Outlier Detection via the Principle of Maximal Coding Rate Reduction

Sixuan Chen
University of Waterloo
200 University Ave, Waterloo, Canada
s743chen@uwaterloo.ca

Vivi Wei Yu
University of Waterloo
200 University Ave, Waterloo, Canada
v22yu@uwaterloo.ca

Abstract

Outliers are data points deviate significantly from distribution of previously seen data. Deep learning techniques for detecting outliers have recently improved the cutting edge of detection performance for complex data sets such as large sets of images and text. These results have inspired new interest in the problem of outlier detection and led to the introduction of several new approaches. This paper will review state-of-the-art outlier detection method Distance-based Multiple Transformation Classification (GOAD) and adjusting its loss function using machine learning method the Principle of Maximal Coding Rate Reduction (MCR2) to improve the results.

1. Introduction

An outlier is an observation that deviates considerably from some concept of normality. The task of Outlier Detection (OD) is to identify abnormal data from large amount of normal data to detect minority, unexpected and rare events. Application of this may include detection of terrorist attacks, abnormal products, credit card frauds, cyber attacks and so on. OD has unique challenges compared to other machine-learning tasks. Therefore, special algorithms has been developed to solve this problem.

In order to solve outlier detection, we should effectively and efficiently learn the distribution from a finite set of i.i.d samples. However, this question is still one ongoing researcher because there are many different types of outliers and challenges related to identify them.

1.1. Types of outliers

Point outliers: A few individual instances are abnormal and most individual instances are normal. For example, normal CPU image and abnormal CPU image.

Conditional outliers: Also known as contextual outlier, it refers to the specific situation in which an individual instance is considered abnormal or normal depending on the situation. For example, as a result of a breaking news, the CPU usage of Twitter server may suddenly increases or decrease; Large number of high transaction of credit card suddenly appears frequently.

Group Outliers: A collection of instances is considered an outlier, but the individual instances in the group is not considered an outlier. In intrusion or fraud detection, outliers correspond to a sequence of multiple data points rather than a single data point. The set formed by fake accounts in social network is regarded as a abnormal subset of the group, but individual nodes in the subset may be indistinguishable to real accounts.

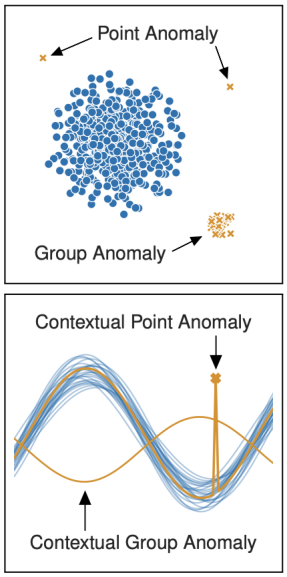

Figure 1. An illustration of the types of outliers. [9]

1.2. The challenge of Outlier detection

Unpredictability: Outliers are associated with many unknowns factors, for example: unknown burst behavior, data structures and distributions. They have unknown characteristics until they actually happen, such as terrorist attacks, scams and hacking activities;

Heterogeneity of anomaly classes: Outliers are irregular, and have characteristics completely different from each other. For example, in video surveillance, unusual events such as robberies, traffic accidents and larceny are all visually very different from each other, therefore, it's not easy to classify them based on some common feature.

Class imbalance: A common challenge common OD algorithm must face is the lack of large amount of out of distribution (OOD) data. Typically, the algorithm must generalize beyond training distribution of achieve high performance. This lack of supervision makes outlier detection more closely related to unsupervised learning.

1.3. Traditional methods of outlier detection

Reconstruction-based methods: The assumption is that outlier points are non-compressible or cannot be effectively reconstructed from lower-dimensional embedding space. This type of methods include PCA (Principal Component Analysis), Robust PCA, Random Projection and other dimensionality reduction methods. [6] [7]

Clustering Analysis methods: Clustering can create a model of the data, and the existence of outliers can distort and destroy the model. This type of methods include Gaussian Mixture Models, K-means and Multivariate Gaussian Models. [15] [8] [13]

One-Class method: A discriminative boundary is established for normal data, and points out of the boundary are considered outliers. Common methods include OC-SVM (One-Class Support Vector Machine) [4] [12]

1.4. Classification of outlier detection task

Supervised outlier detection: Both normal and abnormal instances in the training set have labels. The disadvantage of this method is that data labels are difficult to obtain or data is imbalanced (the number of normal samples is much larger than the number of abnormal samples). [9]

Semi-supervised outlier detection: Only a single classification in a training set (normal) instances, no abnormal instance are used in the training. Most current outlier detection research focus on semi-supervised methods. Some researchers considered unsupervised outlier detection methods to also be partly supervised. The reason is, although the outlier detection process is unsupervised, and the feature used in this process is learned in an unsupervised fashion, but the evaluation method is actually semi-supervised. Therefore, the boundary between unsupervised and semi-supervised OD is not very clear. [9]

Unsupervised outlier detection: In the training set, there is not only normal instance but also possibly abnormal instances, it is assumed that the proportion of normal data is much larger than that of abnormal data, and no label is used in the model training process. [9]

Weakly supervised outlier detection: The algorithm is designed especially for incomplete abnormal instances, coarse-grained labels, and noisy labels. [9]

2. Review of SOTA Outlier detection Algorithms

Currently, the state-of-art-outlier detection method for general data is called Distance-based Multiple Transformation Classification (GOAD) [3]. It is a distance-based multiple transformation classification method which unifies one-class Support Vector Data Description (OC-SVDD) [10] and Geometric-transformation classification (GEOM) methods [5].

The OC-SVDD is classical approach that has been introduced that transforms data to an isotropic feature space and fit the minimal hypersphere of radius R and center c_0 around the features of the normal training data. It poses the following objective:

$$\min \frac{1}{n} \sum_{i=1}^{n} ||f(x_i) - c||^2 + \mathcal{R} \tag{1}$$

Here, the neural network transformation $f(x)$ is learned to minimize the mean squared distance over all data points to center $c \in Z$. A recurring question in deep one-class classification is how to meaningfully regularize against a feature map collapse fc. Because test data is classified as anomalous if the following normality score is positive: $||f(x) - c_0||^2 - \mathcal{R}^2$, it is very easy to achieve the trivial solution of $f(x) = 0 \forall x$. Thus some adjustment need to be made.

Although one-class classification is generally more sample-efficient and more robust to non-representative sampling of the normal data (e.g., a sampling bias towards specific normal modes) [11], is consequentially also less informative.

Because of those limitation of OC-SVDD the authors of GOAD combine it with Geometric-transformation to learn more features of the samples through learning from augmentation. GEOM is done by transforming each image $x \in X$ using M different geometric transformations (rotation, reflection, translation) into $T(x, 1)..T(x, M)$. A significant issue with this methodology, is that the learned classifier $P(m'|T(x, m))$ is only valid for samples $x \in X$ which were found in the training set. Although getting such supervision is possible for some image tasks (where large external datasets can be used) this is not possible in the general case. For example, for tabular data, which exhibits much more variation between datasets. GOAD overcome these issues by using ideas from open-set classification and affine transformation.

GOAD transform X to $X_1...X_M$ and learn a feature extractor $f(x)$ using a neural network, which maps the original input data into a feature representation. Then model

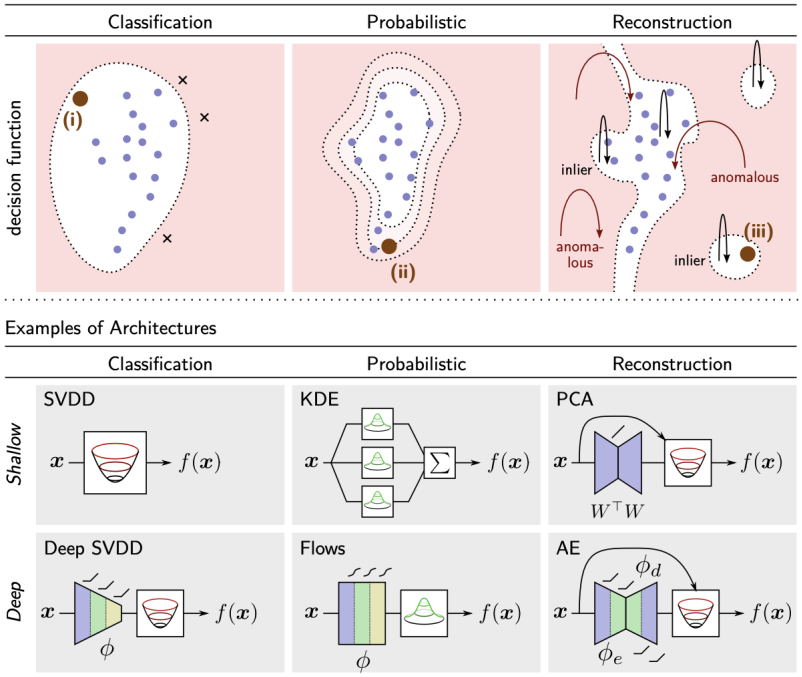

Figure 2. An overview of the different approaches to outlier detection. [9]

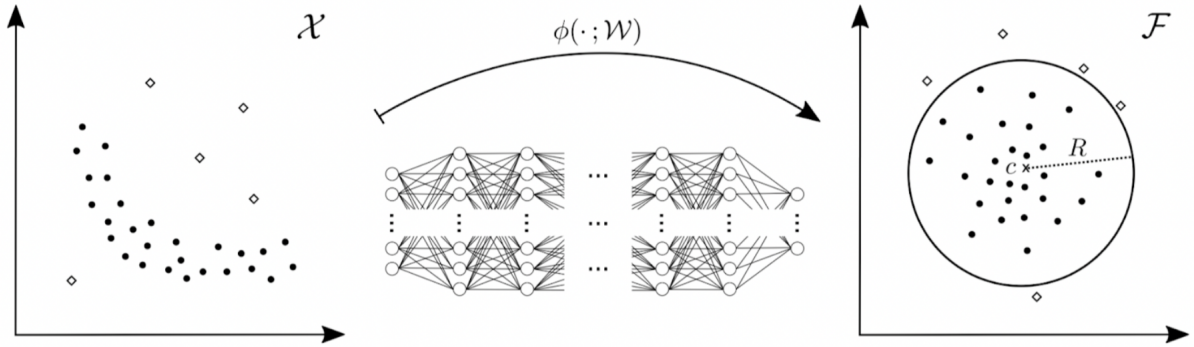

Figure 3. Deep SVDD learns a neural network transformation. [10]

each subspace mapped to the feature space as a sphere with center c_m. The probability of data point x after transformation m is parameterized by

$$P(T(x,m) \in X'_m) = \frac{1}{Z} e^{-(f(T(x,m))-c'_m)^2} \quad (2)$$

The classifier predicts transformation m given a transformed data point:

$$P(m'|T(x,m)) = \frac{e^{-||f(T(x,m))-c'_m||^2}}{\sum_{\tilde{m}} e^{-||f(T(x,m))-c_{\tilde{m}}||^2}} \quad (3)$$

In practice we obtained better results by training f using the center triplet loss (He et al., 2018). GOAD method learns supervised clusters with low intra-class variation, and high inter-class variation by optimizing the following loss function (where s is a margin regularizing the distance between blusters):

$$\text{L}=\sum_i max(||f(T(x_i,m)) - c_m||^2 + s - \min_{m' \neq m} ||f(T(x_i,m)) - c'_m||^2), 0) \quad (4)$$

To add a general prior for uncertainty far from the training set, we add a small regularizing constant $\epsilon$to the probability of each transformation. This ensures equal probabilities for uncertain regions:

$$\tilde{P}(m'|T(x,m)) = \frac{e^{-||f(T(x,m))-c'_m||^2} + \epsilon}{\sum_{\tilde{m}} e^{-||f(T(x,m))-c_{\tilde{m}}||^2} + M \cdot \epsilon} \quad (5)$$

By assuming independence between transformations, the probability that x is normal (i.e. $x \in X$) is the product of the probabilities that all transformed samples are in their

respective subspace. For log-probabilities the total score is given by:

$$Score(x) = -logP(x \in X) \tag{6}$$
$$= -\sum_m log\tilde{P}(T(x,m) \in X_m) \tag{7}$$
$$= -\sum_m log\tilde{P}(m|T(x,m)) \tag{8}$$

The score computes the degree of anomaly of each sample. Higher scores indicate a more anomalous sample.

3. Improvement to GOAD

The previous section, we explained that GOAD has one limitation, that is the data point with the same transformation will collapse to a single point. The feature within class will crumble after the projection to feature space. Information beside the transformation identity will be eliminated, which may remove important information and make it harder for the classifier to generalize to out of distribution data. In order to learn a more expressive low-dimensional structure from high-dimensional data that also discriminate between classes, we modify the GOAD method by a special objective function called the principle of Maximal Coding Rate Reduction (MCR2) [14].

Originally, GOAD use cross entropy loss as the classification objective function during training. For each transformed data $T(x,m)$, its true label will be represented as a one-hot vector $m_i \in R^k$. The mapping from the data $T(x,m)$ to its class label y can be denoted as $f(x,\theta) : x \to y$. Through back-propagation over the deep neural network parameters, we hope to minimize distance between the distribution of ground truth transformation label and predicted transformation probability.

$$\min CE(\theta, T(x,m), m) = -\mathcal{E}\{m, log[f(T(x,m),\theta)]\} \tag{9}$$

Despite cross entropy's effectiveness and enormous popularity, there are two serious limitations with this approach: 1) It aims only to predict the labels y. 2) The intermediate feature learned by the neural network may not capture the intrinsic structures of the data beyond making good classifications.

Here we changed the loss function using MCR2 such that it maximizes the coding rate difference between the whole dataset and the sum of each individual class. We address such limitations of current learning frameworks by reformulating the objective towards learning explicitly meaningful representations for the data x. According to the inventor of MCR2, the learned representation will have three properties:

1. Between-Class Discriminative: Different class or cluster's features should be highly uncorrelated and belong to different low-dimensional linear subspaces.

2. Within-Class Compressible: Features of within class data should be relatively correlated and they belong to a low-dimensional linear subspace.

3. Maximally Diverse Representation: The variance of features for each class/cluster should be as large as possible as long as they stay uncorrelated from the other classes.

Whether the given data X of a mixed distribution D can be effectively classified depends on how separable the component distributions D_j can be made. MCR2 assume that the distribution of each transformed data class has relatively low-dimensional intrinsic structures and has a support on a low-dimensional submanifold, say $\mathcal{M}_j$ with dimension d_j far less than D, and the distribution D of x is supported on the mixture of those submanifolds, $\mathcal{M} = \cup_{j=1}^k \mathcal{M}_j$, in the high-dimensional ambient space $\mathbb{R}^D$, as illustrated in Figure 4. With the manifold assumption in mind, we want to learn a mapping $z = f(x,\theta)$ that maps each of the submanifolds $\mathcal{M}_j \subset \mathbb{R}^D$ to a linear subspace $\mathcal{S}_j \in \mathbb{R}^D$

Something worth mention is that the intrinsic structures of each class may be low-dimensional, but they are not simply linear in their original representation x. Here the subspaces $\mathbb{S}j$ can be viewed as nonlinear generalized principal components for x (2005).

3.1. Our Evaluation Score

Since MCR2 learns subspace structured features, classification will be based on which subspace is the closest to a certain sample. More specifically, let features from a certain class i in the training set be Z_i. We first perform singular value decomposition to Z_i, $Z_i = USV$. Where S is diagonal matrix with singular values of Z_i. We normalize the singular values by it's maximum, and threshold by 0.5. Columns of V that correspond to the singular values higher than the threshold is kept as basis to the subspace corresponding to class i.

During testing, for each sample, projecting to each class subspace is computed, and the lowest cosine similarity between the sample and projections to all subspaces is used as the score of this sample. This is similar in spirit to nearest subspace classifier in [14].

4. Experiment

We perform experiments to validate the effectiveness of our MCR2 approach and compared it with GOAD's results on CIFAR10. We build on the standard protocol that trained on all training images of a single class and tested on all test images together and increased the batch size to 1000. Most of our results are semi-supervised as we assuming that no outlier exist in the training set. We used the same geometric transformations as [5]. In the previous experiment

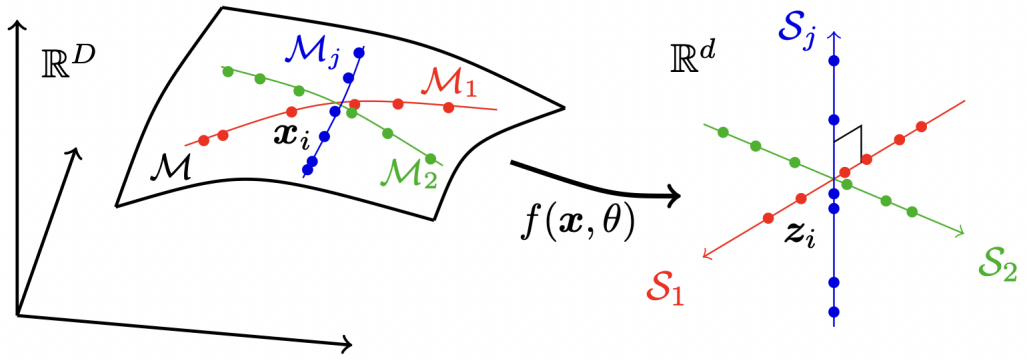

Figure 4. MCR2 loss function learns subspace structured features where samples from different classes form orthogonal subspaces, whose dimensions are maximized. [14]

of GOAD, the researchers used twenty different transformations, because the intuition behind their method is by training the classifier to distinguish between transformed images, it must learn salient geometrical features, some of which are likely to be unique to the single class. However, MCR2 aims to learn more feature within each class, so it cannot handle too many classes. Thus we selected eight transformations, including the combination of flipped or not flipped, and four rotations. We compared the AUROC score between GOAD and MCR2 using the same reduced set of transformations with epsilon =0.2 and gamma =0.8.

5. Results

The results of the experiments are demonstrated in table 1. We run their method using default widen factor 4 and compared with our MCR2 method using widen factor 4, 6, 8, and 10. We believed that the widen the neural network the more feature MCR2 will learn and hence higher AUROC. We selected the highest AURUC among 16 for each class It appears that our MCR2 results outperformed GOAD in class 7 even without wider neural network. Our model are close to the GOAD results especially in class1, class 2 and class 9.

6. Discussion

GOAD is a method for detecting anomalies for general data that does not require knowledge of the data domain. We thought since the representations learned using this MCR2 alone are significantly more robust to label corruptions in classification than those using cross-entropy, and can lead to state-of-the-art results in clustering mixed data from self-learned invariant features, it should increase the AUROC of outlier detection on CIFAR10. However our MCR2 method did not improve all the GOAD's AUROC on CIFAR10's different classes, but only one of them. This is because MCR2 method especially require wider neural network, maybe it will outperform the GOAD method with larger widen factor.

Table 1. Results of Experiments, reported are all AUROC (in %), W: widen factor of the neural network.

Classes	**GOAD W=4**	**W=6**	**W=8**	**W=10**	
1	**75.4**	73.0	69.1	69.6	66.9
2	**94.0**	91.2	92.5	90.7	91.5
3	**82.6**	57.1	63.8	65.3	65.1
4	**76.8**	58.4	59.9	62.2	60.3
5	**80.7**	68.8	72.0	71.3	69.5
6	**86.9**	68.7	73.5	73.7	75.9
7	69.4	**74.7**	**73.5**	**78.8**	**75.1**
8	**94.2**	86.3	87.1	88.2	88.3
9	**92.7**	87.2	87.5	87.2	88.1
10	**90.2**	82.6	83.9	85.6	85.5

In a broader context, the interesting question will be to what extent MCR2 can facilitate the learning of semantic representations? If it is only outperform one class for example frog here, will we confidently say it learn more important feature about frog than GOAD such that it can easily generalize to other data class for example to toad samples? There is some evidence from previous research showed that self-supervised learning improved the detection of semantic anomalies and thus exhibits inductive biases towards semantic representations [1]. On the other hand, there also exists evidence showing that self-supervision method predominantly improves learning of effective feature representations for low-level statistics [2] so no semantic representation are learned. Hence, this research question remains to be a mystery, but there are many improve space for many domains where large amounts of unlabeled data are avail-

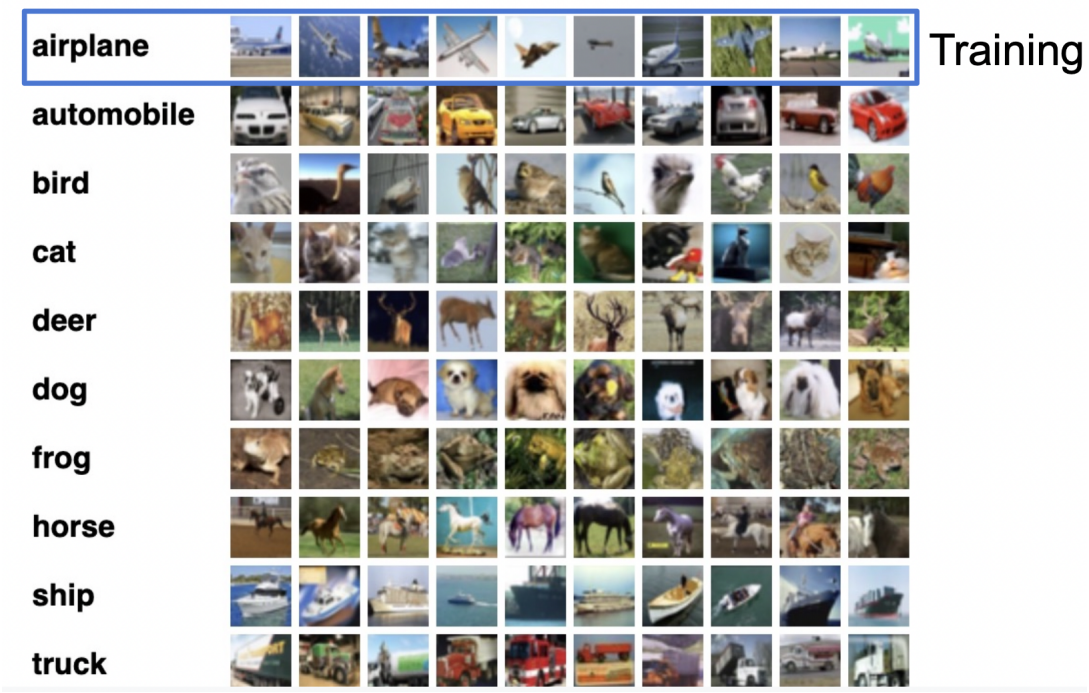

Figure 5. During training, only samples from a single normal class is used.

Figure 6. During testing, samples from all classes are used. The class used during training is considered to be normal class and the rest is considered to be outliers.

able. For example the following:

Intrusion detection: the process of collecting and analyzing information from several key points in a computer network or system to find out whether there are behaviors violating security policies and signs of attacks in the network or system and make appropriate responses. The two most

common intrusion detection systems include host-based intrusion detection systems (HIDS) and network intrusion detection systems (NIDS).

Fraud detection: Monitoring systems that can identify faults when they occur and accurately point out the type and location of faults. Major application areas include bank fraud, mobile cellular network failure, insurance fraud, and health care fraud.

Medical Anomaly Detection: Medical images such as X-ray, MRI, AND CT can be used to detect diseases or quantify anomalies. Time series signals such as EEG and ECG can also be used to detect diseases or give early warning of anomalies.

Internet of Things (IoT) Big Data Anomaly Detection: Detects abnormality from information collected from many small devices.

7. Conclusion

In this paper, we presented a method that replace cross entropy loss with MCR2 that partially improve the outlier detection method GOAD. It is only tested it on CIFAR10, so in the future maybe we can test its generalization properties on other tabular datasets like Arrhythmia and Thyroid for Medical Anomaly Detection .

The recent progress in outlier detection research has also raised more fundamental questions. These include open questions about the out-of-distribution generalization properties of various methods, because we are not sure why changing the random seed would cause so many variance in AUROC. Also, we are not sure about the definition of anomalies in high-dimensional spaces so we can only apply semi-superivised our training.

In general, the push towards MCR2 presents new opportunities to interpret and analyze the outlier detection problem from different theoretical angles and call for more interoperability in deep learning.

8. Accesibility

Our code can be found at GitHub https://github.com/SixuanChen/SYDE675_Outlier_Detection

References

[1] Faruk Ahmed and Aaron Courville. Detecting semantic anomalies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3154–3162, 2020. 5

[2] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. *arXiv preprint arXiv:1904.13132*, 2019. 5

[3] Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. *arXiv preprint arXiv:2005.02359*, 2020. 2

[4] Yunqiang Chen, Xiang Sean Zhou, and Thomas S Huang. One-class svm for learning in image retrieval. In *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, volume 1, pages 34–37. IEEE, 2001. 2

[5] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. *Advances in neural information processing systems*, 31, 2018. 2, 4

[6] Simon Günter, Nicol Schraudolph, S Vishwanathan, et al. Fast iterative kernel principal component analysis. 2007. 2

[7] Peter J Huber. *Robust statistics*, volume 523. John Wiley & Sons, 2004. 2

[8] JooSeuk Kim and Clayton D Scott. Robust kernel density estimation. *The Journal of Machine Learning Research*, 13(1):2529–2565, 2012. 2

[9] 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. 1, 2, 3

[10] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pages 4393–4402. PMLR, 2018. 2, 3

[11] David MJ Tax and Klaus-R Müller. Feature extraction for one-class classification. In *Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003*, pages 342–349. Springer, 2003. 2

[12] Graham Williams, Rohan Baxter, Hongxing He, Simon Hawkins, and Lifang Gu. A comparative study of rnn for outlier detection in data mining. In *2002 IEEE International Conference on Data Mining, 2002. Proceedings.*, pages 709–712. IEEE, 2002. 2

[13] Liang Xiong, Barnabás Póczos, and Jeff Schneider. Group anomaly detection using flexible genre models. *Advances in neural information processing systems*, 24, 2011. 2

[14] Yaodong Yu, Kwan Ho Ryan Chan, Chong You, Chaobing Song, and Yi Ma. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. *Advances in Neural Information Processing Systems*, 33:9422–9434, 2020. 4, 5

[15] Arthur Zimek, Erich Schubert, and Hans-Peter Kriegel. A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 5(5):363–387, 2012. 2