_s(g(\omega'))]}_{\text{(c)}},\end{aligned} \tag{1}$$

where D_l and D_s define the discriminators for the latent features and samples, respectively. $\bar{x}'$ is a prototype of the generated high-confident normal samples $g(\acute{\omega})=\acute{x}$. It is defined by averaging the generated samples as follows: $\bar{x}'=\mathbb{E}_{x'\sim g(N(\mu_{\mathcal{Z}},\sigma I_d))}(\acute{x})$. $\mu_{\mathcal{Z}}$ and I_d denote the averaged latent features and their covariance matrix represented by an identity matrix. w indicates noise signals sampled from the Gaussian distribution $N(\mu_{\mathcal{Z}},I_d)$.

R1-7: The related work is not adequately investigated, including those anomaly detection methods in the unsupervised/semi-supervised. Some direct related work should be discussed and compared, such as 'Feature Encoding with AutoEncoders for Weakly-supervised Anomaly Detection, IEEE Transactions on Neural Networks and Learning Systems, 2022, 33(6): 2454 - 2465.' and 'FedTADBench: Federated Time-Series Anomaly Detection Benchmark, IEEE International Conference on High Performance Computing & Communications (HPCC),2022 Unsupervised Deep Learning for IoT Time Series, IEEE Internet of Things Journal, 2023.'

Response: We appreciate that the reviewers recommended several great related studies. As the reviewer commented, we found out that the current related work section does not provide enough background information for the AD literature. We have conducted additional investigations and added more comprehensive knowledge about AD studies. Not only the studies reviewer mentioned, but we also found several recent studies [9, 10, 11, 12] related to our works and added those into the related work section. The related work has been rewritten to consider the newly added studies.

R1-8: For formula 9, it seems that beta is hard to determined

Response: Thanks for the comments. The β is the threshold and should be decided manually to distinguish whether a given data x is abnormal or not. The AD approach using a manually decided threshold is a common decision process for unsupervised AD because it assumes that data anomalies can not be provided during the training process. The bigger β, the more comprehensive abnormality will be detected. The smaller β, the more precise detection results (less false positives) will be provided. The current description can not provide an understanding of those natural unsupervised AD processes using the threshold. To provide a more clarified description, we have added the following explanation to the last paragraph of section 3.3, as follows:

The AD detection performance depends on the value of β. When β is too small, the AD results will be more precise, but they will be too specific, and many anomalies will be ignored. Hence, if the β is too large, many anomalies will be detected, but it contains many false positives. In other words, The bigger β, the more comprehensive abnormality will be detected. The smaller β, the more precise detection results (less false positives) will be provided. Considering

Table 1: Performance comparison on unsupervised anomaly detection in terms of various contamination ratios ρ. The AUC value is used to evaluate the performance. MNIST, Fashion-MNIST, and CIFAR-10 datasets are used for the comparison. The **bolded** figures indicate the best performances.

Dataset	ρ	OC-SVM	IF	KDE	CAE	Deep SVDD	SSAD	SS-DGM	Deep SAD	Classification	DAGMM	LatentOut	NCAE
MNIST	.00	96.0±2.9	85.4±8.7	95.0±3.3	92.9±5.7	92.8±4.9	**97.9±1.8**	92.2±5.6	96.7±2.4	94.5±4.6	91.7±6.2	96.5±3.2	94.0±4.2
	.01	94.3±3.9	85.2±8.8	91.2±4.9	91.3±6.1	92.1±5.1	96.6±2.4	92.0±6.0	95.5±3.3	91.5±5.9	90.4±5.2	93.9±4.8	**97.2±2.8**
	.05	91.4±5.2	83.9±9.2	85.5±7.1	87.2±7.1	89.4±5.8	93.4±3.4	91.0±6.9	93.5±4.1	86.7±7.4	88.5±9.2	86.4±7.2	**95.0±4.9**
	.10	88.8±6.0	82.3±9.5	82.1±8.5	83.7±8.4	86.5±6.8	90.7±4.4	89.7±7.5	91.2±4.9	83.6±8.2	84.2±6.2	81.6±8.3	**92.6±5.7**
	.20	84.1±7.6	78.7±10.5	77.4±10.9	78.6±10.3	81.5±8.4	87.4±5.6	87.4±8.6	86.6±6.6	79.7±9.4	81.5±7.3	73.7±9.3	**89.8±7.4**
F-MNIST	.00	92.8±4.7	91.6±5.5	92.0±4.9	90.2±5.8	89.2±6.2	**94.0±4.4**	71.4±12.7	90.5±6.5	76.8±13.2	87.6±7.2	91.2±6.2	91.5±8.3
	.01	91.7±5.0	91.5±5.5	89.4±6.3	87.1±7.3	86.3±6.3	92.2±4.9	71.2±14.3	87.2±7.1	67.3±8.1	81.5±5.5	86.2±7.6	**94.5±4.7**
	.05	90.7±5.5	90.9±5.9	85.2±9.1	81.6±9.6	80.6±7.1	88.3±6.2	71.9±14.3	81.5±8.5	59.8±4.6	74.1±9.3	84.2±6.3	**92.4±7.2**
	.10	89.5±6.1	90.2±6.3	81.8±11.2	77.4±11.1	76.2±7.3	85.6±7.0	72.5±15.5	78.2±9.1	56.7±4.1	69.2±5.2	67.3±5.2	**91.5±5.7**
	.20	86.3±7.7	88.4±7.6	77.4±13.6	72.5±12.6	69.3±6.3	81.9±8.1	70.8±16.0	74.8±9.4	53.9±2.9	65.2±11.4	61.3±12.7	**88.9±9.2**
CIFAR-10	.00	63.8±9.0	59.9±6.7	56.1±10.2	56.2±13.2	60.9±9.4	73.3±8.4	50.8±4.7	**77.9±7.2**	63.5±8.0	78.2±7.3	75.4±5.2	73.2±7.3
	.01	63.8±9.3	59.9±6.7	56.3±10.4	56.2±13.1	60.5±9.4	72.8±8.1	51.1±4.7	76.5±7.2	72.9±7.3	66.2±7.2	74.2±6.2	**79.3±3.9**
	.05	62.6±9.2	59.6±6.4	55.6±10.5	55.7±13.3	59.6±9.8	71.5±8.2	50.1±2.9	74.0±6.9	62.2±8.2	69.3±6.4	71.0±6.8	**78.2±3.2**
	.10	62.9±8.2	59.1±6.6	54.9±11.1	55.4±13.3	58.6±10.0	69.8±8.4	50.5±3.6	71.8±7.0	60.6±8.3	64.2±10.2	69.2±9.7	**76.7±5.4**
	.20	61.9±8.1	58.3±6.2	54.2±11.1	54.6±13.3	57.0±10.6	67.9±8.1	50.1±1.7	68.5±7.1	58.5±6.7	58.2±5.2	66.2±8.2	**71.1±6.2**

this evaluation property, we provide the evaluation metric by observing the performance trend regarding the threshold value change instead of the AD performance at a single point. More detailed information is provided in Section 5.

R1-9: for the experiment setting, the reviewer suggest to refer Feature Encoding with AutoEncoders for Weakly-supervised Anomaly Detection, IEEE Transactions on Neural Networks and Learning Systems, 2022, 33(6): 2454 - 2465.

Response: Thanks for suggesting new work. We have cited the mentioned work for the experiment setting section.

R1-10: the datasets used in the experiments are a little weak. More datasets are strongly suggested.

Response: Thanks for the review comments. We use two additional datasets from OODs and also employ CIFAR-10 datasets. For the OODs, we employ the 'Shuttle' and 'Mammography' datasets. In particular, the CIFAR-10 dataset has 60,000 32x32 colour images in 10 classes, with 6,000 images per class. There are 50000 training images and 10000 test images. The dataset is divided into five training batches and one test batch, each with 10000 images. We have conducted additional unsupervised AD experiments using the above dataset and updated the tables. Also, as the tables have been updated, we have added analysis of the updated results. The updated tables using additional datasets and the corresponding analysis are as follows:

Table 1 shows the quantitative performance comparison depending on the contamination ratio ρ. In the comparison using the MNIST dataset, the proposed NCAE achieves the best performances except when the dataset is not contaminated ($\rho = 0.0$). Even compared with semi-supervised approaches (SSAD and Deep SVDD) [4, 13], which use explicit anomaly samples in the training phase, the NCAE shows outstanding performances. This trend is also shown in the performance comparison using the Fashion-MNIST and CIFAR-10 datasets. The NCAE produces the AUC of 94.5 and 88.9 for the Fashion-MNIST dataset with 1% and 20% contamination ratios, respectively. Also, it achieves the AUC of 79.3 and 71.1 for the CIFAR-10 dataset with 1% and 20% contamination ratios, respectively. Those figures perform best among the listed methods when a dataset is contaminated. Compared with other methods, which degrade their performance significantly when the contamination ratio is increased, the NCAE AD performances are relatively robust to the contamination ratios.

Table 2: Performance comparison using AUC-ROC values on unsupervised anomaly detection using ODDs. The **bolded** figures indicate the best performances.

Dataset	Satellite	Cardio	Thyroid	Satimage-2	Shuttle	Mammography
OC-SVM	95.1±0.2	98.5±0.3	98.3±0.9	99.4±0.8	99.4±0.9	71.4±9.7
IF	94.2±0.2	91.4±1.1	95.2±0.3	99.2±1.2	93.4±0.9	91.4±3.6
KDE	66.7±5.8	99.6±1.7	91.6±2.3	99.5±0.2	87.2±7.4	67.2±12.5
DeepSVDD	79.8±4.1	84.8±3.6	72.0±9.7	98.3±1.4	98.7±0.2	97.4±0.5
SSAD	96.2±0.3	98.8±0.3	97.9±1.9	99.9±0.1	98.9±0.4	92.6±2.4
SS-DGM	95.7±0.1	95.2±1.3	95.8±0.7	99.2±0.2	97.5±0.4	96.4±2.3
AE	73.5±9.4	89.6±6.7	95.4±2.7	99.1±1.7	95.6±0.7	82.7±9.4
Deep SAD	91.5±1.1	95.0±1.6	**98.6±0.9**	99.9±0.1	99.3±0.1	93.0±0.5
Classification	87.2±2.1	83.2±9.6	97.8±2.6	99.9±0.1	98.3±0.2	79.5±15.8
DAGMM	73.9±3.1	88.5±3.3	96.4±0.7	99.7±0.2	99.0±0.2	75.9±7.9
LatentOut	92.4±0.5	99.0±0.1	98.1±0.3	99.8±0.1	99.7±0.1	94.2±0.7
NCAE	**97.3±0.2**	**99.2±0.2**	98.5±0.1	**99.9±0.1**	**99.9±0.1**	**98.7±0.2**

R1-11: the compared methods seems to be weak

Response: We investigated recent unsupervised AD methods and found two benchmarks: DAGMM [14] and LatentOut [15]. We have conducted additional experimental results and put the results in Table 2, Table 3, and the new Table 4 (Table for AUC-PR). Also, we provide additional analysis for the experimental results corresponding to those new methods. Section 6 has been updated to reflect the updated experimental results. We believe that those efforts improve our paper.

R1-12: Both AUC-ROC and AUC-PR should be used in the performance evaluation

Response: Thanks for the review comments. We employed the AUC-ROC, because it is the most commonly used evaluation metric to evaluate the performance of unsupervised AD and outlier detection studies. However, as the reviewer mentioned, we acknowledge that we should provide more diverse evaluation metrics to provide a comprehensive and deep insight into our work. As a result, based on the ODDs, we have conducted additional experiments to measure the AUC-PR. Table 4 contains the experimental results. Also, since Table 4 has been updated, We have added an additional description for it to the 3rd paragraph of section 6.2. The updated contents are as follows:

Table 3: Performance comparison using AUC-PR values on unsupervised anomaly detection using ODDs. The **bolded** figures indicate the best performances.

Dataset	Satellite	Cardio	Thyroid	Satimage-2	Shuttle	Mammography
OC-SVM	72.5±5.6	82.6±6.2	78.9±9.8	94.8±1.5	96.2±2.6	43.7±9.5
IF	64.8±8.2	83.2±7.5	61.6±11.6	89.5±3.2	87.2±6.8	28.6± 15.7
KDE	48.2±17.8	86.2±4.3	65.4±9.4	90.3±6.7	73.6±6.1	28.4±3.7
DeepSVDD	64.3±4.2	81.6±2.4	51.4±4.2	91.4±1.2	92.6±1.2	52.6±1.2
SSAD	70.2±1.8	89.4±0.7	89.5±0.5	93.2±7.9	97.2±0.5	42.5±4.5
SS-DGM	69.5±22.6	42.5±9.8	92.6±0.3	94.2±0.5	91.4±2.3	48.2±1.8
AE	41.7±13.9	60.2±11.4	75.2±7.1	84.6±2.6	90.3±0.7	18.2±7.6
Deep SAD	83.3±0.7	96.8±0.8	92.3±0.4	96.2±0.2	95.2±0.3	59.1±4.6
Classification	90.2±1.1	92.3±0.7	90.2±2.6	94.1±0.5	96.2±0.2	60.3±1.1
DAGMM	68.4±2.1	62.3±8.6	87.5±2.7	95.0±0.4	95.9±1.3	12.5±7.0
LatentOut	85.4±0.9	**99.5±0.1**	94.7±0.2	96.1±0.3	96.5±0.1	**64.6±1.2**
NCAE	**90.4±0.5**	99.3±0.2	**95.1±0.6**	**96.4±0.2**	**98.2±0.4**	64.2±0.7

These results show that the NCAE's AD performance is slightly lower than the Deep SAD, but the NCAE's performance is much more stable. For AUC-PR results using the Satellite dataset, LatentOut [15] achieve partially better performances in Cardio and Mannography datasets. LatentOut achieved a 99.5 and 64.6 of AUC-PR, respectively. This result is 0.2 and 0.4 higher than the proposed NCAE. For the variation, the NCAE achieves better variation. The NCAE usually shows smaller variation, which can be interpreted as the performance of the NCAE fluctuating less than other methods.

R1-13: there should be an ablation study on the adoption of the specific designed NCAE architectural

Response: Thanks for the comments. In selecting AE architecture, the AD performance of deep learning-based methods can be changed depending on the number of layers and the kernel size of each convolutional layer. It is deserved that if we choose a deeper and larger network, better performance will be achieved. Based on this principle, since using a deeper layer can be thought of as a way to boost performance, the fairest way for AD research is to employ a well-known and comprehensively used network model. In this work, for the experiments using MNIST and Fashion-MNIST, we employ LeNet-based autoencoder, which was employed for various deep learning-based AD studies [4, 13, 16, 17]. Also, for the experiments using OOD datasets, we use three neural-layer-based encoders and decoders with 32-16-8 units. Nevertheless, We acknowledge that the current description for explaining the architectural design does not provide the above background. To improve the understanding of those nature, we have added additional explanations about architectural design to the 'Implementation' part of section 4, as follows:

Our method can be thought of as one of deep neural network-based AD methods; accordingly, the AD performance of deep learning-based methods can be changed depending on the number of layers and the kernel size of convolutional layers. To this principle, using a deeper layer can be considered a way to boost performance, which is very unfair to other AD methods. The fairest way for deep learning-based AD research is to employ a well-known and comprehensively used network model. In this work, we employ a LeNet-based autoencoder, which was employed for various deep learning-based AD studies [4, 13, 16, 17] on MNIST and Fashion-MNIST datasets, where each convolutional module consists of a convolutional layer followed by leaky ReLU activation functions with leakiness of 0.1. On the outlier detection dataset (OODs) benchmark using Cardio, Satellite, Satimage-2 and Thyroid, we employ standard MLP feed-forward AE structure presented by Ruff et al. *[4]. The MLP for the encoder and decoder is defined by a 3-layer neural network with 32-16-8 units.*

Response to Reviewer 2

R2-1: The paper proposes an anomaly detection method, i.e., normality-calibrated autoencoder, for contaminated datasets. And a contaminated data finding method with generative adversarial learning is proposed. The biggest issue with this paper is that the motivation is not solid enough.

Response: Thanks for the review comments. The first reviewer also mentioned this issue. We acknowledge that the current description of the motivation is not enough to provide our purpose. We have enhanced the description of the research motivation. We have added additional content in the introduction section. The newly added descriptions are as follows:

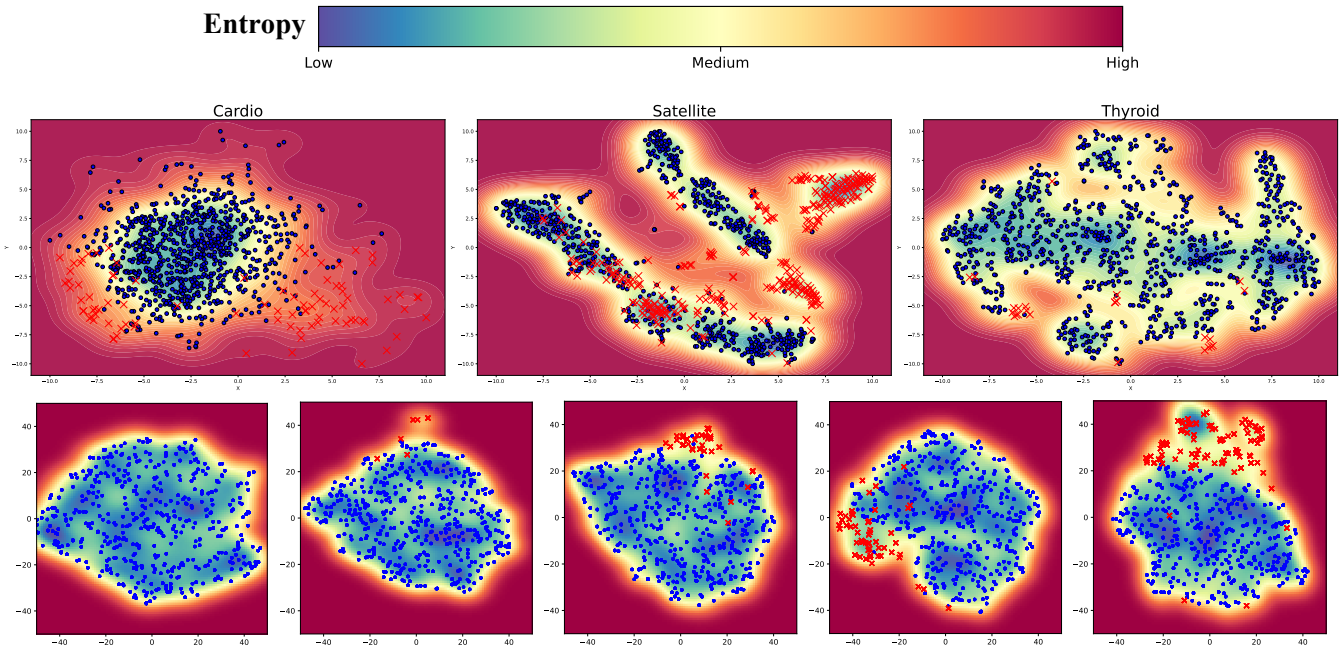
However, it is a more reasonable and practical hypothesis that the contamination ratio cannot be usually estimated. Also, for the semi-supervised approaches [1, 2, 3, 4], we cannot guarantee that the prepared abnormal data will cover all other unobserved data anomalies. Moreover, even if we successfully find out the contamination ratio of a particular domain or we finally find some specific abnormal data which can cover all other data anomalies for the domain, they can only be applied to derive AD models in that particular domain, not others. Consequently, developing more generalised solutions for AD robust to contaminated datasets is still very challenging but important.

To address this issue, AD methods based on contamination sample prediction using geometric distance measurements [5, 6, 7] have been proposed. These methods assume that contaminated data is always distributed far away from the data distribution's centre or in the highest entropy space. Then, the methods sorted the contaminated data in ascending or descending order by distance from the centre of the data distribution or by entropy, and they filtered out data that was contaminated by a specific percentage [5, 6, 7]. However, as shown in Fig. 1, if a training dataset is highly contaminated (like over 10% of data samples on a dataset), the contaminated samples can also form a low entropy space by themselves. As a result, the development of an AD method which does not require prior information about data anomalies and also does not take geometric solid assumptions in finding contaminated samples is essential.

This paper presents a Normality-Calibrated Autoencoder (NCAE), which is robust to the training dataset contamination. Our key idea for the NCAE is to adversarially generate highly confident normal samples from a low entropy feature space and then contrastively compare the generated samples with the input samples to estimate the contamination score. We pay attention to the fact that if adversarial learning inputs data biased to a specific class in the process of optimizing the generator and discriminator, the generator repeatedly generates only specific data instead of generating various data.

R2-2: The finds of Fig 1 which states highly contaminated samples can form a low entropy space by themselves should be justified more clearly. The contamination in Fig 1 is simulated manually and the are collected from training samples. However, is this situation consistent with the real cases? The contaminated datasets always demonstrate an unpredictable distribution. The following method is highly related to this assumption that the contaminated data follows the similar distribution with normal data or are in low entropy latent space. If the authors want the latter method to be reasonable, then this motivation part needs to be more solid.

Response: Thanks for the review comments. We put Fig 1 to justify our hypothesis that when the contamination ratio is very high (like over 15%, abnormal samples can be positioned in some low-entropy space. However, we have overlooked that simulated results can be different from the real situation. To improve our motivation, we have added three additional visualisation results using an outlier detection dataset (OODs) to Fig. 1. The OODs consist of several subsets containing data captured from various domains with a real situation. For example, the 'Cardio' set is composed of data consisting of measurements of fetal heart rate (FHR) and uterine contraction (UC) features on cardiotocograms classified by expert obstetricians. We believe that this revision enhances our hypothesis and motivation. The newly added contents are as follows:



(a) 0% (b) 1% (c) 5% (d) 10% (e) 20%

Figure 1: Visualisation of entropy and distribution of latent features of the various datasets considering different contamination ratios. The first row indicates the visualisation of the outlier detection datasets. The ratios of the data contamination are 9.6%, 31.6%, and 2.5%, respectively. The second row illustrates the visualisation results using a simulated dataset based on the MNIST dataset, considering various contamination ratios. (a) 0% (No contamination), (b) 1%, (c) 5%, (d) 10%, and (e) 20%. The samples on the '5' class on the MNIST dataset are used as normal (blue dots), and contaminated samples (red x-marks) are randomly picked from the training samples of the remaining classes. The entropy of each sample is computed by the probability estimated by the kernel density estimation (KDE). The 500 samples are randomly picked for visualisation. When a dataset is highly contaminated (*i.e.* contamination ratio over 10%), contaminated samples are also located in a low entropy region.

To address this issue, AD methods based on contamination sample prediction using geometric distance measurements [5, 6, 7] have been proposed. These methods assume that contaminated data is always distributed far away from the data distribution's centre or in the highest entropy space. Then, the methods sorted the contaminated data in ascending or descending order by distance from the centre of the data distribution or by entropy, and they filtered out data that was contaminated by a specific percentage [5, 6, 7]. However, as shown in Fig. 1, if a training dataset is highly contaminated (like the Satellite dataset, in which samples of 31.6% are contaminated or over 10% of data

samples on a simulated dataset), the contaminated samples can also form a low entropy space by themselves, even when the contamination ratio is not high enough, abnormal samples can be positioned some low-entropy space. As a result, the development of an AD method which does not require prior information about data anomalies and also does not take geometric solid assumptions in finding contaminated samples is essential.

R2-3: There are a lot of format issues. For example, the title of section 2 should be "Related Work" and there is no legends in some figures. And the format of algorithm 1 need to be improved.

Response: Thanks for the review comments. First of all, we apologise for the formatting issues in our manuscript. As the reviewer mentioned, we have revised the title of section 2 to 'Related works'. Also, we checked all figures and found no legends in Figure 2. We have added legends to the figure 2. Also, we have added extra explanation to the figure 2 to improve readability. About algorithm 1, we have tried to simplify and clarify the process. You can see the revised version of algorithm 1 below. We did our best effort to improve the quality of our manuscript. We believe that those revisions have improved our paper. The updated algorithm and Fig 2 are as follows:

Algorithm 1 Repetitive generation-feedback algorithm for NCAE

Require: The training epoch T, training step S, the batch size m, the contaminated ratio τ, the learning rate γ, and the variance controller σ for generating normal samples

Initialise • $\mu_{\mathcal{Z}}$: $\mu_{\mathcal{Z}} = \frac{1}{n}\sum_{i=1}^{n} z_i$

for $t = 1$ to T epochs **do**

for $s = 1$ to S training steps **do**

• Update the high confident normal sample generation components: $\{g, \mathcal{D}_s\}$

• Sample $\{\acute{\omega}_i\}_{i=1:m} \sim N(\mu_{\mathcal{Z}}, \sigma I_d)$ and $\{x_i\}_{i=1:m} \sim P_{\mathcal{X}}$

• Update the decoder g and the discriminator D_s using following objective:

$$\min_{g}\max_{D_s} \mathbb{E}_{x\sim P_{\mathcal{X}}}[\log D_s(x)] + \mathbb{E}_{\omega'\sim N(\mu_{\mathcal{Z}}, \sigma I_d)}[\log(1 - D_s(g(\omega'))].$$

• Generate high-confidence normal samples $\{\acute{x}_1, \ldots, \acute{x}_m\}$: $\acute{x}_i = g(\acute{\omega}_i)$.

• Predict contaminated samples

· Construct feature dictionary $\mathcal{M} = \{\acute{z}_1, \ldots, \acute{z}_m\}$, where $\acute{z}_i = f(\acute{x}_i)$.

· Compute the contamination score c: $c_i = \frac{1}{m}\sum_{j=1}^{m} f(x_i) \cdot \acute{z}_j^{\mathrm{T}}$

· Predict contamination samples $\mathcal{X}_{\mathrm{C}}$ as follows:

$$\mathcal{X}^{\mathrm{C}} = \{x_t\}_{t\in C[1:\lceil \tau m\rceil]} \quad w.r.t., C = \arg\operatorname*{sort}_{i} c_i, \quad w.r.t., 1 < i < m.$$

• Update f, g and $\mathcal{D}_l$.

• update the f, g, and D_l using $\{\omega_i\}_{i=1:m}$, $\{x_i\}_{i=1:m}$, $\mathcal{X}^{\mathrm{N}}$, and $\mathcal{X}^{\mathrm{C}}$ with the following objective:

$$\begin{aligned}\min_{f,g}\max_{D_l} \quad & \mathbb{E}_{x\sim\mathcal{X}^{\mathrm{N}}}||x - g\circ f(x)||^2 + \mathbb{E}_{x\sim\mathcal{X}^{\mathrm{C}}}||x - \bar{x}'||^2 \\ & + \mathbb{E}_{\omega\sim N(\mu_{\mathcal{Z}}, I_d)}[\log D_l(\omega)] + \mathbb{E}_{x\sim P_{\mathcal{X}}}[\log(1 - D_l(f(x)))]\end{aligned}$$

• Update $\mu_{\mathcal{Z}}$: $\mu_{\mathcal{Z}} = \mu_{\mathcal{Z}} - \gamma\frac{1}{m}\sum_{i=1}^{m}(\mu_{\mathcal{Z}} - f(x_i))$

end for

end for

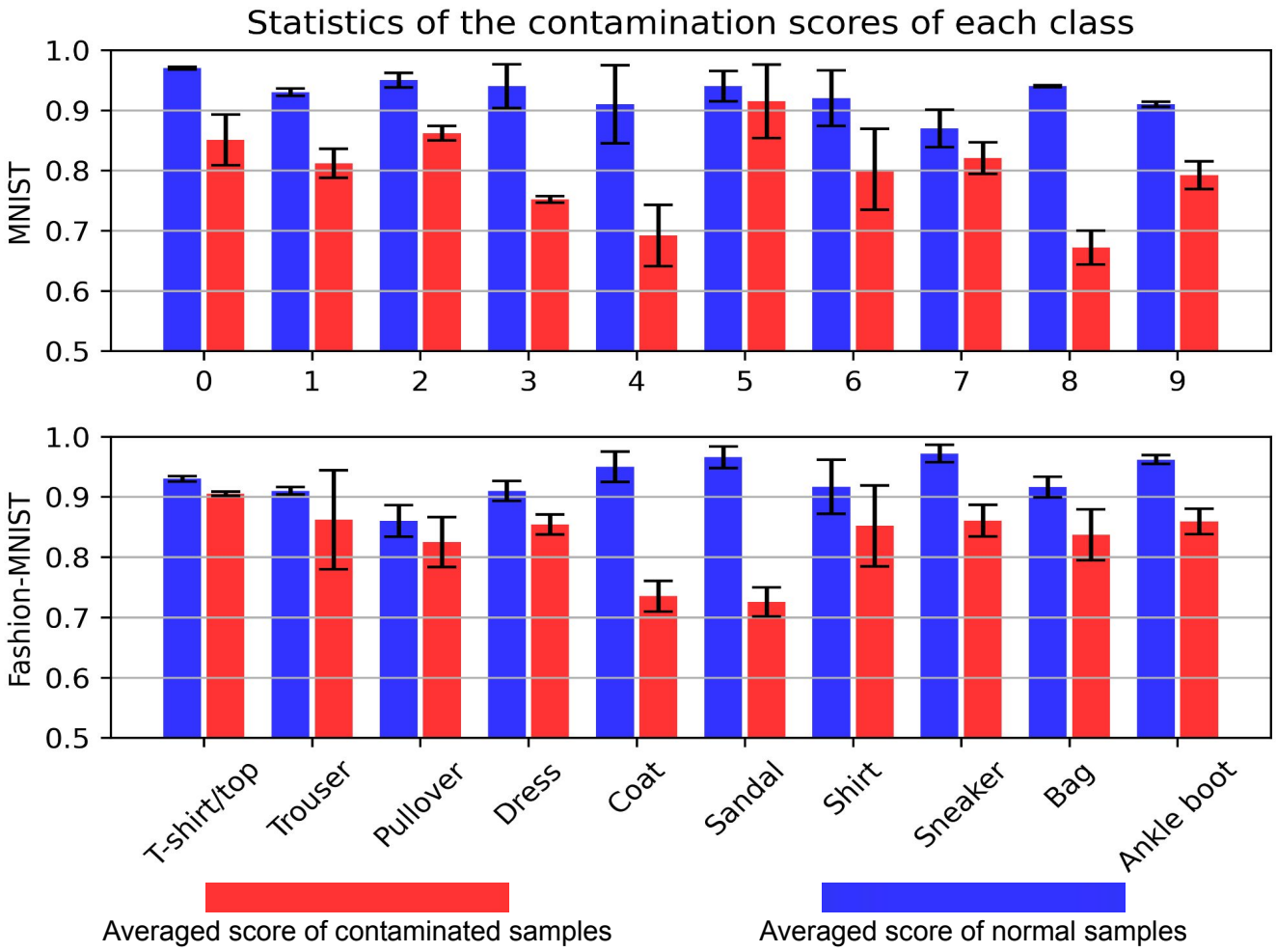


Figure 2: The means and variances of the contaminated scores for 100 times repeated experiments. The upper illustrates the averaged contamination score of contaminated and normal samples on the MNIST dataset. The lower illustrates the averaged scores on the Fashion-MNIST dataset. The black lines of each bar show a variation of the score.

R2-4: There are some grammar errors in the paper.

Response: Thanks for the comments. We have revised our paper very carefully and found some grammar errors. We revised those errors. Also, to improve the readability, we have deleted some duplicate and redundant content, which can make some things misleading. The list that we fix typos and revised contents are as follows:

About the typo(s) and language(s)

- (1) Sec 1, Page 1: *'assumed'* has been revised to *'assume'*
- (2) Sec 1, Page 1: *'are'* has been revised to *'is'*
- (3) Sec 1, Page 1: *'farthest'* has been revised to *'far'*
- (4) Sec 1, Page 1: *'only can'* has been revised to *'can only'*
- (5) Sec 1, Para 3: *'we conduct experiments of AD on contaminated datasets using various datasets having diverse contamination ratios. '* has been revised to *'we conduct AD experiments on contaminated datasets using diverse contamination ratios.'*.
- (6) Sec 2, We fixed formatting issues.
- (7) Sec 3 Page 9, *'low'* has been revised to *'a low'*.
- (8) Sec 3, Page 10, *'However, this formulation is intractable in applying for optimising an AE directly because there is no bound in maximising the error terms.'* has been revised to .*'However, this formulation is intractable when applied to optimise an AE directly because there is no bound to maximise the error terms.'*.
- (9) Sec 3, Page 12, *'to be'* has been revised to *'being'*.
- (9) Sec 3, Page 13, *'high'* has been revised to *'higly'*.

In addition to the above revision, we revise some typos and grammatical errors. Also, we tried to arrange our manuscript to improve readability.

References

[1] Jigang Wang, Predrag Neskovic, and Leon N Cooper. Pattern classification via single spheres. In *International Conference on Discovery Science*, pages 241–252. Springer, 2005. 3, 8

[2] Yi Liu and Yuan F Zheng. Minimum enclosing and maximum excluding machine for pattern description and discrimination. In *ICPR*, pages 129–132, 2006. 3, 8

[3] Nico Görnitz, Marius Kloft, Konrad Rieck, and Ulf Brefeld. Toward supervised anomaly detection. *Journal of Artificial Intelligence Research*, 46:235–262, 2013. 3, 8

[4] Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. *arXiv preprint arXiv:1906.02694*, 2019. 3, 5, 7, 8

[5] Amanda Berg, Jörgen Ahlberg, and Michael Felsberg. Unsupervised learning of anomaly detection from contaminated image data using simultaneous encoder training. *arXiv preprint arXiv:1905.11034*, 2019. 3, 8, 9

[6] Tangqing Li, Zheng Wang, Siying Liu, and Wen-Yan Lin. Deep unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3636–3645, 2021. 3, 8, 9

[7] Chieh-Hsin Lai, Dongmian Zou, and Gilad Lerman. Robust subspace recovery layer for unsupervised anomaly detection. In *International Conference on Learning Representations*, 2020. 3, 8, 9

[8] Sangwoong Yoon, Yung-Kyun Noh, and Frank Park. Autoencoding under normalization constraints. In *International Conference on Machine Learning*, pages 12087–12097. PMLR, 2021. 3

[9] Jinan Fan, Qianru Zhang, Jialei Zhu, Meng Zhang, Zhou Yang, and Hanxiang Cao. Robust deep auto-encoding gaussian process regression for unsupervised anomaly detection. *Neurocomputing*, 376:180–190, 2020. 4

[10] Subutai Ahmad, Alexander Lavin, Scott Purdy, and Zuha Agha. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing*, 262:134–147, 2017. 4

[11] Yingjie Zhou, Xucheng Song, Yanru Zhang, Fanxing Liu, Ce Zhu, and Lingqiao Liu. Feature encoding with autoencoders for weakly supervised anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 33(6):2454–2465, 2021. 4

[12] Fanxing Liu, Cheng Zeng, Le Zhang, Yingjie Zhou, Qing Mu, Yanru Zhang, Ling Zhang, and Ce Zhu. Fedtadbench: Federated time-series anomaly detection benchmark. In *2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*, pages 303–310. IEEE, 2022. 4

[13] Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Lucas Deecke, Shoaib A Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *ICML*, volume 80, pages 4390–4399, 2018. 5, 7

[14] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International Conference on Learning Representations*, 2018. 6

[15] Fabrizio Angiulli, Fabio Fassetti, and Luca Ferragina. Latent o ut: an unsupervised deep anomaly detection approach exploiting latent space distribution. *Machine Learning*, 112(11):4323–4349, 2023. 6, 7

[16] Stanislav Pidhorskyi, Ranya Almohsen, and Gianfranco Doretto. Generative probabilistic novelty detection with adversarial autoencoders. In *Advances in Neural Information Processing Systems*, pages 6822–6833, 2018. 7

[17] Lukas Ruff, Jacob R Kauffmann, Robert A Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G Dietterich, and Klaus-Robert Müller. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 2021. 7

Normality-Calibrated Autoencoder for Unsupervised Anomaly Detection on Data Contamination

Jongmin Yu[a,*], Minkyung Kim[b,c], Junsik Kim[d], Hyeontaek Oh[e]

[a]*Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Wilberforce Rd, Cambridge, CB3 0WA, United Kingdom*
[b]*Division of Surgical Oncology, Department of Otolaryngology-Head and Neck Surgery, Mass Eye and Ear, Boston, MA 02114, U.S.A*
[c]*Department of Otolaryngology-Head and Neck Surgery, Harvard Medical School, Boston, MA 02115, U.S.A*
[d]*School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138, U.S.A*
[e]*Institute for Information Technology Convergence, KAIST, 291 Daehak-ro, Yuseong-gu, Daejeon, 34141, Republic of Korea*

Abstract

Anomaly detection, or outlier detection, refers to identifying rare or abnormal instances or patterns within a dataset that deviate significantly from the expected or normal behaviour. Various methods have been proposed, but most assume that their training datasets take full, complete integrity. However, the innocent integrity of data is not easy to maintain in reality. Existing anomaly detection methods generally see given data as a single class and learn features that can represent it well, but this approach is very vulnerable to data contamination. This paper proposes a Normality-Calibrated Autoencoder (NCAE), which can boost anomaly detection performance on the contaminated datasets without any prior information or explicit abnormal samples in the training phase. The NCAE adversarially generates highly confident normal samples from a latent space having low entropy and leverages them to predict abnormal samples in a training dataset. NCAE is trained to minimise reconstruction errors in uncontaminated samples and maximise reconstruction errors in contaminated samples. The experimental results demonstrate that our method outperforms shallow, hybrid, and deep methods for unsupervised anomaly detection and achieves comparable performance compared with semi-supervised methods using labelled anomaly samples in the training phase.

*The corresponding author

Keywords: Unsupervised anomaly detection, autoencoder, data contamination, data pollution

1. Introduction

Anomaly detection (AD) generally assumes that given data are a single class, focusing on learning features that can represent it well. According to Kim *et al.* [1], this assumption is defined as the *'Normality assumption'*. The model learned in this way compares the features extracted from the data input in the actual anomaly detection step with the learned features to calculate an error or probability, and then it compares the calculated result with a predetermined threshold to detect anomalies in the data. Currently, most models assume complete integrity of the data given to training the models; that is, all data consists of only a single class (normal) and no noise (anomaly) associated with outliers in the data exists.

However, it is not easy to guarantee data integrity in practice. Datasets in the real world are easily *contaminated*, which means that datasets contain both normal and abnormal samples. In particular, as the number of data to be learned increases, the probability of including errors also increases proportionally. The contaminated samples significantly degrade AD models' robustness, reliability, and accuracy and increase the uncertainty of the detection results.

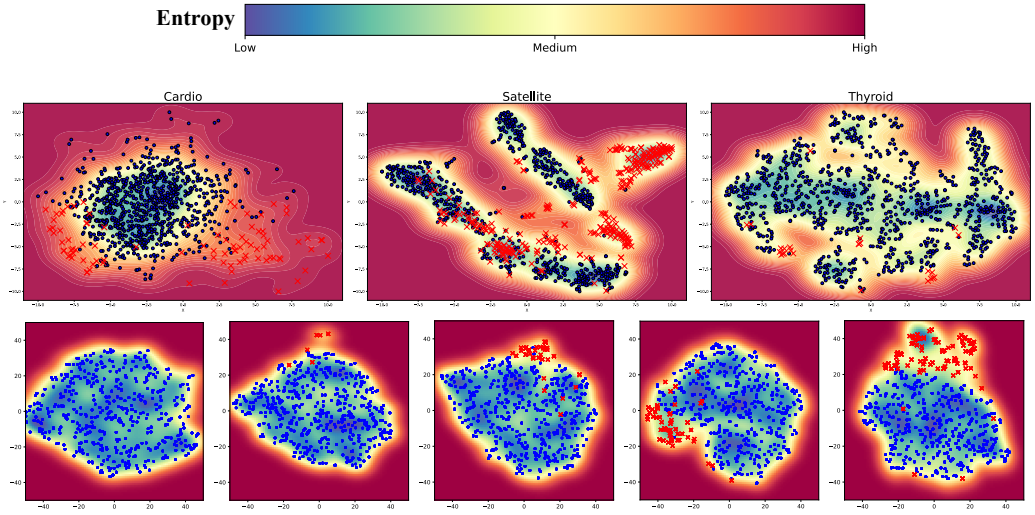
Various methods have been proposed [2, 3, 4, 5, 6, 7] to improve the robustness of AD methods on contaminated datasets. In the beginning, filtering contaminated samples based on contamination ratio [6, 2, 8] and semi-supervised learning approaches that use explicit abnormal samples in the training step [9, 10, 11, 2] have been presented. However, the approaches above are domain or data-type-specific, and their performance highly depends on hyper-parameter settings. For example, the methods using contamination ratios or prepared information about abnormal data work well if precious contamination ratios or abnormal samples can be provided. However, it is a more reasonable and practical hypothesis that the contamination ratio cannot be usually estimated. Also, for the semi-supervised approaches [9, 10, 11, 2], we cannot guarantee that the prepared abnormal data will cover all other unobserved data anomalies. Moreover, even if we successfully find out the contamination ratio of a particular domain or we finally find some specific abnormal data which can cover all other data anomalies for the domain, they can only be applied to derive AD models in that particular domain, not others. Consequently, developing more generalised solutions for AD robust to contaminated datasets is still very challenging but important.

To address this issue, AD methods based on contamination sample prediction using geometric distance measurements [12, 13, 14] have been proposed. These methods assume that contaminated data is always distributed far away from the data distribution's centre or in the highest entropy space. Then, the methods sorted the contaminated data in ascending or descending order by distance from the centre of the data distribution or by entropy, and they filtered out data that was contaminated by a specific percentage [12, 13, 14]. However, as shown in Fig 1, if a training dataset is highly contaminated (like the Satellite dataset, in which samples of 31.6% are contaminated or over 10% of data samples on a simulated dataset), the contaminated samples can also form a low entropy space by themselves, even when the contamination ratio is not high enough, abnormal samples can be positioned some low-entropy space. As a result, the development of an AD method that does not require prior information about data anomalies and does not take geometric solid assumptions when finding contaminated samples is essential.

This paper presents a Normality-Calibrated Autoencoder (NCAE), which is robust to the training dataset contamination. Our key idea for the NCAE is to adversarially generate highly confident normal samples from a low entropy feature space and then contrastively compare the generated samples with the input samples to estimate the contamination score. We pay attention to the fact that if adversarial learning inputs data biased to a specific class in optimizing the generator and discriminator, the generator repeatedly generates only particular data instead of generating various data.

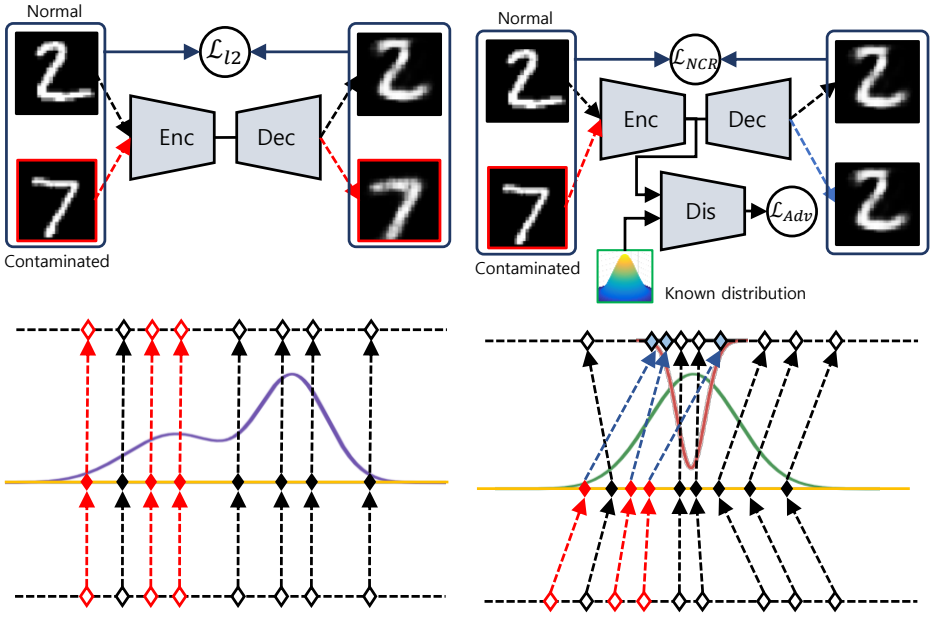
Fig.2 shows the architectural difference between an autoencoder (AE) and the proposed NCAE. The NCAE has a structure that combines an AE and a discriminator for applying adversarial learning. Data is compressed into low-dimensional latent features through the encoder, and the distribution of these latent features is induced into a specific data distribution through adversarial learning (See $\mathcal{L}_{Adv}$ in Fig. 2(b)). This process reduces the uncertainty of the data distribution and minimises the blind spot [15] on the latent feature space. Based on the decoder part of the AE and the specific data distribution used for adversarial learning, samples of the region with the highest probability (*a.k.a.* the lowest entropy) are generated. The generated samples are used to distinguish normal samples from contaminated samples. After identifying the contaminated samples, NCAE calibrates the normality of an AD model by maximising the reconstruction error of the found samples.

To demonstrate the effectiveness of the proposed NCAE as a robust method, we conduct experiments of AD on contaminated datasets using various datasets having diverse contamination ratios. In the performance comparison with the existing



(a) 0% (b) 1% (c) 5% (d) 10% (e) 20%

Figure 1: Visualisation of entropy and distribution of latent features of the various datasets considering different contamination ratios. The first row indicates the visualisation of the 'Cardio', 'Satellite', and 'Thyroid' datasets among the outlier detection datasets. The ratios of the data contamination are 9.6%, 31.6%, and 2.5%, respectively. The second row illustrates the visualisation results using a simulated dataset based on the MNIST dataset, considering various contamination ratios. (a) 0% (No contamination), (b) 1%,(c) 5%, (d) 10%, and (e) 20%. The samples on the '5' class on the MNIST dataset are used as normal (blue dots), and contaminated samples (red x-marks) are randomly picked from the training samples of the remaining classes. The entropy of each sample is computed by the probability estimated by the kernel density estimation (KDE). The 500 samples are randomly picked for visualisation. When a dataset is highly contaminated (*i.e.,* contamination ratio over 10%), contaminated samples are also located in a low entropy region.



(a) Autoencoder (b) NCAE

Figure 2: Comparison of an autoencoder (AE) and the proposed normality-calibrated autoencoder (NCAE). (a) denotes the architectural details and mapping functionality of the AE. (b) indicates the details and functionality of the NCAE. The AE reconstructs the contaminated samples without any concerns (Red dotted lines). By applying adversarial learning $\mathcal{L}_{\text{adv}}$, NCAE removes uncertainty from the latent trait distribution by deriving it from a well-known distribution. After that, the NCAE generate high-confidence normal samples from the centre of the distribution (Red coloured distribution). In computing the reconstruction error, compared with AE, which minimises the reconstruction error of the contaminated samples without any concerns, the NCAE tries to minimise the reconstruction error of contaminated samples with randomly generated high-confidence normal samples (Blue coloured diamonds).

state-of-the-art (SOTA) AD methods, the NCAE achieves better performance and more robustness in data contamination. Particularly, compared to the methods using prior information such as contamination ratio [16, 17] and prepared data anomalies [2], which showed a large performance fluctuation due to errors in the prior information, the proposed method shows robust AD performance against data contamination without prior information.

The contributions of this paper are summarised as follows:

- Normality-calibrated autoencoder (NCAE). A new AD method is robust to

data contamination without any prior knowledge about the data contamination.

- Generative adversarial learning for identifying contaminated data and joint learning scheme for the NCAE. We propose adversarial learning to generate a high-confidence normal data sample and apply it to find contaminated data during the model training. Moreover, we propose an algorithm that efficiently optimises the proposed joint learning models for the NCAE.
- Comprehensive experimental results of AD on data contamination. We provide various ablation studies and performance comparisons with existing SOTA AD methods on data contamination.

The remainder of this paper is organised as follows. Section 2 introduces various related works about AD methods considering data contamination. Section 3 describes the detailed information of the proposed NCAE and its training process. Section 4 provides the experimental settings, and Section 5 presents comprehensive ablation studies for the proposed hyper-parameters. With the best performance hyper-parameters, Section 6 compares AD performance on contaminated data with existing SOTA AD methods. This paper is concluded in Section 7.

2. Related works

Drawing on the research presented in the survey by Ruff *et al.* [18], there are several primary types of anomaly detection methodologies that hinge on the normality assumption. The first of these groups primarily uses one-class classification-based strategies. These strategies work by creating a discerning decision boundary that converts regular data into a succinct representation. This group includes predominant methods such as the One-Class SVM (OC-SVM) [19] and Support Vector Data Description (SVDD) [20], both of which are shallow models. In contrast, methods like OC-NN [21] and Deep SVDD [17] offer deep learning alternatives. These strategies utilise deep neural networks, replacing the shallow models yet maintaining the same goal functions as their OC-SVM and SVDD counterparts. These techniques delineate a hyperplane and a minimum-volume hypersphere, encapsulating the standard data within a latent space. The principle of AD using those methods is simply considering data located outside of the hyperplane or hypersphere as abnormals.

Conversely, there is another type of classification-based methods [22, 23, 24, 25] that experiments with the use of self-supervised learning, especially for image

data. These techniques evaluate normality based on the errors arising from proxy tasks, such as rotation, flipping, or patch rearrangement in augmentation classifications. Furthermore, a novel approach has been introduced recently by Maziarka *et al.* [26], presenting a flow-based one-class classifier aiming to identify the smallest volume bounding area. However, those approaches are only valid for domains that generate anomalies by proxy tasks that are similar to actual anomalies. For instance, up-down flipping can be thought of as an abnormal image, but left-right flipping may be considered data augmentation [27].

Probabilistic model-based methodologies often employ shallow non-parametric density estimators such as Kernel Density Estimation (KDE) [28, 29], and Gaussian Mixture Models (GMM) [30, 6]. However, the above methods can not model complicated data distribution. Recently, there is a growing trend in anomaly detection research to adopt deep generative models like Variational Autoencoders (VAE) [31], Generative Adversarial Networks (GAN) [32, 33], and Anomaly Detection (AD) techniques based on Normalising Flows [34, 35, 36]. These models aim to decipher the latent feature space data distribution. However, VAE sometimes causes posterior collapse, in which the VAE decoder ignores the actual data distribution and generates a sample from a Gaussian distribution. Also, unstable training of GANs is still troublesome.

In the realm of reconstruction-based methods [37, 38, 39], Autoencoders (AE) [40, 41, 42] are predominantly used. Numerous adaptations of this method have been introduced to improve its performance in anomaly detection. Further, attempts have been made to merge the discriminating representation of AE with shallow methods [43, 44, 45, 46]. Unfortunately, the autoencoder has a blind stop. When an AE is trained, we expect that unseen samples output larger reconstruction errors. However, there are several studies that AD still can reconstruct unseen samples with small reconstruction errors [15, 47].

Despite extensive research in anomaly detection using normality modelling, these techniques heavily depend on the availability of a pure training dataset free of anomalous data, which is often difficult to obtain. This dependence often leads to less-than-optimal performance and limits the methods' application. The one-class classification-based methods, although able to handle contaminated datasets by adjusting a contamination ratio as a hyper-parameter, also depend largely on prior knowledge of this ratio. However, in real-world scenarios, this ratio is typically unknown.

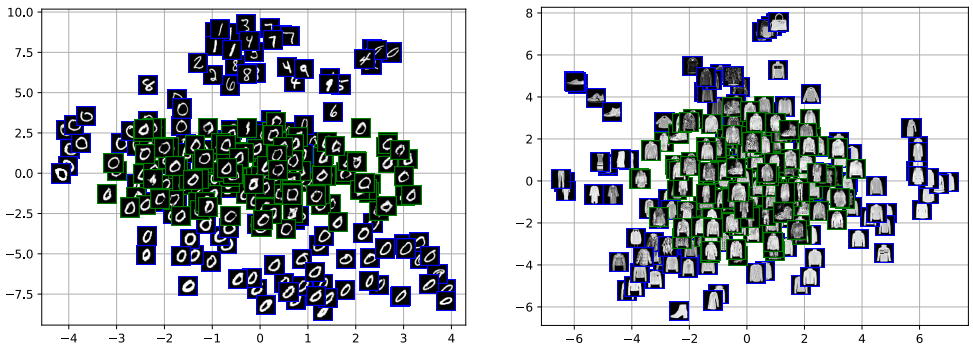
To tackle the problem of learning normality from contaminated datasets, researchers have presented methods using meta-information about the contaminated data such as pollution rate [6, 2, 8], or pre-defined abnormal samples

[9, 10, 11, 2, 48]. However, those methods are domain-specific, and their AD performances are highly dependent on hyper-parameter settings such as pollution ratio. More recently, researchers have proposed robust methods that aim to diminish the adverse effects of anomalous data in such datasets [49, 38, 50, 51, 39]. Most of these studies have used pseudo-labelling techniques to improve the models' robustness. Different methods, including AE [49, 38, 50] or regression-based [51, 39] approaches, have been used for pseudo-labelling.

Xia *et al.* [49] pioneered an approach where pseudo-labelling is conducted using the reconstruction error obtained from an AE. Following this, the AE is iteratively trained to minimize the reconstruction error of pseudo-normal data, incorporating a regularization term that captures the separability of the error distribution. Using a similar framework, Zhou *et al.* [38] and Beggel *et al.* [50] implemented related techniques. Zhou *et al.* [38] used a robust AE inspired by robust Principal Component Analysis (PCA), while Begge *et al.* utilized an adversarial AE [52].

While most of the previous research focused on AE and its variants, a few recent methods proposed the use of regression models [51, 39], trained in an iterative manner. Pang *et al.* [51] and Shim *et al.* [53] presented a two-class ordinal regression model that uses neural networks, while Fan *et al.* [39] used a Gaussian process regression model. In both approaches, an initial anomaly detection stage is carried out by a separate, pre-existing anomaly detector. Following this, the model is iteratively trained using pseudo-labeled data.

A significant limitation of the aforementioned methods is their dependency on hyper-parameters that govern the selection and quantity of data treated as pseudo-normal or pseudo-abnormal samples. However, finding optimal hyper-parameters can be difficult as they tend to fluctuate depending on the dataset and unknown contamination ratios. To tackle this problem, anomaly detection (AD) methods utilizing contamination sample prediction through geometric distance metrics have been introduced [12, 13, 14]. These techniques typically assume that contaminated data points are located far from the centre of the data distribution or within regions of high entropy. Subsequently, the methods rank the contaminated samples either in ascending or descending order based on their distance from the distribution centre or their entropy level, filtering out a specified percentage of contaminated data [12, 13, 14]. However, as illustrated in Fig 1, when a training dataset is heavily contaminated (e.g., more than 10% of the samples), the contaminated samples may also cluster in low-entropy regions. Therefore, it is crucial to develop an AD method that does not rely on prior knowledge of data anomalies and avoids strong geometric assumptions when identifying contaminated samples.



(a) (b)

Figure 3: Visualisation of the input image and the generated image based on the scatter plot for each latent feature and sampled noise. '0' class on MNIST and the 'Coat' class on Fashion-MNIST are determined as normal classes, respectively. Each model is trained with a contamination ratio of 20%. The images with the blue border show the samples on a single training batch, and the images with the green border represent the generated images by adversarial learning.

In this paper, we introduce a Normality-Calibrated Autoencoder (NCAE) designed to be resilient to contamination within the training dataset. The central concept of NCAE involves adversarially generating highly confident normal samples from a low entropy feature space and then using a contrastive comparison between these generated samples and the input data to estimate the contamination score. We observe that when adversarial learning processes data biased toward a particular class during the optimization of the generator and discriminator, the generator tends to repeatedly produce specific data rather than a diverse set of samples [54].

3. Normality-Calibrated Autoencoder

3.1. Learning normality-calibrated autoencoder

For n number of input samples with D dimensions $\mathcal{X} = \{x_i\}_{i=1:n}$, $x \in \mathbb{R}^D$ and the corresponding latent features with d dimensions $\mathcal{Z} = \{z_i\}_{i=1:n}$, $z \in \mathbb{R}^d$, let define an autoencoder (AE) composed of an encoder $f(x) : x \rightarrow z$ and a decoder $g(x) : z \rightarrow \bar{x}$, where $\bar{x}$ denotes the reconstruction result of x. The general objective of the AE is training f and g to minimise an error between input samples x and the reconstruction results $\bar{x}$, as follows:

$$\min_{f,g} \mathbb{E}_{x \sim p_{\mathcal{X}}} ||x - \bar{x}||^2, \quad \bar{x} = g \circ f(x), \tag{1}$$

where $p_{\mathcal{X}}$ denotes the entire input samples. By minimising Eq. 1, the AE compiles a mapping function between a high dimensionality data x to a low dimensionality feature z. Based on this process, the AE can learn much compressed and abstracted information, which can represent various features of the given data. To detect data anomalies, the AE uses the reconstruction error. If anomaly data is given, the learnt abstracted information would not reconstruct the data well so the reconstruction error would be higher than normal ones.

However, an AE is known to have an over-confidence issue [47], *i.e.,* low reconstruction error of unseen samples. This issue can be thought that if the AE takes anomaly data as their input, the reconstruction error of the data would not be high, or even similar to the error of normal data samples. As we mentioned above, AD methods using the AE usually identify abnormal samples using the reconstruction error. Therefore, if the AE takes anomaly samples as inputs, it may not distinguish whether the samples are abnormal or not [47, 55, 56]. This over-confidence issue would be deepened when a training dataset is contaminated.

One straightforward approach to prevent this issue is adding an extra term to maximise reconstruction error for contaminated samples. This is simply done by adding the negative reconstruction error for contaminated samples represented by

$$\min_{f,g}(\mathbb{E}_{x^n \sim p_{\mathcal{X}^{\mathrm{N}}}}||x^n - \bar{x}^n||^2 - \mathbb{E}_{x^{\mathrm{c}} \sim p_{\mathcal{X}^{\mathrm{C}}}}||x^{\mathrm{c}} - \bar{x}^{\mathrm{c}}||^2), \tag{2}$$

where x^{n} and x^{c} indicate normal data and abnormal (or contaminated) data sampled from each distribution *i.e.,* $p_{\mathcal{X}^{\mathrm{N}}}$ and $p_{\mathcal{X}^{\mathrm{C}}}$, respectively. $\bar{x}^{\mathrm{n}}$ and $\bar{x}^{\mathrm{c}}$ denote the reconstruction results corresponding to the normal and abnormal data, respectively.

Since the reconstruction error for contaminated samples $\mathbb{E}_{x^{\mathrm{n}} \sim p_{\mathcal{X}^{\mathrm{N}}}, x^{\mathrm{c}} \sim p_{\mathcal{X}^{\mathrm{C}}}}$ is negative, to minimise the entire loss, the error should be maximised. However, this formulation is intractable when applied to optimising an AE directly because there is no bound in maximising the error terms.

This problem can be avoided by replacing the error maximisation task with a minimising task of the reconstruction error between the contaminated samples and arbitrarily assigned normal samples. Based on this principle, we define normality-calibrated reconstruction (NCR) loss as follows:

$$\min_{f,g}(\mathbb{E}_{x^n \sim p_{\mathcal{X}^{\mathrm{N}}}}||x^n - \bar{x}^n||^2 + \mathbb{E}_{\acute{x}^{\mathrm{n}} \sim p_{\mathcal{X}^{\mathrm{N}}}, x^{\mathrm{c}} \sim p_{\mathcal{X}^{\mathrm{C}}}}||x^{\mathrm{c}} - \acute{x}^{\mathrm{n}}||^2), \tag{3}$$

We transform the maximisation term of Eq. 2 (the second term) to the minimisation term by using the contaminated data and randomly picked normal data ($\acute{x}^{\mathrm{n}}$) sampled from the $p_{\mathcal{X}^{\mathrm{N}}}$. However, to optimise this loss function, we should find out which samples are contaminated to optimise AE using Eq. 3 properly.

3.2. *High-confidence normal samples generation using Generative Adversarial Network*

We find contaminated samples by using highly confident normal samples generated from low entropy latent space. We apply the generative adversarial network (GAN) [57] framework to do this. The high-confidence normal sample generation via the GAN framework is carried out as follows.

Initially, we transform a distribution of all latent features z, which are encoded from input samples x through the encoder f, to a more knowledgeable probabilistic distribution such as Gaussian distribution. And then, we generate samples using noise signals sampled from the centre of the knowledgeable distribution, *i.e.,* the lowest entropy space as you can see in Fig. 1, even though a training dataset is highly contaminated (like over 10%), the dominant data distributed in the lowest entropy space are normal data.

An adversarial loss for transforming a latent feature distribution into a more knowledgeable probabilistic distribution is defined by the following:

$$\min_{f} \max_{D_l} \mathbb{E}_{\omega \sim N(\mu_{\mathcal{Z}}, I_d)}[\log D_l(\omega)] + \mathbb{E}_{x \sim P_{\mathcal{X}}}[\log (1 - D_l(f(x)))], \tag{4}$$

where D_l denotes the discriminator for latent features, and $N(\mu_{\mathcal{Z}}, I_d)$ defines a normal distribution with the mean of latent features $\mu_{\mathcal{Z}} \in \mathbb{R}^d$ and a covariance matrix defined by an identity matrix $I_d \in \mathbb{R}^{d \times d}$. $\mu_{\mathcal{Z}}$ is initialised by the mean value of latent features: $\mu_{\mathcal{Z}} = \frac{1}{n}\sum_{i=1}^{n} z_i$. We would want each component of z to be maximally informative such as each of them to be an independent random variable. Therefore, the $d \times d$ identity matrix determines the covariance matrix.

Variational Autoencoder (VAE) [31] can be an alternative to adversarial learning in deriving the latent feature distribution into a well-known distribution. However, to apply the VAE, it is inevitable to change the network structure since it needs extra networks to sample the means and variances of latent features. We decided to use adversarial learning since it is more flexible in changing the network structure.

Since f and D_l are being updated, $\mu_{\mathcal{Z}}$ would be shifted during the training step. $\mu_{\mathcal{Z}}$ is updated at every training step as follows:

$$\mu_{\mathcal{Z}}^{t+1} = \mu_{\mathcal{Z}}^{t} - \gamma \frac{1}{m} \sum_{i=1}^{m} (\mu_{\mathcal{Z}}^{t} - z_i), \quad \mu_{\mathcal{Z}}^{0} = \frac{1}{n} \sum_{i=1}^{n} z_i^0 \tag{5}$$

where $\mu_{\mathcal{Z}}^{t+1}$ and $\mu_{\mathcal{Z}}^{t}$ denote the $\mu_{\mathcal{Z}}$ on $t+1$-th and t-th training step, respectively. m is the batch size, z_i is i-th latent features on the batch, and γ is a learning rate.

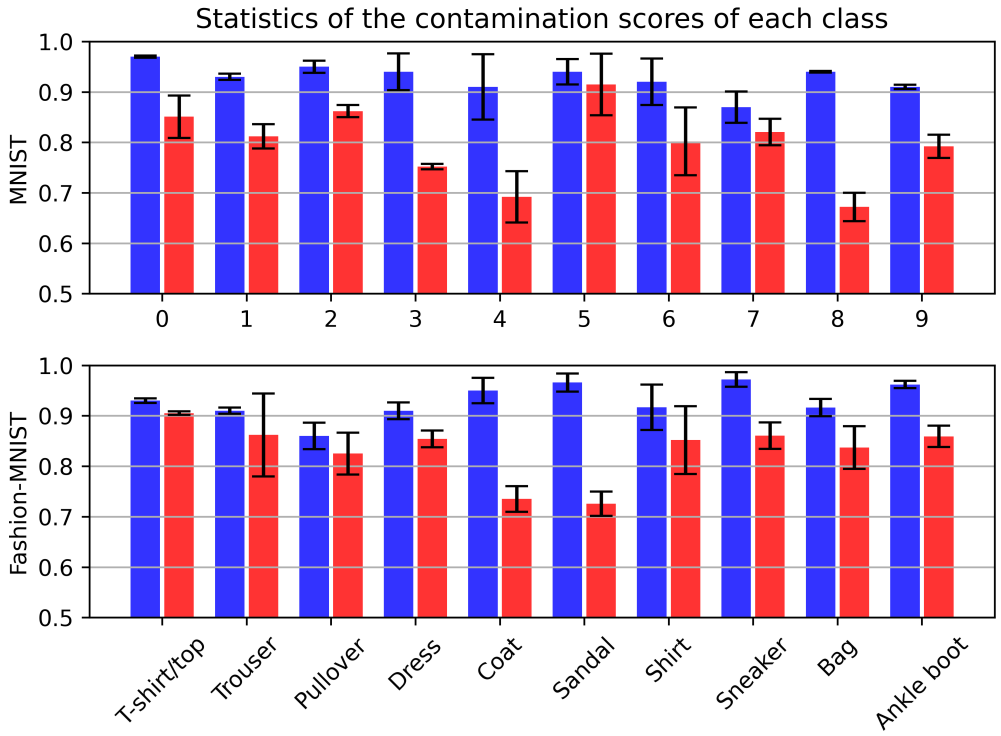


Figure 4: The means and variances of the contaminated scores for 100 times repeated experiments. The upper illustrates the averaged contamination score of contaminated and normal samples on the MNIST dataset. The lower illustrates the averaged scores on the Fashion-MNIST dataset. The black lines of each bar show a variation of the score.

To generate highly confident normal samples, we formulate the following adversarial loss:

$$\min_{g} \max_{D_s} \mathbb{E}_{x \sim P_{\mathcal{X}}}[\log D_s(x)] + \mathbb{E}_{\acute{\omega} \sim N(\mu_{\mathcal{Z}}, \sigma I_d)}[\log(1 - D_s(g(\acute{\omega}))], \tag{6}$$

where D_s denotes the discriminator for samples, and $N(\mu_{\mathcal{Z}}, \sigma I_d)$ is a d-dimensional normal distribution with the mean $\mu_{\mathcal{Z}} \in \mathbb{R}^d$ and the covariance matrix $\sigma I_d \in \mathbb{R}^{d \times d}$. $\mu_{\mathcal{Z}}$ is equivalent to the $\mu_{\mathcal{Z}}$ in Eq. 4. σI_d is defined by multiplication of a scalar value $\sigma \in [0, 1]$ and the identity matrix I_d. σ is a hyper-parameter to control the compactness of random noise for generating samples using the decoder g. The smaller σ can give more chances to generate highly confident normal samples by generating a feature close to the centre of the probability distribution.

We demonstrate the effectiveness of the proposed generative adversarial net-

work (GAN) for generating high-confidence normal samples, we have conducted toy experiments using MNIST and Fashion-MNIST datasets. We have trained the proposed GAN the data of the '0' class on the MNIST dataset and the 'Coat' class on the Fasion-MNIST dataset as a normal class. The contamination ratio is set by 20%, meaning the 20% portion of training data comprises other classes. As shown in Fig. 3, even though the training data is highly contaminated, the GAN trained by our approach generates normal samples.

3.3. Contaminated sample mining and joint learning

To predict contaminated samples, we use the generated high-confident normal samples as a dictionary. With the generation process for high confident normal samples: $g(\acute{\omega}) = \acute{x}$, we construct a latent feature dictionary $\mathcal{M} = [\acute{z}_i]_{i=1:m}$, $\acute{z}_i = f(\acute{x}_i)$ and $\mathcal{M} \in \mathbb{R}^{m \times d}$, where m is the batch size. By leveraging $\mathcal{M}$ and given each training batch $\{x_i\}_{i=1:m}$, we define a pseudo contamination score c_i of each input sample x_i as follow:

$$c_i = \frac{1}{m} \sum_{j=1}^{m} f(x_i) \cdot \acute{z}_j^{\mathrm{T}}, \quad \acute{z}_j \in \mathcal{M}, \tag{7}$$

where T denotes the transpose of the vector. We apply $l2$-normalisation to improve the robustness of the variation of the vector scale of the operation.

We predict the contaminated samples by sorting the score in descending order and picking top-τ% samples among the sorted results as the contaminated samples; thus, the number of predicted contaminated samples is decided by τm that is a multiplication of τ and the batch size m. The above process is represented as follows:

$$\mathcal{X}^{\mathrm{C}} = \{x_t\}_{t \in C[1:\lceil \tau m \rceil]}, \quad C = \arg\operatorname*{sort}_i c_i, \quad \text{with} \quad 1 \leq i \leq m,$$

where C is a set of the sorted indices of input batch samples in descending order of the contamination score (Eq. 7), and $\mathcal{X}^{\mathrm{C}}$ is a set of predicted contaminated samples. $\lceil \cdot \rceil$ denotes the ceiling function. τ affects on deciding the number of predicted contaminated samples, so it directly affects the AD performance of our method.

To demonstrate the validation of the proposed contamination sample mining process, we have conducted toy experiments. Using the MNIST and Fashion-MNIST datasets, we randomly select one class as normal and otherwise all abnormal. We set the batch size of 128 and the contamination ratio of 20%. After that, during the NCAE training, we averaged the contamination score of normal samples and

contaminated samples. Fig. 4 illustrates the distribution of contaminated scores of normal samples and the contaminated samples for 100 iterative experiments. As shown in Fig. 4, the score of contaminated samples is obviously distinguishable from the normal samples. Even if we consider the variation of the score (rectangle of each bar chart), the results can be interpreted that our score-based contaminated sample mining process works well.

The objective function for joint learning of the entire components of our method is as follows:

$$\begin{aligned}\min_{f,g}\max_{D_l,D_s} \quad & \underbrace{\mathbb{E}_{x\sim p_{\mathcal{X}^{\mathrm{N}}}}||x-g\circ f(x)||^2+\mathbb{E}_{x\sim p_{\mathcal{X}^{\mathrm{C}}}}||x-\bar{x}'||^2}_{\text{(a)}} \\ & +\underbrace{\mathbb{E}_{\omega\sim N(\mu_{\mathcal{Z}},I_d)}[\log D_l(\omega)]+\mathbb{E}_{x\sim P_{\mathcal{X}}}[\log\left(1-D_l(f(x))\right)]}_{\text{(b)}} \\ & +\underbrace{\mathbb{E}_{x\sim P_{\mathcal{X}}}[\log D_s(x)]+\mathbb{E}_{\omega'\sim N(\mu_{\mathcal{Z}},\sigma I_d)}[\log(1-D_s(g(\omega'))]}_{\text{(c)}},\end{aligned} \tag{8}$$

where D_l and D_s define the discriminators for the latent features and samples, respectively. $\bar{x}'$ is a prototype of the generated high-confident normal samples $g(\acute{\omega}) = \acute{x}$. It is defined by averaging the generated samples as follows: $\bar{x}' = \mathbb{E}_{x'\sim g(N(\mu_{\mathcal{Z}},\sigma I_d))}(\acute{x})$. $\mu_{\mathcal{Z}}$ and I_d denote the averaged latent features and their covariance matrix represented by an identity matrix. w indicates noise signals sampled from the Gaussian distribution $N(\mu_{\mathcal{Z}}, I_d)$. (a), (b), and (c) denote the NCR loss and the two adversarial losses, respectively.

The encoder of the autoencoder is trained with (a) + (b), and the decoder of the AE is trained with (a) + (c); they share the NCR loss term (a) under joint training. To optimise the above objective efficiently, we propose an alternating algorithm, which optimises model parameters of the encoder f and the decoder g for the AE and the discriminator for the adversarial learning D alternatively, as shown in Algorithm 1. Since the algorithm monotonically and iteratively decreases the objective function, it is guaranteed to converge.

In the algorithm, σ and τ significantly affect the AD performance of the NCAE model. σ decides the range for generating the noise signal to create the high-confidence normal samples. τ is used to decide how many samples would be considered contaminated samples. We will discuss the effectiveness of those hyper-parameters on AD performance in Section 5.

The AD process of the NCAE is equivalent to other AD-based AD methods. When a sample x is given, we produce reconstructed results $g \circ f(x)$ and compute

the reconstruction error (See Eq. 1). After that, anomaly detection is done by comparing the error with a pre-defined threshold β. The AD process is represented as follows:

$$X = \begin{cases} \text{Abnormal} & ||x - g \circ f(x)||^2 > \beta, \\ \text{Normal} & \text{Otherwise.} \end{cases} \tag{9}$$

The AD detection performance depends on the value of β. When β is too small, the AD results will be more precise, but they will be too specific, and many anomalies will be ignored. Hence, if the β is too large, many anomalies will be detected, but it contains many false positives. In other words, The bigger β, the more comprehensive abnormality will be detected. The smaller β, the more precise detection results (less false positives) will be provided. Considering this evaluation property, we provide the evaluation metric by observing the performance trend regarding the threshold value change instead of the AD performance at a single point. More detailed information is provided in Section 5.

4. Experimental settings

Dataset: We use various anomaly detection datasets for our experiment. Primarily, MNIST [58][1], Fashion-MNIST [59][2], and CIFAR-10 [60][3] datasets are used for the experiments. The MNIST dataset consists of a collection of 70,000 grayscale images of handwritten digits. It is divided into a training set of 60,000 examples and a test set of 10,000 examples. Each image has a resolution of 28×28 pixels. The Fashion-MNIST dataset is intended to be a more challenging version of the MNIST dataset. Instead of handwritten digits, it consists of 70,000 grayscale images of 10 different fashion items, including t-shirts, trousers, dresses, shoes, and more. Like MNIST, it is divided into a training set of 60,000 examples and a test set of 10,000 examples. The CIFAR-10 dataset of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images. The dataset is divided into five training batches and one test batch, each with 10000 images [60]. The test batch contains exactly 1000 randomly selected images from each class. The training batches contain the remaining images in random order, but some training batches may contain more images from one class than another. Between them, the training batches contain exactly 5000 images from each class.

[1] https://www.kaggle.com/datasets/hojjatk/mnist-dataset

[2] https://www.kaggle.com/datasets/zalando-research/fashionmnist

[3] https://www.cs.toronto.edu/~kriz/cifar.html

Algorithm 1 Repetitive generation-feedback algorithm for NCAE

Require: The training epoch T, training step S, the batch size m, the contaminated ratio τ, the learning rate γ, and the variance controller σ for generating normal samples

Initialise • $\mu_{\mathcal{Z}} : \mu_{\mathcal{Z}} = \frac{1}{n}\sum_{i=1}^{n} z_i$

for $t = 1$ to T epochs **do**

for $s = 1$ to S training steps **do**

• Update the high confident normal sample generation components: $\{g, \mathcal{D}_s\}$

• Sample $\{\acute{\omega}_i\}_{i=1:m} \sim N(\mu_{\mathcal{Z}}, \sigma I_d)$ and $\{x_i\}_{i=1:m} \sim P_{\mathcal{X}}$

• Update the decoder g and the discriminator D_s using following objective:

$$\min_{g}\max_{D_s} \mathbb{E}_{x\sim P_{\mathcal{X}}}[\log D_s(x)] + \mathbb{E}_{\omega'\sim N(\mu_{\mathcal{Z}},\sigma I_d)}[\log(1 - D_s(g(\omega'))].$$

• Generate high-confidence normal samples $\{\acute{x}_1, \ldots, \acute{x}_m\}$: $\acute{x}_i = g(\acute{\omega}_i)$.

• Predict contaminated samples

· Construct feature dictionary $\mathcal{M} = \{\acute{z}_1, \ldots, \acute{z}_m\}$, where $\acute{z}_i = f(\acute{x}_i)$.

· Compute the contamination score c: $c_i = \frac{1}{m}\sum_{j=1}^{m} f(x_i) \cdot \acute{z}_j^{\mathrm{T}}$

· Predict contamination samples $\mathcal{X}_{\mathrm{C}}$ as follows:

$$\mathcal{X}^{\mathrm{C}} = \{x_t\}_{t\in C[1:\lceil \tau m\rceil]} \quad w.r.t., C = \arg\operatorname*{sort}_{i} c_i, \quad w.r.t., 1 < i < m.$$

• Update f, g and $\mathcal{D}_l$.

• update the f, g, and D_l using $\{\omega_i\}_{i=1:m}$, $\{x_i\}_{i=1:m}$, $\mathcal{X}^{\mathrm{N}}$, and $\mathcal{X}^{\mathrm{C}}$ with the following objective:

$$\begin{aligned}\min_{f,g}\max_{D_l} \quad & \mathbb{E}_{x\sim\mathcal{X}^{\mathrm{N}}}||x - g\circ f(x)||^2 + \mathbb{E}_{x\sim\mathcal{X}^{\mathrm{C}}}||x - \bar{x}'||^2 \\ & + \mathbb{E}_{\omega\sim N(\mu_{\mathcal{Z}},I_d)}[\log D_l(\omega)] + \mathbb{E}_{x\sim P_{\mathcal{X}}}[\log\left(1 - D_l(f(x))\right)]\end{aligned}$$

• Update $\mu_{\mathcal{Z}}$: $\mu_{\mathcal{Z}} = \mu_{\mathcal{Z}} - \gamma\frac{1}{m}\sum_{i=1}^{m}(\mu_{\mathcal{Z}} - f(x_i))$

end for

end for

Table 1: Key properties of the Outlier Detection DataSets (ODDs).

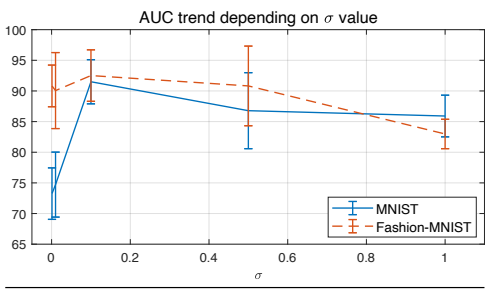
Dataset	Total #	Dim	# outliers (%)
Satimage-2	5,803	36	71 (1.2%)
Thyroid	3,772	6	93 (2.5%)
Cardio	1,831	21	176 (9.6%)
Satellite	6,435	36	2,036 (31.6%)
Shuttle	49,097	9	3,511 (7%)
Mammography	11,183	6	260 (2.32%)

Each image also has a resolution of 28×28 pixels. In addition, we leverage Outlier Detection DataSets[4] (ODDs). Table 1 shows the key properties of the ODDs. In particular, ODDs provide a chance to evaluate the AD performance with various contamination ratios. The contamination ratios of the ODDs have ranged from a minimum of 1.2% to a maximum of 31.6%.

Experiment protocol: We follow the unsupervised AD protocol described by Ruff *et al.* [2] and Zhou *et al.* [48]. We set one of the classes provided by a dataset as normal and others as abnormal. First, we have decided on contamination ratio $\rho = \frac{n_{\mathrm{A}}}{n_{\mathrm{N}}+n_{\mathrm{A}}}$, where n_{N} and n_{A} are the numbers of normal and abnormal samples, respectively. After that, we pick normal samples from the chosen class and contaminated samples from the remaining classes. In the test phase, the samples of the normal class are labelled by 0, and other samples are labelled by 1. The maximum contamination ratio is set by 30%. This ratio has been decided by considering the maximum ratio of outliers among the dataset we used for our experiment (See Table 1). It may consider a higher contamination ratio such as 50% or 60%; however, when we consider the definition of 'abnormal' meaning a pattern rarely observed data which does not follow a dominant data representation [61], the 30%, the maximum contamination ratio of our experiment, is a reasonable choice. For the performance analysis, the Area Under the Receiver Operating Characteristic Curve (AUC) is used. In comparison with other methods, we refer the experimental results presented in the Ruff *et al.* [2]. The experimental results in Table 2 and Table 3 contain the quantitative results of the NCAE and other methods referred from Ruff *et al.* [2].

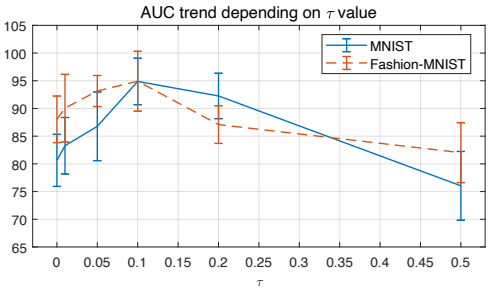
Implimentation: Our method can be thought of as one of deep neural network-

[4]http://odds.cs.stonybrook.edu/



σ	0	0.01	0.1	0.5	1.0
MNIST	73.25±4.2	74.72±5.3	91.50±3.6	86.78±6.2	85.91±3.4
fMNIST	90.82±3.4	90.06±6.2	92.51±4.2	90.82±6.5	82.98±2.4

(a) Effects of σ



τ	0	0.01	0.05	0.1	0.2	0.5
MNIST	80.63±4.7	83.27±5.1	86.78±6.2	94.94±5.0	92.26±4.1	76.04±6.2
fMNIST	88.04±4.2	90.06±6.1	93.15±2.8	94.92±4.9	87.09±3.4	82.02±5.4

(b) Effects of τ

Figure 5: Ablation studies about unsupervised AD performance depending on σ and τ. (a) and (b) represent the trends of AUC with respect to the setting of σ and τ, respectively, on the MNIST and Fashion-MNIST (fMNIST) datasets.

based AD methods; accordingly, the AD performance of deep learning-based methods can be changed depending on the number of layers and the kernel size of convolutional layers. To this principle, using a deeper layer can be considered a way to boost performance, which is very unfair to other AD methods. The fairest way for deep learning-based AD research is to employ a well-known and comprehensively used network model. In this work, we employ a LeNet-based autoencoder, which was employed for various deep learning-based AD studies [47, 17, 2, 18] on MNIST and Fashion-MNIST datasets, where each convolutional module consists of a convolutional layer followed by leaky ReLU activation functions with leakiness of 0.1. On the outlier detection dataset (OODs) benchmark using Cardio, Satellite, Satimage-2 and Thyroid, we employ standard MLP feed-forward AE structure presented by Ruff *et al.* [2]. The MLP for the encoder and decoder is defined by a 3-layer neural network with 32-16-8 units. We use the Adam optimiser with the recommended default hyper-parameters [16]. The batch size is set to 128. The initial learning rate is 0.01 and decayed every ten epochs by multiplying it by 0.1. σ and τ are decided by 0.1 (based on the results from the ablation study), respectively[5].

[5]https://github.com/andreYoo/NCAE_UAD

5. Ablation study

This ablation study provides insights into how to set up the prior parameters required for model training. In addition, we compare the effectiveness of our contaminated sample mining approach based on the generation of high-confidence normal samples with other commonly used filtering or sampling methods to find contaminated data samples. In the ablation study, only the MNIST and Fashion-MNIST data were used for experimental efficiency.

5.1. Hyper-parameter analysis

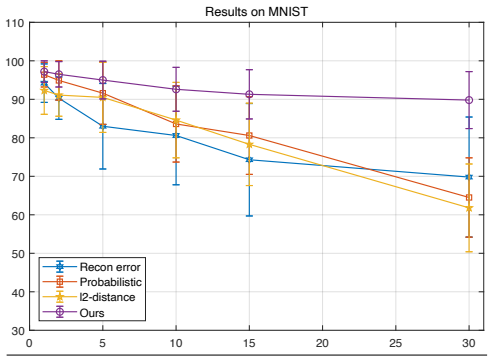
We analyse unsupervised AD performance depending on the setting of σ and τ. MNIST and Fashion-MNIST datasets are used for the ablation study. Ablation studies are conducted based on the experimental protocol described in the previous section. The contamination ratio ρ is fixed to 0.2.

Parameter analysis on σ: When σ is too small, then the distribution of sample noise for generating samples would be too compact so that the generated samples cannot provide comprehensive information to cover the diverse patterns of normal samples. On the other hand, when σ is too large, then there is a possibility that the noise can be sampled from low entropy space (*i.e.,* abnormal samples also can be generated).

Figure 5(a) shows the AUC trends depending on the σ. The AUC increases rapidly in the case of σ is less than 0.1, and then decreases gradually. This can be interpreted as follows. If the sampling space is too compact (*i.e.,* when σ is too small), it means that the generated normal sample does not provide enough information to distinguish the contaminated sample. When sampling space is too broad (*i.e.,* when σ is too large), it also degrades performance, but the impacts of the broader sampling space are relatively less than that of the smaller sampling space (e.g., when $\sigma \leq 0.1$). The best performance is obtained by σ of 0.1.

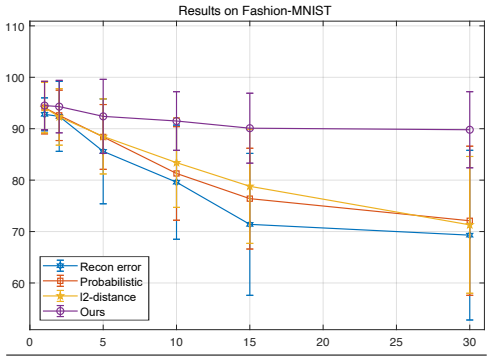
Parameter analysis on τ: τ decides the number of predicted contaminated samples per training batch. The lower τ can provide a more precise prediction performance but may not be enough to provide a more comprehensive prediction performance. In contrast, when τ is too large, the predicted results are possibly more accurate but also may have a lot of false-positive results.

As shown in Figure 5(b), the AUC increases rapidly with τ from 0 to 0.1 and then decreases slowly. The results can be interpreted as follows. Finding contaminated samples themselves has a large impact on AD performance, but the quantity of found samples affects less to AD performance. However, predicting



ρ	.01	.02	.05	.10	.15	.30
Recon	94.2±5.0	90.3±5.5	83.0±11.1	80.6±12.8	74.3±14.6	69.8±15.6
Probabilistic	96.4±3.2	94.9±5.1	91.6±8.1	83.6±9.9	80.6±10.1	64.5± 10.3
l2-distance	92.3±6.2	91.1±5.5	90.5±9.1	84.6±9.8	78.3±10.7	61.8±11.4
Ours	97.2±2.8	96.5±3.3	95.0±4.9	92.6±5.7	91.3±6.4	89.8±7.4

(a)



ρ	.01	.02	.05	.10	.15	.30
Recon	92.8±3.2	92.4±6.8	85.6±10.2	79.6±11.1	71.4±13.8	69.3±16.5
Probabilistic	94.1±5.1	92.6±4.9	88.4±6.3	81.3±9.1	76.4±9.8	72.1± 14.4
l2-distance	94.1±4.9	92.3±5.5	88.5±7.3	83.4±8.7	78.8±11.1	67.3±13.3
Ours	94.5±4.7	94.3±5.1	92.4±7.2	91.5±5.7	90.1±6.8	89.4±7.1

(b)

Figure 6: Trends in AUC values for polluted data sampling methods and changes in contamination degree. (a) shows the experimental results on the MNIST dataset. (b) represents the results on the Fashion-MNIST dataset.

too many samples may degrade AD performance by taking a great number of false positives. The best performance is obtained by τ of 0.1.

5.2. Effectiveness of the contamination sample mining

The contaminated sample mining based on the generated sample can be considered a way of pseudo-labelling based on data sampling. There are several sampling strategies to generate a pseudo-label for contaminated samples. Euclidean distance-based sampling [12, 13, 14, 16, 17] is one of the popular approaches to assigning the pseudo label for contaminated samples. Based on the Euclidean distance between the distribution centre and each data point (*e.g.,* $l2$-distance between the latent feature distribution centre and each latent feature), it designates some samples in order of distance as anomalies.

We have compared our contamination sample mining approach with various contaminated data sampling methods. Since our method is based on AE, we compare our method with reconstruction error-based sampling [49] and latent feature's $l2$-distance-based sampling method [17], and latent feature's probability distribution-based method [16, 47]. For a fair comparison, based on our ablation study (about τ), we set 10% (0.1 of τ) of sample on each batch that would be labelled by the contaminated samples.

Figure 6 shows the trend of AUCs depending on the contamination ratio. The ablation study results show that our contaminated sample mining approach out-

Table 2: Performance comparison on unsupervised anomaly detection in terms of various contamination ratios ρ. The AUC value is used to evaluate the performance. MNIST, Fashion-MNIST, and CIFAR-10 datasets are used for the comparison. The **bolded** figures indicate the best performances.

Dataset	ρ	OC-SVM	IF	KDE	CAE	Deep SVDD	SSAD	SS-DGM	Deep SAD	Classification	DAGMM	LatentOut	NCAE
MNIST	.00	96.0±2.9	85.4±8.7	95.0±3.3	92.9±5.7	92.8±4.9	**97.9±1.8**	92.2±5.6	96.7±2.4	94.5±4.6	91.7±6.2	96.5±3.2	94.0±4.2
	.01	94.3±3.9	85.2±8.8	91.2±4.9	91.3±6.1	92.1±5.1	96.6±2.4	92.0±6.0	95.5±3.3	91.5±5.9	90.4±5.2	93.9±4.8	**97.2±2.8**
	.05	91.4±5.2	83.9±9.2	85.5±7.1	87.2±7.1	89.4±5.8	93.4±3.4	91.0±6.9	93.5±4.1	86.7±7.4	88.5±9.2	86.4±7.2	**95.0±4.9**
	.10	88.8±6.0	82.3±9.5	82.1±8.5	83.7±8.4	86.5±6.8	90.7±4.4	89.7±7.5	91.2±4.9	83.6±8.2	84.2±6.2	81.6±8.3	**92.6±5.7**
	.20	84.1±7.6	78.7±10.5	77.4±10.9	78.6±10.3	81.5±8.4	87.4±5.6	87.4±8.6	86.6±6.6	79.7±9.4	81.5±7.3	73.7±9.3	**89.8±7.4**
F-MNIST	.00	92.8±4.7	91.6±5.5	92.0±4.9	90.2±5.8	89.2±6.2	**94.0±4.4**	71.4±12.7	90.5±6.5	76.8±13.2	87.6±7.2	91.2±6.2	91.5±8.3
	.01	91.7±5.0	91.5±5.5	89.4±6.3	87.1±7.3	86.3±6.3	92.2±4.9	71.2±14.3	87.2±7.1	67.3±8.1	81.5±5.5	86.2±7.6	**94.5±4.7**
	.05	90.7±5.5	90.9±5.9	85.2±9.1	81.6±9.6	80.6±7.1	88.3±6.2	71.9±14.3	81.5±8.5	59.8±4.6	74.1±9.3	84.2±6.3	**92.4±7.2**
	.10	89.5±6.1	90.2±6.3	81.8±11.2	77.4±11.1	76.2±7.3	85.6±7.0	72.5±15.5	78.2±9.1	56.7±4.1	69.2±5.2	67.3±5.2	**91.5±5.7**
	.20	86.3±7.7	88.4±7.6	77.4±13.6	72.5±12.6	69.3±6.3	81.9±8.1	70.8±16.0	74.8±9.4	53.9±2.9	65.2±11.4	61.3±12.7	**88.9±9.2**
CIFAR-10	.00	63.8±9.0	59.9±6.7	56.1±10.2	56.2±13.2	60.9±9.4	73.3±8.4	50.8±4.7	**77.9±7.2**	63.5±8.0	78.2±7.3	75.4±5.2	73.2±7.3
	.01	63.8±9.3	59.9±6.7	56.3±10.4	56.2±13.1	60.5±9.4	72.8±8.1	51.1±4.7	76.5±7.2	72.9±7.3	66.2±7.2	74.2±6.2	**79.3±3.9**
	.05	62.6±9.2	59.6±6.4	55.6±10.5	55.7±13.3	59.6±9.8	71.5±8.2	50.1±2.9	74.0±6.9	62.2±8.2	69.3±6.4	71.0±6.8	**78.2±3.2**
	.10	62.9±8.2	59.1±6.6	54.9±11.1	55.4±13.3	58.6±10.0	69.8±8.4	50.5±3.6	71.8±7.0	60.6±8.3	64.2±10.2	69.2±9.7	**76.7±5.4**
	.20	61.9±8.1	58.3±6.2	54.2±11.1	54.6±13.3	57.0±10.6	67.9±8.1	50.1±1.7	68.5±7.1	58.5±6.7	58.2±5.2	66.2±8.2	**71.1±6.2**

performs than other strategies. In overall, the model trained by the reconstruction error-based approach shows the lowest performance. The sampling approach using $l2$-distance on the latent feature space shows much better performance compared with the reconstruction-error-based method. The probabilistic model-based sampling achieved almost similar performance with $l2$-distance-based method.

However, as the contamination rate of training data increases, the gap between the proposed method and the rest of the sampling methods widens. In particular, when more than 10% of the data is contaminated, the performance of simple Euclidean distance-based or error-based methods quickly degrades. In particular, the performance deviation of methods based on probabilistic models fell more and more rapidly.

This means that when we first simulated (see Fig. 1), if the data were contaminated at a high rate, there would be a high probability that the data would be centred in the distribution of the data, or that the data would itself have a high probability distribution. Experimental results prove that the proposed sampling method is a method for finding contaminated samples that are robust to the degree of data contamination.

6. Comparison with other methods

6.1. Results on MNIST, MNIST Fashion. and CIFAR-10 datasets

We consider the OC-SVM [19], isolation forest (IF) [62], and KDE [63] for shallow unsupervised baselines. For deep unsupervised competitors, we consider

Table 3: Performance comparison using AUC-ROC values on unsupervised anomaly detection using ODDs. The **bolded** figures indicate the best performances.

Dataset	Satellite	Cardio	Thyroid	Satimage-2	Shuttle	Mammography
OC-SVM	95.1±0.2	98.5±0.3	98.3±0.9	99.4±0.8	99.4±0.9	71.4±9.7
IF	94.2±0.2	91.4±1.1	95.2±0.3	99.2±1.2	93.4±0.9	91.4±3.6
KDE	66.7±5.8	99.6±1.7	91.6±2.3	99.5±0.2	87.2±7.4	67.2±12.5
DeepSVDD	79.8±4.1	84.8±3.6	72.0±9.7	98.3±1.4	98.7±0.2	97.4±0.5
SSAD	96.2±0.3	98.8±0.3	97.9±1.9	99.9±0.1	98.9±0.4	92.6±2.4
SS-DGM	95.7±0.1	95.2±1.3	95.8±0.7	99.2±0.2	97.5±0.4	96.4±2.3
AE	73.5±9.4	89.6±6.7	95.4±2.7	99.1±1.7	95.6±0.7	82.7±9.4
Deep SAD	91.5±1.1	95.0±1.6	**98.6±0.9**	99.9±0.1	99.3±0.1	93.0±0.5
Classification	87.2±2.1	83.2±9.6	97.8±2.6	99.9±0.1	98.3±0.2	79.5±15.8
DAGMM	73.9±3.1	88.5±3.3	96.4±0.7	99.7±0.2	99.0±0.2	75.9±7.9
LatentOut	92.4±0.5	99.0±0.1	98.1±0.3	99.8±0.1	99.7±0.1	94.2±0.7
NCAE	**97.3±0.2**	**99.2±0.2**	98.5±0.1	**99.9±0.1**	**99.9±0.1**	**98.7±0.2**

Table 4: Performance comparison using AUC-PR values on unsupervised anomaly detection using ODDs. The **bolded** figures indicate the best performances.

Dataset	Satellite	Cardio	Thyroid	Satimage-2	Shuttle	Mammography
OC-SVM	72.5±5.6	82.6±6.2	78.9±9.8	94.8±1.5	96.2±2.6	43.7±9.5
IF	64.8±8.2	83.2±7.5	61.6±11.6	89.5±3.2	87.2±6.8	28.6± 15.7
KDE	48.2±17.8	86.2±4.3	65.4±9.4	90.3±6.7	73.6±6.1	28.4±3.7
DeepSVDD	64.3±4.2	81.6±2.4	51.4±4.2	91.4±1.2	92.6±1.2	52.6±1.2
SSAD	70.2±1.8	89.4±0.7	89.5±0.5	93.2±7.9	97.2±0.5	42.5±4.5
SS-DGM	69.5±22.6	42.5±9.8	92.6±0.3	94.2±0.5	91.4±2.3	48.2±1.8
AE	41.7±13.9	60.2±11.4	75.2±7.1	84.6±2.6	90.3±0.7	18.2±7.6
Deep SAD	83.3±0.7	96.8±0.8	92.3±0.4	96.2±0.2	95.2±0.3	59.1±4.6
Classification	90.2±1.1	92.3±0.7	90.2±2.6	94.1±0.5	96.2±0.2	60.3±1.1
DAGMM	68.4±2.1	62.3±8.6	87.5±2.7	95.0±0.4	95.9±1.3	12.5±7.0
LatentOut	85.4±0.9	**99.5±0.1**	94.7±0.2	96.1±0.3	96.5±0.1	**64.6±1.2**
NCAE	**90.4±0.5**	99.3±0.2	**95.1±0.6**	**96.4±0.2**	**98.2±0.4**	64.2±0.7

general binary classifier (supervised), convolutional autoencoders (CAE), deep support vector data description (Deep SVDD) [17], semi-supervised anomaly detection (SSAD) [2], semi-supervised deep generative model (SS-DGM) [16], Deep autoencoding gaussian mixture model (DAGMM) [6], LatentOut [64], and deep semi-supervised anomaly detection (Deep SAD) [2]. We repeat this training set generation process ten times per AD set up over all the nine respective anomaly classes and report the average results over the resulting 90 experiments per contamination ratio.

Table 2 shows the quantitative performance comparison depending on the contamination ratio ρ. In the comparison using the MNIST dataset, the proposed NCAE achieves the best performances except when the dataset is not contaminated

($\rho = 0.0$). Even compared with semi-supervised approaches (SSAD and Deep SVDD) [17, 2] which use explicit anomaly samples in the training phase, the NCAE shows outstanding performances. This trend is also shown in the performance comparison using the Fashion-MNIST and CIFAR-10 datasets also. The NCAE produces the AUC of 94.5 and 88.9 for the Fashion-MNIST dataset with 1% and 20% contamination ratios, respectively. Also, it achieves the AUC of 79.3 and 71.1 for the CIFAR-10 dataset with 1% and 20% contamination ratios, respectively. Those figures perform best among the listed methods when a dataset is contaminated. Compared with other methods, which degrade their performance significantly when the contamination ratio is increased, the NCAE AD performances are relatively robust to the contamination ratios. Those figures are the best performance among the listed methods when a dataset is contaminated.

The interpretation of the relatively low performance on the uncontaminated dataset ($\rho = 0.0$) is as follows. Basically, our method is derived under the assumption that a training dataset is contaminated. Therefore, even if the dataset is not contaminated, the NCAE tries to find some anomaly samples and maximise the reconstruction errors of the samples during the model training. The minimisation process is reformulated by the error minimisation between the contaminated samples and generated normal samples (See Eq. 8). This process degrades the performance of our methods as shown in the experimental results. This is a critical defect of our method.

Overall, the comparison results demonstrate the advantage of the proposed NCAE that can detect anomaly samples on data contamination without prior knowledge or explicit abnormal samples in the training phase.

6.2. Results on ODDs

Table 3 shows the Area under the ROC curve (AUC-ROC) of various AD methods on ODDs. Table 4 shows the Area under the precision-recall curve (AUC-PR) of various AD methods on ODDs. The detailed information on OODs used for our experiments is shown in Table 1. The experimental results on Table 3 and Table 4 demonstrate that the proposed NCAE is robust to data contamination. For all categories, the NCAE produces the best performances. For the AUC-ROC, NCAE achieves 97.3, 99.2, 98.5, and 99.9 for Satellite, Cario, Thyroid, and Satimage-2 datasets, respectively. Except for the result using the Thyroid dataset, those results outperform the performances of other methods. For the AUC-PR, NCAE achieves 90.4, 99.3, 95.1, and 96.4 for Satellite, Cario, Thyroid, and Satimage-2 datasets, respectively. For the results using Cardio and Mammography, LatentOut [64] achieves better performances with small margins.

For AUC-ROC results using the Satellite dataset, the method that recorded the second-highest performance was SSAD. SSAD achieved an AUC of 96.2. This result is 1.1 lower than the proposed NCAE, and the variation of performances is also 0.1 higher than the NCAE. Among the experiments using the Cardio dataset, the second highest method is also SSAD. SSAD records an AUC of 98.8 in trials using Cardio. In the experiment using Thyroid, it is the only one that failed to achieve the best performance of the NCAE. The NCAE produces an AUC of 98.5, which is the second-ranked performance. The Deep SAD achieves the best performance on the Thyroid dataset. Deep SAD achieves an AUC of 98.6, which is 0.1 higher than that of NCAE. However, the performance variation of the Deep SAD is 0.9, which is much higher than the 0.1 of NCAE.

These results show that the NCAE's AD performance is slightly lower than the Deep SAD, but the NCAE's performance is much more stable. For AUC-PR results using the Satellite dataset, LatentOut [64] achieve partially better performances in Cardio and Mannography datasets. LatentOut achieved a 99.5 and 64.6 of AUC-PR, respectively. This result is 0.2 and 0.4 higher than the proposed NCAE. For the variation, the NCAE achieves better variation. The NCAE usually shows smaller variation, which can be interpreted as the performance of the NCAE fluctuating less than other methods.

Overall, the experimental results of AUC-ROC and AUC-PR on ODDs show that the proposed NCAE can provide more robust AD performance on data contamination. The quantitative evaluation results prove that the proposed NCAE model shows higher performance than the comparison targets in most of the data on contamination level, and at the same time, the variation in performance is not large, even in repeated experiments. In conclusion, the proposed NCAE can be interpreted as having better AD performance for contaminated data than the comparison methods.

7. Conclusion

In this work, we have proposed a Normality-Calibrated Autoencoder (NCAE) that is a generative method for fully unsupervised anomaly detection on contaminated data. The proposed NCAE extracts latent features based on AE structure and compiles latent feature distribution to a well-known distribution such as Gaussian distribution. After that based on the decoder part and a discriminator, NCAE applies adversarial learning to generate high-confidence normal samples. Based on the generated high-confidence normal sample, the NCAE identifies contaminated data and applies it to the training model for minimising reconstruction error

between the contaminated sample and randomly selected high-confidence normal samples. The experimental results have suggested that the NCAE outperforms existing methods for fully unsupervised anomaly detection by a large margin, and they have also provided competitive performances compared with semi-supervised methods using explicit abnormal samples to train their AD model.

However, there is a drawback that we should solve in our future work. Even though the proposed NCAE achieves state-of-the-art AD performance in various datasets with various contamination ratios. The performance of NCAE is affected by the hyper-parameters τ and σ. Finding optimal values of hyper-parameters is a common issue in machine learning and deep learning studies. Additionally, to train the NCAE, we assume that a dataset is always contaminated. As we mentioned in the performance comparison with other methods, since the NCAE always assumes that some data samples have been contaminated, the experimental results using a non-contaminated dataset are a bit lower than others. This issue will be addressed in our future works.

References

[1] M. Kim, J. Kim, J. Yu, J. K. Choi, Active anomaly detection based on deep one-class classification, Pattern Recognition Letters 167 (2023) 18–24.

[2] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, M. Kloft, Deep semi-supervised anomaly detection, arXiv preprint arXiv:1906.02694 (2019).

[3] H. Song, Z. Jiang, A. Men, B. Yang, A hybrid semi-supervised anomaly detection model for high-dimensional data, Computational Intelligence and Neuroscience (2017).

[4] S. Akcay, A. Atapour-Abarghouei, T. P. Breckon, GANomaly: Semi-supervised anomaly detection via adversarial training, in: ACCV, 2018, pp. 622–637.

[5] R. Chalapathy, S. Chawla, Deep learning for anomaly detection: A survey, arXiv preprint arXiv:1901.03407 (2019).

[6] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho, H. Chen, Deep autoencoding gaussian mixture model for unsupervised anomaly detection, in: International Conference on Learning Representations, 2018.

[7] X. Xia, X. Pan, N. Li, X. He, L. Ma, X. Zhang, N. Ding, Gan-based anomaly detection: A review, Neurocomputing 493 (2022) 497–535.

[8] W. Chen, L. Tian, B. Chen, L. Dai, Z. Duan, M. Zhou, Deep variational graph convolutional recurrent network for multivariate time series anomaly detection, in: International Conference on Machine Learning, PMLR, 2022, pp. 3621–3633.

[9] J. Wang, P. Neskovic, L. N. Cooper, Pattern classification via single spheres, in: International Conference on Discovery Science, Springer, 2005, pp. 241–252.

[10] Y. Liu, Y. F. Zheng, Minimum enclosing and maximum excluding machine for pattern description and discrimination, in: ICPR, 2006, pp. 129–132.

[11] N. Görnitz, M. Kloft, K. Rieck, U. Brefeld, Toward supervised anomaly detection, Journal of Artificial Intelligence Research 46 (2013) 235–262.

[12] A. Berg, J. Ahlberg, M. Felsberg, Unsupervised learning of anomaly detection from contaminated image data using simultaneous encoder training, arXiv preprint arXiv:1905.11034 (2019).

[13] T. Li, Z. Wang, S. Liu, W.-Y. Lin, Deep unsupervised anomaly detection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 3636–3645.

[14] C.-H. Lai, D. Zou, G. Lerman, Robust subspace recovery layer for unsupervised anomaly detection, in: International Conference on Learning Representations, 2020.
URL https://openreview.net/forum?id=rylb3eBtwr

[15] S. Yoon, Y.-K. Noh, F. Park, Autoencoding under normalization constraints, in: International Conference on Machine Learning, PMLR, 2021, pp. 12087–12097.

[16] D. P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, in: ICLR, 2015.

[17] L. Ruff, R. A. Vandermeulen, N. Görnitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, M. Kloft, Deep one-class classification, in: ICML, Vol. 80, 2018, pp. 4390–4399.

[18] L. Ruff, J. R. Kauffmann, R. A. Vandermeulen, G. Montavon, W. Samek, M. Kloft, T. G. Dietterich, K.-R. Müller, A unifying review of deep and shallow anomaly detection, Proceedings of the IEEE (2021).

[19] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, R. C. Williamson, Estimating the support of a high-dimensional distribution, Neural computation 13 (7) (2001) 1443–1471.

[20] D. M. Tax, R. P. Duin, Support vector data description, Machine learning 54 (1) (2004) 45–66.

[21] R. Chalapathy, A. K. Menon, S. Chawla, Anomaly detection using one-class neural networks, arXiv preprint arXiv:1802.06360 (2018).

[22] I. Golan, R. El-Yaniv, Deep anomaly detection using geometric transformations, in: Advances in Neural Information Processing Systems, 2018.

[23] D. Hendrycks, M. Mazeika, S. Kadavath, D. Song, Using self-supervised learning can improve model robustness and uncertainty, in: Advances in Neural Information Processing Systems, 2019, pp. 15637–15648.

[24] L. Bergman, Y. Hoshen, Classification-based anomaly detection for general data, arXiv preprint arXiv:2005.02359 (2020).

[25] S. Ahmad, A. Lavin, S. Purdy, Z. Agha, Unsupervised real-time anomaly detection for streaming data, Neurocomputing 262 (2017) 134–147.

[26] Ł. Maziarka, M. Śmieja, M. Sendera, Ł. Struski, J. Tabor, P. Spurek, Oneflow: One-class flow for anomaly detection based on a minimal volume region, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (11) (2021) 8508–8519.

[27] X. Wang, G.-J. Qi, Contrastive learning with stronger augmentations, IEEE transactions on pattern analysis and machine intelligence 45 (5) (2022) 5549–5560.

[28] E. Parzen, On estimation of a probability density function and mode, The annals of mathematical statistics 33 (3) (1962) 1065–1076.

[29] D. W. Scott, Multivariate density estimation: theory, practice, and visualization, John Wiley & Sons, 2015.

[30] S. Roberts, L. Tarassenko, A probabilistic resource allocating network for novelty detection, Neural Computation 6 (2) (1994) 270–284.

[31] D. P. Kingma, M. Welling, Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114 (2013).

[32] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, G. Langs, Unsupervised anomaly detection with generative adversarial networks to guide marker discovery, in: International conference on information processing in medical imaging, 2017.

[33] M. Z. Zaheer, A. Mahmood, M. H. Khan, M. Segu, F. Yu, S.-I. Lee, Generative cooperative learning for unsupervised video anomaly detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 14744–14754.

[34] J. An, S. Cho, Variational autoencoder based anomaly detection using reconstruction probability, Special Lecture on IE 2 (1) (2015) 1–18.

[35] P. Perera, R. Nallapati, B. Xiang, Ocgan: One-class novelty detection using gans with constrained latent representations, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[36] B. Nachman, D. Shih, Anomaly detection with density estimation, Physical Review D 101 (7) (2020) 075042.

[37] M. Sakurada, T. Yairi, Anomaly detection using autoencoders with nonlinear dimensionality reduction, in: Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, 2014.

[38] C. Zhou, R. C. Paffenroth, Anomaly detection with robust deep autoencoders, in: International Conference on Knowledge Discovery and Data Mining, 2017.

[39] J. Fan, Q. Zhang, J. Zhu, M. Zhang, Z. Yang, H. Cao, Robust deep autoencoding gaussian process regression for unsupervised anomaly detection, Neurocomputing 376 (2020) 180–190.

[40] G. E. Hinton, R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, science 313 (5786) (2006) 504–507.

[41] J. Masci, U. Meier, D. Cireşan, J. Schmidhuber, Stacked convolutional auto-encoders for hierarchical feature extraction, in: International conference on artificial neural networks, 2011.

[42] M. Thill, W. Konen, H. Wang, T. Bäck, Temporal convolutional autoencoder for unsupervised anomaly detection in time series, Applied Soft Computing 112 (2021) 107751.

[43] J. T. Andrews, E. J. Morton, L. D. Griffin, Detecting anomalous data using auto-encoders, International Journal of Machine Learning and Computing 6 (1) (2016) 21.

[44] S. M. Erfani, S. Rajasegarar, S. Karunasekera, C. Leckie, High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning, Pattern Recognition 58 (2016) 121–134.

[45] T. Ergen, S. S. Kozat, Unsupervised anomaly detection with lstm neural networks, IEEE transactions on neural networks and learning systems 31 (8) (2019) 3127–3141.

[46] F. Liu, C. Zeng, L. Zhang, Y. Zhou, Q. Mu, Y. Zhang, L. Zhang, C. Zhu, Fedtadbench: Federated time-series anomaly detection benchmark, in: 2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys), IEEE, 2022, pp. 303–310.

[47] S. Pidhorskyi, R. Almohsen, G. Doretto, Generative probabilistic novelty detection with adversarial autoencoders, in: Advances in Neural Information Processing Systems, 2018, pp. 6822–6833.

[48] Y. Zhou, X. Song, Y. Zhang, F. Liu, C. Zhu, L. Liu, Feature encoding with autoencoders for weakly supervised anomaly detection, IEEE Transactions on Neural Networks and Learning Systems 33 (6) (2021) 2454–2465.

[49] Y. Xia, X. Cao, F. Wen, G. Hua, J. Sun, Learning discriminative reconstructions for unsupervised outlier removal, in: Proceedings of the IEEE International Conference on Computer Vision, 2015.

[50] L. Beggel, M. Pfeiffer, B. Bischl, Robust anomaly detection in images using adversarial autoencoders, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2019.

[51] G. Pang, C. Yan, g. C. Shen, A. v. d. Hengel, X. Bai, Self-trained deep ordinal regression for end-to-end video anomaly detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2020.

[52] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, B. Frey, Adversarial autoencoders, arXiv preprint arXiv:1511.05644 (2015).

[53] I. Shim, T.-H. Oh, I. S. Kweon, High-fidelity depth upsampling using the self-learning framework, Sensors 19 (1) (2018) 81.

[54] Z. Lin, A. Khetan, G. Fanti, S. Oh, Pacgan: The power of two samples in generative adversarial networks, Advances in neural information processing systems 31 (2018).

[55] J. Yu, Y. Lee, K. C. Yow, M. Jeon, W. Pedrycz, Abnormal event detection and localization via adversarial event prediction, IEEE Transactions on Neural Networks and Learning Systems (2021).

[56] N.-C. Ristea, N. Madan, R. T. Ionescu, K. Nasrollahi, F. S. Khan, T. B. Moeslund, M. Shah, Self-supervised predictive convolutional attentive block for anomaly detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 13576–13586.

[57] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative Adversarial Nets, in: NIPS, 2014, pp. 2672–2680.

[58] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (11) (1998) 2278–2324.

[59] H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, arXiv preprint arXiv:1708.07747 (2017).

[60] A. Krizhevsky, G. Hinton, et al., Learning multiple layers of features from tiny images (2009).

[61] M. A. Pimentel, D. A. Clifton, L. Clifton, L. Tarassenko, A review of novelty detection, Signal processing 99 (2014) 215–249.

[62] F. T. Liu, K. M. Ting, Z.-H. Zhou, Isolation Forest, in: ICDM, 2008, pp. 413–422.

[63] E. Parzen, On Estimation of a Probability Density Function and Mode, The annals of mathematical statistics 33 (3) (1962) 1065–1076.

[64] F. Angiulli, F. Fassetti, L. Ferragina, Latent o ut: an unsupervised deep anomaly detection approach exploiting latent space distribution, Machine Learning 112 (11) (2023) 4323–4349.

Normality-Calibrated Autoencoder for Unsupervised Anomaly Detection on Data Contamination

Jongmin Yu[a,e,*], Minkyung Kim[b,c], Junsik Kim[d], Hyeontaek Oh[e]

[a]*Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Wilberforce Rd, Cambridge, CB3 0WA, United Kingdom*
[b]*Division of Surgical Oncology, Department of Otolaryngology-Head and Neck Surgery, Mass Eye and Ear, Boston, MA 02114, U.S.A*
[c]*Department of Otolaryngology-Head and Neck Surgery, Harvard Medical School, Boston, MA 02115, U.S.A*
[d]*School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138, U.S.A*
[e]*ProjectG.AI, 291 Daehak-ro, Yuseong-gu, Daejeon, 34130, Republic of Korea*

Abstract

Anomaly detection, or outlier detection, refers to identifying rare or abnormal instances or patterns within a dataset that deviate significantly from the expected or normal behaviour. Various methods have been proposed, but most assume that their training datasets take full, complete integrity. However, the innocent integrity of data is not easy to maintain in reality. Existing anomaly detection methods generally see given data as a single class and learn features that can represent it well, but this approach is very vulnerable to data contamination. This paper proposes a Normality-Calibrated Autoencoder (NCAE), which can boost anomaly detection performance on the contaminated datasets without any prior information or explicit abnormal samples in the training phase. The NCAE adversarially generates highly confident normal samples from a latent space with low entropy and leverages them to predict abnormal samples in a training dataset. NCAE is trained to minimise reconstruction errors in uncontaminated samples and maximise reconstruction errors in contaminated samples. The experimental results demonstrate that our method outperforms shallow, hybrid, and deep methods for unsupervised anomaly detection and achieves comparable performance compared with semi-supervised methods using labelled anomaly samples in the training phase.

Keywords: Unsupervised anomaly detection, normality calibration, autoencoder,

*The corresponding author

data contamination, data pollution

1. Introduction

Anomaly detection (AD) generally assumes that given data are a single class, focusing on learning features that can represent it well. According to Kim *et al.* [1], this assumption is defined as the *'Normality assumption'*. The model learned in this way compares the features extracted from the data input in the actual anomaly detection step with the learned features to calculate an error or probability, and then it compares the calculated result with a predetermined threshold to detect anomalies in the data. Currently, most models assume complete integrity of the data given to training the models; that is, all data consists of only a single class (normal) and no noise (anomaly) associated with outliers in the data exists.

However, it is not easy to guarantee data integrity in practice. Datasets in the real world are easily *contaminated*, which means that datasets contain both normal and abnormal samples. In particular, as the number of data to be learned increases, the probability of including errors also increases proportionally. The contaminated samples significantly degrade AD models' robustness, reliability, and accuracy and increase the uncertainty of the detection results.

Various methods have been proposed [2, 3, 4, 5, 6, 7] to improve the robustness of AD methods on contaminated datasets. In the beginning, filtering contaminated samples based on contamination ratio [6, 2, 8] and semi-supervised learning approaches that use explicit abnormal samples in the training step [9, 10, 11, 2] have been presented. However, the approaches above are domain or data-type-specific, and their performance highly depends on hyper-parameter settings. For example, the methods using contamination ratios or prepared information about abnormal data work well if precious contamination ratios or abnormal samples can be provided. However, it is a more reasonable and practical hypothesis that the contamination ratio cannot be usually estimated. Also, for the semi-supervised approaches [9, 10, 11, 2], we cannot guarantee that the prepared abnormal data will cover all other unobserved data anomalies. Moreover, even if we successfully find out the contamination ratio of a particular domain or we finally find some specific abnormal data which can cover all other data anomalies for the domain, they can only be applied to derive AD models in that particular domain, not others. Consequently, developing more generalised solutions for AD robust to contaminated datasets is still very challenging but important.

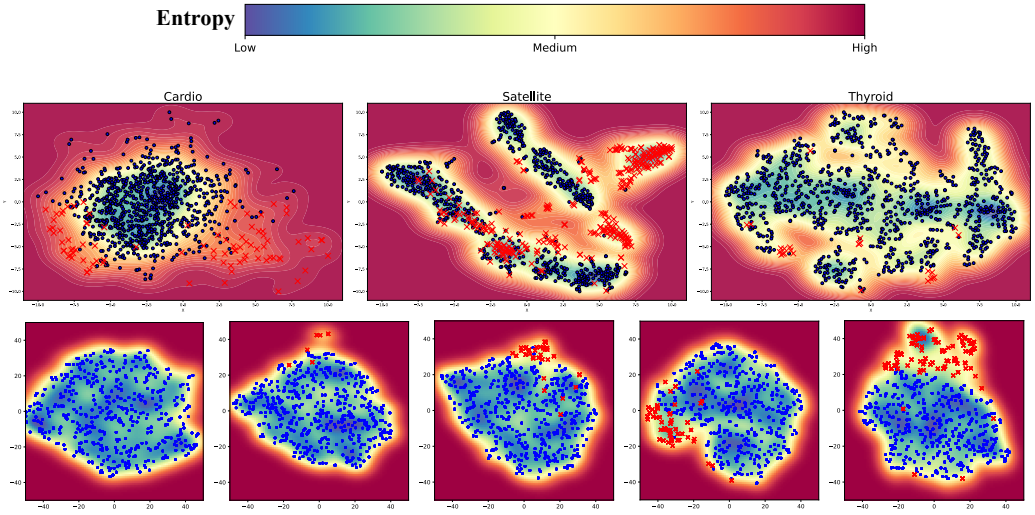
To address this issue, AD methods based on contamination sample prediction using geometric distance measurements [12, 13, 14] have been proposed. These

methods assume that contaminated data is always distributed far away from the data distribution's centre or in the highest entropy space. Then, the methods sorted the contaminated data in ascending or descending order by distance from the centre of the data distribution or by entropy, and they filtered out data that was contaminated by a specific percentage [12, 13, 14]. However, as shown in Fig 1, if a training dataset is highly contaminated (like the Satellite dataset, in which samples of 31.6% are contaminated or over 10% of data samples on a simulated dataset), the contaminated samples can also form a low entropy space by themselves, even when the contamination ratio is not high enough, abnormal samples can be positioned some low-entropy space. As a result, the development of an AD method that does not require prior information about data anomalies and does not take geometric solid assumptions when finding contaminated samples is essential.

This paper presents a Normality-Calibrated Autoencoder (NCAE), which is robust to the training dataset contamination. Our key idea for the NCAE is to adversarially generate highly confident normal samples from a low entropy feature space and then contrastively compare the generated samples with the input samples to estimate the contamination score. We pay attention to the fact that if adversarial learning inputs data biased to a specific class in optimizing the generator and discriminator, the generator repeatedly generates only particular data instead of generating various data.

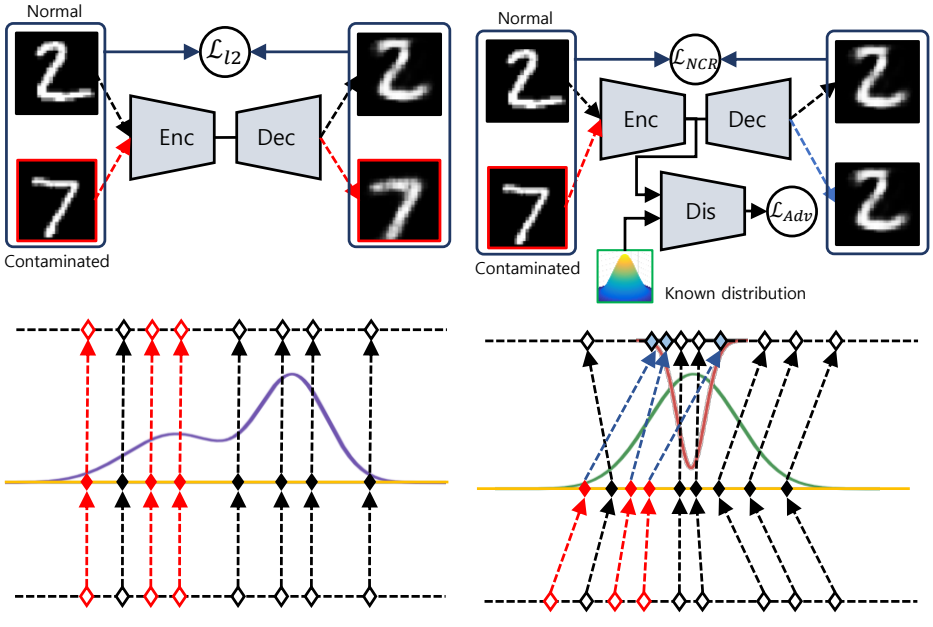
Fig.2 shows the architectural difference between an autoencoder (AE) and the proposed NCAE. The NCAE has a structure that combines an AE and a discriminator for applying adversarial learning. Data is compressed into low-dimensional latent features through the encoder, and the distribution of these latent features is induced into a specific data distribution through adversarial learning (See $\mathcal{L}_{Adv}$ in Fig. 2(b)). This process reduces the uncertainty of the data distribution and minimises the blind spot [15] on the latent feature space. Based on the decoder part of the AE and the specific data distribution used for adversarial learning, samples of the region with the highest probability (*a.k.a.* the lowest entropy) are generated. The generated samples are used to distinguish normal samples from contaminated samples. After identifying the contaminated samples, NCAE calibrates the normality of an AD model by maximising the reconstruction error of the found samples.

To demonstrate the effectiveness of the proposed NCAE as a robust method, we conduct experiments of AD on contaminated datasets using various datasets having diverse contamination ratios. In the performance comparison with the existing state-of-the-art (SOTA) AD methods, the NCAE achieves better performance and more robustness in data contamination. Particularly, compared to the methods



(a) 0% (b) 1% (c) 5% (d) 10% (e) 20%

Figure 1: Visualisation of entropy and distribution of latent features of the various datasets considering different contamination ratios. The first row indicates the visualisation of the 'Cardio', 'Satellite', and 'Thyroid' datasets among the outlier detection datasets. The ratios of the data contamination are 9.6%, 31.6%, and 2.5%, respectively. The second row illustrates the visualisation results using a simulated dataset based on the MNIST dataset, considering various contamination ratios. (a) 0% (No contamination), (b) 1%, (c) 5%, (d) 10%, and (e) 20%. The samples on the '5' class on the MNIST dataset are used as normal (blue dots), and contaminated samples (red x-marks) are randomly picked from the training samples of the remaining classes. The entropy of each sample is computed by the probability estimated by the kernel density estimation (KDE). The 500 samples are randomly picked for visualisation. When a dataset is highly contaminated (*i.e.,* contamination ratio over 10%), contaminated samples are also located in a low entropy region.



(a) Autoencoder (b) NCAE

Figure 2: Comparison of an autoencoder (AE) and the proposed normality-calibrated autoencoder (NCAE). (a) denotes the architectural details and mapping functionality of the AE. (b) indicates the details and functionality of the NCAE. The AE reconstructs the contaminated samples without any concerns (Red dotted lines). By applying adversarial learning $\mathcal{L}_{\text{adv}}$, NCAE removes uncertainty from the latent trait distribution by deriving it from a well-known distribution. After that, the NCAE generate high-confidence normal samples from the centre of the distribution (Red coloured distribution). In computing the reconstruction error, compared with AE, which minimises the reconstruction error of the contaminated samples without any concerns, the NCAE tries to minimise the reconstruction error of contaminated samples with randomly generated high-confidence normal samples (Blue coloured diamonds).

using prior information such as contamination ratio [16, 17] and prepared data anomalies [2], which showed a large performance fluctuation due to errors in the prior information, the proposed method shows robust AD performance against data contamination without prior information.

The contributions of this paper are summarised as follows:

- Normality-calibrated autoencoder (NCAE). A new AD method is robust to data contamination without any prior knowledge about the data contamination.

- Generative adversarial learning for identifying contaminated data and joint learning scheme for the NCAE. We propose adversarial learning to generate a high-confidence normal data sample and apply it to find contaminated data during the model training. Moreover, we propose an algorithm that efficiently optimises the proposed joint learning models for the NCAE.
- Comprehensive experimental results of AD on data contamination. We provide various ablation studies and performance comparisons with existing SOTA AD methods on data contamination.

The remainder of this paper is organised as follows. Section 2 introduces various related works about AD methods considering data contamination. Section 3 describes the detailed information of the proposed NCAE and its training process. Section 4 provides the experimental settings, and Section 5 presents comprehensive ablation studies for the proposed hyper-parameters. With the best performance hyper-parameters, Section 6 compares AD performance on contaminated data with existing SOTA AD methods. This paper is concluded in Section 7.

2. Related works

Drawing on the research presented in the survey by Ruff *et al.* [18], there are several primary types of anomaly detection methodologies that hinge on the normality assumption. The first of these groups primarily uses one-class classification-based strategies. These strategies work by creating a discerning decision boundary that converts regular data into a succinct representation. This group includes predominant methods such as the One-Class SVM (OC-SVM) [19] and Support Vector Data Description (SVDD) [20], both of which are shallow models. In contrast, methods like OC-NN [21] and Deep SVDD [17] offer deep learning alternatives. These strategies utilise deep neural networks, replacing the shallow models yet maintaining the same goal functions as their OC-SVM and SVDD counterparts. These techniques delineate a hyperplane and a minimum-volume hypersphere, encapsulating the standard data within a latent space. The principle of AD using those methods is simply considering data located outside of the hyperplane or hypersphere as abnormals.

Conversely, there is another type of classification-based methods [22, 23, 24, 25] that experiments with the use of self-supervised learning, especially for image data. These techniques evaluate normality based on the errors arising from proxy tasks, such as rotation, flipping, or patch rearrangement in augmentation classifications. Furthermore, a novel approach has been introduced recently by Maziarka *et*

al. [26], presenting a flow-based one-class classifier aiming to identify the smallest volume bounding area. However, those approaches are only valid for domains that generate anomalies by proxy tasks that are similar to actual anomalies. For instance, up-down flipping can be thought of as an abnormal image, but left-right flipping may be considered data augmentation [27].

Probabilistic model-based methodologies often employ shallow non-parametric density estimators such as Kernel Density Estimation (KDE) [28, 29] and Gaussian Mixture Models (GMM) [30, 6]. However, the above methods can not model complicated data distribution. Recently, there is a growing trend in anomaly detection research to adopt deep generative models like Variational Autoencoders (VAE) [31], Generative Adversarial Networks (GAN) [32, 33], and Anomaly Detection (AD) techniques based on Normalising Flows [34, 35, 36]. These models aim to decipher the latent feature space data distribution. However, VAE sometimes causes posterior collapse, in which the VAE decoder ignores the actual data distribution and generates a sample from a Gaussian distribution. Also, unstable training of GANs is still troublesome.

In the realm of reconstruction-based methods [37, 38, 39], Autoencoders (AE) [40, 41, 42] are predominantly used. Numerous adaptations of this method have been introduced to improve its performance in anomaly detection. Further, attempts have been made to merge the discriminating representation of AE with shallow methods [43, 44, 45, 46]. Unfortunately, the autoencoder has a blind stop. When an AE is trained, we expect that unseen samples output larger reconstruction errors. However, there are several studies that AD still can reconstruct unseen samples with small reconstruction errors [15, 47].

Despite extensive research in anomaly detection using normality modelling, these techniques heavily depend on the availability of a pure training dataset free of anomalous data, which is often difficult to obtain. This dependence often leads to less-than-optimal performance and limits the methods' application. The one-class classification-based methods, although able to handle contaminated datasets by adjusting a contamination ratio as a hyper-parameter, also depend largely on prior knowledge of this ratio. However, in real-world scenarios, this ratio is typically unknown.

To tackle the problem of learning normality from contaminated datasets, researchers have presented methods using meta-information about the contaminated data such as pollution rate [6, 2, 8], or pre-defined abnormal samples [9, 10, 11, 2, 48]. However, those methods are domain-specific, and their AD performances are highly dependent on hyper-parameter settings such as pollution ratio. More recently, researchers have proposed robust methods that aim to diminish

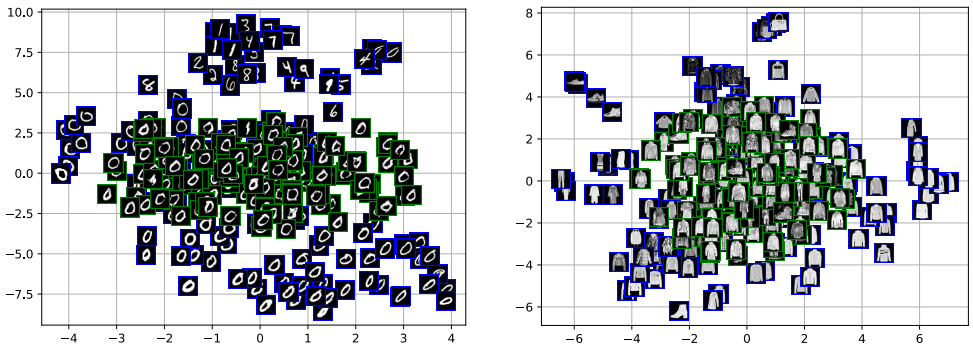
the adverse effects of anomalous data in such datasets [49, 38, 50, 51, 39]. Most of these studies have used pseudo-labelling techniques to improve the models' robustness. Different methods, including AE [49, 38, 50] or regression-based [51, 39] approaches, have been used for pseudo-labelling.

Xia et al *et al.* [49] pioneered an approach where pseudo-labelling is conducted using the reconstruction error obtained from an AE. Following this, the AE is iteratively trained to minimize the reconstruction error of pseudo-normal data, incorporating a regularization term that captures the separability of the error distribution. Using a similar framework, Zhou *et al.* [38] and Beggel *et al.* [50] implemented related techniques. Zhou *et al.* used a robust AE inspired by robust Principal Component Analysis (PCA), while Begge *et al.* utilized an adversarial AE [52].

While most of the previous research focused on AE and its variants, a few recent methods proposed the use of regression models [51, 39], trained in an iterative manner. Pang *et al.* [51] and Shim *et al.* [53] presented a two-class ordinal regression model that uses neural networks, while Fan *et al.* [39] used a Gaussian process regression model. In both approaches, an initial anomaly detection stage is carried out by a separate, pre-existing anomaly detector. Following this, the model is iteratively trained using pseudo-labeled data.

A significant limitation of the aforementioned methods is their dependency on hyper-parameters that govern the selection and quantity of data treated as pseudo-normal or pseudo-abnormal samples. However, finding optimal hyper-parameters can be difficult as they tend to fluctuate depending on the dataset and unknown contamination ratios. To tackle this problem, anomaly detection (AD) methods utilizing contamination sample prediction through geometric distance metrics have been introduced [12, 13, 14]. These techniques typically assume that contaminated data points are located far from the centre of the data distribution or within regions of high entropy. Subsequently, the methods rank the contaminated samples either in ascending or descending order based on their distance from the distribution centre or their entropy level, filtering out a specified percentage of contaminated data [12, 13, 14]. However, as illustrated in Fig. 1, when a training dataset is heavily contaminated (e.g., more than 10% of the samples), the contaminated samples may also cluster in low-entropy regions. Therefore, it is crucial to develop an AD method that does not rely on prior knowledge of data anomalies and avoids strong geometric assumptions when identifying contaminated samples.

In this paper, we introduce a Normality-Calibrated Autoencoder (NCAE) designed to be resilient to contamination within the training dataset. The central concept of NCAE involves adversarially generating highly confident normal sam-



(a) (b)

Figure 3: Visualisation of the input image and the generated image based on the scatter plot for each latent feature and sampled noise. '0' class on MNIST and the 'Coat' class on Fashion-MNIST are determined as normal classes, respectively. Each model is trained with a contamination ratio of 20%. The images with the blue border show the samples on a single training batch, and the images with the green border represent the generated images by adversarial learning.

ples from a low entropy feature space and then using a contrastive comparison between these generated samples and the input data to estimate the contamination score. We focus on the observation that when adversarial learning processes data biased toward a particular class during the optimization of the generator and discriminator, the generator tends to repeatedly produce specific data rather than a diverse set of samples [54].

3. Normality-Calibrated Autoencoder

3.1. Learning normality-calibrated autoencoder

For n number of input samples with D dimensions $\mathcal{X} = \{x_i\}_{i=1:n}$, $x \in \mathbb{R}^D$ and the corresponding latent features with d dimensions $\mathcal{Z} = \{z_i\}_{i=1:n}$, $z \in \mathbb{R}^d$, let define an autoencoder (AE) composed of an encoder $f(x) : x \rightarrow z$ and a decoder $g(x) : z \rightarrow \bar{x}$, where $\bar{x}$ denotes the reconstruction result of x. The general objective of the AE is training f and g to minimise an error between input samples x and the reconstruction results $\bar{x}$, as follows:

$$\min_{f,g} \mathbb{E}_{x \sim p_{\mathcal{X}}} ||x - \bar{x}||^2, \quad \bar{x} = g \circ f(x), \tag{1}$$

where $p_{\mathcal{X}}$ denotes the entire input samples. By minimising Eq. 1, the AE compiles a mapping function between a high dimensionality data x to a low dimensionality

feature z. Based on this process, the AE can learn much compressed and abstracted information, which can represent various features of the given data. To detect data anomalies, the AE uses the reconstruction error. If anomaly data is given, the learnt abstracted information would not reconstruct the data well so the reconstruction error would be higher than normal ones.

However, an AE is known to have an over-confidence issue [47], *i.e.,* low reconstruction error of unseen samples. This issue can be thought of as follows: if the AE takes anomaly data as their input, the reconstruction error of the data would not be high, or even similar to the error of normal data samples. As we mentioned above, AD methods using the AE usually identify abnormal samples using the reconstruction error. Therefore, if the AE takes anomaly samples as inputs, it may not distinguish whether the samples are abnormal or not [47, 55, 56]. This over-confidence issue would be deepened when a training dataset is contaminated.

One straightforward approach to prevent this issue is adding an extra term to maximise reconstruction error for contaminated samples. This is simply done by adding the negative reconstruction error for contaminated samples represented by

$$\min_{f,g}(\mathbb{E}_{x^n \sim p_{\mathcal{X}^{\mathrm{N}}}}||x^n - \bar{x}^n||^2 - \mathbb{E}_{x^{\mathrm{c}} \sim p_{\mathcal{X}^{\mathrm{C}}}}||x^{\mathrm{c}} - \bar{x}^c||^2), \tag{2}$$

where x^{n} and x^{c} indicate normal data and abnormal (or contaminated) data sampled from each distribution *i.e.,* $p_{\mathcal{X}^{\mathrm{N}}}$ and $p_{\mathcal{X}^{\mathrm{C}}}$, respectively. $\bar{x}^{\mathrm{n}}$ and $\bar{x}^{\mathrm{c}}$ denote the reconstruction results corresponding to the normal and abnormal data, respectively.

Since the reconstruction error for contaminated samples $\mathbb{E}_{x^{\mathrm{n}} \sim p_{\mathcal{X}^{\mathrm{N}}}, x^{\mathrm{c}} \sim p_{\mathcal{X}^{\mathrm{C}}}}$ is negative, to minimise the entire loss, the error should be maximised. However, this formulation is intractable when applied to optimising an AE directly because there is no bound in maximising the error terms.

This problem can be avoided by replacing the error maximisation task with a minimising task of the reconstruction error between the contaminated samples and arbitrarily assigned normal samples. Based on this principle, we define normality-calibrated reconstruction (NCR) loss as follows:

$$\min_{f,g}(\mathbb{E}_{x^n \sim p_{\mathcal{X}^{\mathrm{N}}}}||x^n - \bar{x}^n||^2 + \mathbb{E}_{\acute{x}^{\mathrm{n}} \sim p_{\mathcal{X}^{\mathrm{N}}}, x^{\mathrm{c}} \sim p_{\mathcal{X}^{\mathrm{C}}}}||x^{\mathrm{c}} - \acute{x}^{\mathrm{n}}||^2), \tag{3}$$

We transform the maximisation term of Eq. 2 (the second term) to the minimisation term by using the contaminated data and randomly picked normal data ($\acute{x}^{\mathrm{n}}$) sampled from the $p_{\mathcal{X}^{\mathrm{N}}}$. However, to optimise this loss function, we should find out which samples are contaminated to optimise AE using Eq. 3 properly.

3.2. *High-confidence normal samples generation using Generative Adversarial Network*

We find contaminated samples by using highly confident normal samples generated from low entropy latent space. We apply the generative adversarial network (GAN) [57] framework to do this. The high-confidence normal sample generation via the GAN framework is carried out as follows.

Initially, we transform a distribution of all latent features z, which are encoded from input samples x through the encoder f, to a more knowledgeable probabilistic distribution such as Gaussian distribution. And then, we generate samples using noise signals sampled from the centre of the knowledgeable distribution, *i.e.,* the lowest entropy space as you can see in Fig. 1, even though a training dataset is highly contaminated (like over 10%), the dominant data distributed in the lowest entropy space are normal data.

An adversarial loss for transforming a latent feature distribution into a more knowledgeable probabilistic distribution is defined by the following:

$$\min_{f} \max_{D_l} \mathbb{E}_{\omega \sim N(\mu_{\mathcal{Z}}, I_d)}[\log D_l(\omega)] + \mathbb{E}_{x \sim P_{\mathcal{X}}}[\log (1 - D_l(f(x)))], \tag{4}$$

where D_l denotes the discriminator for latent features, and $N(\mu_{\mathcal{Z}}, I_d)$ defines a normal distribution with the mean of latent features $\mu_{\mathcal{Z}} \in \mathbb{R}^d$ and a covariance matrix defined by an identity matrix $I_d \in \mathbb{R}^{d \times d}$. $\mu_{\mathcal{Z}}$ is initialised by the mean value of latent features: $\mu_{\mathcal{Z}} = \frac{1}{n}\sum_{i=1}^{n} z_i$. We would want each component of z to be maximally informative such as each of them to be an independent random variable. Therefore, the $d \times d$ identity matrix determines the covariance matrix.

Variational Autoencoder (VAE) [31] can be an alternative to adversarial learning in deriving the latent feature distribution into a well-known distribution. However, to apply the VAE, it is inevitable to change the network structure since extra networks are needed to sample the means and variances of latent features. We decided to use adversarial learning since it is more flexible when changing the network structure.

Since f and D_l are being updated, $\mu_{\mathcal{Z}}$ would be shifted during the training step. $\mu_{\mathcal{Z}}$ is updated at every training step as follows:

$$\mu_{\mathcal{Z}}^{t+1} = \mu_{\mathcal{Z}}^{t} - \gamma \frac{1}{m} \sum_{i=1}^{m} (\mu_{\mathcal{Z}}^{t} - z_i), \quad \mu_{\mathcal{Z}}^{0} = \frac{1}{n} \sum_{i=1}^{n} z_i^0 \tag{5}$$

where $\mu_{\mathcal{Z}}^{t+1}$ and $\mu_{\mathcal{Z}}^{t}$ denote the $\mu_{\mathcal{Z}}$ on $t+1$-th and t-th training step, respectively. m is the batch size, z_i is i-th latent features on the batch, and γ is a learning rate.

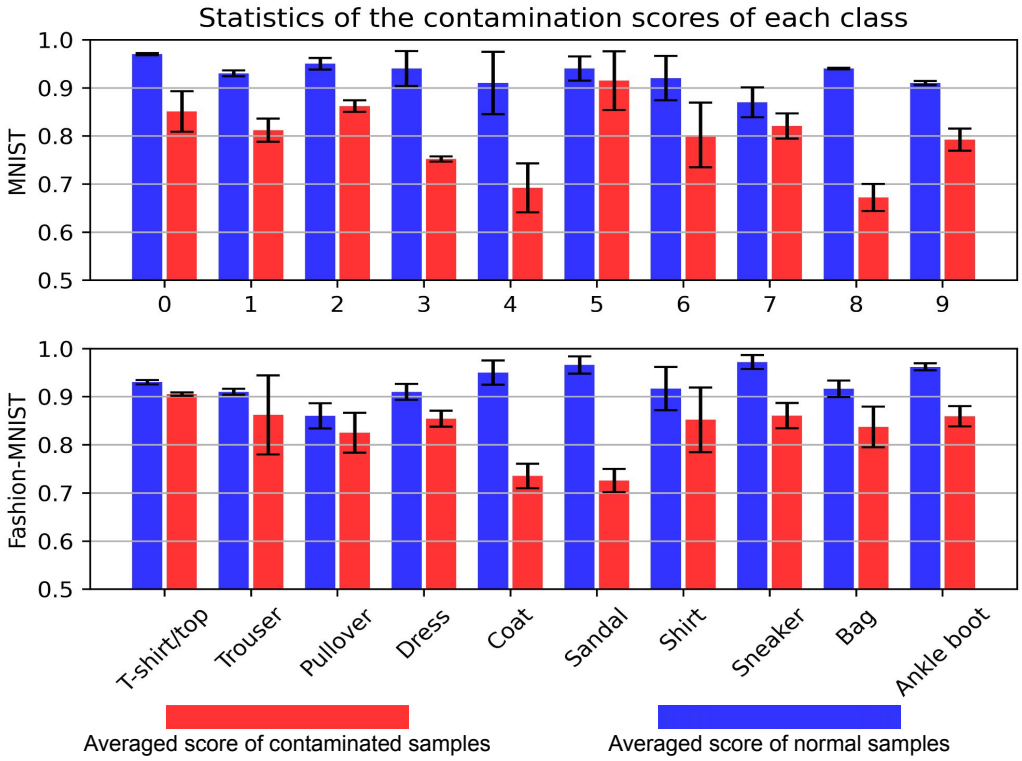


Figure 4: The means and variances of the contaminated scores for 100 times repeated experiments. The upper illustrates the averaged contamination score of contaminated and normal samples on the MNIST dataset. The lower illustrates the averaged scores on the Fashion-MNIST dataset. The black lines of each bar show a variation of the score.

To generate highly confident normal samples, we formulate the following adversarial loss:

$$\min_{g} \max_{D_s} \mathbb{E}_{x \sim P_{\mathcal{X}}}[\log D_s(x)] + \mathbb{E}_{\acute{\omega} \sim N(\mu_{\mathcal{Z}}, \sigma I_d)}[\log(1 - D_s(g(\acute{\omega}))], \tag{6}$$

where D_s denotes the discriminator for samples, and $N(\mu_{\mathcal{Z}}, \sigma I_d)$ is a d-dimensional normal distribution with the mean $\mu_{\mathcal{Z}} \in \mathbb{R}^d$ and the covariance matrix $\sigma I_d \in \mathbb{R}^{d \times d}$. $\mu_{\mathcal{Z}}$ is equivalent to the $\mu_{\mathcal{Z}}$ in Eq. 4. σI_d is defined by multiplication of a scalar value $\sigma \in [0, 1]$ and the identity matrix I_d. σ is a hyper-parameter to control the compactness of random noise for generating samples using the decoder g. The smaller σ can give more chances to generate highly confident normal samples by generating a feature close to the centre of the probability distribution.

We demonstrate the effectiveness of the proposed generative adversarial network (GAN) for generating high-confidence normal samples, we have conducted

toy experiments using MNIST and Fashion-MNIST datasets. We have trained the proposed GAN the data of the '0' class on the MNIST dataset and the 'Coat' class on the Fasion-MNIST dataset as a normal class. The contamination ratio is set by 20%, meaning the 20% portion of training data comprises other classes. As shown in Fig. 3, even though the training data is highly contaminated, the GAN trained by our approach generates normal samples.

3.3. Contaminated sample mining and joint learning

To predict contaminated samples, we use the generated high-confident normal samples as a dictionary. With the generation process for high confident normal samples: $g(\acute{\omega}) = \acute{x}$, we construct a latent feature dictionary $\mathcal{M} = [\acute{z}_i]_{i=1:m}$, $\acute{z}_i = f(\acute{x}_i)$ and $\mathcal{M} \in \mathbb{R}^{m \times d}$, where m is the batch size. By leveraging $\mathcal{M}$ and given each training batch $\{x_i\}_{i=1:m}$, we define a pseudo contamination score c_i of each input sample x_i as follow:

$$c_i = \frac{1}{m} \sum_{j=1}^{m} f(x_i) \cdot \acute{z}_j^{\mathrm{T}}, \quad \acute{z}_j \in \mathcal{M}, \tag{7}$$

where T denotes the transpose of the vector. We apply $l2$-normalisation to improve the robustness of the variation of the vector scale of the operation.

We predict the contaminated samples by sorting the score in descending order and picking top-τ% samples among the sorted results as the contaminated samples; thus, the number of predicted contaminated samples is decided by τm that is a multiplication of τ and the batch size m. The above process is represented as follows:

$$\mathcal{X}^{\mathrm{C}} = \{x_t\}_{t \in C[1:\lceil \tau m \rceil]}, \quad C = \arg\operatorname*{sort}_i c_i, \quad \text{with} \quad 1 \leq i \leq m,$$

where C is a set of the sorted indices of input batch samples in descending order of the contamination score (Eq. 7), and $\mathcal{X}^{\mathrm{C}}$ is a set of predicted contaminated samples. $\lceil \cdot \rceil$ denotes the ceiling function. τ affects on deciding the number of predicted contaminated samples, so it directly affects the AD performance of our method.

To demonstrate the validation of the proposed contamination sample mining process, we have conducted toy experiments. Using the MNIST and Fashion-MNIST datasets, we randomly select one class as normal and otherwise all abnormal. We set the batch size of 128 and the contamination ratio of 20%. After that, during the NCAE training, we averaged the contamination score of normal samples and contaminated samples. Fig. 4 illustrates the distribution of contaminated scores of

normal samples and the contaminated samples for 100 iterative experiments. As shown in Fig. 4, the score of contaminated samples is obviously distinguishable from the normal samples. Even if we consider the variation of the score (rectangle of each bar chart), the results can be interpreted that our score-based contaminated sample mining process works well.

The objective function for joint learning of the entire components of our method is as follows:

$$\begin{aligned}\min_{f,g}\max_{D_l,D_s} \quad & \underbrace{\mathbb{E}_{x\sim p_{\mathcal{X}^{\mathrm{N}}}}||x-g\circ f(x)||^2+\mathbb{E}_{x\sim p_{\mathcal{X}^{\mathrm{C}}}}||x-\bar{x}'||^2}_{\text{(a)}} \\ & +\underbrace{\mathbb{E}_{\omega\sim N(\mu_{\mathcal{Z}},I_d)}[\log D_l(\omega)]+\mathbb{E}_{x\sim P_{\mathcal{X}}}[\log\left(1-D_l(f(x))\right)]}_{\text{(b)}} \\ & +\underbrace{\mathbb{E}_{x\sim P_{\mathcal{X}}}[\log D_s(x)]+\mathbb{E}_{\omega'\sim N(\mu_{\mathcal{Z}},\sigma I_d)}[\log(1-D_s(g(\omega'))]}_{\text{(c)}},\end{aligned} \tag{8}$$

where D_l and D_s define the discriminators for the latent features and samples, respectively. $\bar{x}'$ is a prototype of the generated high-confident normal samples $g(\acute{\omega}) = \acute{x}$. It is defined by averaging the generated samples as follows: $\bar{x}' = \mathbb{E}_{x'\sim g(N(\mu_{\mathcal{Z}},\sigma I_d))}(\acute{x})$. $\mu_{\mathcal{Z}}$ and I_d denote the averaged latent features and their covariance matrix represented by an identity matrix. w indicates noise signals sampled from the Gaussian distribution $N(\mu_{\mathcal{Z}}, I_d)$. (a), (b), and (c) denote the NCR loss and the two adversarial losses, respectively.

The encoder of the autoencoder is trained with (a) + (b), and the decoder of the AE is trained with (a) + (c); they share the NCR loss term (a) under joint training. To optimise the above objective efficiently, we propose an alternating algorithm, which optimises model parameters of the encoder f and the decoder g for the AE and the discriminator for the adversarial learning D alternatively, as shown in Algorithm 1. Since the algorithm monotonically and iteratively decreases the objective function, it is guaranteed to converge.

In the algorithm, σ and τ significantly affect the AD performance of the NCAE model. σ decides the range for generating the noise signal to create the high-confidence normal samples. τ is used to decide how many samples would be considered contaminated samples. We will discuss the effectiveness of those hyper-parameters on AD performance in Section 5.

The AD process of the NCAE is equivalent to other AD-based AD methods. When a sample x is given, we produce reconstructed results $g \circ f(x)$ and compute the reconstruction error (See Eq. 1). After that, anomaly detection is done by

comparing the error with a pre-defined threshold β. The AD process is represented as follows:

$$X = \begin{cases} \text{Abnormal} & ||x - g \circ f(x)||^2 > \beta, \\ \text{Normal} & \text{Otherwise.} \end{cases} \tag{9}$$

The AD detection performance depends on the value of β. When β is too small, the AD results will be more precise, but they will be too specific, and many anomalies will be ignored. Hence, if the β is too large, many anomalies will be detected, but it contains many false positives. In other words, The bigger β, the more comprehensive abnormality will be detected. The smaller β, the more precise detection results (less false positives) will be provided. Considering this evaluation property, we provide the evaluation metric by observing the performance trend regarding the threshold value change instead of the AD performance at a single point. More detailed information is provided in Section 5.

4. Experimental settings

Dataset: We use various anomaly detection datasets for our experiment. Primarily, MNIST [58][1], Fashion-MNIST [59][2], and CIFAR-10 [60][3] datasets are used for the experiments. The MNIST dataset consists of a collection of 70,000 grayscale images of handwritten digits. It is divided into a training set of 60,000 examples and a test set of 10,000 examples. Each image has a resolution of 28×28 pixels. The Fashion-MNIST dataset is intended to be a more challenging version of the MNIST dataset. Instead of handwritten digits, it consists of 70,000 grayscale images of 10 different fashion items, including t-shirts, trousers, dresses, shoes, and more. Like MNIST, it is divided into a training set of 60,000 examples and a test set of 10,000 examples. The CIFAR-10 dataset of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images. The dataset is divided into five training batches and one test batch, each with 10000 images [60]. The test batch contains exactly 1000 randomly selected images from each class. The training batches contain the remaining images in random order, but some training batches may contain more images from one class than another. Between them, the training batches contain exactly 5000 images from each class.

[1] https://www.kaggle.com/datasets/hojjatk/mnist-dataset

[2] https://www.kaggle.com/datasets/zalando-research/fashionmnist

[3] https://www.cs.toronto.edu/~kriz/cifar.html

Algorithm 1 Repetitive generation-feedback algorithm for NCAE

Require: The training epoch T, training step S, the batch size m, the contaminated ratio τ, the learning rate γ, and the variance controller σ for generating normal samples

Initialise • $\mu_{\mathcal{Z}} : \mu_{\mathcal{Z}} = \frac{1}{n}\sum_{i=1}^{n} z_i$

for $t = 1$ to T epochs **do**

for $s = 1$ to S training steps **do**

• Update the high confident normal sample generation components: $\{g, \mathcal{D}_s\}$

• Sample $\{\acute{\omega}_i\}_{i=1:m} \sim N(\mu_{\mathcal{Z}}, \sigma I_d)$ and $\{x_i\}_{i=1:m} \sim P_{\mathcal{X}}$

• Update the decoder g and the discriminator D_s using following objective:

$$\min_{g}\max_{D_s} \mathbb{E}_{x\sim P_{\mathcal{X}}}[\log D_s(x)] + \mathbb{E}_{\omega'\sim N(\mu_{\mathcal{Z}},\sigma I_d)}[\log(1 - D_s(g(\omega'))].$$

• Generate high-confidence normal samples $\{\acute{x}_1, \ldots, \acute{x}_m\}$: $\acute{x}_i = g(\acute{\omega}_i)$.

• Predict contaminated samples

· Construct feature dictionary $\mathcal{M} = \{\acute{z}_1, \ldots, \acute{z}_m\}$, where $\acute{z}_i = f(\acute{x}_i)$.

· Compute the contamination score c: $c_i = \frac{1}{m}\sum_{j=1}^{m} f(x_i) \cdot \acute{z}_j^{\mathrm{T}}$

· Predict contamination samples $\mathcal{X}_{\mathrm{C}}$ as follows:

$$\mathcal{X}^{\mathrm{C}} = \{x_t\}_{t\in C[1:\lceil \tau m\rceil]} \quad w.r.t., C = \arg\operatorname*{sort}_{i} c_i, \quad w.r.t., 1 < i < m.$$

• Update f, g and $\mathcal{D}_l$.

• update the f, g, and D_l using $\{\omega_i\}_{i=1:m}$, $\{x_i\}_{i=1:m}$, $\mathcal{X}^{\mathrm{N}}$, and $\mathcal{X}^{\mathrm{C}}$ with the following objective:

$$\begin{aligned}\min_{f,g}\max_{D_l} \quad & \mathbb{E}_{x\sim\mathcal{X}^{\mathrm{N}}}||x - g\circ f(x)||^2 + \mathbb{E}_{x\sim\mathcal{X}^{\mathrm{C}}}||x - \bar{x}'||^2 \\ & + \mathbb{E}_{\omega\sim N(\mu_{\mathcal{Z}},I_d)}[\log D_l(\omega)] + \mathbb{E}_{x\sim P_{\mathcal{X}}}[\log\left(1 - D_l(f(x))\right)]\end{aligned}$$

• Update $\mu_{\mathcal{Z}}$: $\mu_{\mathcal{Z}} = \mu_{\mathcal{Z}} - \gamma\frac{1}{m}\sum_{i=1}^{m}(\mu_{\mathcal{Z}} - f(x_i))$

end for

end for

Table 1: Key properties of the Outlier Detection DataSets (ODDs).

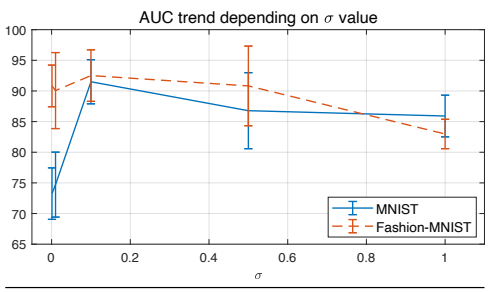
Dataset	Total #	Dim	# outliers (%)
Satimage-2	5,803	36	71 (1.2%)
Thyroid	3,772	6	93 (2.5%)
Cardio	1,831	21	176 (9.6%)
Satellite	6,435	36	2,036 (31.6%)
Shuttle	49,097	9	3,511 (7%)
Mammography	11,183	6	260 (2.32%)

Each image also has a resolution of 28×28 pixels. In addition, we leverage Outlier Detection DataSets[4] (ODDs). Table 1 shows the key properties of the ODDs. In particular, ODDs provide a chance to evaluate the AD performance with various contamination ratios. The contamination ratios of the ODDs have ranged from a minimum of 1.2% to a maximum of 31.6%.

Experiment protocol: We follow the unsupervised AD protocol described by Ruff *et al.* [2] and Zhou *et al.* [48]. We set one of the classes provided by a dataset as normal and others as abnormal. First, we have decided on contamination ratio $\rho = \frac{n_{\mathrm{A}}}{n_{\mathrm{N}}+n_{\mathrm{A}}}$, where n_{N} and n_{A} are the numbers of normal and abnormal samples, respectively. After that, we pick normal samples from the chosen class and contaminated samples from the remaining classes. In the test phase, the samples of the normal class are labelled by 0, and other samples are labelled by 1. The maximum contamination ratio is set by 30%. This ratio has been decided by considering the maximum ratio of outliers among the dataset we used for our experiment (See Table 1). It may consider a higher contamination ratio such as 50% or 60%; however, when we consider the definition of 'abnormal' meaning a pattern rarely observed data which does not follow a dominant data representation [61], the 30%, the maximum contamination ratio of our experiment, is a reasonable choice. For the performance analysis, the Area Under the Receiver Operating Characteristic Curve (AUC) is used. In comparison with other methods, we refer the experimental results presented in the Ruff *et al.* [2]. The experimental results in Table 2 and Table 3 contain the quantitative results of the NCAE and other methods referred from Ruff *et al.* [2].

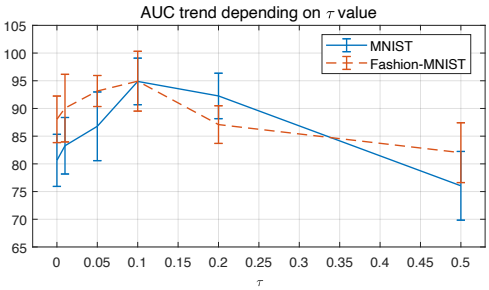
Implimentation: Our method can be thought of as one of deep neural network-

[4]http://odds.cs.stonybrook.edu/



σ	0	0.01	0.1	0.5	1.0
MNIST	73.25±4.2	74.72±5.3	91.50±3.6	86.78±6.2	85.91±3.4
fMNIST	90.82±3.4	90.06±6.2	92.51±4.2	90.82±6.5	82.98±2.4

(a) Effects of σ



τ	0	0.01	0.05	0.1	0.2	0.5
MNIST	80.63±4.7	83.27±5.1	86.78±6.2	94.94±5.0	92.26±4.1	76.04±6.2
fMNIST	88.04±4.2	90.06±6.1	93.15±2.8	94.92±4.9	87.09±3.4	82.02±5.4

(b) Effects of τ

Figure 5: Ablation studies about unsupervised AD performance depending on σ and τ. (a) and (b) represent the trends of AUC with respect to the setting of σ and τ, respectively, on the MNIST and Fashion-MNIST (fMNIST) datasets.

based AD methods; accordingly, the AD performance of deep learning-based methods can be changed depending on the number of layers and the kernel size of convolutional layers. To this principle, using a deeper layer can be considered a way to boost performance, which is very unfair to other AD methods. The fairest way for deep learning-based AD research is to employ a well-known and comprehensively used network model. In this work, we employ a LeNet-based autoencoder, which was employed for various deep learning-based AD studies [47, 17, 2, 18] on MNIST and Fashion-MNIST datasets, where each convolutional module consists of a convolutional layer followed by leaky ReLU activation functions with leakiness of 0.1. On the outlier detection dataset (OODs) benchmark using Cardio, Satellite, Satimage-2 and Thyroid, we employ standard MLP feed-forward AE structure presented by Ruff *et al.* [2]. The MLP for the encoder and decoder is defined by a 3-layer neural network with 32-16-8 units. We use the Adam optimiser with the recommended default hyper-parameters [16]. The batch size is set to 128. The initial learning rate is 0.01 and decayed every ten epochs by multiplying it by 0.1. σ and τ are decided by 0.1 (based on the results from the ablation study), respectively[5].

[5]https://github.com/andreYoo/NCAE_UAD

5. Ablation study

This ablation study provides insights into how to set up the prior parameters required for model training. In addition, we compare the effectiveness of our contaminated sample mining approach based on the generation of high-confidence normal samples with other commonly used filtering or sampling methods to find contaminated data samples. In the ablation study, only the MNIST and Fashion-MNIST data were used for experimental efficiency.

5.1. Hyper-parameter analysis

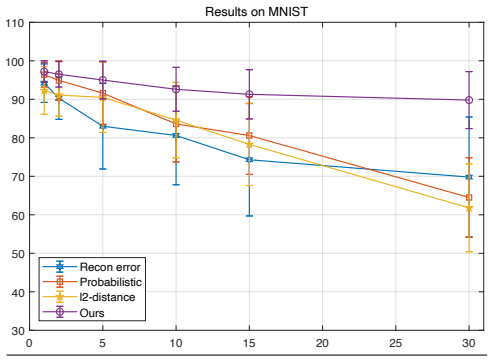
We analyse unsupervised AD performance depending on the setting of σ and τ. MNIST and Fashion-MNIST datasets are used for the ablation study. Ablation studies are conducted based on the experimental protocol described in the previous section. The contamination ratio ρ is fixed to 0.2.

Parameter analysis on σ: When σ is too small, then the distribution of sample noise for generating samples would be too compact so that the generated samples cannot provide comprehensive information to cover the diverse patterns of normal samples. On the other hand, when σ is too large, then there is a possibility that the noise can be sampled from low entropy space (*i.e.,* abnormal samples also can be generated).

Figure 5(a) shows the AUC trends depending on the σ. The AUC increases rapidly in the case of σ is less than 0.1, and then decreases gradually. This can be interpreted as follows. If the sampling space is too compact (*i.e.,* when σ is too small), it means that the generated normal sample does not provide enough information to distinguish the contaminated sample. When sampling space is too broad (*i.e.,* when σ is too large), it also degrades performance, but the impacts of the broader sampling space are relatively less than that of the smaller sampling space (e.g., when $\sigma \leq 0.1$). The best performance is obtained by σ of 0.1.

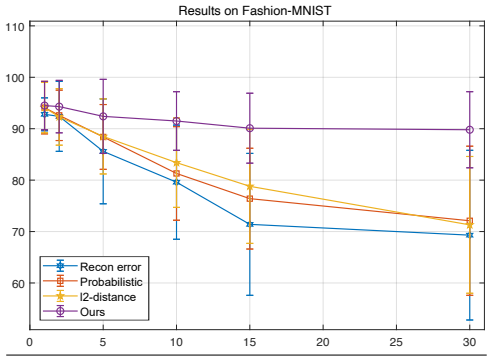
Parameter analysis on τ: τ decides the number of predicted contaminated samples per training batch. The lower τ can provide a more precise prediction performance but may not be enough to provide a more comprehensive prediction performance. In contrast, when τ is too large, the predicted results are possibly more accurate but also may have a lot of false-positive results.

As shown in Figure 5(b), the AUC increases rapidly with τ from 0 to 0.1 and then decreases slowly. The results can be interpreted as follows. Finding contaminated samples themselves has a large impact on AD performance, but the quantity of found samples affects AD performance. However, predicting too many



ρ	.01	.02	.05	.10	.15	.30
Recon	94.2±5.0	90.3±5.5	83.0±11.1	80.6±12.8	74.3±14.6	69.8±15.6
Probabilistic	96.4±3.2	94.9±5.1	91.6±8.1	83.6±9.9	80.6±10.1	64.5± 10.3
l2-distance	92.3±6.2	91.1±5.5	90.5±9.1	84.6±9.8	78.3±10.7	61.8±11.4
Ours	97.2±2.8	96.5±3.3	95.0±4.9	92.6±5.7	91.3±6.4	89.8±7.4

(a)



ρ	.01	.02	.05	.10	.15	.30
Recon	92.8±3.2	92.4±6.8	85.6±10.2	79.6±11.1	71.4±13.8	69.3±16.5
Probabilistic	94.1±5.1	92.6±4.9	88.4±6.3	81.3±9.1	76.4±9.8	72.1± 14.4
l2-distance	94.1±4.9	92.3±5.5	88.5±7.3	83.4±8.7	78.8±11.1	67.3±13.3
Ours	94.5±4.7	94.3±5.1	92.4±7.2	91.5±5.7	90.1±6.8	89.4±7.1

(b)

Figure 6: Trends in AUC values for polluted data sampling methods and changes in contamination degree. (a) shows the experimental results on the MNIST dataset. (b) represents the results on the Fashion-MNIST dataset.

samples may degrade AD performance by taking a great number of false positives. The best performance is obtained by τ of 0.1.

5.2. Effectiveness of the contamination sample mining

The contaminated sample mining based on the generated sample can be considered a way of pseudo-labelling based on data sampling. There are several sampling strategies to generate a pseudo-label for contaminated samples. Euclidean distance-based sampling [12, 13, 14, 16, 17] is one of the popular approaches to assigning the pseudo label for contaminated samples. Based on the Euclidean distance between the distribution centre and each data point (*e.g.,* $l2$-distance between the latent feature distribution centre and each latent feature), it designates some samples in order of distance as anomalies.

We have compared our contamination sample mining approach with various contaminated data sampling methods. Since our method is based on AE, we compare our method with reconstruction error-based sampling [49] and latent feature's $l2$-distance-based sampling method [17], and latent feature's probability distribution-based method [16, 47]. For a fair comparison, based on our ablation study (about τ), we set 10% (0.1 of τ) of sample on each batch that would be labelled by the contaminated samples.

Figure 6 shows the trend of AUCs depending on the contamination ratio. The ablation study results show that our contaminated sample mining approach

Table 2: Performance comparison on unsupervised anomaly detection in terms of various contamination ratios ρ. The AUC value is used to evaluate the performance. MNIST, Fashion-MNIST, and CIFAR-10 datasets are used for the comparison. The **bolded** figures indicate the best performances.

Dataset	ρ	OC-SVM	IF	KDE	CAE	Deep SVDD	SSAD	SS-DGM	Deep SAD	Classification	DAGMM	LatentOut	NCAE
MNIST	.00	96.0±2.9	85.4±8.7	95.0±3.3	92.9±5.7	92.8±4.9	**97.9±1.8**	92.2±5.6	96.7±2.4	94.5±4.6	91.7±6.2	96.5±3.2	94.0±4.2
	.01	94.3±3.9	85.2±8.8	91.2±4.9	91.3±6.1	92.1±5.1	96.6±2.4	92.0±6.0	95.5±3.3	91.5±5.9	90.4±5.2	93.9±4.8	**97.2±2.8**
	.05	91.4±5.2	83.9±9.2	85.5±7.1	87.2±7.1	89.4±5.8	93.4±3.4	91.0±6.9	93.5±4.1	86.7±7.4	88.5±9.2	86.4±7.2	**95.0±4.9**
	.10	88.8±6.0	82.3±9.5	82.1±8.5	83.7±8.4	86.5±6.8	90.7±4.4	89.7±7.5	91.2±4.9	83.6±8.2	84.2±6.2	81.6±8.3	**92.6±5.7**
	.20	84.1±7.6	78.7±10.5	77.4±10.9	78.6±10.3	81.5±8.4	87.4±5.6	87.4±8.6	86.6±6.6	79.7±9.4	81.5±7.3	73.7±9.3	**89.8±7.4**
F-MNIST	.00	92.8±4.7	91.6±5.5	92.0±4.9	90.2±5.8	89.2±6.2	**94.0±4.4**	71.4±12.7	90.5±6.5	76.8±13.2	87.6±7.2	91.2±6.2	91.5±8.3
	.01	91.7±5.0	91.5±5.5	89.4±6.3	87.1±7.3	86.3±6.3	92.2±4.9	71.2±14.3	87.2±7.1	67.3±8.1	81.5±5.5	86.2±7.6	**94.5±4.7**
	.05	90.7±5.5	90.9±5.9	85.2±9.1	81.6±9.6	80.6±7.1	88.3±6.2	71.9±14.3	81.5±8.5	59.8±4.6	74.1±9.3	84.2±6.3	**92.4±7.2**
	.10	89.5±6.1	90.2±6.3	81.8±11.2	77.4±11.1	76.2±7.3	85.6±7.0	72.5±15.5	78.2±9.1	56.7±4.1	69.2±5.2	67.3±5.2	**91.5±5.7**
	.20	86.3±7.7	88.4±7.6	77.4±13.6	72.5±12.6	69.3±6.3	81.9±8.1	70.8±16.0	74.8±9.4	53.9±2.9	65.2±11.4	61.3±12.7	**88.9±9.2**
CIFAR-10	.00	63.8±9.0	59.9±6.7	56.1±10.2	56.2±13.2	60.9±9.4	73.3±8.4	50.8±4.7	**77.9±7.2**	63.5±8.0	78.2±7.3	75.4±5.2	73.2±7.3
	.01	63.8±9.3	59.9±6.7	56.3±10.4	56.2±13.1	60.5±9.4	72.8±8.1	51.1±4.7	76.5±7.2	72.9±7.3	66.2±7.2	74.2±6.2	**79.3±3.9**
	.05	62.6±9.2	59.6±6.4	55.6±10.5	55.7±13.3	59.6±9.8	71.5±8.2	50.1±2.9	74.0±6.9	62.2±8.2	69.3±6.4	71.0±6.8	**78.2±3.2**
	.10	62.9±8.2	59.1±6.6	54.9±11.1	55.4±13.3	58.6±10.0	69.8±8.4	50.5±3.6	71.8±7.0	60.6±8.3	64.2±10.2	69.2±9.7	**76.7±5.4**
	.20	61.9±8.1	58.3±6.2	54.2±11.1	54.6±13.3	57.0±10.6	67.9±8.1	50.1±1.7	68.5±7.1	58.5±6.7	58.2±5.2	66.2±8.2	**71.1±6.2**

outperforms other strategies. Overall, the model trained by the reconstruction error-based approach shows the lowest performance. The sampling approach using $l2$-distance on the latent feature space shows much better performance compared with the reconstruction-error-based method. The probabilistic model-based sampling achieved almost similar performance with $l2$-distance-based method.

However, as the contamination rate of training data increases, the gap between the proposed method and the rest of the sampling methods widens. In particular, when more than 10% of the data is contaminated, the performance of simple Euclidean distance-based or error-based methods quickly degrades. In particular, the performance deviation of methods based on probabilistic models fell more and more rapidly.

This means that when we first simulated (see Fig. 1) if the data were contaminated at a high rate, there would be a high probability that the data would be centred in the distribution of the data or that the data would itself have a high probability distribution. Experimental results prove that the proposed sampling method is a method for finding contaminated samples that are robust to the degree of data contamination.

6. Comparison with other methods

6.1. Results on MNIST, MNIST Fashion. and CIFAR-10 datasets

We consider the OC-SVM [19], isolation forest (IF) [62], and KDE [63] for shallow unsupervised baselines. For deep unsupervised competitors, we consider

Table 3: Performance comparison using AUC-ROC values on unsupervised anomaly detection using ODDs. The **bolded** figures indicate the best performances.

Dataset	Satellite	Cardio	Thyroid	Satimage-2	Shuttle	Mammography
OC-SVM	95.1±0.2	98.5±0.3	98.3±0.9	99.4±0.8	99.4±0.9	71.4±9.7
IF	94.2±0.2	91.4±1.1	95.2±0.3	99.2±1.2	93.4±0.9	91.4±3.6
KDE	66.7±5.8	99.6±1.7	91.6±2.3	99.5±0.2	87.2±7.4	67.2±12.5
DeepSVDD	79.8±4.1	84.8±3.6	72.0±9.7	98.3±1.4	98.7±0.2	97.4±0.5
SSAD	96.2±0.3	98.8±0.3	97.9±1.9	99.9±0.1	98.9±0.4	92.6±2.4
SS-DGM	95.7±0.1	95.2±1.3	95.8±0.7	99.2±0.2	97.5±0.4	96.4±2.3
AE	73.5±9.4	89.6±6.7	95.4±2.7	99.1±1.7	95.6±0.7	82.7±9.4
Deep SAD	91.5±1.1	95.0±1.6	**98.6±0.9**	99.9±0.1	99.3±0.1	93.0±0.5
Classification	87.2±2.1	83.2±9.6	97.8±2.6	99.9±0.1	98.3±0.2	79.5±15.8
DAGMM	73.9±3.1	88.5±3.3	96.4±0.7	99.7±0.2	99.0±0.2	75.9±7.9
LatentOut	92.4±0.5	99.0±0.1	98.1±0.3	99.8±0.1	99.7±0.1	94.2±0.7
NCAE	**97.3±0.2**	**99.2±0.2**	98.5±0.1	**99.9±0.1**	**99.9±0.1**	**98.7±0.2**

Table 4: Performance comparison using AUC-PR values on unsupervised anomaly detection using ODDs. The **bolded** figures indicate the best performances.

Dataset	Satellite	Cardio	Thyroid	Satimage-2	Shuttle	Mammography
OC-SVM	72.5±5.6	82.6±6.2	78.9±9.8	94.8±1.5	96.2±2.6	43.7±9.5
IF	64.8±8.2	83.2±7.5	61.6±11.6	89.5±3.2	87.2±6.8	28.6± 15.7
KDE	48.2±17.8	86.2±4.3	65.4±9.4	90.3±6.7	73.6±6.1	28.4±3.7
DeepSVDD	64.3±4.2	81.6±2.4	51.4±4.2	91.4±1.2	92.6±1.2	52.6±1.2
SSAD	70.2±1.8	89.4±0.7	89.5±0.5	93.2±7.9	97.2±0.5	42.5±4.5
SS-DGM	69.5±22.6	42.5±9.8	92.6±0.3	94.2±0.5	91.4±2.3	48.2±1.8
AE	41.7±13.9	60.2±11.4	75.2±7.1	84.6±2.6	90.3±0.7	18.2±7.6
Deep SAD	83.3±0.7	96.8±0.8	92.3±0.4	96.2±0.2	95.2±0.3	59.1±4.6
Classification	90.2±1.1	92.3±0.7	90.2±2.6	94.1±0.5	96.2±0.2	60.3±1.1
DAGMM	68.4±2.1	62.3±8.6	87.5±2.7	95.0±0.4	95.9±1.3	12.5±7.0
LatentOut	85.4±0.9	**99.5±0.1**	94.7±0.2	96.1±0.3	96.5±0.1	**64.6±1.2**
NCAE	**90.4±0.5**	99.3±0.2	**95.1±0.6**	**96.4±0.2**	**98.2±0.4**	64.2±0.7

general binary classifier (supervised), convolutional autoencoders (CAE), deep support vector data description (Deep SVDD) [17], semi-supervised anomaly detection (SSAD) [2], semi-supervised deep generative model (SS-DGM) [16], Deep autoencoding gaussian mixture model (DAGMM) [6], LatentOut [64], and deep semi-supervised anomaly detection (Deep SAD) [2]. We repeat this training set generation process ten times per AD set up over all the nine respective anomaly classes and report the average results over the resulting 90 experiments per contamination ratio.

Table 2 shows the quantitative performance comparison depending on the contamination ratio ρ. In the comparison using the MNIST dataset, the proposed NCAE achieves the best performances except when the dataset is not contaminated

($\rho = 0.0$). Even compared with semi-supervised approaches (SSAD and Deep SVDD) [17, 2], which use explicit anomaly samples in the training phase, the NCAE shows outstanding performances. This trend is also shown in the performance comparison using the Fashion-MNIST and CIFAR-10 datasets. The NCAE produces the AUC of 94.5 and 88.9 for the Fashion-MNIST dataset with 1% and 20% contamination ratios, respectively. Also, it achieves the AUC of 79.3 and 71.1 for the CIFAR-10 dataset with 1% and 20% contamination ratios, respectively. Those figures perform best among the listed methods when a dataset is contaminated. Compared with other methods, which degrade their performance significantly when the contamination ratio is increased, the NCAE AD performances are relatively robust to the contamination ratios. Those figures are the best performance among the listed methods when a dataset is contaminated.

The interpretation of the relatively low performance on the uncontaminated dataset ($\rho = 0.0$) is as follows. Basically, our method is derived under the assumption that a training dataset is contaminated. Therefore, even if the dataset is not contaminated, the NCAE tries to find some anomaly samples and maximise the reconstruction errors of the samples during the model training. The minimisation process is reformulated by the error minimisation between the contaminated samples and generated normal samples (See Eq. 8). This process degrades the performance of our methods, as shown in the experimental results. This is a critical defect of our method.

Overall, the comparison results demonstrate the advantage of the proposed NCAE that can detect anomaly samples on data contamination without prior knowledge or explicit abnormal samples in the training phase.

6.2. Results on ODDs

Table 3 shows the Area under the ROC curve (AUC-ROC) of various AD methods on ODDs. Table 4 shows the Area under the precision-recall curve (AUC-PR) of various AD methods on ODDs. The detailed information on OODs used for our experiments is shown in Table 1. The experimental results on Table 3 and Table 4 demonstrate that the proposed NCAE is robust to data contamination. For all categories, the NCAE produces the best performances. For the AUC-ROC, NCAE achieves 97.3, 99.2, 98.5, and 99.9 for Satellite, Cario, Thyroid, and Satimage-2 datasets, respectively. Except for the result using the Thyroid dataset, those results outperform the performances of other methods. For the AUC-PR, NCAE achieves 90.4, 99.3, 95.1, and 96.4 for Satellite, Cario, Thyroid, and Satimage-2 datasets, respectively. For the results using Cardio and Mammography, LatentOut [64] achieves better performances with small margins.

For AUC-ROC results using the Satellite dataset, the method that recorded the second-highest performance was SSAD. SSAD achieved an AUC of 96.2. This result is 1.1 lower than the proposed NCAE, and the variation of performances is also 0.1 higher than the NCAE. Among the experiments using the Cardio dataset, the second highest method is also SSAD. SSAD records an AUC of 98.8 in trials using Cardio. In the experiment using Thyroid, it is the only one that failed to achieve the best performance of the NCAE. The NCAE produces an AUC of 98.5, which is the second-ranked performance. The Deep SAD achieves the best performance on the Thyroid dataset. Deep SAD achieves an AUC of 98.6, which is 0.1 higher than that of NCAE. However, the performance variation of the Deep SAD is 0.9, which is much higher than the 0.1 of NCAE.

These results show that the NCAE's AD performance is slightly lower than the Deep SAD, but the NCAE's performance is much more stable. For AUC-PR results using the Satellite dataset, LatentOut [64] achieve partially better performances in Cardio and Mannography datasets. LatentOut achieved 99.5 and 64.6 of AUC-PR, respectively. This result is 0.2 and 0.4 higher than the proposed NCAE. For the variation, the NCAE achieves better variation. The NCAE usually shows smaller variation, which can be interpreted as the performance of the NCAE fluctuating less than other methods.

Overall, the experimental results of AUC-ROC and AUC-PR on ODDs show that the proposed NCAE can provide more robust AD performance on data contamination. The quantitative evaluation results prove that the proposed NCAE model shows higher performance than the comparison targets in most of the data on contamination level, and at the same time, the variation in performance is not large, even in repeated experiments. In conclusion, the proposed NCAE can be interpreted as having better AD performance for contaminated data than the comparison methods.

7. Conclusion

In this work, we have proposed a Normality-Calibrated Autoencoder (NCAE) that is a generative method for fully unsupervised anomaly detection on contaminated data. The proposed NCAE extracts latent features based on AE structure and compiles latent feature distribution to a well-known distribution such as Gaussian distribution. After that, based on the decoder part and a discriminator, NCAE applies adversarial learning to generate high-confidence normal samples. Based on the generated high-confidence normal sample, the NCAE identifies contaminated data and applies it to the training model to minimise reconstruction error between

the contaminated sample and randomly selected high-confidence normal samples. The experimental results have suggested that the NCAE outperforms existing methods for fully unsupervised anomaly detection by a large margin, and they have also provided competitive performances compared with semi-supervised methods using explicit abnormal samples to train their AD model.

However, there are drawbacks that we should solve in our future work. Even though the proposed NCAE achieves state-of-the-art AD performance in various datasets with various contamination ratios. The performance of NCAE is affected by the hyper-parameters τ and σ. Finding optimal values of hyper-parameters is a common issue in machine learning and deep learning studies. Additionally, to train the NCAE, we assume that a dataset is always contaminated. As we mentioned in the performance comparison with other methods, since the NCAE always assumes that some data samples have been contaminated, the experimental results using a non-contaminated dataset are a bit lower than others. This issue will be addressed in our future works.

References

[1] M. Kim, J. Kim, J. Yu, J. K. Choi, Active anomaly detection based on deep one-class classification, Pattern Recognition Letters 167 (2023) 18–24.

[2] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, M. Kloft, Deep semi-supervised anomaly detection, arXiv preprint arXiv:1906.02694 (2019).

[3] H. Song, Z. Jiang, A. Men, B. Yang, A hybrid semi-supervised anomaly detection model for high-dimensional data, Computational Intelligence and Neuroscience (2017).

[4] S. Akcay, A. Atapour-Abarghouei, T. P. Breckon, GANomaly: Semi-supervised anomaly detection via adversarial training, in: ACCV, 2018, pp. 622–637.

[5] R. Chalapathy, S. Chawla, Deep learning for anomaly detection: A survey, arXiv preprint arXiv:1901.03407 (2019).

[6] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho, H. Chen, Deep autoencoding gaussian mixture model for unsupervised anomaly detection, in: International Conference on Learning Representations, 2018.

[7] X. Xia, X. Pan, N. Li, X. He, L. Ma, X. Zhang, N. Ding, Gan-based anomaly detection: A review, Neurocomputing 493 (2022) 497–535.

[8] W. Chen, L. Tian, B. Chen, L. Dai, Z. Duan, M. Zhou, Deep variational graph convolutional recurrent network for multivariate time series anomaly detection, in: International Conference on Machine Learning, PMLR, 2022, pp. 3621–3633.

[9] J. Wang, P. Neskovic, L. N. Cooper, Pattern classification via single spheres, in: International Conference on Discovery Science, Springer, 2005, pp. 241–252.

[10] Y. Liu, Y. F. Zheng, Minimum enclosing and maximum excluding machine for pattern description and discrimination, in: ICPR, 2006, pp. 129–132.

[11] N. Görnitz, M. Kloft, K. Rieck, U. Brefeld, Toward supervised anomaly detection, Journal of Artificial Intelligence Research 46 (2013) 235–262.

[12] A. Berg, J. Ahlberg, M. Felsberg, Unsupervised learning of anomaly detection from contaminated image data using simultaneous encoder training, arXiv preprint arXiv:1905.11034 (2019).

[13] T. Li, Z. Wang, S. Liu, W.-Y. Lin, Deep unsupervised anomaly detection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 3636–3645.

[14] C.-H. Lai, D. Zou, G. Lerman, Robust subspace recovery layer for unsupervised anomaly detection, in: International Conference on Learning Representations, 2020.
URL https://openreview.net/forum?id=rylb3eBtwr

[15] S. Yoon, Y.-K. Noh, F. Park, Autoencoding under normalization constraints, in: International Conference on Machine Learning, PMLR, 2021, pp. 12087–12097.

[16] D. P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, in: ICLR, 2015.

[17] L. Ruff, R. A. Vandermeulen, N. Görnitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, M. Kloft, Deep one-class classification, in: ICML, Vol. 80, 2018, pp. 4390–4399.

[18] L. Ruff, J. R. Kauffmann, R. A. Vandermeulen, G. Montavon, W. Samek, M. Kloft, T. G. Dietterich, K.-R. Müller, A unifying review of deep and shallow anomaly detection, Proceedings of the IEEE (2021).

[19] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, R. C. Williamson, Estimating the support of a high-dimensional distribution, Neural computation 13 (7) (2001) 1443–1471.

[20] D. M. Tax, R. P. Duin, Support vector data description, Machine learning 54 (1) (2004) 45–66.

[21] R. Chalapathy, A. K. Menon, S. Chawla, Anomaly detection using one-class neural networks, arXiv preprint arXiv:1802.06360 (2018).

[22] I. Golan, R. El-Yaniv, Deep anomaly detection using geometric transformations, in: Advances in Neural Information Processing Systems, 2018.

[23] D. Hendrycks, M. Mazeika, S. Kadavath, D. Song, Using self-supervised learning can improve model robustness and uncertainty, in: Advances in Neural Information Processing Systems, 2019, pp. 15637–15648.

[24] L. Bergman, Y. Hoshen, Classification-based anomaly detection for general data, arXiv preprint arXiv:2005.02359 (2020).

[25] S. Ahmad, A. Lavin, S. Purdy, Z. Agha, Unsupervised real-time anomaly detection for streaming data, Neurocomputing 262 (2017) 134–147.

[26] Ł. Maziarka, M. Śmieja, M. Sendera, Ł. Struski, J. Tabor, P. Spurek, Oneflow: One-class flow for anomaly detection based on a minimal volume region, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (11) (2021) 8508–8519.

[27] X. Wang, G.-J. Qi, Contrastive learning with stronger augmentations, IEEE transactions on pattern analysis and machine intelligence 45 (5) (2022) 5549–5560.

[28] E. Parzen, On estimation of a probability density function and mode, The annals of mathematical statistics 33 (3) (1962) 1065–1076.

[29] D. W. Scott, Multivariate density estimation: theory, practice, and visualization, John Wiley & Sons, 2015.

[30] S. Roberts, L. Tarassenko, A probabilistic resource allocating network for novelty detection, Neural Computation 6 (2) (1994) 270–284.

[31] D. P. Kingma, M. Welling, Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114 (2013).

[32] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, G. Langs, Unsupervised anomaly detection with generative adversarial networks to guide marker discovery, in: International conference on information processing in medical imaging, 2017.

[33] M. Z. Zaheer, A. Mahmood, M. H. Khan, M. Segu, F. Yu, S.-I. Lee, Generative cooperative learning for unsupervised video anomaly detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 14744–14754.

[34] J. An, S. Cho, Variational autoencoder based anomaly detection using reconstruction probability, Special Lecture on IE 2 (1) (2015) 1–18.

[35] P. Perera, R. Nallapati, B. Xiang, Ocgan: One-class novelty detection using gans with constrained latent representations, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[36] B. Nachman, D. Shih, Anomaly detection with density estimation, Physical Review D 101 (7) (2020) 075042.

[37] M. Sakurada, T. Yairi, Anomaly detection using autoencoders with nonlinear dimensionality reduction, in: Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, 2014.

[38] C. Zhou, R. C. Paffenroth, Anomaly detection with robust deep autoencoders, in: International Conference on Knowledge Discovery and Data Mining, 2017.

[39] J. Fan, Q. Zhang, J. Zhu, M. Zhang, Z. Yang, H. Cao, Robust deep autoencoding gaussian process regression for unsupervised anomaly detection, Neurocomputing 376 (2020) 180–190.

[40] G. E. Hinton, R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, science 313 (5786) (2006) 504–507.

[41] J. Masci, U. Meier, D. Cireşan, J. Schmidhuber, Stacked convolutional auto-encoders for hierarchical feature extraction, in: International conference on artificial neural networks, 2011.

[42] M. Thill, W. Konen, H. Wang, T. Bäck, Temporal convolutional autoencoder for unsupervised anomaly detection in time series, Applied Soft Computing 112 (2021) 107751.

[43] J. T. Andrews, E. J. Morton, L. D. Griffin, Detecting anomalous data using auto-encoders, International Journal of Machine Learning and Computing 6 (1) (2016) 21.

[44] S. M. Erfani, S. Rajasegarar, S. Karunasekera, C. Leckie, High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning, Pattern Recognition 58 (2016) 121–134.

[45] T. Ergen, S. S. Kozat, Unsupervised anomaly detection with lstm neural networks, IEEE transactions on neural networks and learning systems 31 (8) (2019) 3127–3141.

[46] F. Liu, C. Zeng, L. Zhang, Y. Zhou, Q. Mu, Y. Zhang, L. Zhang, C. Zhu, Fedtadbench: Federated time-series anomaly detection benchmark, in: 2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys), IEEE, 2022, pp. 303–310.

[47] S. Pidhorskyi, R. Almohsen, G. Doretto, Generative probabilistic novelty detection with adversarial autoencoders, in: Advances in Neural Information Processing Systems, 2018, pp. 6822–6833.

[48] Y. Zhou, X. Song, Y. Zhang, F. Liu, C. Zhu, L. Liu, Feature encoding with autoencoders for weakly supervised anomaly detection, IEEE Transactions on Neural Networks and Learning Systems 33 (6) (2021) 2454–2465.

[49] Y. Xia, X. Cao, F. Wen, G. Hua, J. Sun, Learning discriminative reconstructions for unsupervised outlier removal, in: Proceedings of the IEEE International Conference on Computer Vision, 2015.

[50] L. Beggel, M. Pfeiffer, B. Bischl, Robust anomaly detection in images using adversarial autoencoders, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2019.

[51] G. Pang, C. Yan, g. C. Shen, A. v. d. Hengel, X. Bai, Self-trained deep ordinal regression for end-to-end video anomaly detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2020.

[52] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, B. Frey, Adversarial autoencoders, arXiv preprint arXiv:1511.05644 (2015).

[53] I. Shim, T.-H. Oh, I. S. Kweon, High-fidelity depth upsampling using the self-learning framework, Sensors 19 (1) (2018) 81.

[54] Z. Lin, A. Khetan, G. Fanti, S. Oh, Pacgan: The power of two samples in generative adversarial networks, Advances in neural information processing systems 31 (2018).

[55] J. Yu, Y. Lee, K. C. Yow, M. Jeon, W. Pedrycz, Abnormal event detection and localization via adversarial event prediction, IEEE Transactions on Neural Networks and Learning Systems (2021).

[56] N.-C. Ristea, N. Madan, R. T. Ionescu, K. Nasrollahi, F. S. Khan, T. B. Moeslund, M. Shah, Self-supervised predictive convolutional attentive block for anomaly detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 13576–13586.

[57] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative Adversarial Nets, in: NIPS, 2014, pp. 2672–2680.

[58] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (11) (1998) 2278–2324.

[59] H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, arXiv preprint arXiv:1708.07747 (2017).

[60] A. Krizhevsky, G. Hinton, et al., Learning multiple layers of features from tiny images (2009).

[61] M. A. Pimentel, D. A. Clifton, L. Clifton, L. Tarassenko, A review of novelty detection, Signal processing 99 (2014) 215–249.

[62] F. T. Liu, K. M. Ting, Z.-H. Zhou, Isolation Forest, in: ICDM, 2008, pp. 413–422.

[63] E. Parzen, On Estimation of a Probability Density Function and Mode, The annals of mathematical statistics 33 (3) (1962) 1065–1076.

[64] F. Angiulli, F. Fassetti, L. Ferragina, Latent o ut: an unsupervised deep anomaly detection approach exploiting latent space distribution, Machine Learning 112 (11) (2023) 4323–4349.

Jongmin Yu is a principal scientist in ProjectG.AI and a research associate of the image analysis group at the department of applied mathematics and theoretical physics at University of Cambridge. He was a research associate at the King's College London. He received the Ph.D. from the School of Electrical Engineering and Computer Science in Gwangju Institute of Science and Technology (GIST), Gwangju, Korea, Republic of. He was a visiting researcher at the School of Electrical Engineering, Computing and Mathematical Sciences at Curtin University, Perth, Western Australia, Australia. Presently, His research interests include artificial intelligence, machine learning, pattern recognition, and mathematical understanding of these.

Minkyung Kim received a B.E. degree in Electrical and Computer Engineering from the University of Seoul, South Korea in 2016, and M.S. and Ph.D. degrees in Electrical Engineering from KAIST, South Korea in 2018 and 2023, respectively. She is currently a postdoctoral researcher in the Department of Otolaryngology-Head and Neck Surgery at Massachusetts Eye and Ear, which is a teaching hospital of Harvard Medical School. Her research interests include anomaly detection, unsupervised learning, and active learning.

Junsik Kim received the BS, MS, and Ph.D. degrees in Electrical Engineering Department, KAIST, South Korea, in 2013, 2015, and 2020 respectively. He is currently a postdoctoral researcher in the School of Engineering and Applied Sciences with the Harvard University. Before joining Harvard, he was a postdoctoral researcher with KAIST. His research interest includes computer vision problems, especially with data imbalance and scarcity problems. He was a research intern with Hikvision Research America, Santa Clara, in 2018. He was a recipient of the Qualcomm Innovation award in 2018.

Hyeontaek Oh (S'14, M'20) is currently a team leader at Institute for IT Convergence in Korea Advanced Institute of Science and Technology (KAIST). He received the BS degree in computer science (summa cum laude), MS degree, and PhD degree in electrical engineering from KAIST in 2012, 2014, and 2020, respectively. His research interests in trust in ICT environments, personal data ecosystem, Internet of Things (IoT), and Web technologies. He has actively participated in nationally-funded research projects for ICT environment. He also has contributed the International Telecommunication Union Telecommunication Standardization Sector Study Group 13/20 as contributors and editors since 2015.

Contributor Roles Taxonomy

Jongmin Yu
Roles: Conceptualization, Methodology, Validation, Formal analysis, Writing - Original Draft, Writing - Review & Editing, Visualization, Supervision

Minkyung Kim
Roles: Data Curation, Validation, Writing - Original Draft

Junsik Kim
Roles: Validation, Writing - Review & Editing

Hyeontaek Oh
Roles: Validation, Writing - Review & Editing

Declaration of interests

☒The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: