Journal of Dynamics, Monitoring and Diagnostics, 2023, 2, 144-154
https://doi.org/10.37965/jdmd.2023.234

RESEARCH ARTICLE

Constraint-Guided Autoencoders to Enforce a Predefined Threshold on Anomaly Scores: An Application in Machine Condition Monitoring

Maarten Meire,[1,2,3] Quinten Van Baelen,[1,2,3] Ted Ooijevaar,[4] and Peter Karsmakers[1,2,3]

[1]Department of Computer Science, KU Leuven, ADVISE-DTAI, Kleinhoefstraat 4 B-2440 Geel, Belgium
[2]Leuven.AI – KU Leuven institute for AI, 3000 Leuven, Belgium
[3]Flanders Make @ KU Leuven, 3000 Leuven, Belgium
[4]Flanders Make vzw, CoreLab MotionS, Leuven 3001, Belgium

(Received 31 March 2023; Revised 15 May 2023; Accepted 05 June 2023; Published online 05 June 2023)

Abstract: Anomaly detection (AD) is an important task in a broad range of domains. A popular choice for AD are Deep Support Vector Data Description models. When learning such models, normal data is mapped close to and anomalous data is mapped far from a center, in some latent space, enabling the construction of a sphere to separate both types of data. Empirically, it was observed: (i) that the center and radius of such sphere largely depend on the training data and model initialization which leads to difficulties when selecting a threshold, and (ii) that the center and radius of this sphere strongly impact the model AD performance on unseen data. In this work, a more robust AD solution is proposed that (i) defines a sphere with a fixed radius and margin in some latent space and (ii) enforces the encoder, which maps the input to a latent space, to encode the normal data in a small sphere and the anomalous data outside a larger sphere, with the same center. Experimental results indicate that the proposed algorithm attains higher performance compared to alternatives, and that the difference in size of the two spheres has a minor impact on the performance.

Keywords: anomaly detection; autoencoders; deep learning

I. INTRODUCTION

An important task when looking at real-life datasets is detecting instances that behave differently than the majority of the dataset; these instances are termed *anomalies*, and this task is, therefore, termed anomaly detection (AD). It is used in a broad range of domains including, but not limited to, finance [1], medical diagnosis [2], network intrusion detection [3], and machine condition monitoring [4]. Research in this area has already been active for decades with some classic approaches such as Principal Component Analysis (PCA) [5], One-Class Support Vector Machines (OC-SVM) [6], and Support Vector Data Description (SVDD) [7]. These approaches are all shallow and, thus, experience limitations when scaling to larger datasets and typically need application-dependent features, which is usually very time-consuming.

In recent years, deep learning (DL) alternatives have increased and overcome these limitations [8]. The most common approach within these DL alternatives is an autoencoder (AE) or an AE-based variant [8]. These approaches attempt to encode normal data into a low-dimensional representation in a way that allows the normal data to be reconstructed as well as possible [9]. As the aim of an AE is to work as well as possible on normal data, it is expected that the reconstruction of anomalous data will be worse [10]. To improve the performance of AEs, they are often combined with a traditional AD method. This is typically done by using the AE to extract features in the encoded representation, which are then used as an input for the AD method. For example, a random forest can be trained using these extracted features [11]. However, the features are not necessarily extracted in the encoded representation. Another AD method uses the reconstruction of Automatic Dependent Surveillance-Broadcast (ADS-B) time series in comparison with the input times series [12]. The difference is then passed to the SVDD algorithm for AD.

The previously discussed approaches allow only normal data in its training set and are mainly used as the amount of available anomalous data is generally limited, which makes fully supervised approaches less suitable. However, if anomalous data is available, it can provide valuable information. Hence, approaches that can exploit this additional data are of interest [13]. Nevertheless, fully supervised methods might not generalize well to unseen anomalies as they might differ too much from previously observed anomalies. Semi-supervised AD methods allow for better generalization to unseen anomalies since they use normal and anomalous data in their training procedure. An example generalizing (traditional) SVDD can be found in [14]. A comparison between an unsupervised, semi-supervised, and supervised method for network intrusion detection is made in [15], indicating that, if there are no unknown anomalies, supervised methods do have the best performance. In general, the semi-supervised methods outperform the unsupervised methods.

The semi-supervised AD methods often work using a two-step approach: first, by learning the feature extractor and, second, by training the AD model. This could possibly lead to a disconnect between the learned features and the AD model if the objectives do not align properly. To

Corresponding author: Maarten Meire (e-mail: maarten.meire@kuleuven.be).

alleviate this disconnect, hybrid alternatives have been proposed, where the learning of the feature extractor and AD model are integrated into one objective. Examples of these hybrid AD methods are as follows: (i) Soft-Boundary Deep Support Vector Data Description (SB-DSVDD) that minimizes the radius of the sphere as well as the SVDD objective on the encodings where the normal points inside the sphere are assigned an objective value of 0 for the SVDD objective [16], (ii) Deep Support Vector Data Description (DSVDD) that is generalized to work in a semi-supervised setting [17], and (iii) the OC-SVM objective that is adapted to fit into the DL framework [18]. SB-DSVDD yields an improvement in performance in comparison with prior work. Nevertheless, the method uses only normal data. Moreover, the objective corresponding to the adjusted SVDD objective is in spirit a fuzzy translation of an implication [19]. More specific, if the point is healthy, then the distance to the center should be smaller than the radius R. However, it was already mentioned in [20] that this translation leads to a loss in precise meaning and should be avoided. While approaches (ii)–(iii) indicate a better performance in comparison with existing methods, both suffer from drawbacks as well. For example, the adaptation of the OC-SVM objective to DL results in the objective becoming nonconvex [18]. The adapted DSVDD objective causes the network to be subjected to restrictions as to avoid trivial solutions [16,17], causing a loss of modeling flexibility. These restrictions are as follows: (i) the center used by DSVDD cannot be 0, (ii) biases cannot be used in the network, and (iii) bounded activation functions cannot be used in the network.

A combination of an AE and DSVDD was proposed in [21], where the objectives of both methods were combined into a multitask learning objective. In this way, the model will learn to map normal data close to a center, while preventing overlap between different data points due to the reconstruction objective. Similar work was done in [22,23], where AE or variants were used as feature learner in combination with DSVDD. It is also shown that the restrictions mentioned above are no longer present for these methods [22,23].

As mentioned earlier, AD is applicable in a broad range of domains; however, this work will focus on machine condition monitoring, and more specifically on rotating machinery. The cause of most system failures in this type of machinery is rolling element bearings (REB) [24]. A common approach to monitor the condition of REB is to measure vibrations produced by the rotating machines [25]. The raw vibration signal was used with an AE in [26] to detect bearing faults and was compared with a traditional neural network, an SVM, and the combination of both with the AE. In [27], two approaches are discussed that enable variational AEs to learn with labeled data, showing a better performance than a fully supervised classifier. A multi-scale DSVDD [28], which refers to transforming the original sample space to multiple subspaces in combination with multiple DSVDD centers, was used to detect incipient faults in bearings.

A critical step in using AD methods in practice is determining a threshold on the anomaly score provided by the AD method [29]. More specific, if the AD method is used for determining whether maintenance needs to be performed on a machine, then this requires a binary classification of normal or anomalous. This can be done by putting a threshold on the anomaly score. However, determining the point where this threshold needs to be set is typically a nontrivial problem and can be dependent on the application, since in some cases false negatives are preferred over false positives or vice versa. Some common methods used to set this threshold rely on: (i) statistics obtained on the training dataset, such as mean, standard deviation, or percentiles of the normal data [27,30], (ii) an additional set of data, the test set for example, to set a threshold based on a chosen metric [31], and (iii) dynamic thresholding that uses the evaluated sample [32]. In this work, methods are targeted that avoid case-specific adaptation. In other words, methods for which the model parameters are determined only based on training data, without needing any additional information of the test data. It should be stressed that the methods statistics obtained on the training data determine a threshold after training. Empirically it was observed that the value of such threshold largely depends on the training dataset and model initialization when, for example, DL methods are used, and the choice of this threshold largely impacts the model performance on unseen data. Therefore, more robust methods are desired that make the position of the threshold less dependent on the data and the initialization of the model parameters.

In this work, we argue that more robust AD solutions can be found by defining a fixed threshold in some latent space and enforcing the encoder that maps the input to the latent space to position the normal and anomalous data in the latent space on the correct side of the threshold. The proposed method Constraint-Guided AutoEncoder (CG-AE) learns an AE for the normal data under the constraints that the encodings of the normal data are inside a sphere with a fixed radius and the encodings of the anomalous points are outside a sphere with a larger fixed radius. The constraints are employed to better discriminate between normal and anomalous data.

The main contributions of this work are as follows:

- The design of a novel training method that allows the use of anomalous data in the learning of an AE under constraints such that the normal and anomalous data are separated.
- A proposal and comparison of a novel thresholding method with traditional methods that only use statistics obtained on the training data.

The remainder of this text is structured as follows. First, the related methods that will be used are discussed. Then, the proposed CG-AE method will be discussed in detail together with a comparison with prior work. Afterward, the experimental setup, including the setup of an ablation study, will be described followed by a discussion about the results of the comparison with state-of-the-art methods and the ablation study. Finally, some directions for future work are proposed before concluding the text.

II. METHODS

This section introduces the methods that will be used to compare CG-AE with.

A. DEEP SUPPORT VECTOR DATA DESCRIPTION

DSVDD [17] uses the encoder of an AE to classify points as normal or anomalous. This is done by considering a point c, referred to as the center, in the latent space and computing the distance between the encoding of a sample and this center. The resulting distance is compared to a threshold, and if the distance exceeds this threshold, then the sample is considered anomalous. In the other case, that is, when the distance between the encoding and the center is smaller than the threshold, then the sample is considered normal. The neural network is learned by decreasing the distance of the encoded normal points to the center and at the same time increasing the distance between the encoding of anomalous points to the center.

In the context of this work, labels are available for each point, both normal and anomalous, in the training set. Hence, a supervised version of DSVDD is used. This yields the following learning objective of supervised DSVDD for the weights θ_ε of an encoder ε:

$$\min_{\theta_\varepsilon} \sum_{i=1}^{N} \eta^{\frac{1+y_i}{2}} (\|\varepsilon(x_i|\theta_\varepsilon) - c\|^2)^{-y_i}, \tag{1}$$

where N is the number of training samples, $\eta > 0$ is a hyperparameter, y_i is the label of the i-training sample x_i, and $\|\cdot\|$ denotes the L_2-norm on $\mathbb{R}^n$, the n-dimension real space [33]. The labels in this work are considered to be -1 and 1 denoting a normal and anomalous point, respectively.

B. AUTOENCODER DEEP SUPPORT VECTOR DATA DESCRIPTION

AutoEncoder Deep Support Vector Data Description (AE-DSVDD) [21] combines AEs with the DSVDD learning objective. This results in the learning objective being defined as reconstructing all healthy points as good as possible as well as performing as good as possible with respect to the DSVDD objective. In other words, AE-DSVDD learns the weights $\theta_\varepsilon, \theta_\mathcal{D}$ of an AE, defined by an encoder ε and a decoder $\mathcal{D}$, by solving the following learning objective:

$$\min_{\theta_\varepsilon, \theta_\mathcal{D}} \frac{1}{N_n} \sum_{j=1}^{N_n} \|x_{n_j} - \hat{x}_{n_j}\|^2 + \sum_{i=1}^{N} \eta^{\frac{1+y_i}{2}} (\|\varepsilon(x_i|\theta_\varepsilon) - c\|^2)^{-y_i}, \tag{2}$$

where N_n is number of normal samples, x_{n_j} is the j-th normal sample, and $\hat{x}_{n_j}$ is the reconstruction of x_{n_j} by the AE defined by the encoder ε and the decoder $\mathcal{D}$. It should be noted that the original objective described in [21] only uses normal data, while objective (2) extends the learning objective to both normal and anomalous data.

III. CONSTRAINT-GUIDED AUTOENCODERS

In this section, the proposed method CG-AE is introduced. CG-AE learns the parameters of an AE model by minimizing the reconstruction error of normal data points subject to the constraints: (i) all normal data points should be encoded inside a sphere of radius R_1 around the origin, and (ii) all anomalous data points should be encoded outside a sphere of radius R_2 around the origin. An assumption is made that a least a few anomalous examples are available during training. The constraints are used to create a decision boundary in the latent space that can be used to distinguish normal from anomalous data. Therefore, the assumption $R_1 < R_2$ is made such that the space of possible normal encodings is separated from the space of possible anomalous encodings, if the model is feasible. This could lead to a potential better generalization to unseen data. In other words, an AE is learned with the constrained learning objective:

$$\min_{\theta_\varepsilon, \theta_\mathcal{D}} \frac{1}{N_n} \sum_{j=1}^{N_n} \|x_{n_j} - \hat{x}_{n_j}\|^2, \tag{3}$$

$$s.t.\ \forall x_n : \varepsilon(x_n|\theta_\varepsilon) \in B[0,R_1], \quad \forall x_a : \varepsilon(x_a|\theta_\varepsilon) \notin B[0,R_2],$$

where $x_n$is a normal sample, x_a is an anomalous sample, and $B[0,R_1]$ and $B[0,R_2]$ are the closed balls around the origin of radius R_1 and R_2, respectively. Observe that the reconstruction objective uses only normal training points, while the constraints use both normal and anomalous training samples. The constraints are visualized in Fig. 1.

The constrained optimization problem is solved using Constraint-Guided Gradient Descent (CGGD) [34], where the direction of the constraints is defined as the vector defined by the origin and the encoding itself or the opposite. Observe that the direction of the constraints is the shortest path to the feasible region (FR). The FR is defined as a set of model parameters, for which the predictions of all training data satisfy all constraints. This method treats the constraints as hard constraints, meaning that the constraints should be satisfied for every example and approximates a model that satisfies the constraints on the training set. CGGD yields this result by assigning more weight to optimizing the constraints compared to optimizing the loss function during every step of the constrained optimization problem. The loss function in this case is the reconstruction objective of the AE for the normal data.

By considering the optimization problem and the method for solving the optimization problem in this manner, the learning will prioritize finding discriminative features over generative features. However, this does not imply that only discriminative features will be constructed in the latent space. Moreover, the optimal threshold on the latent space is expected to be somewhere in the interval $[R_1,R_2]$. Hence, the threshold for determining if data is anomalous or not is defined as:

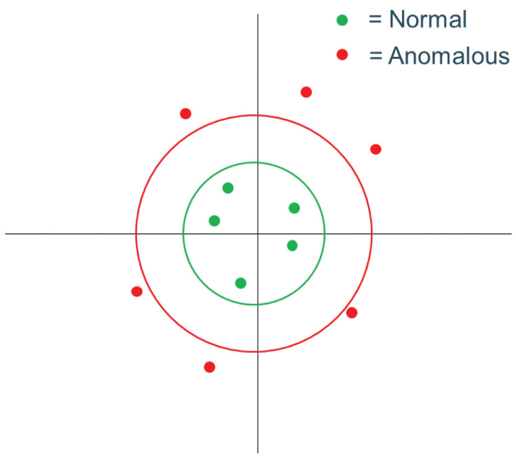


Fig. 1. A 2D illustration of the constraints used in CG-AE. The green (smallest) circle represents the sphere with radius R_1. The red (largest) circle represents the sphere with radius R_2.

$$T := R_1 + (R_2 - R_1)\frac{N_a}{N_n + N_a}, \tag{4}$$

where N_a and N_n are the number of anomalous and normal data in the training set, respectively. A point is classified as normal if the norm of its encoding is smaller or equal than T and is classified as anomalous if the norm of its encoding is (strictly) larger than T.

In this setting, the constraints which are applied to a given training example are conditioned on the corresponding label, that is, if the example is normal or anomalous. Therefore, it is not possible to construct a network that guarantees satisfaction of all constraints on all data, as it requires a perfect anomaly classification on unseen data. In [35], an overview of many approaches that use constraints on neural networks is given. Observe that all methods therein that guarantee satisfaction of the constraints would require access to the label of the points or are not applicable in this case since they do not support this type of constraint. In the CGGD framework, the constraints are considered to be hard constraints because the training can only be converged if all the constraints are satisfied on the training set, which leads to a perfect AD model on the training data.

IV. THEORETICAL COMPARISON

In this section, a theoretical comparison is made between the methods mentioned above. In particular, the main advantages of CG-AE are compared to the previous methods. First, models obtained from DSVDD with the center set at the origin can lead to the trivial solution where all the weights are set to 0 [16]. This is not possible for AE-DSVDD and CG-AE, because the reconstruction objectives (2) and (3) will not be locally optimal for the corresponding learning objectives. Moreover, under the assumption that anomalous data is available during the training procedure, it follows by the definition of CGGD that the model will be updated, even when initialized with all the weights put to 0. Therefore, the restriction that the center cannot be set as 0 and biases are not allowed in DSVDD are in fact not a restriction for AE-DSVDD and CG-AE.

Second, as mentioned earlier in some applications, it is required to have a threshold that determines if a given example is anomalous or not. Therefore, for these applications this threshold needs to be determined before employing the model. Hence, the threshold should be part of the learning cycle of the model or should be determined using the training set after learning. For DSVDD and AE-DSVDD, there are methods that can compute a threshold. However, it depends highly on the training of the model and the dataset resulting in thresholds that vary significantly between different datasets. For CG-AE, it is expected to find a good threshold in the interval $[R_1, R_2]$ for models that have converged and obtained a high satisfaction ratio [34], which is the ratio of the number of satisfied constraints over the number of constraints, on the training set. Since these models are likely to separate normal from anomalous data points on the test set if the train and test distributions are similar enough. Moreover, if the difference between R_2 and R_1 is large, then a relatively large difference in encodings of the training and test samples needs to occur in order to misclassify a sample.

Thirdly, both DSVDD and AE-DSVDD can converge to a model that assigns a wrong label to a training point due to the fact that the objective is locally optimal. For DSVDD, this can be a result of the gradient of a normal point being pointed outward and the gradient of an anomalous point being pointed inward in the opposite direction, which results in a gradient of 0. For AE-DSVDD, the same phenomenon can occur as well as the case where the gradient of the reconstruction and the DSVDD are opposite to each other, which results once more in a gradient of 0. Observe that by definition of the CGGD optimization procedure, this phenomenon cannot occur for a training procedure that has converged to a model with a perfect satisfaction ratio.

Fourth, the Soft-Boundary extension of DSVDD [16] is similar in terms of learning objective compared to CG-AE. However, the soft-boundary is added as a regularization term to the learning objective which leads to no guarantee, even on the training set, in terms of how many examples are consistent with the soft-boundary. This phenomenon was already illustrated in [34].

Lastly, the parameters R_1 and R_2 determine a margin, in which, for models that have converged, a good threshold is expected to be found. Note that if the model is sufficiently flexible to model the task, it is likely to converge. The performance of the model with respect to the constraints is measured by the satisfaction ratio. If this is insufficiently high, then the AE can be made more complex or other choices can be made for R_1 and R_2. Observe that if R_1 is increased, then the size of the volume of the latent space in which the normal points can be mapped is increased.

V. EXPERIMENTAL SETUP

In this section, the dataset, the preprocessing, the model architecture, the setting of the hyperparameters, the performed experiments, and the relevant metrics are discussed.

A. DATASET

The dataset used in the experiments was collected by Flanders Make and consists of accelerometer data measured during accelerated lifetime tests of bearings and was previously used in [33]. A bearing test rig setup is shown in Fig. 2. Radial accelerations were measured at a sampling frequency of 50 kHz by an accelerometer attached to the bearing housing. All measurements were performed under

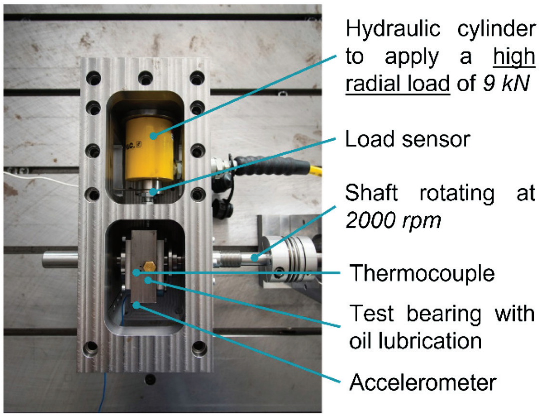
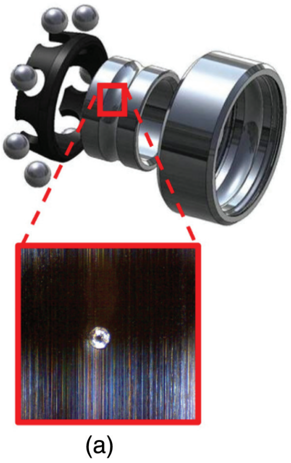

Hydraulic cylinder to apply a high radial load of 9 kN
Load sensor
Shaft rotating at 2000 rpm
Thermocouple
Test bearing with oil lubrication
Accelerometer

Fig. 2. Example of a bearing test rig setup.

fixed operating conditions of a high radial load of 9 kN and with the shaft of the setup rotating at 2000 rpm. To only retain data that was collected under these fixed conditions, data where the rotating speed was lower than 2000 rpm was omitted. The stopping condition for the tests was set at 20 g peak acceleration.

A total of 49 accelerated lifetime tests were performed on a fleet of 7 bearing test rigs, starting with a small initial indentation in the inner race of the bearing until a severe surface fatigue fault was introduced. Additionally, 21 *healthy* tests without this initial indentation in the race were performed for a short duration and not resulting in actual bearing failure. Due to various reasons, explained in [33], only 34 *faulty* tests were retained, resulting in a total of 55 remaining tests. Although these are called *faulty* test and have the small initial indentation, the beginning of these tests are considered as healthy data for the experiments, as will be explained in the next paragraph. Figure 3 shows an example of this initial indentation and the resulting severe surface fatigue fault at the end of the accelerated lifetime test.

No exact ground truth was provided with this dataset; hence, manual annotation was performed to obtain three segments on the time axis for each test. An example of this annotation can be found in Fig. 4, where cutoff points p_h and p_f are shown. Data prior to p_h is considered as normal, and data after p_f is considered as anomalous. The data between both cutoff points will not be used in experiments as the condition of the bearing is undefined. A detailed explanation about the selection of the cutoff points can be found in [33].



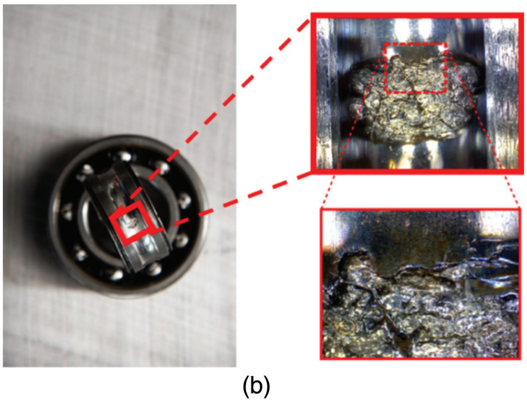


Fig. 3. Initial indentation (a) with a diameter of 300 μm at the inner race and the resulting surface fatigue fault (b) at the end of the accelerated lifetime test.

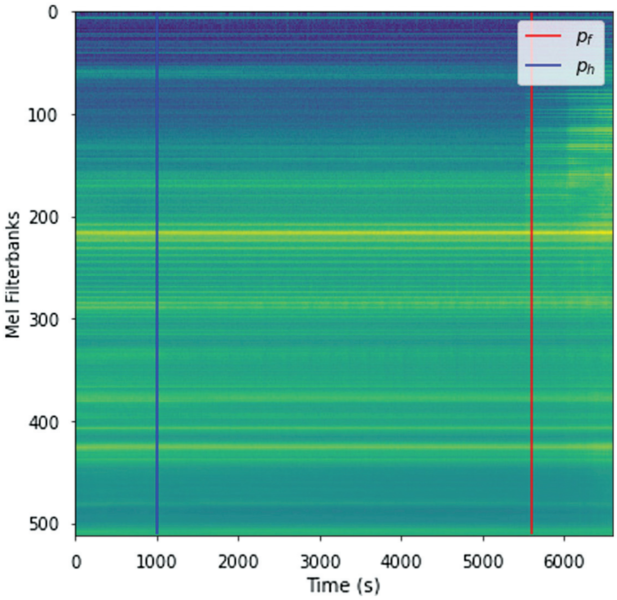


Fig. 4. Example of manual annotation. Points to the left of the blue line (around 1000 s) are normal. Points to the right side of the red line (around 5500 s) are anomalous.

B. PREPROCESSING

Prior to being provided as input to the DL model, the acceleration signals were first transformed to the log mel spectra, as was done in [33,36]. These were calculated using a window size and hop size of 1s into 512 mel bands. Finally, to provide some temporal information, 8 consecutive seconds were combined to create a single frame to input to the DL model. These 8 seconds will receive the same anomaly score during evaluation. As the different tests show some dissimilarity, each test is separately standardized so that the data prior to p_h has zero mean and unit variance after the previous preprocessing step.

C. EXPERIMENTAL DETAILS

All models used during training use the same AE architecture, with DSVDD omitting the biases due to the algorithm restrictions. The architecture of the encoder consists of 3 convolutional layers, with 64, 64, and 32 filters, respectively, and a fully connected layer with 16 neurons that acts as the encoding layer. The decoder consists of a fully connected layer, with 1024 neurons to recreate the input shape to the encoding layer, followed by 3 convolutional layers, with 32, 64, and 64 filters, respectively, and finally a convolutional layer with 1 filter as the output layer.

All models were trained for 150 epochs using the Adam optimizer [37] with a learning rate of $1e^{-3}$. If the model did not improve the learning rate was halved, with a limit at $1e^{-6}$. The training performance of the model is measured with learning objectives (1), (2), and (3). Additionally, for CG-AE the satisfaction ratio is also monitored. An improvement corresponds to the lowering of the loss function and, in the case of CG-AE, an increase in the satisfaction ratio or being sufficiently high. Sufficiently high means, in this context, that 95% of the constraints are satisfied. Data was provided to the model in batches of 128 input frames.

D. EXPERIMENTS

As discussed in the dataset description, 15 *faulty* tests were not used and the 21 *healthy* tests do not contain any anomalous data and, therefore, they cannot be used to evaluate a model. This leads to 34 test folds that are used in a leave-one-test-out scheme, where the healthy runs are added during training. In each of the folds a single test was used as test set, so the model must generalize to an unseen test, and the remaining tests were used to construct the training and validation sets by randomly sampling 75% and 25%, respectively. Additionally, the anomalous data from various amounts of tests was used, more specifically 5%, 25%, 50%, and 100%. For the results on these folds, the mean and standard deviation is computed over the different folds.

The performance of CG-AE with regard to the ability to discriminate normal from anomalous behavior is studied in comparison with DSVDD and AE-DSVDD. For each method, a single fixed threshold is used. Recall that for CG-AE this is done by (4). Moreover, this threshold is determined using only information about the training data. Additionally, a comparison will be made to investigate the similarity of this fixed threshold is to an optimal threshold, and this will be explained in detail later. To this end, an experiment was performed with the leave-one-test-out scheme described above. For this experiment. The radii R_1 and R_2 used by CG-AE were set to 3 and 5, respectively. No fine-tuning was performed for the choice of these radii.

Next to studying the performance of CG-AE in comparison with DSVDD and AE-DSVDD, an ablation study was performed to investigate the effect of the radii chosen for CG-AE. A total of three different settings for R_1 and R_2 were evaluated, more specific, $(R_1,R_2) \in \{(3,5),(1,2),(1,5)\}$.

E. METRICS

The evaluation of the studied methods will be split into two objectives: (i) the discriminative performance between normal and anomalous data, and (ii) the difference in thresholds between the various thresholding methods.

First, the performance with regard to classification of normal and anomalous samples is evaluated using the F1 score (5) and the balanced accuracy (BA) (6). The former is the harmonic mean of the precision (7) and recall (8), and the latter is the mean of the recall for each class, normal and anomalous in this context, that is,

$$precision = \frac{tp}{(tp + fp)}, \tag{5}$$

$$recall = \frac{tp}{(tp + fn)}, \tag{6}$$

$$\mathrm{F1} = \frac{2(precision \times recall)}{(precison + recall)}, \tag{7}$$

$$BA = \frac{1}{2}\left(\frac{tn}{(tn + fp)} + \frac{tp}{(tp + fn)}\right), \tag{8}$$

where anomalous is the positive class, and tp, fp, tn, and fn being the amount of true positive, false positive, true negative, and false negative samples, respectively.

The F1 score rewards classify more samples as the positive class because this increases the numerator. Moreover, if this does not lead to a large increase of false positives, then the F1 score increases. Since this is not desirable in every application, the BA is considered as well. The BA is well suited for unbalanced classification problems, which AD is by default.

To evaluate the above metrics, a threshold is needed. For both DSVDD and AE-DSVDD, three different methods are considered for determining this threshold. The first method uses the test set itself to find an *optimal* threshold (T_{opt}), for the considered fold, where the above metrics are maximized, and this is done for both metrics separately. The performance using this threshold will serve as an upper limit to the attainable performance but will not be evaluated in detail as this work focuses on thresholds obtained using only information about the training data.

The second method [30] computes the mean μ_n of the anomaly score for the normal data in the training set and the standard deviation σ_n of this anomaly score. The threshold (T_{train}) is then set to $\mu_n + 3\sigma_n$, for the considered fold.

The third method [38] fits a sigmoid function on the anomaly score for the normal data in the training set so that the minimum, median, and 99th percentile of this anomaly score match to 0.01, 0.25, and 0.5 on the sigmoid, respectively. This emulates the behavior of a sigmoid activation on the output layer of a DL model. The threshold is set to 0.6 after mapping the anomaly score of the test set using the obtained sigmoid function. By inverting the sigmoid function, the value of this threshold ($T_{sigmoid}$) prior to the mapping can be determined for the considered fold.

Second, an evaluation is performed regarding the thresholds obtained using these various methods. This evaluation will make a comparison using the relative difference between the *optimal* threshold and the threshold obtained using the other methods. The relative difference is defined by the function:

$$T_{diff,m} : \mathbb{R}^+ \to \mathbb{R} : T \mapsto \frac{T_{opt} - T}{T_{opt}}, \tag{9}$$

where T_{opt} is the optimal threshold for the method for which the threshold T is used, m is the name of the method, and $\mathbb{R}^+$ is the set of positive real numbers. To evaluate this function for CG-AE, T_{opt} is computed in the same way as for DSVDD and AE-DSVDD. However, this threshold is not used to compute the F1 and BA metric.

Observe that the value T_{opt} changes for the different methods and, thus, it could happen that

$$T_{diff,DSVDD}(T^*) \neq T_{diff,AE-DSVDD}(T^*), \tag{10}$$

for some threshold T^*.

In this way, an evaluation can be made how closely the thresholds determined using the training set match the possible "optimal" threshold. Observe that negative values can occur and denote that the chosen threshold is higher than the optimal threshold.

VI. RESULTS

In this section, the results of the performed experiments will be discussed. First, the performance of CG-AE in comparison with DSVDD and AE-DSVDD will be evaluated. Second, an ablation study with regard to different choices for R_1 and R_2 will be discussed.

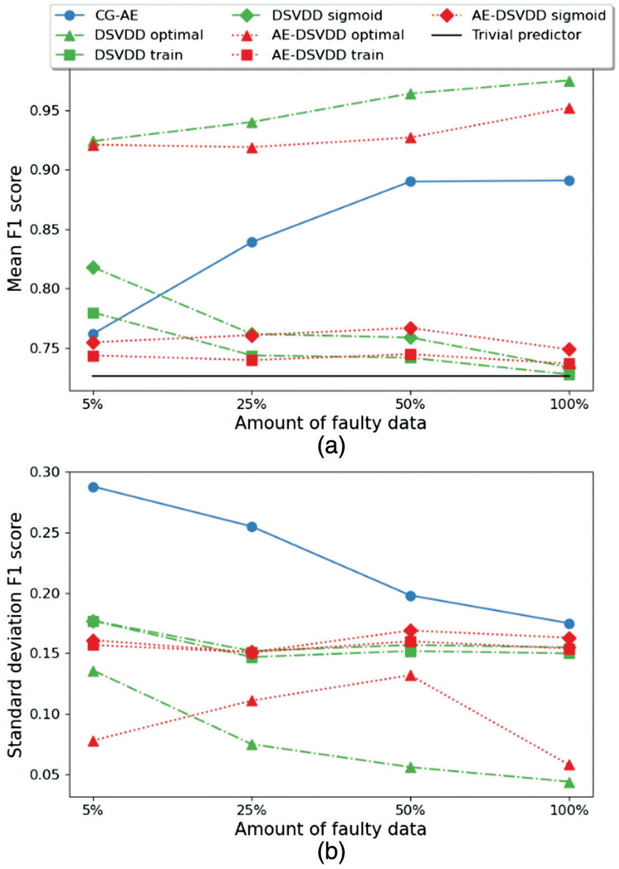
A. DIFFERENT METHODS

First, the obtained F1 scores are computed for models obtained when training with the three previously discussed methods, that is, CG-AE, DSVDD, and AE-DSVDD. Note that the optimal versions of (AE-)DSVDD uses the test data to determine the threshold. Their other counterparts and also CG-AE do not use this information. The mean and standard deviation of the three methods and different manners of determining a threshold are shown in Fig. 5. The trivial predictor is added as a lower bound. This predictor predicts every example as anomalous, as this is considered the positive class; hence, several predictions will be incorrect as the testing data also contains examples of normal data, as discussed in the dataset description.

It is immediately clear that in these experiments, the performance of CG-AE is in between the upper bound of the performance of DSVDD and AE-DSVDD and the performance of DSVDD and AE-DSVDD where only the training data can be used for determining a threshold. This shows that there could be a small drop-off between CG-AE with its fixed threshold and the other methods with information about the test set. However, if the test set could be used to adjust the threshold, then this performance of CG-AE can likely be improved as well. Nevertheless, when at least 25% of the anomalous data is used, there is a significant difference in mean F1 score of CG-AE compared with the mean F1 score of DSVDD and AE-DSVDD with a threshold determined on the training set. The standard deviation of CG-AE is relatively high. But, the performance of DSVDD and AE-DSVDD is only slightly higher than a trivial predictor that predicts every example as anomalous. Hence, this could be the main reason of the relatively small standard deviation of DSVDD and AE-DSVDD. Note that this implies that these methods have not learned well, because there is only a small improvement compared to the trivial predictor.

Second, we will investigate how the threshold changes from fold to fold during the experiments. As was mentioned in the section Metrics, the relative difference between the optimal threshold for that fold and the threshold used for the method is computed. This should be, in absolute value, on average small as well as for the standard deviation, because this means that the threshold varies only slightly between different folds. Moreover, this could imply that the threshold is less likely to change between different datasets. The results are shown in Fig. 6.

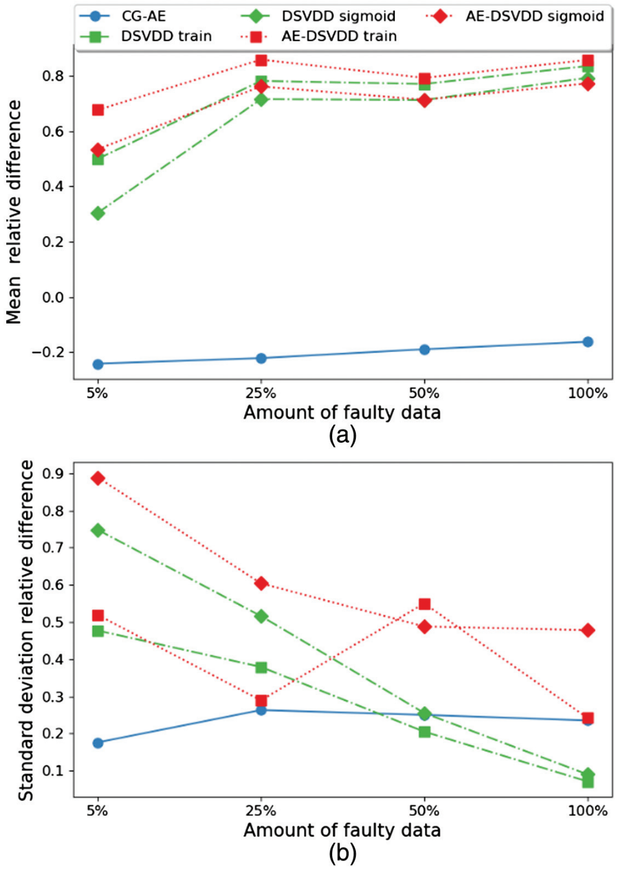
Figure 6 indicates that on average, CG-AE has the lowest mean difference in absolute value. Moreover, this difference decreases in absolute value when the amount of anomalous data points in the training set is increased. The standard deviation of CG-AE is almost always equal to 0.2



(a)

(b)

Fig. 5. The mean (a) and standard deviation (b) of the F1 score for different methods and different choices of threshold.



(a)

(b)

Fig. 6. The mean (a) and standard deviation (b) of the relative difference in norm between the threshold and the corresponding optimal threshold for the F1 score and for the different methods.

for the different amounts of anomalous data points in the training set. It is observed that, when the number of anomalous points is increased, the standard deviation decreases and the mean increases in absolute value for DSVDD and, albeit not as much, AE-DSVDD. This phenomenon can likely be attributed to the objective of these methods that aim at mapping normal data close to a center and anomalous data further away from the same center during training. As more anomalous data is added, its weight on the loss function with respect to the weight of normal data is increased. In the latent space, this might cause all data to be projected further away from the center. As a result, the optimal threshold as well as the training statistics increase as a function of the amount of anomalous data used during training. By (9), the increase in relative difference is attributed to a larger increase in optimal threshold compared to the increase in training statistics. This shows that there is no fixed relation between the optimal threshold and a threshold based on the training statistics for these methods. However, this does not imply that the thresholds of DSVDD do not vary a lot between the different folds because the mean difference is approximately 0.7. Recall that this is a relative difference, meaning that if the optimal threshold would be 3, then the mean difference between the optimal threshold and the training threshold would be 2.1, which is relatively large.

Thirdly, we will investigate the BA when the problem is considered as a classification problem. Similarly, as for the F1 score, the mean and standard deviation of the BA is computed for the different methods and different ways of determining a threshold. The corresponding results are shown in Fig. 7.

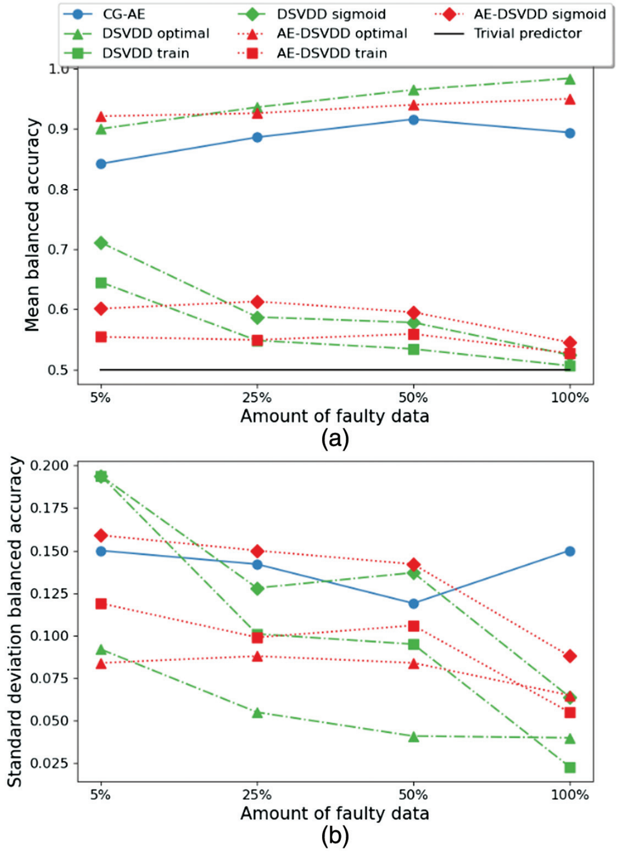


Fig. 7. The mean (a) and standard deviation (b) of the BA for the different methods and different thresholds.

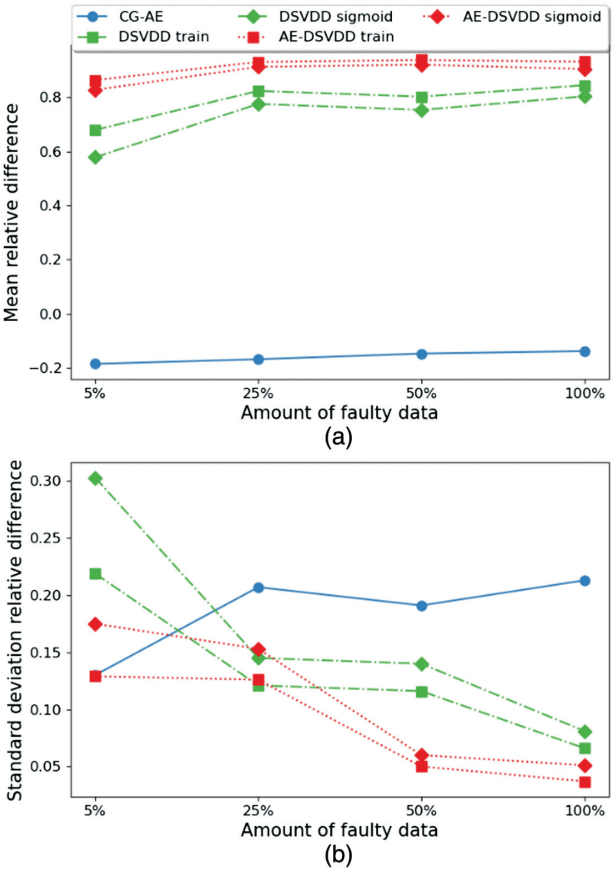


Fig. 8. The mean (a) and standard deviation (b) of the relative difference between the threshold and the corresponding optimal threshold for the BA and the different methods.

The BA of CG-AE is on average very close to the upper bound on the performance of DSVDD and AE-DSVDD. This means that on average, CG-AE performs very well in terms of BA. The standard deviation of CG-AE is slightly higher than the other methods. However, the standard deviation of DSVDD and AE-DSVDD with the thresholds determined on the training set are low because they only slightly outperform the trivial predictor. Meaning that almost all data is projected to values outside of the sphere determined by the threshold. Relatively small changes to this threshold will not change much to the BA. This phenomenon was also observed for the F1 score.

Lastly, we are interested in the relative difference between the thresholds leading to the optimal BA for each fold and the considered methods of determining a threshold for the different methods. The relevant results are shown in Fig. 8.

The results are very similar as for the relative difference between the thresholds for the F1 score. The main argument for this is that only the optimal threshold differs. In particular, the optimal threshold for the F1 score is lower than the optimal threshold for the BA, since the F1 rewards labeling more points as anomalous while the BA does not. It is remarkable that the standard deviation for CG-AE does not vary a lot between different amounts of anomalous points.

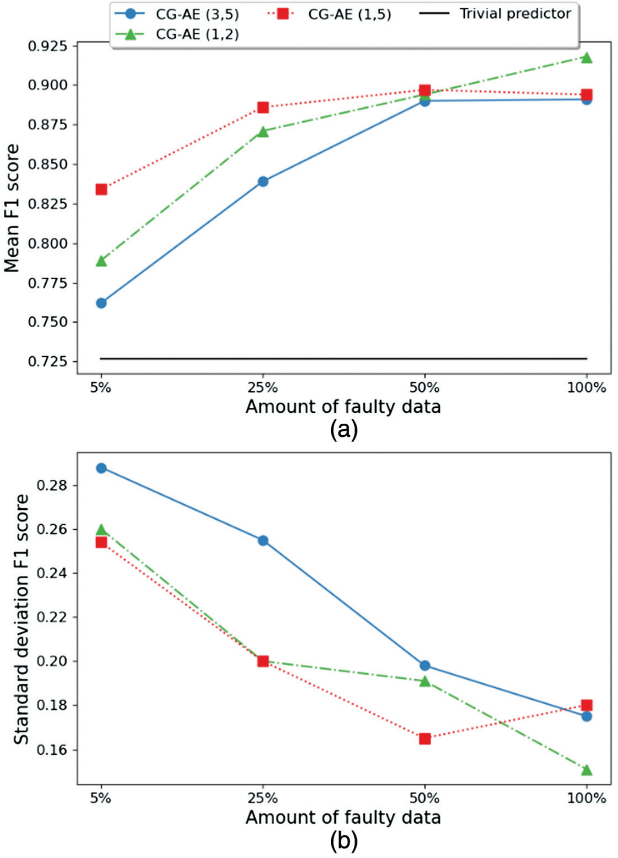


Fig. 9. The mean (a) and standard deviation (b) of the F1 score for CG-AE for different choices of radii.

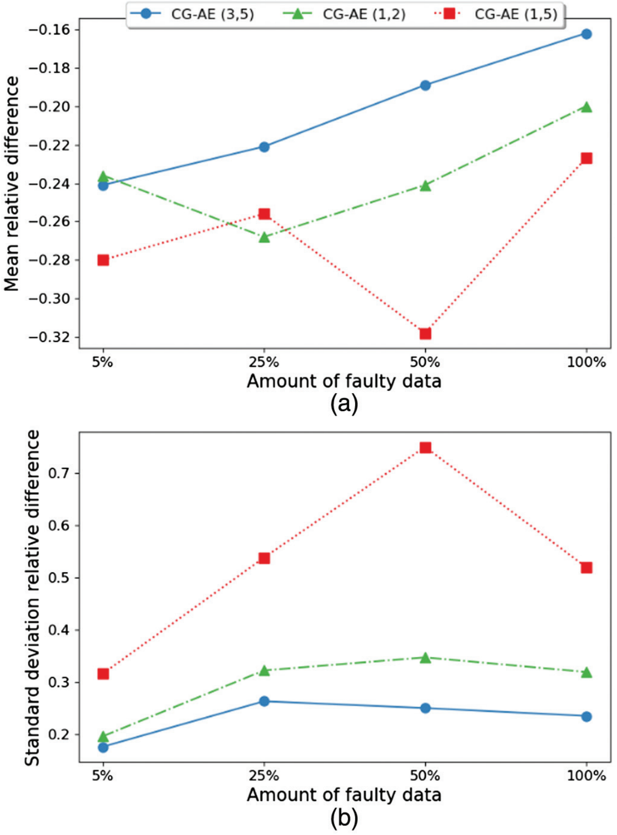


Fig. 10. The mean (a) and standard deviation (b) of the relative difference between the threshold and the corresponding optimal threshold for the F1 score for CG-AE and different choices of radii.

Therefore, this shows once more that the thresholds do not vary a lot between different folds for CG-AE.

B. DIFFERENT RADII

First, the F1 score is considered for different choices of the radii in CG-AE. The related results are shown in Fig. 9. These results indicate that there is only a small difference in terms of mean and standard deviation. Observe that the choice $(R_1, R_2) = (3{,}5)$ yields in fact the worst performance, both on average and in terms of standard deviation. It is clearly visible that the mean increases when the number of anomalous points in the training set are increased. At the same time, the standard deviation decreases.

Second, the relative difference between the threshold and the corresponding optimal threshold for the F1 score is considered. Figure 10 indicates that the proposed threshold is often smaller than the optimal threshold. Nevertheless, the difference between the different choices is small. Except for CG-AE (1,5) and 50% of the anomalies, which does not follow the same pattern as the other choices. A possible explanation could be that the margin defined as the difference between R_2 and R_1 is too large.

Thirdly, the mean and standard deviation of the BA are computed, and the results are shown in Fig. 11. On average, there are only small differences between the different choices. The standard deviation varies more except that the largest difference is still 0.04 across the different choices of radii and different number of anomalous points in the training set.

Lastly, the mean and standard deviation of the relative difference between the threshold and the optimal threshold are shown in Fig. 12. On average, there are no large differences between the different choices. It seems that the difference does not always decrease in absolute value when the number of anomalous points in the training set is increased. The standard deviation of the choice (1,5) is larger than the standard deviation of the other choices. This could be a consequence of the network being insufficiently complex to make such a large difference between normal and anomalous points in the training set.

It should be stressed that even though the thresholds can vary differently for different choices of the radii of the spheres, the obtained performance in terms of F1 score and BA remain the same. Therefore, the proposed formula for computing a threshold seems a robust manner for determining the threshold. Moreover, different choices of the radii do not have a large impact on the performance for the considered metrics for AD.

VII. FUTURE WORK

First, an obvious extension of CG-AE is to include anomalous data in the reconstruction objective as well. This could lead to additional structure on the latent space and better generalization due to generative features also being learned for anomalous data.

Second, during inference on the test set, it is possible to use a dynamic threshold mechanism [32]. This could lead to

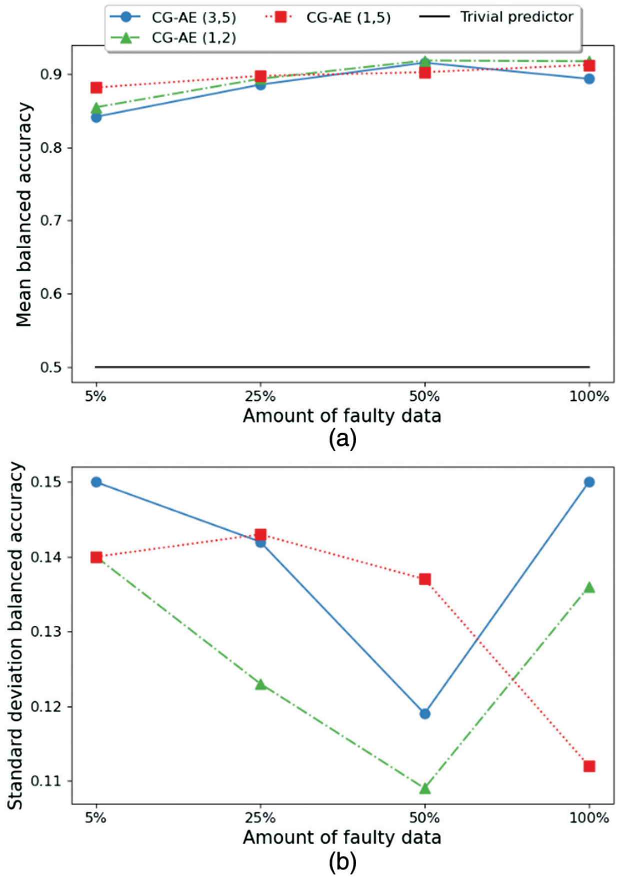


Fig. 11. The mean (a) and standard deviation (b) of the BA for CG-AE and different choices of radii.

an improvement when applying the obtained models to unseen data.

Thirdly, the function (4) that determines the threshold tends to be too high compared to the best possible threshold for small amounts of anomalies. Hence, a more advanced function could be proposed. Other insights might also be incorporated, such as considering if false positives or false negatives are less or more important for the application.

Lastly, the chosen threshold and model could be fine-tuned when some data of the test set is available in a similar fashion as is possible for DSVDD [33] and AE-DSVDD.

VIII. CONCLUSION

The proposed method CG-AE determines a threshold only depending on the amount of normal and anomalous data. Moreover, this threshold is independent of the distribution of the training data in the latent space. Moreover, the obtained performance of the thresholds does not vary between different choices of the radii of the spheres that define the constraints and different folds in the experiments. The performance of the model increases when more anomalous data is added in the training set.

Acknowledgments

This research received funding from the Flemish Government (AI Research Program). This research has received support of Flanders Make, the strategic research center for the manufacturing industry.

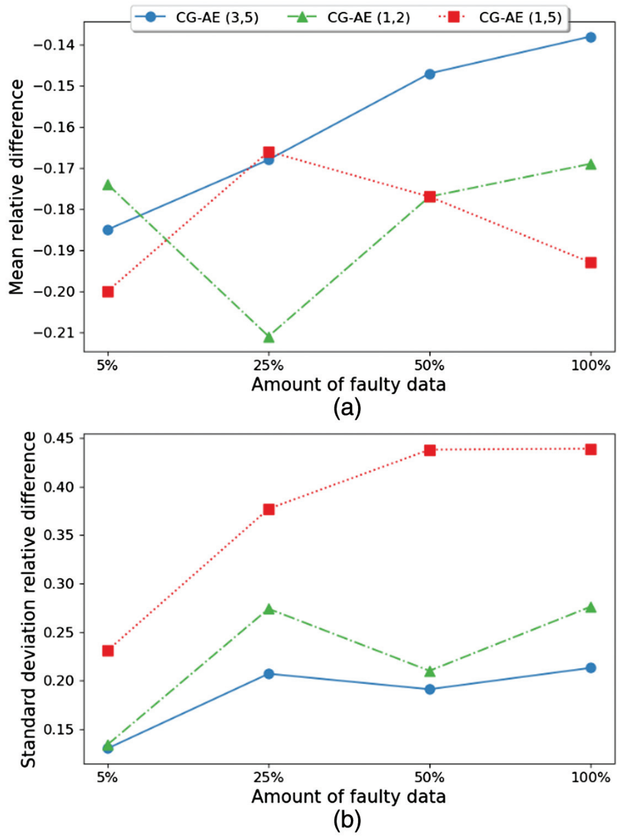


Fig. 12. The mean (a) and standard deviation (b) of the relative difference of the threshold and the corresponding optimal threshold for the balanced accuracy for CG-AE and different choices of radii.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

References

[1] M. Ahmed, A. N. Mahmood, and R. Islam, "A survey of anomaly detection techniques in financial domain," *Future Gener. Comput. Syst.*, vol. 55, pp. 278–288, 2016. DOI: 10.1016/j.future.2015.01.001.

[2] T. Fernando, H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, "Deep learning for medical anomaly detection – a survey," *ACM Comput. Surveys (CSUR)*, vol. 54, no. 7, Jul. 2021. DOI: 10.1145/3464423.

[3] Z. Liu, N. Thapa, A. Shaver, K. Roy, X. Yuan, and S. Khorsandroo, "Anomaly detection on lot network intrusion using machine learning," in *2020 Int. Conf. Artif. Intell., Big Data, Comput. Data Commun. Syst., icABCD 2020 – Proc.*, Aug. 2020. DOI: 10.1109/ICABCD49160.2020.9183842.

[4] P. Kamat and R. Sugandhi, "Anomaly detection for predictive maintenance in industry 4.0- a survey," *E3S Web Conf.*, vol. 170, p. 02007, May 2020. DOI: 10.1051/E3SCONF/202017002007.

[5] D. H. Hoang and H. D. Nguyen, "A PCA-based method for IoT network traffic anomaly detection," *Int. Conf. Adv. Commun. Technol., ICACT*, vol. 2018, pp. 381–386, Mar. 2018. DOI: 10.23919/ICACT.2018.8323766.

[6] B. Scholkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt, "Support vector method for novelty detection," *Adv. Neural Inf. Process. Syst.*, vol. 12, pp. 582–588, 1999.

[7] D. M. J. Tax and R. P. W. Duin, "Support vector data description," *Mach. Learn.*, vol. 54, no. 1, pp. 45–66, Jan. 2004. DOI: 10.1023/B:MACH.0000008084.60811.49/METRICS.

[8] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: a survey," *ArXiv*, p. arXiv:1901.03407, 2019.

[9] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006. DOI: 10.1126/SCIENCE.1127647.

[10] D. Y. Oh and I. D. Yun, "Residual error based anomaly detection using auto-encoder in SMD machine sound," *Sensors 2018*, vol. 18, no. 5, p. 1308, Apr. 2018. DOI: 10.3390/S18051308.

[11] T. H. Lin and J. R. Jiang, "Anomaly detection with autoencoder and random forest," in *Proc. – 2020 Int. Comput. Symp., ICS 2020*, pp. 96–99, Dec. 2020. DOI: 10.1109/ICS51289.2020.00028.

[12] P. Luo, B. Wang, T. Li, and J. Tian, "ADS-B anomaly data detection model based on VAE-SVDD," *Comput. Secur.*, vol. 104, May 2021. DOI: 10.1016/J.COSE.2021.102213.

[13] P. Primus, V. Haunschmid, P. Praher, and G. Widmer, "Anomalous sound detection as a simple binary classification problem with careful selection of proxy outlier examples," in *Proc. Detect. Classifi. Acoust. Scen. Event. 2020 Workshop (DCASE2020)*, pp. 170–174, Nov. 2020.

[14] N. Goernitz, M. Kloft, K. Rieck, and U. Brefeld, "Toward supervised anomaly detection," *J. Artif. Intell. Res.*, vol. 46, pp. 235–262, Jan. 2014. DOI: 10.1613/jair.3623.

[15] N. Arunraj, R. Hable, M. Fernandes, K. Leidl, and M. Heigl, "Comparison of supervised, semi-supervised and unsupervised learning methods in network intrusion detection system (NIDS) application," *Anwend. Konzepte Wirtsch.*, vol. 6, pp. 10–19, 2017.

[16] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. Siddiqui, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," *PMLR*, vol. 80, pp. 4393–4402, Jul. 2018.

[17] L. Ruff, R. Vandermeulen, N. Goernitz, A. Binder, M. Müller, K. Müller, and M. Kloft, "Deep semi-supervised anomaly detection," in *8th Int. Conf. Learn. Rep. (ICLR)*, vol. 8, 2020.

[18] R. Chalapathy, A. K. Menon, and S. Chawla, "Anomaly detection using one-class neural networks," *ArXiv*, p. arXiv:1802.06360, 2018.

[19] F. Giannini, M. Diligenti, M. Maggini, M. Gori, and G. Marra, "T-norms driven loss functions for machine learning," *Appl. Intel.*, pp. 1–15, Jul. 2019. DOI: 10.1007/S10489-022-04383-6.

[20] J. Xu, Z. Zhang, T. Friedman, Y. Liang, and G. Van den Broeck, "A semantic loss function for deep learning with symbolic knowledge," *35th Int. Conf. Mach. Learn., ICML 2018*, vol. 12, pp. 8752–8760, Nov. 2017.

[21] Z. Zhang and X. Deng, "Anomaly detection using improved deep SVDD model with data structure preservation," *Pattern Recognit. Lett.*, vol. 148, pp. 1–6, Aug. 2021. DOI: 10.1016/J.PATREC.2021.04.020.

[22] Y. Zhou, X. Liang, W. Zhang, L. Zhang, and X. Song, "VAE-based Deep SVDD for anomaly detection," *Neurocomputing*, vol. 453, pp. 131–140, Sep. 2021. DOI: 10.1016/J.NEUCOM.2021.04.089.

[23] H. Hojjati and N. Armanfard, "DASVDD: deep autoencoding support vector data descriptor for anomaly detection," *ArXiv*, p. arXiv:2106.05410v2, Jun. 2021.

[24] N. M. Nabhan, A. Ghazaly, and M. M. O. Samy, "Bearing fault detection techniques-a review," *Turkish J. Eng., Sci. Technol.*, vol. 3, pp. 1–18, Jan. 2015.

[25] D. T. Hoang and H. J. Kang, "A survey on deep learning based bearing fault diagnosis," *Neurocomputing*, vol. 335, pp. 327–335, Mar. 2019. DOI: 10.1016/J.NEUCOM.2018.06.078.

[26] W. Cheng, Z. Li, and F. Cheng, "Deep robust autoencoder based framework for bearing fault detection," in *2022 Global Reliab. Prog. Health Manag. Conf., PHM-Yantai 2022*, 2022. DOI: 10.1109/PHM-YANTAI55411.2022.9941789.

[27] S. Zhang, F. Ye, B. Wang, and T. G. Habetler, "Semi-supervised learning of bearing anomaly detection via deep variational autoencoders," *ArXiv*, p. arXiv:1912.01096v2, Dec. 2019.

[28] L. Kou, J. Chen, Y. Qin, and W. Mao, "The robust multi-scale deep-SVDD model for anomaly online detection of rolling bearings," *Sensors 2022*, vol. 22, no. 15, p. 5681, Jul. 2022. DOI: 10.3390/S22155681.

[29] C. Liu and K. Gryllias, "A semi-supervised Support Vector Data Description-based fault detection method for rolling element bearings based on cyclic spectral analysis," *Mech. Syst. Signal Process.*, vol. 140, p. 106682, Jun. 2020. DOI: 10.1016/J.YMSSP.2020.106682.

[30] A. Giannoulidis, A. Gounaris, N. Nikolaidis, A. Naskos, and D. Caljouw, "Investigating thresholding techniques in a real predictive maintenance scenario," *ACM SIGKDD Explor. Newslett.*, vol. 24, no. 2, pp. 86–95, Dec. 2022. DOI: 10.1145/3575637.3575651.

[31] A. Garg, W. Zhang, J. Samaran, S. Ramasamy, and C.-S. Foo, "An evaluation of anomaly detection and diagnosis in multivariate time series," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 6, pp. 2508–2517, Sep. 2021. DOI: 10.1109/TNNLS.2021.3105827.

[32] D. Jia, X. Zhang, J. T. Zhou, P. Lai, and Y. Wei, "Dynamic thresholding for video anomaly detection," *IET Image Process*, vol. 16, no. 11, pp. 2973–2982, Sep. 2022. DOI: 10.1049/IPR2.12532.

[33] M. Meire, R. Brijder, G. Dekkers, and P. Karsmakers, "Accelerometer-based bearing condition indicator estimation using semi-supervised adaptive DSVDD," *Annu. Conf. PHM Soc.*, vol. 14, no. 1, Oct. 2022. DOI: 10.36001/PHMCONF.2022.V14I1.3173.

[34] Q. Van Baelen and P. Karsmakers, "Constraint guided gradient descent: guided training with inequality constraints," in *Proc. 30th Eur. Symp. Artif. Neural Netwk., Comput. Intell. Mach. Learn.*, Louvain-la-Neuve, Belgium: Ciaco – i6doc.com, 2022, pp. 175–180. DOI: 10.14428/esann/2022.ES2022-105.

[35] E. Giunchiglia, M. C. Stoian, and T. Lukasiewicz, "Deep learning with logical constraints," in *Proc. Thirty-First Int. Joint Conf. Artif. Intell.*, California: International Joint Conferences on Artificial Intelligence Organization, Jul. 2022, pp. 5478–5485. doi: 10.24963/ijcai.2022/767.

[36] S. Lee, H. Yu, H. Yang, I. Song, J. Choi, J. Yang, G. Lim, K. Kim, B. Choi, and J. Kwon, "A study on deep learning application of vibration data and visualization of defects for predictive maintenance of gravity acceleration equipment," *Appl. Sci.*, vol. 11, no. 4, p. 1564, Feb. 2021. DOI: 10.3390/APP11041564.

[37] D. P. Kingma and J. L. Ba, "Adam: a method for stochastic optimization," in *3rd Int. Conf. Learn. Rep., ICLR 2015 – Conf. Track Proc.*, San Diego, CA, USA, May 7–9, 2015.

[38] C. Wu, F. Feng, S. Wu, P. Jiang, and J. Wang, "A method for constructing rolling bearing lifetime health indicator based on multi-scale convolutional neural networks," *J. Braz. Soc. Mech. Sci. Eng.*, vol. 41, no. 11, pp. 1–11, Nov. 2019. DOI: 10.1007/S40430-019-2010-6/TABLES/5.