A Unifying Review of Deep and Shallow Anomaly Detection

Lukas Ruff, Jacob R. Kauffmann, Robert A. Vandermeulen, Grégoire Montavon, Wojciech Samek, *Member, IEEE,* Marius Kloft*, Thomas G. Dietterich*, Klaus-Robert Müller* *Member, IEEE,*

Abstract—**Deep learning approaches to anomaly detection have recently improved the state of the art in detection performance on complex datasets such as large collections of images or text. These results have sparked a renewed interest in the anomaly detection problem and led to the introduction of a great variety of new methods. With the emergence of numerous such methods, including approaches based on generative models, one-class classification, and reconstruction, there is a growing need to bring methods of this field into a systematic and unified perspective. In this review we aim to identify the common underlying principles as well as the assumptions that are often made implicitly by various methods. In particular, we draw connections between classic 'shallow' and novel deep approaches and show how this relation might cross-fertilize or extend both directions. We further provide an empirical assessment of major existing methods that is enriched by the use of recent explainability techniques, and present specific worked-through examples together with practical advice. Finally, we outline critical open challenges and identify specific paths for future research in anomaly detection.**

Index Terms—**Anomaly detection, deep anomaly detection, deep learning, interpretability, kernel methods, neural networks, novelty detection, one-class classification, out-of-distribution detection, outlier detection, unsupervised learning**

I. Introduction

An *anomaly* is an observation that deviates considerably from some concept of normality. Also known as *outlier* or *novelty*, such an observation may be termed unusual, irregular, atypical, inconsistent, unexpected, rare, erroneous, faulty, fraudulent, malicious, unnatural, or simply strange — depending on the situation. *Anomaly detection* (or *outlier detection* or *novelty detection*) is the research area that studies the detection of such anomalous observations through methods, models, and algorithms based on data. Classic approaches to anomaly detection include Principal Component Analysis (PCA) [1]–[5], the One-Class Support Vector Machine (OC-SVM) [6], Support Vector Data Description (SVDD) [7], nearest neighbor algorithms [8]–[10], and Kernel Density Estimation (KDE) [11], [12].

What the above methods have in common is that they are all *unsupervised*, which constitutes the predominant approach to anomaly detection. This is because labeled anomalous data is often non-existent. When available, it is usually insufficient to represent the diversity of all potential anomalies. This prohibits or renders a supervised approach ineffective. Instead, a central idea in anomaly detection is to learn a model of normality from normal data in an unsupervised manner, so that anomalies become detectable through deviations from such a model.

The study of anomaly detection has a long history and spans multiple disciplines including engineering, machine learning, data mining, and statistics. While the first formal definitions of so-called 'discordant observations' date back to the 19th century [13], the problem of anomaly detection has likely been studied informally even earlier, since anomalies are phenomena that naturally occur in diverse academic disciplines such as medicine and the natural sciences. Anomalous data may be useless, for example when caused by measurement errors, or may be extremely informative and hold the key to new insights, such as very long surviving cancer patients. Kuhn [14] claims that persistent anomalies drive scientific revolutions (cf., section VI 'Anomaly and the Emergence of Scientific Discoveries' in [14]).

Anomaly detection today has numerous applications across a variety of domains. Examples include intrusion detection in cybersecurity [15]–[20], fraud detection in finance, insurance, healthcare, and telecommunication [21]–[27], industrial fault and damage detection [28]–[36], the monitoring of infrastructure [37], [38] and stock markets [39], [40], acoustic novelty detection [41]–[45], medical diagnosis [46]–[61] and disease outbreak detection [47], [62], event detection in the earth sciences [63]–[68], and scientific discovery in chemistry [69], [70], bioinformatics [71], genetics [72], [73], physics [74], [75], and astronomy [76]–[79]. The data available in these domains is continually growing in size. It is also growing to include complex data types such as images, video, audio, text, graphs, multivariate time series, and biological sequences, among others. For applications to be successful on such complex and high-dimensional data, a meaningful representation of the data is crucial [80].

Deep learning [81]–[83] follows the idea of *learning* effective representations from the data itself by training flexible, multi-layered ('deep') neural networks and has greatly improved the state of the art in many applications that involve complex data types. Deep neural networks provide

* Corresponding authors: M. Kloft, T. G. Dietterich, and K.-R. Müller.

L. Ruff, J. R. Kauffmann, R. A. Vandermeulen, and G. Montavon are with the ML group, Technische Universität Berlin, 10587 Berlin, Germany.

W. Samek is with the Fraunhofer Heinrich Hertz Institute, 10587 Berlin, Germany.

M. Kloft is with the Dept. of Computer Science, Technische Universität Kaiserslautern, 67653 Kaiserslautern, Germany (e-mail: kloft@cs.uni-kl.de).

T. G. Dietterich is with the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR 97331, USA (e-mail: tgd@cs.orst.edu).

K.-R. Müller is with Google Research, Brain team, Berlin, Germany and the ML group, Technische Universität Berlin, 10587 Berlin, Germany, and also with the Dept. of Artificial Intelligence, Korea University, Seoul 136-713, South Korea and Max Planck Institute for Informatics, 66123 Saarbrücken, Germany (e-mail: klaus-robert.mueller@tu-berlin.de).

the most successful solutions for many tasks in domains such as computer vision [84]–[93], speech recognition [94]–[103], or natural language processing [104]–[113], and have contributed to the sciences [114]–[123]. Methods based on deep neural networks are able to exploit the hierarchical or latent structure that is often inherent to data through their multi-layered, distributed feature representations. Advances in parallel computation, stochastic gradient descent optimization, and automated differentiation make it possible to apply deep learning at scale using large datasets.

Recently, there has been a rapidly growing interest in developing deep learning approaches for anomaly detection. This is motivated by a lack of effective methods for anomaly detection tasks which involve complex data, for instance cancer detection from multi-gigapixel whole-slide images in histopathology. As in other adoptions of deep learning, the ambition of *deep anomaly detection* is to mitigate the burden of manual feature engineering and to enable effective as well as scalable solutions. However, unlike supervised deep learning, it is less clear what characterizes an effective learning objective for anomaly detection and which signals should be used for learning a representation due to the mostly unsupervised nature of the problem.

The major approaches to deep anomaly detection include deep autoencoder variants [44], [52], [55], [124]–[134], deep one-class classification [135]–[144], methods based on deep generative models such as Generative Adversarial Networks (GANs) [51], [57], [145]–[152], as well as recent self-supervised methods [153]–[156]. In comparison to traditional anomaly detection methods, where a feature representation is fixed a priori (e.g., via a kernel feature map), these approaches aim to learn a feature map of the data $\phi_\omega : \boldsymbol{x} \mapsto \phi_\omega(\boldsymbol{x})$, a deep neural network parameterized with weights ω, as part of their learning objective.

Due to the long history and diversity of anomaly detection, there exists a wealth of review and survey literature [157]–[176] as well as books [177]–[179] on the topic. Some very recent surveys focus specifically on deep anomaly detection [180]–[182], but these works exclusively consider the deep learning approaches themselves. An integrated treatment of deep learning methods in the overall context of anomaly detection research — in particular its kernel-based learning part [6], [183], [184] — is still missing.

In this review paper, our aim is to exactly fill this gap by presenting a unifying view that connects traditional shallow and novel deep learning approaches. We will summarize recent exciting developments, present different classes of anomaly detection methods, provide theoretical insights, and highlight the current best practices when applying anomaly detection. Note finally, that we do not attempt an encyclopedic treatment of all available anomaly detection literature; rather, we present a slightly biased point of view illustrating the main ideas (and in doing so we often draw from the work of the authors) and providing ample reference to related work for further reading.

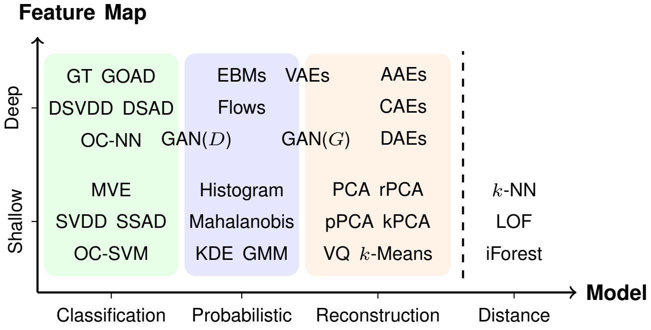

Fig. 1. Anomaly detection approaches placed in the plane spanned by two major components of our unifying view. Besides *Model* and *Feature Map*, we identify *Loss*, *Regularization*, and *Inference Mode* as other important modeling components of the anomaly detection problem.

II. An Introduction to Anomaly Detection

A. Why Should We Care About Anomaly Detection?

Anomaly detection is part of our daily lives. Operating mostly unnoticed, anomaly detection algorithms are continuously monitoring our credit card payments, our login behaviors, and companies' communication networks. If they detect an abnormally expensive purchase made on our credit card, several unsuccessful login attempts made from an alien device in a distant country, or unusual ftp requests made to our computer, they will issue an alarm. While warnings such as "someone is trying to login to your account" can be annoying when you are on a business trip abroad and just want to check your e-mails from the hotel computer, the ability to detect such anomalous patterns is vital for a large number of today's applications and services and even small improvements in anomaly detection can lead to immense monetary savings[1].

In addition, the ability to detect anomalies is also considered an important ingredient in ensuring fail-safe and robust design of deep learning-based systems, e.g. in medical applications or autonomous driving. Various international standardization initiatives have been launched towards this goal (e.g., ITU/WHO FG-AI4H, ISO/IEC CD TR 24029-1, or IEEE P7009).

Despite its importance, discovering a reliable distinction between 'normal' and 'anomalous' events is a challenging task. First, the variability within the normal data can be very large, resulting in misclassifying normal samples as being anomalous (type I error) or not identifying the anomalous ones (type II error). Especially in biological or biomedical datasets, the variability between the normal data (e.g., person-to-person variability) is often as large or even larger than the distance to anomalous samples (e.g., patients). Preprocessing, normalization, and feature selection are potential means to reduce this variability and improve detectability. Second, anomalous events are often very rare, which results in highly imbalanced training datasets. Even worse, in most cases the datasets are unlabeled, so that it remains unclear which data points are regarded anomalies and why. Hence, the anomaly

[1] In 2019, UK's online banking fraud has been estimated to be 111.8 million GBP (source: https://www.statista.com/).

detection problem reduces to an unsupervised learning task with the goal to learn a valid model of the majority of data points. Finally, anomalies themselves can be very diverse, so that it becomes difficult to learn a complete model for them. Also here the solution is to learn a model for the normal samples and treat deviations from it as anomalies. However, this approach can be problematic if the distribution of the (normal) data changes (non-stationarity), either intrinsically or due to environmental changes (e.g., lighting conditions, recording devices from different manufacturers, etc.).

As exemplified and discussed above, we note that anomaly detection has a broad practical relevance and impact. Moreover, (accidentally) detecting the *unknown unknowns* [185] has always been a strong driving force in the sciences. If applied to these disciplines, anomaly detection can help us to identify new, previously unknown patterns in data, which can lead to novel scientific insights and hypotheses.

B. A Formal Definition of the Problem

In the following, we formally introduce the anomaly detection problem. We first define in probabilistic terms what an anomaly is, explain what types of anomalies there are, and delineate the subtle differences between an anomaly, an outlier, and a novelty. Finally we present a fundamental principle in anomaly detection — the so-called concentration assumption — and give a theoretical problem formulation that corresponds to density level set estimation.

1) What is an Anomaly?: We opened this review with the following definition:

> *An anomaly is an observation that deviates considerably from some concept of normality.*

To formalize this definition, we here specify two aspects more precisely: a 'concept of normality' and what 'deviates considerably' signifies. Following many previous authors [13], [177], [186]–[188], we rely on probability theory.

Let $\mathcal{X} \subseteq \mathbb{R}^D$ be the data space given by some task or application. We define a concept of normality as the distribution $\mathbb{P}^+$ on $\mathcal{X}$ that captures the *ground-truth law of normal behavior* in a given task or application. An observation that deviates considerably from such a law of normality —*an anomaly*— is then a data point $\boldsymbol{x} \in \mathcal{X}$ (or set of points) that lies in a low probability region under $\mathbb{P}^+$. Assuming that $\mathbb{P}^+$ has a corresponding probability density function (pdf) $p^+(\boldsymbol{x})$, we can define the *set of anomalies* as

$$\mathcal{A} = \{\boldsymbol{x} \in \mathcal{X} \mid p^+(\boldsymbol{x}) \leq \tau\}, \quad \tau \geq 0, \tag{1}$$

where $\tau \geq 0$ is some threshold such that the probability of $\mathcal{A}$ under $\mathbb{P}^+$ is 'sufficiently small' which we will discuss in further detail below.

2) Types of Anomalies: Various types of anomalies have been identified in the literature [161], [179]. These include point anomalies, conditional or contextual anomalies [169], [171], [190]–[194], and group or collective anomalies [146], [192], [195]–[198]. We extend these three established types by further adding low-level, sensory anomalies and high-level, semantic anomalies [199], a distinction that is particularly relevant for choosing between deep and shallow feature maps.

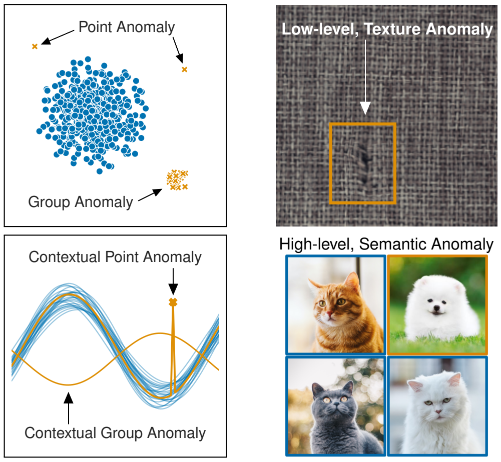

Fig. 2. An illustration of the types of anomalies: A *point anomaly* is a single anomalous point. A *contextual point anomaly* occurs if a point deviates in its local context, here a spike in an otherwise normal time series. A *group anomaly* can be a cluster of anomalies or some series of related points that is anomalous under the joint series distribution (*contextual group anomaly*). Note that both contextual anomalies have values that fall into the global (time-integrated) range of normal values. A low-level, sensory anomaly deviates in the low-level features, here a cut in the fabric texture of a carpet [189]. A semantic anomaly deviates in high-level factors of variation or semantic concepts, here a dog among the normal class of cats. Note that the white cat is more similar to the dog than to the other cats in low-level pixel space.

A *point anomaly* is an individual anomalous data point $\boldsymbol{x} \in \mathcal{A}$, for example an illegal transaction in fraud detection or an image of a damaged product in manufacturing. This is arguably the most commonly studied type in anomaly detection research.

A *conditional* or *contextual anomaly* is a data instance that is anomalous in a specific context such as time, space, or the connections in a graph. A price of $1 per Apple Inc. stock might have been normal before 1997, but as of today (2020) would be an anomaly. A mean daily temperature below freezing point would be an anomaly in the Amazon rainforest, but not in the Antarctic desert. For this anomaly type, the normal law $\mathbb{P}^+$ is more precisely a conditional distribution $\mathbb{P}^+ \equiv \mathbb{P}^+_{X|T}$ with conditional pdf $p^+(\boldsymbol{x} \mid t)$ that depends on some contextual variable T. Time series anomalies [169], [194], [200]–[203] are the most prominent example of contextual anomalies. Other examples include spatial [204], [205], spatio-temporal [191], or graph-based [171], [206], [207] anomalies.

A *group* or *collective anomaly* is a *set* of related or dependent points $\{\boldsymbol{x}_j \in \mathcal{X} \mid j \in J\}$ that is anomalous, where $J \subseteq \mathbb{N}$ is an index set that captures some relation or dependency. A cluster of anomalies such as similar or related network attacks in cybersecurity form a collective anomaly for instance [18], [207], [208]. Often, collective anomalies are also contextual such as anomalous time (sub-)series or biological (sub-)sequences, for example, some series or sequence $\{\boldsymbol{x}_t, \dots, \boldsymbol{x}_{t+s-1}\}$ of length $s \in \mathbb{N}$. It is important to note that although each individual point $\boldsymbol{x}_j$ in such a series or sequence might be normal under the time-integrated marginal $p^+(\boldsymbol{x}) = \int p^+(\boldsymbol{x}, t)\,\mathrm{d}t$ or under the sequence-integrated, time-conditional marginal $p^{+}(\boldsymbol{x}\,|\,t)$ given by

$$\int\cdots\int p^{+}(\boldsymbol{x}_t,\dots,\boldsymbol{x}_{t+s-1}\,|\,t)\,\mathrm{d}\boldsymbol{x}_t\cdots\mathrm{d}\boldsymbol{x}_{j-1}\,\mathrm{d}\boldsymbol{x}_{j+1}\cdots\mathrm{d}\boldsymbol{x}_{t+s-1}$$

the full series or sequence $\{\boldsymbol{x}_t,\dots,\boldsymbol{x}_{t+s-1}\}$ can be anomalous under the *joint* conditional density $p^{+}(\boldsymbol{x}_t,\dots,\boldsymbol{x}_{t+s-1}\,|\,t)$, which properly describes the distribution of the collective series or sequences.

In the wake of deep learning, the distinction between *low-level, sensory anomalies* and *high-level, semantic anomalies* [199] has become important. Low and high here refer to the level in the feature hierarchy of some hierarchical distribution, for instance, the hierarchy from pixel-level features such as edges and textures to high-level objects and scenes in images or the hierarchy from individual characters and words to semantic concepts and topics in texts. It is commonly assumed that data with such a hierarchical structure is generated from some semantic latent variables Z and Y that describe higher-level factors of variation Z (e.g., the shape, size or orientation of an object) and concepts Y (e.g., the object class identity) [80], [209]. We can express this via a normal law with conditional pdf $p^{+}(\boldsymbol{x}\,|\,\boldsymbol{z},y)$, where we usually assume Z to be continuous and Y to be discrete. Low-level anomalies can for example be texture defects or artifacts in images, or character typos in words. In comparison, semantic anomalies can be images of objects from non-normal classes [199], for instance, or misposted reviews and news articles [139]. Note that semantic anomalies can be very close to normal instances in the raw feature space $\mathcal{X}$. For example a dog with a fur texture and color similar to that of some cat can be more similar in raw pixel space than various cat breeds among themselves (cf., Fig. 2). Similarly, low-level background statistics can also result in a high similarity in raw pixel space even when objects in the foreground are completely different [199]. Detecting semantic anomalies is thus innately tied to finding a semantic feature representation (e.g., extracting the semantic features of cats such as whiskers, slit pupils, triangular snout, etc.), which is an inherently difficult task in an unsupervised setting [209].

3) Anomaly, Outlier, or Novelty?: Some works make a more subtle distinction between what is an anomaly, an outlier, or a novelty. While all three refer to instances from low probability regions under $\mathbb{P}^{+}$ (i.e., are elements of $\mathcal{A}$), an anomaly is often characterized as being an instance from a distinct distribution other than $\mathbb{P}^{+}$ (e.g., when anomaly points are generated by a different process than the normal points), an outlier as being a rare or low-probability instance from $\mathbb{P}^{+}$, and a novelty as being an instance from some new region or mode of an evolving, non-stationary $\mathbb{P}^{+}$. Under the distribution $\mathbb{P}^{+}$ of cats, for instance, a dog would be an anomaly, a rare breed of cats such as the LaPerm would be an outlier, and a new breed of cats would be a novelty. Such a distinction between anomaly, outlier, and novelty may reflect slightly different objectives in an application: whereas anomalies are often the data points of interest (e.g., a long-term survivor of a disease), outliers are frequently regarded as 'noise' or 'measurement error' that should be removed in a data preprocessing step ('outlier removal'), and novelties are new observations that require models to be updated to the 'new normal'. The methods for detecting points from low probability regions, whether termed anomaly, outlier, or novelty, are usually the same however. For this reason, we do not make such a distinction here and refer to any instance $\boldsymbol{x}\in\mathcal{A}$ as an anomaly.

4) The Concentration Assumption: In general, the data space $\mathcal{X}\subseteq\mathbb{R}^D$ can be unbounded. A fundamental assumption in anomaly detection however is that the region where the normal data lives can be bounded. That is, that there exists some threshold $\tau\geq 0$ such that

$$\mathcal{X}\setminus\mathcal{A}=\{\boldsymbol{x}\in\mathcal{X}\mid p^{+}(\boldsymbol{x})>\tau\} \qquad (2)$$

is non-empty and small (typically in the Lebesgue-measure sense). This is known as the so-called *concentration* or *cluster assumption* [210]–[212]. Note that the concentration assumption does not imply that the full support $\operatorname{supp}(p^{+})=\{\boldsymbol{x}\in\mathcal{X}\,|\,p^{+}(\boldsymbol{x})>0\}$ of the normal law $\mathbb{P}^{+}$ must be bounded; only that some high-density subset of the support is bounded. A standard univariate Gaussian is supported on the full real axis, for example, but approximately 95% of the most likely region is covered by the bounded interval $[-1.96, 1.96]$. In contrast, the set of anomalies $\mathcal{A}$ need not be concentrated and can be unbounded.

5) Density Level Set Estimation: A law of normality $\mathbb{P}^{+}$ is only known in a few application settings, such as for certain laws of physics. Sometimes a concept of normality might also be user-specified (as in juridical laws). In most cases, however, the ground-truth law of normality $\mathbb{P}^{+}$ is unknown because the underlying process is too complex. For this reason, we must estimate $\mathbb{P}^{+}$ from data.

Let $\mathbb{P}$ be the *ground-truth data-generating distribution* on data space $\mathcal{X}\subseteq\mathbb{R}^D$ with corresponding density $p(\boldsymbol{x})$. For now, we assume that this data-generating distribution exactly matches the normal data distribution, i.e. $\mathbb{P}\equiv\mathbb{P}^{+}$ and $p\equiv p^{+}$. This assumption is often invalid in practice, of course, as the data-generating process might be subject to noise or contamination as we will discuss in the next section.

Given data points $\boldsymbol{x}_1,\dots,\boldsymbol{x}_n\in\mathcal{X}$ generated by $\mathbb{P}$ (usually assumed to be drawn from i.i.d. random variables following $\mathbb{P}$), the goal of anomaly detection is to learn a model that allows us to predict whether a new test instance $\tilde{\boldsymbol{x}}\in\mathcal{X}$ (or set of test instances) is an anomaly or not, i.e. whether $\tilde{\boldsymbol{x}}\in\mathcal{A}$. Thus the anomaly detection objective is to (explicitly or implicitly) estimate the low-density regions (or equivalently high-density regions) in data space $\mathcal{X}$ under the normal law $\mathbb{P}^{+}$. We can formally express this objective as the problem of *density level set estimation* [213]–[216] which is an instance of *minimum volume set estimation* [217]–[219] for the special case of density-based sets. The density level set of $\mathbb{P}$ for some threshold $\tau\geq 0$ is given by $C=\{\boldsymbol{x}\in\mathcal{X}\,|\,p(\boldsymbol{x})>\tau\}$. For some fixed level $\alpha\in[0,1]$, the *α-density level set* C_α of distribution $\mathbb{P}$ is then defined as the smallest density level set C that has a probability of at least $1-\alpha$ under $\mathbb{P}$, i.e.

$$\begin{aligned} C_\alpha &= \underset{C}{\operatorname{arginf}}\,\{\lambda(C)\,|\,\mathbb{P}(C)\geq 1-\alpha\} \\ &= \{\boldsymbol{x}\in\mathcal{X}\,|\,p(\boldsymbol{x})>\tau_\alpha\} \end{aligned} \qquad (3)$$

where $\tau_\alpha\geq 0$ denotes the corresponding threshold and λ typically is the Lebesgue measure, which is the standard

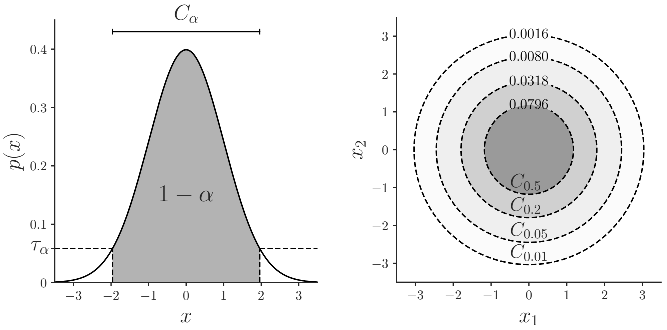

Fig. 3. An illustration of the density level sets of a univariate (left) and bivariate (right) standard Gaussian distribution.

measure of volume in Euclidean space. The extreme cases of $\alpha = 0$ and $\alpha \to 1$ result in the full support $C_0 = \{\boldsymbol{x} \in \mathcal{X} \,|\, p(\boldsymbol{x}) > 0\} = \operatorname{supp}(p)$ and the most likely modes $\operatorname{argmax}_{\boldsymbol{x}}\, p(\boldsymbol{x})$ of $\mathbb{P}$ respectively. If the aforementioned concentration assumption holds, there always exists some level α such that a corresponding level set C_α exists and can be bounded. Fig. 3 illustrates some density level sets for the case that $\mathbb{P}$ is the familiar standard Gaussian distribution. Given a level set C_α, we can define a corresponding threshold anomaly detector $c_\alpha : \mathcal{X} \to \{\pm 1\}$ as

$$c_\alpha(\boldsymbol{x}) = \begin{cases} +1 & \text{if } \boldsymbol{x} \in C_\alpha, \\ -1 & \text{if } \boldsymbol{x} \notin C_\alpha. \end{cases} \tag{4}$$

6) Density Estimation for Level Set Estimation: An obvious approach to density level set estimation is through density estimation. Given some estimated density model $\hat{p}(\boldsymbol{x}) = \hat{p}(\boldsymbol{x};\, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) \approx p(\boldsymbol{x})$ and some target level $\alpha \in [0, 1]$, one can estimate a corresponding threshold $\hat{\tau}_\alpha$ via the empirical p-value function:

$$\hat{\tau}_\alpha = \inf_\tau \big\{ \tau \geq 0 \;\big|\; \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{[0, \hat{p}(\boldsymbol{x}_i))}(\tau) \geq 1 - \alpha \big\}, \tag{5}$$

where $\mathbb{1}_A(\cdot)$ denotes the indicator function for some set A. Using $\hat{\tau}_\alpha$ and $\hat{p}(\boldsymbol{x})$ in (3) yields the plug-in density level set estimator $\hat{C}_\alpha$ which in turn can be used in (4) to obtain the plug-in threshold detector $\hat{c}_\alpha(\boldsymbol{x})$. Note however that density estimation is generally the most costly approach to density level set estimation (in terms of samples required), since estimating the full density is equivalent to first estimating the *entire family* of level sets $\{C_\alpha : \alpha \in [0, 1]\}$ from which the desired level set for some fixed $\alpha \in [0, 1]$ is then selected [220], [221]. If there are insufficient samples, this density estimate can be biased. This has also motivated the development of one-class classification methods that aim to estimate subfamilies [221] or single level sets [6], [7], [183], [222] directly, which we will explain in section IV in more detail.

7) Threshold vs. Score: The previous approach to level set estimation through density estimation is more costly, yet generally results in a more informative model that can rank inliers and anomalies according to their estimated density. In comparison, a pure threshold detector as in (4) only yields a binary prediction. Menon and Williamson [223] propose a compromise by learning a density outside the level set boundary. Many anomaly detection methods also target some strictly increasing transformation $T : [0, \infty) \to \mathbb{R}$ of the density for estimating a model (e.g., log-likelihood instead of likelihood). The resulting target $T(p(\boldsymbol{x}))$ is often no longer a proper density but still preserves the density order [224], [225]. An *anomaly score* $s : \mathcal{X} \to \mathbb{R}$ can then be defined by using an additional order-reversing transformation, for example $s(\boldsymbol{x}) = -T(p(\boldsymbol{x}))$ (e.g., negative log-likelihood), so that high scores reflect low density values and vice versa. Having such a score that indicates the 'degree of anomalousness' is important in many anomaly detection applications. As for the density in (5), of course, we can always derive a threshold from the empirical distribution of anomaly scores if needed.

8) Selecting a Level α: As we will show, there are many degrees of freedom when attacking the anomaly detection problem outlined in this section which inevitably requires making various modeling assumptions and choices. Setting the level α is one of these choices and depends on the specific application. As the value of α increases, the anomaly detector focuses only on the most likely regions of $\mathbb{P}$. Such a detector can be desirable in applications where missed anomalies are costly (e.g., in medical diagnosis or fraud detection). On the other hand, a large α will result in high false alarm rates, which can be undesirable in online settings where lots of data is generated (e.g., in monitoring tasks). We will provide practical guidelines for selecting α in Section VIII. Choosing α also involves further assumptions about the data-generating process $\mathbb{P}$, which we have assumed here to match the normal data distribution $\mathbb{P}^+$. In the next section, we discuss the data settings that can occur in anomaly detection that may alter this assumption.

C. Dataset Settings and Data Properties

The dataset settings and data properties that occur in real-world anomaly detection problems can be diverse. We here characterize these settings which may range from the most common unsupervised to a semi-supervised as well as a supervised setting and list further data properties that are relevant for modeling an anomaly detection problem. But before we elaborate on these, we first observe that the assumptions made about the distribution of anomalies (often implicitly) are also crucial to the problem.

1) A Distribution of Anomalies?: Let $\mathbb{P}^-$ denote the *ground-truth anomaly distribution* also on $\mathcal{X} \subseteq \mathbb{R}^D$. As mentioned above, the common concentration assumption implies that some high-density regions of the normal data distribution are concentrated whereas anomalies are assumed to be *not* concentrated [210], [211]. This assumption may be modeled by an anomaly distribution $\mathbb{P}^-$ that follows a uniform distribution over the (bounded[2]) data space $\mathcal{X}$ [183]. Some well-known unsupervised methods such as KDE [12] or the OC-SVM [6], for example, implicitly make this assumption that $\mathbb{P}^-$ follows a uniform which can be interpreted as a default uninformative

[2] Strictly speaking, we here assume that there always exists some data-enclosing hypercube of numerically meaningful values such that the data space $\mathcal{X}$ is bounded and the uniform distribution is well-defined.

prior on the anomalous distribution [211]. This prior assumes that there are no anomalous modes and that anomalies are equally likely to occur over the valid data space $\mathcal{X}$. Semi-supervised or supervised anomaly detection approaches often depart from this uninformed prior and try to make a more informed a-priori assumption about the anomalous distribution $\mathbb{P}^-$ [211]. If faithful to $\mathbb{P}^-$, such a model based on a more informed anomaly prior can achieve better detection performance. Modeling anomalous modes also can be beneficial in certain applications, for example, for typical failure modes in industrial machines or known disorders in medical diagnosis. We remark that these prior assumptions about the anomaly distribution $\mathbb{P}^-$ are often expressed only implicitly in the literature, though such assumptions are critical to an anomaly detection model.

2) The Unsupervised Setting: The unsupervised anomaly detection setting is the case in which *only unlabeled data*

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X} \tag{6}$$

is available for training a model. This setting is arguably the most common setting in anomaly detection [159], [161], [165], [168]. We will usually assume that the data points have been drawn in an i.i.d. fashion from the data-generating distribution $\mathbb{P}$. For simplicity, we have so far assumed that the data-generating distribution is the same as the normal data distribution $\mathbb{P} \equiv \mathbb{P}^+$. This is often summarized by the statement that the training data is 'clean'. In practice, however, the data-generating distribution $\mathbb{P}$ might be subject to *noise* and *contamination* [183].

Noise, in the classical sense, is some inherent source of randomness ε that is added to the actual signal in the data-generating process, that is, samples from $\mathbb{P}$ have added noise $\boldsymbol{x}+\varepsilon$ where $\boldsymbol{x} \sim \mathbb{P}^+$. Noise might be present due to irreducible measurement uncertainties in an application, for example. The greater the noise, the harder it becomes to accurately estimate the ground-truth level sets of $\mathbb{P}^+$, since characteristic normal features get obfuscated [165]. This is because added noise generally expands the regions covered by the observed data in input space $\mathcal{X}$. A standard assumption about noise is that it is symmetric and unbiased $\mathbb{E}[\varepsilon] = 0$.

In addition to noise, the *contamination* or *pollution* of the unlabeled data with undetected anomalies is another critical source of disturbance. For instance, some unnoticed anomalous errors of a machine might have already occurred during the data collection process. In this case the data-generating distribution $\mathbb{P}$ is a mixture of the normal data and the anomaly distribution, i.e., $\mathbb{P} \equiv (1-\eta)\,\mathbb{P}^+ + \eta\,\mathbb{P}^-$ for some contamination or pollution rate $\eta \in (0, 1)$. The greater the contamination, the more likely the normal data decision boundary will be damaged by including the anomalous points.

In summary, a more general and realistic assumption is that samples from the the data-generating distribution $\mathbb{P}$ have the form of $\boldsymbol{x} + \varepsilon$ where $\boldsymbol{x} \sim (1 - \eta)\,\mathbb{P}^+ + \eta\,\mathbb{P}^-$ and ε is random noise. Assumptions on both, the noise distribution ε and contamination rate η, are crucial for modeling a specific anomaly detection problem. Robust methods [5], [126], [226] specifically aim to account for these sources of disturbance. Note also that by increasing the level α in the density level set definition above, a corresponding model generally becomes more robust, since the target decision boundary becomes tighter and excludes the contamination.

3) The Semi-Supervised Setting: The semi-supervised anomaly detection setting is the case in which both *unlabeled and labeled data*

$$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X} \quad \text{and} \quad (\tilde{\boldsymbol{x}}_1, \tilde{y}_1), \ldots, (\tilde{\boldsymbol{x}}_m, \tilde{y}_m) \in \mathcal{X} \times \mathcal{Y} \tag{7}$$

are available for training a model with $\mathcal{Y} = \{\pm 1\}$, where we denote $\tilde{y} = +1$ for normal and $\tilde{y} = -1$ for anomalous points respectively. Usually, we have $m \ll n$ in the semi-supervised setting, that is, mainly unlabeled and only a few labeled instances are available, since labels are often costly to obtain in terms of resources (time, money, etc.). Labeling might for instance require domain experts such as medical professionals (e.g., pathologists) or technical experts (e.g., aerospace engineers). Anomalous instances in particular are also infrequent by nature (e.g., rare medical conditions) or very expensive (e.g., the failure of some industrial machine). The deliberate generation of anomalies is rarely an option. However, including known anomalous examples, if available, can significantly improve the detection performance of a model [143], [183], [227]–[230]. Labels are also sometimes available in the online setting where alarms raised by the anomaly detector have been investigated to determine whether they were correct. Some unsupervised anomaly detection methods can be incrementally updated when such labels become available [231]. Verifying unlabeled samples as indeed being normal can often be easier due to the more frequent nature of normal data. For this reason, among others, the special semi-supervised case of *Learning from Positive and Unlabeled Examples* (LPUE) [232]–[234], i.e., labeled normal and unlabeled examples, is also studied specifically in the anomaly detection literature [148], [161], [235]–[237].

Previous work [161] has also referred to the special case of learning exclusively from positive examples as the semi-supervised anomaly detection setting, which is confusing terminology. Although meticulously curated normal data can sometimes be available (e.g., in open category detection [238]), such a setting in which entirely (and confidently) labeled normal examples are available is rather rare in practice. The analysis of this setting is rather again justified by the *assumption* that most of the given (unlabeled) training data is normal, but not the absolute certainty thereof. This makes this setting effectively equivalent to the unsupervised setting from a modeling perspective, apart from maybe weakened assumptions on the level of noise or contamination, which previous works also point out [161]. We therefore refer to the more general setting as presented in (7) as the semi-supervised anomaly detection setting, which incorporates both labeled normal as well as anomalous examples in addition to unlabeled instances, since this setting is relevant and occurs in practice. If some labeled anomalies are available, the modeling assumptions about the anomalous distribution $\mathbb{P}^-$, as mentioned in section II-C1, become critical for effectively incorporating anomalies into training. These include for instance whether modes or clusters are expected among the anomalies (e.g., group anomalies).

TABLE I
DATA PROPERTIES RELEVANT IN ANOMALY DETECTION.

Data Property	Description
Size $n+m$	Is scalability in dataset size critical? Are there labeled samples ($m>0$) for (semi-)supervision?
Dimension D	Low- or high-dimensional? Truly high-dimensional or embedded in some higher dimensional ambient space?
Type	Continuous, discrete, or categorical?
Scales	Are features uni- or multi-scale?
Modality	Uni- or multi-modal (classes and clusters) distribution?
Convexity	Is the data support convex or non-convex?
Correlation	Are features (linearly or non-linearly) correlated?
Manifold	Has the data a (linear, locally linear, or non-linear) subspace or manifold structure? Are there invariances (translation, rotation, etc.)?
Hierarchy	Is there a natural feature hierarchy (e.g., images, video, text, speech, etc.)? Are low-level or high-level (semantic) anomalies relevant?
Context	Are there contextual features (e.g., time, space, sequence, graph, etc.)? Can anomalies be contextual?
Stationarity	Is the distribution stationary or non-stationary? Is a domain or covariate shift expected?
Noise	Is the noise level ε large or small? Is the noise type Gaussian or more complex?
Contamination	Is the data contaminated with anomalies? What is the contamination rate η?

4) The Supervised Setting: The supervised anomaly detection setting is the case in which *completely labeled data*

$$(\tilde{\boldsymbol{x}}_1, \tilde{y}_1), \ldots, (\tilde{\boldsymbol{x}}_m, \tilde{y}_m) \in \mathcal{X} \times \mathcal{Y} \tag{8}$$

is available for training a model, where again $\mathcal{Y} = \{\pm 1\}$ with $\tilde{y} = +1$ denoting normal instances and $\tilde{y} = -1$ denoting anomalies respectively. If both the normal and anomalous data points are assumed to be representative for the normal data distribution $\mathbb{P}^+$ and anomaly distribution $\mathbb{P}^-$ respectively, this learning problem is equivalent to supervised binary classification. Such a setting would thus not be an anomaly detection problem in the strict sense, but rather a classification task. Although anomalous modes or clusters might exist, i.e., some anomalies might be more likely to occur than others, *anything* not normal is by definition an anomaly. Labeled anomalies are therefore rarely representative of some 'anomaly class'. This distinction is also reflected in modeling: whereas in classification the objective is to learn a (well generalizing) decision boundary that best separates the data according to some (closed set of) class labels, the objective in anomaly detection remains the estimation of the normal density level set boundaries. Hence, we should interpret supervised anomaly detection problems as label-informed density level set estimation in which confident normal (in-distribution) and anomalous (out-of-distribution) training examples are available. Due to the costs that are usually involved with labeling, as mentioned before, the supervised anomaly detection setting is the most uncommon setting in practice.

5) Further Data Properties: Besides the settings described above, the intrinsic properties of the data itself are also crucial for modeling a specific anomaly detection problem. We give a list of relevant data properties in Table I and present a toy dataset with a specific realization of these properties in Fig. 4 which will serve us as a running example. The assumptions about these properties should be reflected in the modeling choices such as adding context or deciding among suitable deep or shallow feature maps which can be challenging. We outline these and further challenges in anomaly detection next.

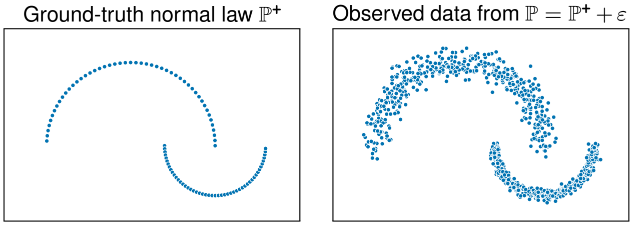

Fig. 4. A two-dimensional *Big Moon, Small Moon* toy example with real-valued ground-truth normal law $\mathbb{P}^+$ that is composed of two one-dimensional manifolds (bimodal, two-scale, non-convex). The unlabeled training data ($n =$ 1,000, $m = 0$) is generated from $\mathbb{P} = \mathbb{P}^+ + \varepsilon$ which is subject to Gaussian noise ε. This toy data is non-hierarchical, context-free, and stationary. Anomalies are off-manifold points that may occur uniformly over the displayed range.

D. Challenges in Anomaly Detection

We conclude our introduction by briefly highlighting some notable challenges in anomaly detection, some of which directly arise from the definition and data characteristics detailed above. Certainly, the fundamental challenge in anomaly detection is the mostly unsupervised nature of the problem, which necessarily requires assumptions to be made about the specific task, the domain, and the given data. These include assumptions about the relevant types of anomalies (cf., II-B2), possible prior assumptions about the anomaly distribution (cf., II-C1) and, if available, the challenge of how to incorporate labeled data instances in a generalizing way (cf., II-C3 and II-C4). Further questions include if a specific task requires an anomaly score or a threshold (cf., II-B7)? What level α (cf., II-B8) strikes a balance between false alarms and missed anomalies that is reasonable for the task? Is the data-generating process subject to noise or contamination (cf., II-C2), i.e. is robustness a critical aspect? Moreover, identifying and including the data properties given in Table I into a method and model can pose challenges as well. The computational complexity in both the dataset size $n+m$ and dimensionality D as well as the memory cost of a model at training time, but also at test time can be a limiting factor (e.g., for data streams or in real-time monitoring). Is the data-generating process assumed to be non-stationary [239]–[241] and are there distributional shifts expected at test time? For (truly) high-dimensional data, the curse of dimensionality and resulting concentration of distances can be a major issue [165]. Here, finding a representation that captures the features that are relevant for the task and meaningful for the data and domain becomes vital. Deep anomaly detection methods further entail new challenges such as an increased number of hyperparameters, for example the selection of a suitable network architecture or specification of optimization parameters (learning rate, batch sizes, etc.). In addition, the more complex the data or a model is, the greater the challenges of interpretability (e.g., [242]–[245]), transparency, and explaining anomalies become. We illustrate these various practical challenges and provide guidelines with worked-through examples in section VIII.

Given all these facets of the anomaly detection problem we covered in this introduction, it is not surprising that there is such a wealth of literature and approaches on the topic. We turn to these approaches in the following sections, where we first examine density estimation and probabilistic models (section III), followed by one-class classification methods (section IV), and finally reconstruction models (section V). In these sections, we will point out the connections between deep and shallow methods. Afterwards, we present our unifying view in section VI, which will enable us to systematically identify open challenges and paths for future research.

III. Density Estimation and Probabilistic Models

The first category of methods predict anomalies by taking the intermediate step of estimating the whole probability distribution. A wealth of existing probability models are therefore direct candidates for the task of anomaly detection. This includes classic density estimation methods [246] as well as deep statistical models. In the following, we describe the adaptation of these techniques to anomaly detection.

A. Classic Density Estimation

One of the most basic approaches to multivariate anomaly detection is to compute the Mahalanobis distance from a test point to the training data mean [247]. This is equivalent to fitting a multivariate Gaussian distribution to the training data and evaluating the log-likelihood of a test point according to that model [248]. Compared to modeling each dimension of the data independently, fitting a multivariate Gaussian can capture linear interactions between multiple dimensions. To model more complex distributions, nonparametric density estimators have been introduced, including kernel density estimators (KDE) [12], [246], histogram estimators, and Gaussian mixture models (GMMs) [249], [250]. The kernel density estimator is arguably the most widely used nonparametric density estimator due to theoretical advantages over histograms [251] and the practical issues with fitting and parameter selection for GMMs [252]. The standard kernel density estimator, along with a more recent adaptation that can deal with modest levels of outliers in the training data [253], [254], is therefore a popular approach to anomaly detection.

While classic nonparametric density estimators perform fairly well for low dimensional problems, they suffer notoriously from the curse of dimensionality: the sample size required to attain a fixed level of accuracy grows exponentially in the dimension of the feature space. One goal of deep statistical models is to overcome this challenge.

B. Energy-Based Models

Some of the earliest deep statistical models are energy based models (EBMs) [255]–[257]. An EBM is a model whose density is characterized by an energy function $E_\theta(\boldsymbol{x})$ as

$$p_\theta(\boldsymbol{x}) = \frac{1}{Z(\theta)} \exp\left(-E_\theta(\boldsymbol{x})\right), \tag{9}$$

where $Z(\theta) = \int \exp\left(-E_\theta(\boldsymbol{x})\right) \mathrm{d}\boldsymbol{x}$ is the so-called partition function that ensures that p_θ integrates to 1. These models are typically trained via gradient descent, and approximating the log-likelihood gradient $\nabla_\theta \log p_\theta(\boldsymbol{x})$ via Markov chain Monte Carlo (MCMC) [258] or Stochastic Gradient Langevin Dynamics (SGLD) [259], [260]. While one typically cannot evaluate the density p_θ directly due to the intractability of the partition function $Z(\theta)$, the function E_θ can be used as an anomaly score since it is monotonically decreasing as the density p_θ increases.

Early deep EBMs such as Deep Belief Networks [261] and Deep Boltzmann Machines [262] are graphical models consisting of layers of latent states followed by an observed output layer that models the training data. Here, the energy function depends not only on the input $\boldsymbol{x}$, but also on the latent state $\boldsymbol{z}$ so the energy function has the form $E_\theta(\boldsymbol{x}, \boldsymbol{z})$. While these approaches can richly model latent probabilistic dependencies in data distributions, they are not particularly amenable to anomaly detection since one must marginalize out the latent variables to recover some value related to the likelihood. Later works replaced the probabilistic latent layers with deterministic ones [263] allowing for the practical use of $E_\theta(\boldsymbol{x})$ as an anomaly score. This sort of model has been successfully used for deep anomaly detection [145]. Recently, EBMs have also been suggested as a framework to reinterpret deep classifiers where the energy-based training has shown to improve robustness and out-of-distribution detection performance [260].

C. Neural Generative Models (VAEs and GANs)

Neural generative models aim to learn a neural network that maps vectors sampled from a simple predefined source distribution $\mathbb{Q}$, usually a Gaussian or uniform distribution, to the actual input distribution $\mathbb{P}^{+}$. More formally, the objective is to train the network so that $\phi_\omega\left(\mathbb{Q}\right) \approx \mathbb{P}^{+}$ where $\phi_\omega\left(\mathbb{Q}\right)$ is the distribution that results from pushing the source distribution $\mathbb{Q}$ forward through neural network ϕ_ω. The two most established neural generative models are variational autoencoders (VAEs) [264]–[266] and generative adversarial networks (GANs) [267].

1) VAEs: A variational autoencoder learns a deep latent-variable model where the data points $\boldsymbol{x}$ are parameterized on latent samples $\boldsymbol{z} \sim \mathbb{Q}$ via some neural network so it learns a distribution $p_\theta(\boldsymbol{x} \,|\, \boldsymbol{z})$ such that $p_\theta(\boldsymbol{x}) \approx p^{+}(\boldsymbol{x})$. For example, a common instantiation of this is to let $\mathbb{Q}$ be an isotropic multivariate Gaussian distribution and let the neural network $\phi_{d,\omega} = (\boldsymbol{\mu}_\omega, \boldsymbol{\sigma}_\omega)$ (the *decoder*) with weights ω, parameterize the mean and variance of an isotropic Gaussian, so $p_\theta(\boldsymbol{x} \,|\, \boldsymbol{z}) \sim \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_\omega(\boldsymbol{z}), \boldsymbol{\sigma}^2_\omega(\boldsymbol{z}) I)$. Performing maximum likelihood estimation on θ is typically intractable. To remedy this an additional neural network $\phi_{e,\omega'}$ (the *encoder*) is introduced to parameterize a variational distribution $q_{\theta'}(\boldsymbol{z} \,|\, \boldsymbol{x})$, with θ' encapsulated by the output of $\phi_{e,\omega'}$, to approximate the latent posterior $p(\boldsymbol{z} \,|\, \boldsymbol{x})$. The full model is then optimized via the evidence lower bound (ELBO) in a variational Bayes manner:

$$\max_{\theta,\theta'} \; -D_{\mathrm{KL}}\left(q_{\theta'}(\boldsymbol{z}|\boldsymbol{x}) \| p(\boldsymbol{z})\right) + \mathbb{E}_{q_{\theta'}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})\right]. \tag{10}$$

Optimization proceeds using Stochastic Gradient Variational Bayes [264]. Given a trained VAE, one can estimate $p_\theta(\boldsymbol{x})$ via a Monte Carlo sampling from the prior $p(\boldsymbol{z})$ and computing $\mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}\left[p_\theta(\boldsymbol{x}\,|\,\boldsymbol{z})\right]$. Using this score directly for anomaly detection has a nice theoretical interpretation, but experiments have shown that it tends to perform worse [268], [269] than alternatively using the *reconstruction probability* [270] which conditions on $\boldsymbol{x}$ to estimate $\mathbb{E}_{q_{\theta'}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_\theta\left(\boldsymbol{x}|\boldsymbol{z}\right)\right]$.

2) GANs: GANs pose the problem of learning the target distribution as a zero-sum-game: a generative model is trained in competition with an adversary that challenges it to generate samples whose distribution is similar to the training distribution. A GAN consists of two neural networks, a *generator* network $\phi_\omega : \mathcal{Z} \to \mathcal{X}$ and a *discriminator* network $\psi_{\omega'} : \mathcal{X} \to (0,1)$ which are pitted against each other so that the discriminator is trained to discriminate between $\phi_\omega(\boldsymbol{z})$ and $\boldsymbol{x} \sim \mathbb{P}^+$ where $\boldsymbol{z} \sim \mathbb{Q}$. The generator is trained to fool the discriminator network thereby encouraging the generator to produce samples more similar to the target distribution. This is done using the following objective:

$$\begin{aligned}\min_{\omega}\max_{\omega'} \quad & \mathbb{E}_{\boldsymbol{x}\sim\mathbb{P}^+}\left[\log \psi_{\omega'}(\boldsymbol{x})\right] \\ & + \mathbb{E}_{\boldsymbol{z}\sim\mathbb{Q}}\left[\log(1-\psi_{\omega'}(\phi_\omega(\boldsymbol{z})))\right].\end{aligned} \tag{11}$$

Training is typically done via an alternating optimization scheme which is notoriously finicky [271]. There exist many GAN variants, including the Wasserstein GAN [272], [273], which is frequently used for anomaly detection methods using GANs, and StyleGAN, which has produced impressive high-resolution photorealistic images [274].

Due to their construction, GAN models offer no way to assign a likelihood to points in the input space. Using the discriminator directly has been suggested as one approach to use GANs for anomaly detection [137]. Other approaches apply optimization to find a point $\tilde{\boldsymbol{z}}$ in latent space $\mathcal{Z}$ such that $\tilde{\boldsymbol{x}} \approx \phi_\omega(\tilde{\boldsymbol{z}})$ for a test point $\tilde{\boldsymbol{x}}$. The authors of AnoGAN [51] recommend using an intermediate layer of the discriminator, $f_{\omega'}$, and setting the anomaly score to be a convex combination of the reconstruction loss $\|\tilde{\boldsymbol{x}} - \phi_\omega(\tilde{\boldsymbol{z}})\|$ and the discrimination loss $\|f_{\omega'}(\tilde{\boldsymbol{x}}) - f_{\omega'}(\phi_\omega(\tilde{\boldsymbol{z}}))\|$. In AD-GAN [147], the authors recommend initializing the search for latent points multiple times to find a collection of m latent points $\tilde{\boldsymbol{z}}_1, \ldots, \tilde{\boldsymbol{z}}_m$ while simultaneously adapting the network parameters ω_i individually for each $\tilde{\boldsymbol{z}}_i$ to improve the reconstruction and using the mean reconstruction loss as an anomaly score:

$$\frac{1}{m}\sum_{i=1}^{m}\|\tilde{\boldsymbol{x}} - \phi_{\omega_i}(\tilde{\boldsymbol{z}}_i)\|. \tag{12}$$

Other adaptations include an encoder network which is trained to find the latent point $\tilde{\boldsymbol{z}}$ and is used in a variety of ways, usually incorporating the reconstruction error [57], [148], [151], [152].

D. Normalizing Flows

Like neural generative models, normalizing flows [275]–[277] attempt to map data points from a source distribution $\boldsymbol{z} \sim \mathbb{Q}$ (usually called *base distribution* for normalizing flows) so that $\boldsymbol{x} := \phi_\omega(\boldsymbol{z})$ is distributed according to p^+. The crucial distinguishing characteristic of normalizing flows is that the latent samples are D-dimensional, so they have the same dimensionality as the input space, and the network consists of L layers $\phi_{i,\omega_i} : \mathbb{R}^D \to \mathbb{R}^D$ so $\phi_\omega = \phi_{L,\omega_L} \circ \cdots \circ \phi_{1,\omega_1}$ where each ϕ_{i,ω_i} is designed to be invertible for all ω_i, thereby making the entire network invertible. The benefit of this formulation is that the probability density of $\boldsymbol{x}$ can be calculated exactly via a change of variables

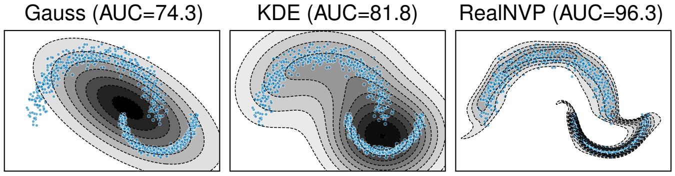

Fig. 5. Density estimation models on the *Big Moon, Small Moon* toy example (cf., Fig. 4). The parametric Gauss model is limited to an ellipsoidal (convex, unimodal) density. KDE with a RBF kernel is more flexible, yet tends to underfit the (multi-scale) distribution due a uniform kernel scale. RealNVP is the most flexible model, yet flow architectures induce biases as well, here a connected support caused by affine coupling layers in RealNVP.

$$p_{\boldsymbol{x}}(\boldsymbol{x}) = p_{\boldsymbol{z}}(\phi_\omega^{-1}(\boldsymbol{x}))\prod_{i=1}^{L}\left|\det J\phi_{i,\omega_i}^{-1}\left(\boldsymbol{x}_i\right)\right| \tag{13}$$

where $\boldsymbol{x}_L = \boldsymbol{x}$ and $\boldsymbol{x}_i = \phi_{i+1}^{-1} \circ \cdots \circ \phi_L^{-1}(\boldsymbol{x})$ otherwise. normalizing flow models are typically optimized to maximize the likelihood of the training data. Evaluating each layer's Jacobian and its determinant can be very expensive for general flow models. Consequently, the networks of flow models are usually designed so that the Jacobian is guaranteed to be upper (or lower) triangular, or have some other nice structure, such that one does not need to compute the full Jacobian and evaluating the determinant is efficient [275], [278], [279]; see [280] for an application in physics.

An advantage of these models over other methods is that one can calculate the likelihood of a point directly without any approximation while also being able to sample reasonably efficiently. Because the density $p_{\boldsymbol{x}}(\boldsymbol{x})$ can be computed exactly, normalizing flow models can be applied directly for anomaly detection [281], [282].

A drawback of these models is that they do not perform any dimensionality reduction, which argues against applying them to images where the true (effective) dimensionality is much smaller than the image dimensionality. It has been observed that these models often assign high likelihood to anomalous instances [269]. Despite present limits, we have included them here because we believe that they may provide an elegant and promising direction for future anomaly detection methods. We will come back to this in our outlook in section IX.

E. Discussion

While we have focused on the case of density estimation on i.i.d. samples of low dimensional data and images, it is worth noting that there exist many deep statistical models for other settings. When performing conditional anomaly detection, for example, one can use GAN [283], VAE [284], and normalizing flow [285] variants which perform conditional density estimation. Likewise there exist many deep generative models for virtually all data types including time series data [284], [286], text [287], [288], and graphs [289]–[291], all of which may potentially be used for anomaly detection.

It has been argued that full density estimation is not needed for solving the anomaly detection problem, since one learns all density level sets simultaneously when one really only needs a single density level set. This violates Vapnik's Principle: "[W]hen limited amount of data is available, one should avoid solving a more general problem as an intermediate step to solve the original problem" [292]. The methods in the next section seek to compute only a single density level set, that is, they perform one-class classification.

IV. One-Class Classification

One-class classification [183], [222], [293]–[295], occasionally also called *single-class classification* [296], [297], adopts a discriminative approach to anomaly detection. Methods based on one-class classification try to avoid a full estimation of the density as an intermediate step. Instead, these methods aim to directly learn a decision boundary that corresponds to a desired density level set of the normal data distribution $\mathbb{P}^+$, or more generally, to produce a decision boundary that yields a low cost when applied to unseen data.

A. The One-Class Classification Objective

We can see one-class classification as a particularly tricky classification problem, namely as binary classification where we only have (or almost only have) access to data from one class — the normal class. Given this imbalanced setting, the one-class classification objective is to learn a one-class decision boundary that minimizes (i) falsely raised alarms for true normal instances (i.e., the false alarm rate or type I error), and (ii) undetected or missed true anomalies (i.e., the miss rate or type II error). Achieving a low (or zero) false alarm rate, is conceptually simple: given enough normal data points, one could just draw some boundary that encloses all the points, for example a sufficiently large ball that contains all data instances. The crux here is, of course, to simultaneously keep the miss rate low, that is, to not draw this boundary too loosely. For this reason, one usually *a priori* specifies some target false alarm rate $\alpha \in [0, 1]$ for which the miss rate is then sought to be minimized. Note that this precisely corresponds to the idea of estimating an α-density level set for some a priori fixed level $\alpha \in [0, 1]$. The key question in one-class classification thus is how to minimize the miss rate for some given target false alarm rate with access to no (or only few) anomalies.

We can express the rationale above in terms of the binary classification risk [211], [221]. Let $Y \in \{\pm 1\}$ be the class random variable, where again $Y = +1$ denotes normal and $Y = -1$ denotes anomalous points, so we can then identify the normal data distribution as $\mathbb{P}^+ \equiv \mathbb{P}_{X|Y=+1}$ and the anomaly distribution as $\mathbb{P}^- \equiv \mathbb{P}_{X|Y=-1}$ respectively. Furthermore, let $\ell : \mathbb{R} \times \{\pm 1\} \to \mathbb{R}$ be a binary classification loss and $f : \mathcal{X} \to \mathbb{R}$ be some real-valued score function. The classification risk of scorer f under loss ℓ is then given by:

$$R(f) = \mathbb{E}_{X \sim \mathbb{P}^+}[\ell(f(X), +1)] + \mathbb{E}_{X \sim \mathbb{P}^-}[\ell(f(X), -1)]. \quad (14)$$

Minimizing the second term — the expected loss of classifying true anomalies as normal — corresponds to minimizing the (expected) miss rate. Given some unlabeled data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$, and potentially some additional labeled data $(\tilde{\boldsymbol{x}}_1, \tilde{y}_1), \ldots, (\tilde{\boldsymbol{x}}_m, \tilde{y}_m)$, we can apply the principle of empirical risk minimization to obtain

$$\min_{f} \quad \frac{1}{n} \sum_{i=1}^{n} \ell(f(\boldsymbol{x}_i), +1) + \frac{1}{m} \sum_{j=1}^{m} \ell(f(\tilde{\boldsymbol{x}}_j), \tilde{y}_j) + \mathcal{R}. \quad (15)$$

This solidifies the empirical one-class classification objective. Note that the second term is an empty sum in the unsupervised setting. Without any additional constraints or regularization, the empirical objective (15) would then be ill-posed. We add $\mathcal{R}$ as an additional term to denote and capture regularization which may take various forms depending on the assumptions about f, but critically also about $\mathbb{P}^-$. Generally, the regularization $\mathcal{R} = \mathcal{R}(f)$ aims to minimize the miss rate (e.g., via volume minimization and assumptions about $\mathbb{P}^-$) and improve generalization (e.g., via smoothing of f). Further note, that the pseudo-labeling of $y = +1$ in the first term incorporates the assumption that the n unlabeled training data points are normal. This assumption can be adjusted, however, through specific choices of the loss (e.g., hinge) and regularization. For example, requiring some fraction of the unlabeled data to get misclassified to include an assumption about the contamination rate η or achieve some target false alarm rate α as we will see below.

B. One-Class Classification in Input Space

As an illustrative example that conveys useful intuition, consider the previous simple idea of fitting a data-enclosing ball as a one-class model. Given $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$, we can define the following objective:

$$\begin{aligned} \min_{R, \boldsymbol{c}, \boldsymbol{\xi}} \quad & R^2 + \frac{1}{\nu n} \sum_{i=1}^{n} \xi_i \\ \text{s.t.} \quad & \|\boldsymbol{x}_i - \boldsymbol{c}\|^2 \leq R^2 + \xi_i, \quad \xi_i \geq 0, \quad \forall i. \end{aligned} \quad (16)$$

In words, we aim to find a hypersphere with radius $R > 0$ and center $\boldsymbol{c} \in \mathcal{X}$ that encloses the data ($\|\boldsymbol{x}_i - \boldsymbol{c}\|^2 \leq R^2$). To control the miss rate, we minimize the volume of this hypersphere by minimizing R^2 to achieve a tight spherical boundary. Slack variables $\xi_i \geq 0$ allow some points to fall outside the sphere, thus making the boundary soft, where hyperparameter $\nu \in (0, 1]$ balances this trade-off.

Objective (16) exactly corresponds to Support Vector Data Description (SVDD) applied in the input space $\mathcal{X}$, motivated above as in [7], [183], [222]. Equivalently, we can derive (16) from the binary classification risk. Consider the (shifted, cost-weighted) hinge loss $\ell(s, y)$ defined by $\ell(s, +1) = \frac{1}{1+\nu} \max(0, s)$ and $\ell(s, -1) = \frac{\nu}{1+\nu} \max(0, -s)$ [221]. Then, for a hypersphere model $f_\theta(\boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{c}\|^2 - R^2$ with parameters $\theta = (R, \boldsymbol{c})$, the corresponding classification risk objective (14) is given by

$$\begin{aligned} \min_{\theta} \quad & \mathbb{E}_{X \sim \mathbb{P}^+}[\max(0, \|X - \boldsymbol{c}\|^2 - R^2)] \\ & + \nu \, \mathbb{E}_{X \sim \mathbb{P}^-}[\max(0, R^2 - \|X - \boldsymbol{c}\|^2)]. \end{aligned} \quad (17)$$

We can estimate the first term in (17) empirically from $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, again assuming (most of) these points have been drawn from $\mathbb{P}^+$. If labeled anomalies are absent, we can still make *an assumption* about their distribution $\mathbb{P}^-$. Following the basic, uninformed prior assumption that anomalies may occur uniformly on $\mathcal{X}$ (i.e., $\mathbb{P}^- \equiv \mathcal{U}(\mathcal{X})$), we can examine the expected value in the second term analytically:

$$\begin{aligned} & \mathbb{E}_{X\sim\mathcal{U}(\mathcal{X})}[\max(0, R^2 - \|X - \boldsymbol{c}\|^2)] \\ = \;& \frac{1}{\lambda(\mathcal{X})} \int_{\mathcal{X}} \max(0, R^2 - \|\boldsymbol{x} - \boldsymbol{c}\|^2)\,\mathrm{d}\lambda(\boldsymbol{x}) \\ \leq \;& R^2\, \frac{\lambda(\mathcal{B}_R(\boldsymbol{c}))}{\lambda(\mathcal{X})} \sim R^2, \end{aligned} \tag{18}$$

where $\mathcal{B}_R(\boldsymbol{c})$ denotes the ball centered at $\boldsymbol{c}$ with radius R and λ is again the standard (Lebesgue) measure of volume.[3] This shows that the minimum volume principle [217], [219] naturally arises in one-class classification through seeking to minimize the risk of missing anomalies, here illustrated for an assumption that the anomaly distribution $\mathbb{P}^-$ follows a uniform distribution. Overall, from (17) we thus can derive the empirical objective

$$\min_{R,\boldsymbol{c}} \quad R^2 + \frac{1}{\nu n}\sum_{i=1}^{n} \max(0, \|\boldsymbol{x}_i - \boldsymbol{c}\|^2 - R^2), \tag{19}$$

which corresponds to (16) with the constraints directly incorporated into the objective function. We remark that the cost-weighting hyperparameter $\nu \in (0, 1]$ is purposefully chosen here, since it is an upper bound on the ratio of points outside and a lower bound on the ratio of points inside or on the boundary of the sphere [6], [136]. We can therefore see ν as an approximation of the false alarm rate, that is $\nu \approx \alpha$.

A sphere in the input space $\mathcal{X}$ is of course a very limited model and only matches a limited class of distributions $\mathbb{P}^+$ (e.g., an isotropic Gaussian). Minimum Volume Ellipsoids (MVE) [178], [298] and the Minimum Covariance Determinant (MCD) estimator [299] are a generalization to non-isotropic distributions with elliptical support. Nonparametric methods such as One-Class Neighbor Machines [300] provide additional freedom to model multi-modal distributions having non-convex support. Extending the objective and principles above to general feature spaces (e.g., [210], [292], [301]) further increases the flexibility of one-class models and enables decision boundaries for more complex distributions.

C. Kernel-based One-Class Classification

The kernel-based OC-SVM [6], [302] and SVDD [7], [183] are perhaps the most well-known one-class classification methods. Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be some positive semi-definite (PSD) kernel with associated RKHS $\mathcal{F}_k$ and corresponding feature map $\phi_k : \mathcal{X} \to \mathcal{F}_k$, so $k(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \langle \phi_k(\boldsymbol{x}), \phi_k(\tilde{\boldsymbol{x}}) \rangle$ for all $\boldsymbol{x}, \tilde{\boldsymbol{x}} \in \mathcal{X}$. The objective of (kernel) SVDD is again to find a data-enclosing hypersphere of minimum volume. The SVDD primal problem is the one given in (16), but with the hypersphere model $f_\theta(\boldsymbol{x}) = \|\phi_k(\boldsymbol{x}) - \boldsymbol{c}\|^2 - R^2$ defined in feature space $\mathcal{F}_k$ instead. In comparison, the OC-SVM objective is to find a *hyperplane* $\boldsymbol{w} \in \mathcal{F}_k$ that separates the data in feature space $\mathcal{F}_k$ with maximum margin from the origin:

$$\begin{aligned} \min_{\boldsymbol{w},\rho,\boldsymbol{\xi}} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 - \rho + \frac{1}{\nu n}\sum_{i=1}^{n} \xi_i \\ \text{s.t.} \quad & \rho - \langle \phi_k(\boldsymbol{x}_i), \boldsymbol{w} \rangle \leq \xi_i, \quad \xi_i \geq 0, \quad \forall i. \end{aligned} \tag{20}$$

So the OC-SVM uses a linear model $f_\theta(\boldsymbol{x}) = \rho - \langle \phi_k(\boldsymbol{x}), \boldsymbol{w} \rangle$ in feature space $\mathcal{F}_k$ with model parameters $\theta = (\boldsymbol{w}, \rho)$. The margin to the origin is given by $\frac{\rho}{\|\boldsymbol{w}\|}$ which is maximized via maximizing ρ, where $\|\boldsymbol{w}\|$ acts as a normalizer.

The OC-SVM and SVDD both can be solved in their respective dual formulations which are quadratic programs that only involve dot products (the feature map ϕ_k is implicit). For the standard Gaussian kernel (or any kernel with constant norm $k(\boldsymbol{x}, \boldsymbol{x}) = c > 0$), the OC-SVM and SVDD are equivalent [183]. In this case, the corresponding density level set estimator defined by

$$\hat{C}_\nu = \{\boldsymbol{x} \in \mathcal{X} \mid f_\theta(\boldsymbol{x}) < 0\} \tag{21}$$

is in fact an asymptotically consistent ν-density level set estimator [303]. The solution paths of hyperparameter ν have been analyzed for both the OC-SVM [304] and SVDD [305].

Kernel-induced feature spaces considerably improve the expressive power of one-class methods and allow to learn well-performing models in multi-modal, non-convex, and non-linear data settings. Many variants of kernel one-class classification have been proposed and studied over the years such as hierarchical formulations for nested density level set estimation [306], [307], Multi-Sphere SVDD [308], Multiple Kernel Learning for OC-SVM [309], [310], OC-SVM for group anomaly detection [196], boosting via L_1-norm regularized OC-SVM [311], One-class Kernel Fisher Discriminants [312]–[314], Bayesian Data Description [315], or robust variants [316].

D. Deep One-Class Classification

Selecting kernels and hand-crafting relevant features can be challenging and quickly become impractical for complex data. Deep one-class classification methods aim to overcome these challenges by learning meaningful neural network feature maps $\phi_\omega : \mathcal{X} \to \mathcal{Z}$ from the data or transferring such networks from related tasks. Deep SVDD [136], [143], [144], [317] and deep OC-SVM variants [135], [223] employ a hypersphere model $f_\theta(\boldsymbol{x}) = \|\phi_\omega(\boldsymbol{x}) - \boldsymbol{c}\|^2 - R^2$ and linear model $f_\theta(\boldsymbol{x}) = \rho - \langle \phi_\omega(\boldsymbol{x}), \boldsymbol{w} \rangle$ with explicit neural feature maps $\phi_\omega(\cdot)$ in (16) and (20) respectively. These methods are typically optimized with Stochastic Gradient Descent variants [318], [319], which, combined with GPU parallelization, makes them scale to large datasets.

As a simpler variant compared to using a neural hypersphere model in (16), the *One-Class Deep SVDD* [136], [320] has been introduced which poses the following objective:

$$\min_{\omega,\boldsymbol{c}} \quad \frac{1}{n}\sum_{i=1}^{n} \|\phi_\omega(\boldsymbol{x}_i) - \boldsymbol{c}\|^2 + \mathcal{R}. \tag{22}$$

[3] Again note that we assume $\lambda(\mathcal{X}) < \infty$ here, i.e., that the data space $\mathcal{X}$ can be bounded to numerically meaningful values.

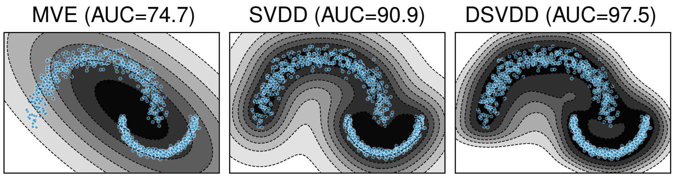

Fig. 6. One-class classification models on the *Big Moon, Small Moon* toy example (cf., Fig. 4). A Minimum Volume Ellipsoid (MVE) in input space is limited to enclose an ellipsoidal, convex region. By (implicitly) fitting a hypersphere in kernel feature space, SVDD enables non-convex support estimation. Deep SVDD learns an (explicit) neural feature map (here with smooth ELU activations) that extracts multiple data scales to fit a hypersphere model in feature space for support description.

Here, the neural network transformation $\phi_\omega(\cdot)$ is learned to minimize the mean squared distance over *all* data points to center $\boldsymbol{c} \in \mathcal{Z}$. Optimizing this simplified objective has been found to converge faster and be effective in many situations [136], [143], [143]. In light of our unifying view, we will see that we may also interpret One-Class Deep SVDD as a single-prototype deep learning method (cf., sections V-A2 and V-D).

A recurring question in deep one-class classification is how to meaningfully regularize against a feature map collapse $\phi_\omega \equiv \boldsymbol{c}$. Without regularization, minimum volume or maximum margin objectives such as (16), (20), or (22) could be trivially solved with a constant mapping [136], [321]. Possible solutions for this include reconstruction or architectural constraints [136], [317], freezing the embedding [135], [138], [139], [141], [322], inversely penalizing the embedding variance [323], using true [143], [324], auxiliary [138], [320], [325], [326], or artificial [326] negative examples in training, pseudo-labeling [153], [154], [156], [323], or integrating some manifold assumption [321]. Further variants include multi-modal extensions [144] and methods that employ adversarial learning [137], [140], [327] or transfer learning [138], [141].

Deep one-class classification methods generally offer a greater modeling flexibility and enable learning or transfer of task-relevant features for complex data. They usually require more data to be effective though, or must rely on some informative domain prior (e.g., some pre-trained network). The underlying principle of one-class classification methods — targeting a discriminative one-class boundary in learning — remains unaltered, regardless of whether a deep or shallow feature map is used.

E. Negative Examples

One-class classifiers can usually incorporate labeled negative examples ($y = -1$) in a direct manner due to their close connection to binary classification as explained above. Such negative examples can facilitate an empirical estimation of the miss rate (cf., (14) and (15)). We here recognize three qualitative types of negative examples that have been studied in the literature, that we distinguish as *artificial*, *auxiliary*, and *true* negative examples which increase in their informativeness in this order.

The idea to approach unsupervised learning problems through generating artificial data points has been around for some time (cf., section 14.2.4 in [328]). If we assume that the anomaly distribution $\mathbb{P}^-$ has some form that we can generate examples from, one idea would be to simply train a binary classifier to discern between the normal and the artificial negative examples. For the uniform prior $\mathbb{P}^- \equiv \mathcal{U}(\mathcal{X})$, this approach yields an asymptotically consistent density level set estimator [211]. Classification against uniformly drawn points from a hypercube, however, quickly becomes ineffective in higher dimensions. To improve over artificial uniform sampling, more informed sampling strategies have been proposed [329] such as resampling schemes [330], manifold sampling [331], and sampling based on local density estimation [332], [333] as well as active learning strategies [334]–[336]. Another recent idea is to treat the enormous quantities of data that are publicly available in some domains as auxiliary negative examples [325], for example images from photo sharing sites for computer vision tasks and the English Wikipedia for NLP tasks. Such auxiliary examples provide more informative domain knowledge, for instance about the distribution of natural images or the English language in general, as opposed to sampling random pixels or words. This approach, called *Outlier Exposure* [325], which trains on known anomalies can significantly improve deep anomaly detection performance in some domains [154], [325]. Finally, the most informative labeled negative examples are true anomalies, for example verified by some domain expert. Access to even a few labeled anomalies has been shown to improve detection performance significantly [143], [183], [228]. There also have been active learning algorithms proposed that include subjective user feedback (e.g., from an expert) to learn about the user-specific informativeness of particular anomalies in an application [337].

V. Reconstruction Models

Models that are trained on a reconstruction objective are among the earliest [338], [339] and most common [180], [182] neural network approaches to anomaly detection. Reconstruction-based methods learn a model that is optimized to well-reconstruct normal data instances, thereby aiming to detect anomalies by *failing* to accurately reconstruct them under the learned model. Most of these methods have a purely geometric motivation (e.g., PCA or deterministic autoencoders), yet some probabilistic variants reveal a connection to density (level set) estimation. In this section, we define the general reconstruction learning objective, highlight common underlying assumptions, as well as present standard reconstruction-based methods and discuss their variants.

A. The Reconstruction Objective

Let $\phi_\theta : \mathcal{X} \to \mathcal{X}, \boldsymbol{x} \mapsto \phi_\theta(\boldsymbol{x})$ be a feature map from the data space $\mathcal{X}$ onto itself that is composed of an *encoding* function $\phi_e : \mathcal{X} \to \mathcal{Z}$ (the *encoder*) and a *decoding* function $\phi_d : \mathcal{Z} \to \mathcal{X}$ (the *decoder*), that is, $\phi_\theta \equiv (\phi_d \circ \phi_e)_\theta$ where θ holds the parameters of both the encoder and decoder. We call $\mathcal{Z}$ the *latent space* and $\phi_e(\boldsymbol{x}) = \boldsymbol{z}$ the *latent representation* (or *embedding* or *code*) of $\boldsymbol{x}$. The reconstruction objective then is to learn ϕ_θ such that $\phi_\theta(\boldsymbol{x}) = \phi_d(\phi_e(\boldsymbol{x})) = \hat{\boldsymbol{x}} \approx \boldsymbol{x}$, that is, to find some encoding and decoding transformation so that $\boldsymbol{x}$ is reconstructed with minimal error, usually measured in

Euclidean distance. Given unlabeled data $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n \in \mathcal{X}$, the reconstruction objective is given by

$$\min_{\theta} \quad \frac{1}{n}\sum_{i=1}^{n} \|\boldsymbol{x}_i - (\phi_d \circ \phi_e)_\theta(\boldsymbol{x}_i)\|^2 + \mathcal{R}, \qquad (23)$$

where $\mathcal{R}$ again denotes the different forms of regularization that various methods introduce, for example on the parameters θ, the structure of the encoding and decoding transformations, or the geometry of latent space $\mathcal{Z}$. Without any restrictions, the reconstruction objective (23) would be optimally solved by the identity map $\phi_\theta \equiv \mathrm{id}$, but then of course nothing would be learned from the data. In order to learn something useful, structural assumptions about the data-generating process are therefore necessary. We here identify two principal assumptions: the manifold and the prototype assumptions.

1) The Manifold Assumption: The manifold assumption asserts that the data lives (approximately) on some lower-dimensional (possibly non-linear and non-convex) manifold $\mathcal{M}$ that is embedded within the data space $\mathcal{X}$ — that is $\mathcal{M} \subset \mathcal{X}$ with $\dim(\mathcal{M}) < \dim(\mathcal{X})$. In this case $\mathcal{X}$ is sometimes also called the *ambient* or *observation space*. For natural images observed in pixel space, for instance, the manifold captures the structure of scenes as well as variation due to rotation and translation, changes in color, shape, size, texture, and so on. For human voices observed in audio signal space, the manifold captures variation due to the words being spoken as well as person-to-person variation in the anatomy and physiology of the vocal folds. The (approximate) manifold assumption implies that there exists a lower-dimensional latent space $\mathcal{Z}$ and functions $\phi_e : \mathcal{X} \mapsto \mathcal{Z}$ and $\phi_d : \mathcal{Z} \mapsto \mathcal{X}$ such that for all $x \in \mathcal{X}$, $x \approx \phi_d(\phi_e(x))$. Consequently, the generating distribution $\mathbb{P}$ can be represented as the push-forward through ϕ_d of a latent distribution $\mathbb{P}_Z$. Equivalently, the latent distribution $\mathbb{P}_Z$ is the push-forward of $\mathbb{P}$ through ϕ_e.

The goal of learning is therefore to learn the pair of functions ϕ_e and ϕ_d so that $\phi_d(\phi_e(\mathcal{X})) \approx \mathcal{M} \subset \mathcal{X}$. Methods that incorporate the manifold assumption usually restrict the latent space $\mathcal{Z} \subseteq \mathbb{R}^d$ to have much lower dimensionality d than the data space $\mathcal{X} \subseteq \mathbb{R}^D$ (i.e., $d \ll D$). The manifold assumption is also widespread in related unsupervised learning tasks such as manifold learning itself [340], [341], dimensionality reduction [3], [342]–[344], disentanglement [209], [345], and representation learning in general [80], [346].

2) The Prototype Assumption: The prototype assumption asserts that there exists a finite number of prototypical elements in the data space $\mathcal{X}$ that characterize the data well. We can model this assumption in terms of a data-generating distribution that depends on a discrete latent categorical variable $Z \in \mathcal{Z} = \{1, \dots, K\}$ that captures some K prototypes or modes of the data distribution. This prototype assumption is also common in clustering and classification when we assume a collection of prototypical instances represent clusters or classes well. With the reconstruction objective under the prototype assumption, we aim to learn an encoding function that for $\boldsymbol{x} \in \mathcal{X}$ identifies a $\phi_e(\boldsymbol{x}) = k \in \{1, \dots, K\}$ and a decoding function $k \mapsto \phi_d(k) = \boldsymbol{c}_k$ that maps to some k-th prototype (or some prototypical distribution or mixture of prototypes more generally) such that the reconstruction error $\|\boldsymbol{x} - \boldsymbol{c}_k\|$ becomes minimal. In contrast to the manifold assumption where we aim to describe the data by some continuous mapping, under the (most basic) prototype assumption we characterize the data by a discrete set of vectors $\{\boldsymbol{c}_1, \dots, \boldsymbol{c}_K\} \subseteq \mathcal{X}$. The method of representing a data distribution by a set of prototype vectors is also known as Vector Quantization (VQ) [347], [348].

3) The Reconstruction Anomaly Score: A model that is trained on the reconstruction objective must extract salient features and characteristic patterns from the data in its encoding — subject to imposed model assumptions — so that its decoding from the compressed latent representation achieves low reconstruction error (e.g., feature correlations and dependencies, recurring patterns, cluster structure, statistical redundancy, etc.). Assuming that the training data $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n \in \mathcal{X}$ includes mostly normal points, we therefore expect a reconstruction-based model to produce a *low* reconstruction error for normal instances and a *high* reconstruction error for anomalies. For this reason, the anomaly score is usually also directly defined by the reconstruction error:

$$s(\boldsymbol{x}) = \|\boldsymbol{x} - (\phi_d \circ \phi_e)_\theta(\boldsymbol{x})\|^2. \qquad (24)$$

For models that have learned some truthful manifold structure or prototypical representation, a high reconstruction error would then detect off-manifold or non-prototypical instances.

Most reconstruction methods do not follow any probabilistic motivation, and a point $\boldsymbol{x}$ gets flagged anomalous simply because it does not conform to its 'idealized' representation $\phi_d(\phi_e(\boldsymbol{x})) = \hat{\boldsymbol{x}}$ under the encoding and decoding process. However, some reconstruction methods also have probabilistic interpretations, for instance PCA [349], or are even derived from probabilistic objectives such as Bayesian PCA [350] or VAEs [264]. Such methods are again related to density (level set) estimation (under specific assumptions about some latent structure), usually in the sense that a high reconstruction error indicates low density regions and vice versa.

B. Principal Component Analysis

A common way to formulate the Principal Component Analysis (PCA) objective is to seek an orthogonal basis W in data space $\mathcal{X} \subseteq \mathbb{R}^D$ that maximizes the empirical variance of the (centered) data $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n \in \mathcal{X}$:

$$\max_{W} \sum_{i=1}^{n} \|W\boldsymbol{x}_i\|^2 \quad \text{s.t. } WW^\top = I. \qquad (25)$$

Solving this objective results in a well-known eigenvalue problem, since the optimal basis is given by the eigenvectors of the empirical covariance matrix where the respective eigenvalues correspond to the component-wise variances [351]. The $d \le D$ components that explain most of the variance — the principal components — are then given by the d eigenvectors that have the largest eigenvalues.

Several works have adapted PCA for anomaly detection [77], [352]–[357], which can be considered the default reconstruction baseline. From a reconstruction perspective, the objective to find an orthogonal projection $W^\top W$ to a d-dimensional linear subspace (which is the case for $W \in \mathbb{R}^{d \times D}$

with $WW^\top = I$) such that the mean squared reconstruction error is minimized,

$$\min_W \sum_{i=1}^n \|\boldsymbol{x}_i - W^\top W \boldsymbol{x}_i\|^2 \quad \text{s.t. } WW^\top = I, \tag{26}$$

yields exactly the same PCA solution. So PCA optimally solves the reconstruction objective (23) for a linear encoder $\phi_e(\boldsymbol{x}) = W\boldsymbol{x} = \boldsymbol{z}$ and transposed linear decoder $\phi_d(\boldsymbol{z}) = W^\top \boldsymbol{z}$ with constraint $WW^\top = I$. For linear PCA, we can also readily identify its probabilistic interpretation [349], namely that the data distribution follows from the linear transformation $X = W^\top Z + \varepsilon$ of a d-dimensional latent Gaussian $Z \sim \mathcal{N}(\mathbf{0}, I)$, possibly with added noise $\varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 I)$, so that $\mathbb{P} \equiv \mathcal{N}(\mathbf{0}, W^\top W + \sigma^2 I)$. Maximizing the likelihood of this Gaussian over the encoding and decoding parameter W again yields PCA as the optimal solution [349]. Hence, PCA assumes the data lives on a d-dimensional ellipsoid embedded in data space $\mathcal{X} \subseteq \mathbb{R}^D$. Standard PCA therefore provides an illustrative example for the connections between density estimation and reconstruction.

Of course linear PCA is limited to data encodings that can only exploit linear feature correlations. Kernel PCA [3] introduced a non-linear generalization of component analysis by extending the PCA objective to non-linear kernel feature maps and taking advantage of the 'kernel trick'. For a PSD kernel $k(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ with feature map $\phi_k : \mathcal{X} \to \mathcal{F}_k$, kernel PCA solves the reconstruction objective (26) in feature space $\mathcal{F}_k$,

$$\min_W \sum_{i=1}^n \|\phi_k(\boldsymbol{x}_i) - W^\top W \phi_k(\boldsymbol{x}_i)\|^2 \quad \text{s.t. } WW^\top = I, \tag{27}$$

which results in an eigenvalue problem of the kernel matrix [3]. For kernel PCA, the reconstruction error can again serve as an anomaly score. It can be computed implicitly via the dual [4]. This reconstruction from linear principal components in feature space $\mathcal{F}_k$ corresponds to a reconstruction from some non-linear subspace or manifold in input space $\mathcal{X}$ [358]. Replacing the reconstruction $W^\top W \phi_k(\boldsymbol{x})$ in (27) with a prototype $\boldsymbol{c} \in \mathcal{F}_k$ yields a reconstruction model that considers the squared error to the kernel mean, since the prototype is optimally solved by $\boldsymbol{c} = \frac{1}{n}\sum_{i=1}^n \phi(\boldsymbol{x}_i)$ for the L^2-distance. For RBF kernels, this prototype model is (up to a multiplicative constant) equivalent to kernel density estimation [4], which provides a link between kernel reconstruction and nonparametric density estimation methods. Finally, Robust PCA variants have been introduced as well [359]–[362], which extend PCA to account for data contamination or noise (cf., II-C2).

C. Autoencoders

Autoencoders are reconstruction models that use neural networks for the encoding and decoding of data. They were originally introduced during the 80s [363]–[366] primarily as methods to perform non-linear dimensionality reduction [367], [368], yet they have also been studied early on for anomaly detection [338], [339]. Today, deep autoencoders are among the most widely adopted methods for deep anomaly detection in the literature [44], [52], [55], [124]–[134] likely owing to their long history and easy-to-use standard variants. The standard autoencoder objective is given by

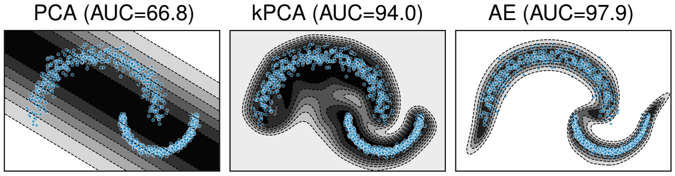

Fig. 7. Reconstruction models on the *Big Moon, Small Moon* toy example (cf., Fig. 4). PCA finds the linear subspace with the lowest reconstruction error under an orthogonal projection of the data. Kernel PCA solves (linear) component analysis in kernel feature space which enables an optimal reconstruction from (kernel-induced) non-linear components in input space. An autoencoder (AE) with one-dimensional latent code learns a one-dimensional, non-linear manifold in input space having minimal reconstruction error.

$$\min_\omega \quad \frac{1}{n}\sum_{i=1}^n \|\boldsymbol{x}_i - (\phi_d \circ \phi_e)_\omega(\boldsymbol{x}_i)\|^2 + \mathcal{R}, \tag{28}$$

where the optimization is carried out over the neural network weights ω of the encoder and decoder. A common way to regularize autoencoders is by mapping to a lower dimensional 'bottleneck' representation $\phi_e(\boldsymbol{x}) = \boldsymbol{z} \in \mathcal{Z}$ through the encoder network, which enforces data compression and effectively limits the dimensionality of the manifold or subspace to be learned. If linear networks are used, such an autoencoder in fact recovers the same optimal subspace as spanned by the PCA eigenvectors [369], [370]. Apart from a 'bottleneck', a number of different ways to regularize autoencoders have been introduced in the literature. Following ideas of sparse coding [371]–[374], sparse autoencoders [375], [376] regularize the (possibly higher-dimensional, over-complete) latent code towards sparsity, for example via L^1 Lasso penalization [377]. Denoising autoencoders (DAEs) [378], [379] explicitly feed noise-corrupted inputs $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \varepsilon$ into the network which is then trained to reconstruct the original inputs $\boldsymbol{x}$. DAEs thus provide a way to specify a noise model for ε (cf., II-C2), which has been applied for noise-robust acoustic novelty detection [42], for instance. For situations in which the training data is already corrupted with noise or unknown anomalies, robust deep autoencoders [126], which split the data into well-represented and corrupted parts similar to robust PCA [361], have been proposed. Contractive autoencoders (CAEs) [380] propose to penalize the Frobenius norm of the Jacobian of the encoder activations with respect to the inputs to obtain a smoother and more robust latent representation. Such ways of regularization influence the geometry and shape of the subspace or manifold that is learned by an autoencoder, for example by imposing some degree of smoothness or introducing invariances towards certain types of input corruptions or transformations [130]. Hence, these regularization choices should again reflect the specific assumptions of a given anomaly detection task.

Besides the deterministic variants above, probabilistic autoencoders have also been proposed, which again establish a connection to density estimation. The most explored class of probabilistic autoencoders are Variational Autoencoders (VAEs) [264]–[266], as introduced in section III-C1 through the lens of neural generative models, which approximately maximize the data likelihood (or evidence) by maximizing the evidence lower bound (ELBO). From a reconstruction perspective, VAEs adopt a stochastic autoencoding process, which is realized by encoding and decoding the parameters of distributions (e.g., Gaussians) through the encoder and decoder networks, from which the latent code and reconstruction then can be sampled. For a standard Gaussian VAE, for example, where $q(\boldsymbol{z}|\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{x}}, \operatorname{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{x}}))$, $p(\boldsymbol{z}) \sim \mathcal{N}(\boldsymbol{0}, I)$, and $p(\boldsymbol{x}|\boldsymbol{z}) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{z}}, I)$ with encoder $\phi_{e,\omega'}(\boldsymbol{x}) = (\boldsymbol{\mu}_{\boldsymbol{x}}, \boldsymbol{\sigma}_{\boldsymbol{x}})$ and decoder $\phi_{d,\omega}(\boldsymbol{z}) = \boldsymbol{\mu}_{\boldsymbol{z}}$, the empirical ELBO objective (10) becomes

$$\begin{aligned}\min_{\omega,\omega'} \; \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{M} \; &\Big[\tfrac{1}{2}\|\boldsymbol{x}_i-\boldsymbol{\mu}_{\boldsymbol{z}_{ij}}\|^2 \\ &+ D_{\mathrm{KL}}\big(\mathcal{N}(\boldsymbol{z}_{ij};\boldsymbol{\mu}_{\boldsymbol{x}_i},\operatorname{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{x}_i}))\|\mathcal{N}(\boldsymbol{z}_{ij};\boldsymbol{0},I)\big)\Big],\end{aligned} \tag{29}$$

where $\boldsymbol{z}_{i1}, \ldots, \boldsymbol{z}_{iM}$ are M Monte Carlo samples drawn from the encoding distribution $\boldsymbol{z} \sim q(\boldsymbol{z}|\boldsymbol{x}_i)$ of $\boldsymbol{x}_i$. Hence, such a VAE is trained to minimize the mean reconstruction error over samples from an encoded latent Gaussian that is regularized to be close to a standard isotropic Gaussian. VAEs have been used in various forms for anomaly detection [268], [270], [381], for instance on multimodal sequential data with LSTM networks for anomaly detection in robot-assisted feeding [382] and for new physics mining at the Large Hadron Collider [74]. Another class of probabilistic autoencoders that has been applied to anomaly detection are Adversarial Autoencoders (AAEs) [44], [52], [383]. By employing an adversarial loss to regularize and match the latent encoding distribution, AAEs can employ any arbitrary prior $p(\boldsymbol{z})$, as long as sampling is feasible.

Finally, other autoencoder variants that have been applied to anomaly detection include RNN-based autoencoders [193], [230], [384], [385], convolutional autoencoders [55], autoencoder ensembles [125], [385] and variants that actively control the topology of the latent code [386]. Autoencoders also have been employed in two-step approaches that utilize autoencoders for dimensionality reduction and apply traditional methods on the learned embeddings [135], [387], [388].

D. Prototypical Clustering

Clustering methods that make the prototype assumption provide another approach to reconstruction-based anomaly detection. As mentioned above, the reconstruction error here is usually given by the distance of a point to its nearest prototype, which ideally has been learned to represent a distinct mode of the normal data distribution. Prototypical clustering methods [389] include the well-known Vector Quantization (VQ) algorithms k-means, k-medians, and k-medoids, which define a Voronoi partitioning [390], [391] over the metric space where they are applied — typically the input space $\mathcal{X}$. Kernel variants of k-means have also been studied [392] and considered for anomaly detection [308]. More recently, deep learning approaches to clustering have also been introduced [393]–[396], some also based on k-means [397], and adopted for anomaly detection [128], [387], [398]. As in deep one-class classification (cf., section IV-D), a persistent question in deep clustering is how to effectively regularize against a feature map collapse [399]. Note that whereas for deep clustering methods the reconstruction error is measured in latent space $\mathcal{Z}$, for deep autoencoders it is measured in the input space $\mathcal{X}$ after decoding. Thus, a latent feature collapse (i.e., a constant encoder $\phi_e \equiv \boldsymbol{c} \in \mathcal{Z}$) would result in a constant decoding (the data mean at optimum) for an autoencoder, which generally is a suboptimal solution of (28). For this reason, autoencoders seem less susceptible to a feature collapse, though they have also been observed to converge to bad local optima under SGD optimization, specifically if they employ bias terms [136].

VI. A Unifying View of Anomaly Detection

In this section, we present a unifying view on the anomaly detection problem. We identify specific anomaly detection modeling components that allow us to organize and characterize the vast collection of discussed anomaly detection methods in a systematic way. Importantly, this view shows connections that enable the transfer of algorithmic ideas between existing anomaly detection methods. Thus it reveals promising directions for future research such as transferring concepts and ideas from kernel-based anomaly detection to deep methods and vice versa.

A. Modeling Dimensions of the Anomaly Detection Problem

We identify the following five components or *modeling dimensions* for anomaly detection:

D1 **Loss**	$\ell : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}, (s, y) \mapsto \ell(s, y)$
D2 **Model**	$f_\theta : \mathcal{X} \to \mathbb{R}, \boldsymbol{x} \mapsto f_\theta(\boldsymbol{x})$
D3 **Feature Map**	$\boldsymbol{x} \mapsto \phi(\boldsymbol{x})$
D4 **Regularization**	$\mathcal{R}(f, \phi, \theta)$
D5 **Inference Mode**	Frequentist or Bayesian $\theta \sim p(\theta)$

Dimension D1 **Loss** is the (scalar) loss function that is applied to the output of some model $f_\theta(\boldsymbol{x})$. Semi-supervised or supervised methods apply loss functions that also incorporate labels, but for the many unsupervised anomaly detection methods we usually have $\ell(s, y) = \ell(s)$. D2 **Model** defines the specific model f_θ that maps an input $\boldsymbol{x} \in \mathcal{X}$ to some scalar value that is evaluated by the loss. We have aligned our previous three sections along this major modeling dimension where we covered certain groups of methods that formulate models based on common principles, namely probabilistic modeling, one-class classification, and reconstruction. Due to the close link between anomaly detection and density estimation (cf., II-B5), many of the methods formulate a likelihood model $f_\theta(\boldsymbol{x}) = p_\theta(\boldsymbol{x} \,|\, \mathcal{D}_n)$ with negative log-loss $\ell(s) = -\log(s)$, that is they have a negative log-likelihood objective, where $\mathcal{D}_n = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ denotes the training data. Dimension D3 captures the **Feature Map** $\boldsymbol{x} \mapsto \phi(\boldsymbol{x})$ that is used in a model. This could be an (implicit) feature map $\phi_k(\boldsymbol{x})$ defined by some given kernel k, for example, or an (explicit) neural network feature map $\phi_\omega(\boldsymbol{x})$ that is learned and parameterized with network weights ω. With dimension D4 **Regularization**, we

TABLE II
ANOMALY DETECTION METHODS IDENTIFIED WITH OUR UNIFYING VIEW (LAST COLUMN CONTAINS REPRESENTATIVE REFERENCES).

Method	Loss $\ell(s,y)$	Model $f_\theta(\boldsymbol{x})$	Feature Map $\phi(\boldsymbol{x})$		Parameter θ	Regularization $\mathcal{R}(f,\phi,\theta)$	Bayes?	References
Parametric Density	$-\log(s)$	$p(\boldsymbol{x} \mid \theta)$	$\boldsymbol{x}$	(input)	θ	choice of density class $\{p_\theta \mid \theta \in \Theta\}$	✗	[400], [401]
Gaussian/Mahalanobis	$-\log(s)$	$\mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu},\Sigma)$	$\boldsymbol{x}$	(input)	$(\boldsymbol{\mu},\Sigma)$	–	✗	[400], [401]
GMM	$-\log(s)$	$\sum_k \pi_k \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_k,\Sigma_k)$	$\boldsymbol{x}$	(input)	$(\pi,\boldsymbol{\mu},\Sigma)$	number of mixture components K	latent	[400], [401]
KDE	s	$\|\phi_k(\boldsymbol{x})-\boldsymbol{\mu}\|^2$	$\phi_k(\boldsymbol{x})$	(kernel)	$\boldsymbol{\mu}$	kernel hyperparameters (e.g., bandwidth h)	✗	[249], [250]
EBMs	$-\log(s)$	$\frac{1}{Z(\theta)}\exp(-E(\phi(\boldsymbol{x}),\boldsymbol{z};\theta))$		(various)	θ	latent prior $p(\boldsymbol{z})$	latent	[145], [257]
Normalizing Flows	$-\log(s)$	$p_{\boldsymbol{z}}(\phi_\omega^{-1}(\boldsymbol{x}))\,\lvert\det J_{\phi_\omega^{-1}}(\boldsymbol{x})\rvert$	$\phi_\omega(\boldsymbol{x})$	(neural)	(ω)	base distribution $p_{\boldsymbol{z}}(\boldsymbol{z})$; diffeomorphism architecture	✗	[276], [281]
GAN (D-based)	$-\log(s)$	$\sigma(\langle\boldsymbol{w},\psi_\omega(\boldsymbol{x})\rangle)$	$\psi_\omega(\boldsymbol{x})$	(neural)	$(\boldsymbol{w},\omega)$	adversarial training	✗	[57], [327]
Min. Vol. Sphere	$\max(0,s)$	$\|\boldsymbol{x}-\boldsymbol{c}\|^2-R^2$	$\boldsymbol{x}$	(input)	$(\boldsymbol{c},R)$	νR^2	✗	[183]
Min. Vol. Ellipsoid	$\max(0,s)$	$(\boldsymbol{x}-\boldsymbol{c})^\top\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{c})-R^2$	$\boldsymbol{x}$	(input)	$(\boldsymbol{c},R,\Sigma)$	$\nu(\frac{1}{2}\|\Sigma\|_{\mathrm{Fr}}^2+R^2)$	✗	[299]
SVDD	$\max(0,s)$	$\|\phi_k(\boldsymbol{x})-\boldsymbol{c}\|^2-R^2$	$\phi_k(\boldsymbol{x})$	(kernel)	$(\boldsymbol{c},R)$	νR^2	✗	[7]
Semi-Sup. SVDD	$\max(0,ys)$	$\|\phi_k(\boldsymbol{x})-\boldsymbol{c}\|^2-R^2$	$\phi_k(\boldsymbol{x})$	(kernel)	$(\boldsymbol{c},R)$	νR^2	✗	[7], [228]
Soft Deep SVDD	$\max(0,s)$	$\|\phi_\omega(\boldsymbol{x})-\boldsymbol{c}\|^2-R^2$	$\phi_\omega(\boldsymbol{x})$	(neural)	$(\boldsymbol{c},R,\omega)$	$\nu R^2)$; weight decay; collapse reg. (various)	✗	[136]
OC Deep SVDD	s	$\|\phi_\omega(\boldsymbol{x})-\boldsymbol{c}\|^2$	$\phi_\omega(\boldsymbol{x})$	(neural)	$(\boldsymbol{c},\omega)$	weight decay; collapse reg. (various)	✗	[136]
Deep SAD	s^y	$\|\phi_\omega(\boldsymbol{x})-\boldsymbol{c}\|^2$	$\phi_\omega(\boldsymbol{x})$	(neural)	$(\boldsymbol{c},\omega)$	weight decay	✗	[143]
OC-SVM	$\max(0,s)$	$\rho-\langle\boldsymbol{w},\phi_k(\boldsymbol{x})\rangle$	$\phi_k(\boldsymbol{x})$	(kernel)	$(\boldsymbol{w},\rho)$	$\nu(\frac{1}{2}\|\boldsymbol{w}\|^2-\rho)$	✗	[6]
OC-NN	$\max(0,s)$	$\rho-\langle\boldsymbol{w},\phi_\omega(\boldsymbol{x})\rangle$	$\phi_\omega(\boldsymbol{x})$	(neural)	$(\boldsymbol{w},\rho,\omega)$	$\nu(\frac{1}{2}(\|\boldsymbol{w}\|^2-\rho)$; weight decay	✗	[223]
Bayesian DD	$\max(0,s)$	$\|\phi_k(\boldsymbol{x})-\boldsymbol{c}\|^2-R^2$	$\phi_k(\boldsymbol{x})$	(kernel)	$(\boldsymbol{c},R)$	$\boldsymbol{c}=\sum_i\alpha_i\phi_k(\boldsymbol{x}_i)$ with prior $\alpha\sim\mathcal{N}(\boldsymbol{\mu},\Sigma)$	fully	[315]
GT	$-\log(s)$	$\prod_k\sigma_k(\langle\boldsymbol{w},\phi_\omega(T_k(\boldsymbol{x}))\rangle)$	$\phi_\omega(\boldsymbol{x})$	(neural)	$(\boldsymbol{w},\omega)$	transformations $\mathcal{T}=\{T_1,\dots,T_K\}$ for self-labeling	✗	[153], [154]
GOAD (CE)	$-\log(s)$	$\prod_k\sigma_k(-\|\phi_\omega(T_k(\boldsymbol{x}))-\boldsymbol{c}_k\|^2)$	$\phi_\omega(\boldsymbol{x})$	(neural)	$(\boldsymbol{c}_1,\dots,\boldsymbol{c}_K,\omega)$	transformations $\mathcal{T}=\{T_1,\dots,T_K\}$ for self-labeling	✗	[156]
BCE (supervised)	$-y\log(s)-\frac{1-y}{2}\log(1-s)$	$\sigma(\langle\boldsymbol{w},\phi_\omega(\boldsymbol{x})\rangle)$	$\phi_\omega(\boldsymbol{x})$	(neural)	$(\boldsymbol{w},\omega)$	weight decay	✗	[320]
BNN (supervised)	$-y\log(s)-\frac{1-y}{2}\log(1-s)$	$\sigma(\langle\boldsymbol{w},\phi_\omega(\boldsymbol{x})\rangle)$	$\phi_\omega(\boldsymbol{x})$	(neural)	$(\boldsymbol{w},\omega)$	prior $p(\boldsymbol{w},\omega)$	fully	[402], [403]
PCA	s	$\|\boldsymbol{x}-W^\top W\boldsymbol{x}\|_2^2$	$\boldsymbol{x}$	(input)	W	$WW^\top=I$	✗	[352]
Robust PCA	s	$\|\boldsymbol{x}-W^\top W\boldsymbol{x}\|_1$	$\boldsymbol{x}$	(input)	W	$WW^\top=I$	✗	[359]
Probabilistic PCA	$-\log(s)$	$\mathcal{N}(\boldsymbol{x} \mid 0,W^\top W+\sigma^2 I)$	$\boldsymbol{x}$	(input)	(W,σ^2)	linear latent Gauss model $\boldsymbol{x}=W^\top\boldsymbol{z}+\varepsilon$	latent	[349]
Bayesian PCA	$-\log(s)$	$\mathcal{N}(\boldsymbol{x} \mid 0,W^\top W+\sigma^2 I)\,p(W \mid \boldsymbol{\alpha})$	$\boldsymbol{x}$	(input)	(W,σ^2)	linear latent Gauss model with prior $p(W \mid \boldsymbol{\alpha})$	fully	[350]
Kernel PCA	s	$\|\phi_k(\boldsymbol{x})-W^\top W\phi_k(\boldsymbol{x})\|^2$	$\phi_k(\boldsymbol{x})$	(kernel)	W	$WW^\top=I$	✗	[3], [4]
Autoencoder	s	$\|\boldsymbol{x}-\phi_\omega(\boldsymbol{x})\|_2^2$	$\phi_\omega(\boldsymbol{x})$	(neural)	ω	advers. (AAE), contract. (CAE), denois. (DAE), etc.	✗	[126], [134]
VAE	$-\log(s)$	$p_{\phi_\omega}(\boldsymbol{x} \mid \boldsymbol{z})$	$\phi_\omega(\boldsymbol{x})$	(neural)	ω	latent prior $p(\boldsymbol{z})$	latent	[266], [270]
GAN (G-based)	$-\log(s)$	$p_{\phi_\omega}(\boldsymbol{x} \mid \boldsymbol{z})$	$\phi_\omega(\boldsymbol{x})$	(neural)	ω	adversarial training and latent prior $p(\boldsymbol{z})$	latent	[51], [147]
k-means	s	$\|\boldsymbol{x}-\operatorname{argmin}_{\boldsymbol{c}_k}\|\boldsymbol{x}-\boldsymbol{c}_k\|_2\|_2^2$	$\boldsymbol{x}$	(input)	$(\boldsymbol{c}_1,\dots,\boldsymbol{c}_K)$	number of prototypes K	✗	[389]
k-medians	s	$\|\boldsymbol{x}-\operatorname{argmin}_{\boldsymbol{c}_k}\|\boldsymbol{x}-\boldsymbol{c}_k\|_1\|_1$	$\boldsymbol{x}$	(input)	$(\boldsymbol{c}_1,\dots,\boldsymbol{c}_K)$	number of prototypes K	✗	[389]
VQ	s	$\|\boldsymbol{x}-\phi_d(\operatorname{argmin}_{\boldsymbol{c}_k}\|\phi_e(\boldsymbol{x})-\boldsymbol{c}_k\|)\|$		(various)	$(\boldsymbol{c}_1,\dots,\boldsymbol{c}_K)$	number of prototypes K	✗	[347], [348]

capture various forms of regularization $\mathcal{R}(f,\phi,\theta)$ of the model f_θ, the feature map ϕ, and their parameters θ in a broader sense. Note that θ here may include both model parameters as well as feature map parameters, that is $\theta = (\theta_f, \theta_\phi)$ in general. θ_f could be the distributional parameters of a parametric density model, for instance, and θ_ϕ the weights of a neural network. Our last modeling dimension D5 describes the **Inference Mode**, specifically whether a method performs Bayesian inference.

The identification of the above modeling dimensions enables us to formulate a general anomaly detection learning objective that applies to a broad range of anomaly detection methods:

$$\boxed{\min_{\theta} \quad \frac{1}{n}\sum_{i=1}^{n} \ell(f_\theta(\boldsymbol{x}_i), y_i) + \mathcal{R}(f,\phi,\theta). \qquad (*)}$$

Denoting the minimum of $(*)$ by θ^*, the anomaly score of a test input $\tilde{\boldsymbol{x}}$ is computed via the model $f_{\theta^*}(\tilde{\boldsymbol{x}})$. In the Bayesian case, when the objective in $(*)$ is the negative log-likelihood of a posterior $p(\theta \mid \mathcal{D}_n)$ induced by a prior distribution $p(\theta)$, we can predict in a fully Bayesian fashion via the expected model $\mathbb{E}_{\theta\sim p(\theta \mid \mathcal{D}_n)} f_\theta(\boldsymbol{x})$. We describe many well-known anomaly detection methods within our unified view in Table II.

B. *Distance-based Anomaly Detection*

Our unifying view focuses on anomaly detection methods that formulate some learning objective. Apart from these methods, there also exists a rich literature on purely 'distance-based' anomaly detection methods and algorithms that have been studied extensively in the data mining community in particular. Many of these algorithms follow a *lazy learning* paradigm, in which there is no a priori training phase of learning a model, but instead new test points are evaluated with respect to the training instances only as they occur. We here group these methods as 'distance-based' without further granularity, but remark that various taxonomies for these types of methods have been proposed [161], [179]. Examples of such methods include nearest-neighbor-based methods [8], [9], [404]–[406] such as LOF [10] and partitioning tree-based methods [407] such as Isolation Forest [408], [409]. These methods usually also aim to capture the high-density regions of the data in some manner, for instance by scaling distances in relation to local neighborhoods [10], and thus are mostly consistent with the formal anomaly detection problem definition presented in section II. The majority of these algorithms have been studied and applied in the original input space $\mathcal{X}$. Few of them have been considered in the context of deep learning, but some hybrid anomaly detection approaches apply distance-based algorithms on top of deep neural feature maps from pre-trained networks (e.g., [410]).

VII. EVALUATION AND EXPLANATION

The theoretical considerations and unifying view above provide useful insights about the characteristics and underlying modeling assumptions of the different anomaly detection methods. What matters the most to the practitioner, however, is to evaluate how well an anomaly detection method performs on real data. In this section, we present different aspects of evaluation, in particular, the problem of *building* a dataset that includes meaningful anomalies, and the problem of robustly *evaluating* an anomaly detection model on the collected data.

TABLE III
EXISTING ANOMALY DETECTION BENCHMARKS.

k-classes-out	(Fashion-)MNIST, CIFAR-10, STL-10, ImageNet
Synthetic	MNIST-C [412], ImageNet-C [413], ImageNet-P [413], ImageNet-O [418]
Real-world	*Industrial:* MVTec-AD [189], PCB [419] *Medical:* CAMELYON16 [61], [420], NIH Chest X-ray [61], [421], MOOD [422], HCP/BRATS [52], Neuropathology [60], [423] *Security:* Credit-card-fraud [424], URL [425], UNSW-NB15 [426] *Time series:* NAB [427], Yahoo [428] *Misc.:* Emmott [417], ELKI [429], ODDS [430], UCI [431], [432]

In a second step, we will look at the limitations of classical evaluation techniques, specifically, their inability to directly inspect and verify the exact strategy employed by some model for detection, for instance, which input variables a model uses for prediction. We then present 'Explainable AI' approaches for enabling such deeper inspection of the model.

A. Building Anomaly Detection Benchmarks

Unlike standard supervised datasets, there is an intrinsic difficulty in building anomaly detection benchmarks: Anomalies are rare and some of them may have never been observed before they manifest themselves in practice. Existing anomaly benchmarks typically rely on one of the following strategies:

1) *k-classes-out:* Start from a binary or multi-class dataset and declare one or more classes to be normal and the rest to be anomalous. Due to the semantic homogeneity of the resulting 'anomalies,' such a benchmark may not be a good simulacrum of real anomalies. For example, simple low-level anomalies (e.g., additive noise) may not be tested for.
2) *Synthetic:* Start from an existing supervised or unsupervised dataset and generate synthetic anomalies (e.g., [411]–[413]). Having full control over anomalies is desirable from a statistical view point, to get robust error estimates. However, the characteristics of real anomalies may be unknown or difficult to generate.
3) *Real-world:* Consider a dataset that contains anomalies and have them labeled by a human expert. This is the ideal case. In addition to the anomaly label, the human can augment a sample with an annotation of which exact features are responsible for the anomaly (e.g., a segmentation mask in the context of image data).

We provide examples of anomaly detection benchmarks and datasets falling into these three categories in Table III.

Although all three approaches are capable of producing anomalous data, we note that real anomalies may exhibit much wider and finer variations compared to those in the dataset. In *adversarial cases*, anomalies may be designed maliciously to avoid detection (e.g., in fraud and cybersecurity scenarios [203], [335], [414]–[417]).

B. Evaluating Anomaly Detectors

Most applications come with different costs for false alarms (type I error) and missed anomalies (type II error). Hence, it is common to consider the decision function

$$\text{decide} \begin{cases} \text{anomaly} & \text{if} \quad s(\boldsymbol{x}) \geq \tau \\ \text{inlier} & \text{if} \quad s(\boldsymbol{x}) < \tau, \end{cases} \tag{30}$$

where s denotes the anomaly score, and adjust the decision threshold τ in a way that (i) minimizes the costs associated to the type I and type II errors on the collected validation data, or (ii) accommodates the hard constraints of the environment in which the anomaly detection system will be deployed.

To illustrate this, consider an example in financial fraud detection: anomaly alarms are typically sent to a fraud analyst who must decide whether to open an investigation into the potentially fraudulent activity. There is typically a fixed number of analysts. Suppose they can only handle k alarms per day, that is, the k examples with the highest predicted anomaly score. In this scenario, the measure to optimize is the 'precision@k', since we want to maximize the number of anomalies contained in those k alarms.

In contrast, consider a credit card company that places an automatic hold on a credit card when an anomaly alarm is reported. False alarms result in angry customers and reduced revenue, so the goal is to maximize the number of true alarms subject to a constraint on the percentage of false alarms. The corresponding measure is to maximize 'recall@k' — where k is the number of false alarms.

However, it is often the case that application-related costs and constraints are not fully specified or vary over time. With such restrictions, it is desirable to have a measure that evaluates the performance of anomaly detection models under a broad range of possible application scenarios, or analogously, a broad range of decision thresholds τ. The Area Under the ROC Curve (AUROC or simply AUC) computes the fraction of detected anomalies, averaged over the full range of decision thresholds. AUC is the standard performance measure used in anomaly detection [429], [433]–[436]. Another commonly employed measure is the Area Under the Precision-Recall Curve (AUPRC) [199].

C. A Comparison on MNIST-C and MVTec-AD

In the following, we apply the AUC measure to compare a selection of anomaly detection methods from the three major approaches (probabilistic, one-class, reconstruction) and three types of feature representation (raw input, kernel, and neural network). We perform the comparison on the synthetic MNIST-C and real-world MVTec-AD datasets. MNIST-C is MNIST extended with a set of fifteen types of corruptions (e.g., blurring, added stripes, impulse noise, etc). MVTec-AD consists of fifteen image sets from industrial production, where anomalies correspond to manufacturing defects. These images sometimes take the form of textures (e.g., wood, grid) or objects (e.g., toothbrush, screw). For MNIST-C, models are trained on the standard MNIST training set and then tested on each corruption separately. We measure the AUC separating the corrupted from the uncorrupted test set. For MVTec-AD, we train distinct models on each of the fifteen image sets and measure the AUC on the corresponding test set. Results for each model are shown in Tables IV and V. We provide the training details of each model in Appendix B.

TABLE IV
AUC DETECTION PERFORMANCE ON MNIST-C.

	Gauss	MVE	PCA	KDE	SVDD	KPCA	AGAN	DOCC	AE
brightness	**100.0**	99.0	**100.0**	**100.0**	100.0	**100.0**	**100.0**	13.7	**100.0**
canny edges	99.4	68.4	**100.0**	78.9	96.3	99.9	100.0	97.9	100.0
dotted line	99.9	62.9	99.3	68.5	70.0	92.6	91.5	86.4	**100.0**
fog	100.0	89.6	98.1	62.1	92.3	91.3	**100.0**	17.4	100.0
glass blur	79.5	34.7	70.7	8.0	49.1	27.1	**100.0**	31.1	99.6
impulse noise	**100.0**	69.0	**100.0**	98.0	99.7	**100.0**	**100.0**	97.5	**100.0**
motion blur	38.1	43.4	24.3	8.1	50.2	18.3	**100.0**	70.7	95.1
rotate	31.3	54.7	24.9	37.1	57.7	38.7	**93.2**	65.5	53.4
scale	7.5	20.7	14.5	5.0	36.5	19.6	68.1	**79.8**	40.4
shear	63.7	58.1	55.5	49.9	58.2	54.1	**94.9**	64.6	70.6
shot noise	94.9	43.2	97.1	41.6	63.4	81.5	96.7	51.5	**99.7**
spatter	**99.8**	52.6	85.0	44.5	57.3	64.5	99.0	68.2	97.4
stripe	**100.0**	99.9	**100.0**	**100.0**	100.0	**100.0**	**100.0**	100.0	**100.0**
translate	94.5	73.9	96.3	76.2	91.8	94.8	97.3	**98.8**	92.2
zigzag	99.9	72.5	100.0	84.0	87.7	99.4	98.3	94.3	**100.0**

TABLE V
AUC DETECTION PERFORMANCE ON MVTEC-AD.

		Gauss	MVE	PCA	KDE	SVDD	KPCA	AGAN	DOCC	AE
Textures	carpet	48.8	63.5	45.6	34.8	48.7	41.9	83.1	**90.6**	36.8
	grid	60.6	67.8	81.8	71.7	80.4	76.7	**91.7**	52.4	74.6
	leather	39.6	49.5	60.3	41.5	57.3	61.1	58.6	**78.3**	64.0
	tile	68.5	79.7	56.4	68.9	73.3	63.2	74.1	**96.5**	51.8
	wood	54.0	80.1	90.4	**94.7**	94.1	90.6	74.5	91.6	88.5
Objects	bottle	78.9	67.0	97.4	83.3	89.3	96.3	90.6	**99.6**	95.0
	cable	56.5	71.9	77.6	66.9	73.1	75.6	69.7	**90.9**	57.3
	capsule	71.6	65.1	75.7	56.2	61.3	71.5	60.7	**91.0**	52.5
	hazelnut	67.6	80.4	89.1	69.9	74.3	83.8	**96.4**	95.0	90.5
	metal nut	54.7	45.1	56.4	33.3	54.3	59.0	79.3	**85.2**	45.5
	pill	65.5	71.5	**82.5**	69.1	76.2	80.7	64.6	80.4	76.0
	screw	53.5	35.5	67.9	36.9	8.6	46.7	**99.6**	86.9	77.9
	toothbrush	93.9	76.1	**98.3**	93.3	96.1	**98.3**	70.8	96.4	49.4
	transistor	70.2	64.8	81.8	72.4	74.8	80.0	78.8	**90.8**	51.2
	zipper	50.1	65.2	82.8	61.4	68.6	81.0	69.7	**92.4**	35.0

A first striking observation is the heterogeneity in performance of the various methods on the different corruptions and defect classes. For example, the AGAN performs generally well on MNIST-C but is systematically outperformed by the Deep One-Class Classification model (DOCC) on MVTec-AD. Also, the more powerful nonlinear models are not better on every class, and simple 'shallow' models occasionally outperform their deeper counterparts. For instance, the simple Gaussian model reaches top performance on MNIST-C:Spatter, linear PCA ranks highest on MVTec-AD:Toothbrush, and KDE ranks highest on MVTec-AD:Wood. The fact that some of the simplest models sometimes perform well highlights the strong differences in modeling structure of each anomaly detection model.

However, what is still unclear is whether the measured model performance faithfully reflects the performance on a broader set of anomalies (i.e., the generalization performance) or whether some methods only benefit from the specific (possibly non-representative) types of anomalies that have been collected in the test set. In other words, assuming that all models achieve 100% test accuracy (e.g., MNIST-C:stripes), can we conclude that all models will perform well on a broader range of such anomalies? This problem was already highlighted in the context of supervised learning, and explanation methods can be applied to uncover potential hidden weaknesses of models, also known as 'Clever Hanses' [244].

D. Explaining Anomalies

To gain further insight into the detection strategies used by different anomaly models, and in turn to also address some of the limitations of classical validation procedures, many practitioners wish to augment anomaly predictions with an 'explanation.' Producing explanations of model predictions is already common in supervised learning, and this field is often referred to as Explainable AI (or XAI) [245]. Popular XAI methods include LIME [437], (Guided) Grad-CAM [438], integrated gradients [439], [440], and Layer-wise Relevance Propagation (LRP) [441]. Grad-CAM and LRP rely on the structure of the network to produce a robust explanation.

Explainable AI has recently also been brought to unsupervised learning, in particular, anomaly detection [38], [322], [326], [442]–[444]. Unlike supervised learning, which is largely dominated by neural networks [81], [84], [445], state-of-the-art methods for unsupervised learning are much more heterogeneous, including neural networks but also kernel-based, centroid-based, or probability-based models. In such a heterogeneous setting, it is difficult to build explanation methods that allow for a consistent comparison of detection strategies of the multiple anomaly detection models. Two directions to achieve such consistent explanations are particularly promising:

1) Model-agnostic explanation techniques (e.g., sampling-based) that apply transparently to any model, whether it is a neural network or something different (e.g., [442]).
2) A conversion of non-neural network models into functionally equivalent neural networks, or '*neuralization*', so that existing approaches for explaining neural networks, e.g. LRP [441], can be applied [322], [444].

In the following, we demonstrate a neuralization approach. It has been shown that numerous anomaly detection models, in particular kernel-based models such as KDE or one-class SVMs, can be rewritten as strictly equivalent neural networks [322], [444]. Examples of neuralized models are shown in Fig. 8. They typically organize into a 3-layer architecture, from left to right: feature extraction, distance computation, and pooling.

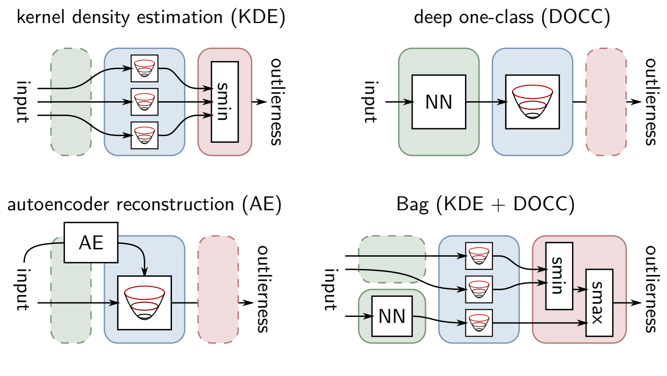

Fig. 8. Three-stage neural network architecture that can be used to formulate in a strictly equivalent manner a variety of non-neural-network anomaly detection models.

For example, the KDE model, usually expressed as $f(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^{n} \exp(-\|\boldsymbol{x}-\boldsymbol{x}_i\|^2)$, can have its negative log-likelihood

$s(\boldsymbol{x}) = -\log f(\boldsymbol{x})$ rewritten as a two-layer network:

$$h_j = \|\boldsymbol{x} - \boldsymbol{x}_j\|^2 + \log n \qquad \text{(layer 1)}$$
$$s(\boldsymbol{x}) = \text{smin}_j\{h_j\} \qquad \text{(layer 2)}$$

where *smin* is a soft min-pooling of the type logsumexp.

Once the model has been converted into a neural network, we can apply explanation techniques such as LRP [441] to produce an explanation of the anomaly prediction. In this case, the LRP algorithm will take the score at the output of the model, propagate to 'winners' in the pool, then assign the score to directions in the input or feature space that contribute the most to the distance, and if necessary propagate the signal further down the feature hierarchy (cf., the Supplement of [322] for how this is done exactly).

Fig. 9 shows from left to right an anomaly from the MNIST-C dataset, the ground-truth explanation (the squared difference between the digit before and after corruption) as well as LRP explanations for three anomaly detection models (KDE, DOCC, and AE).

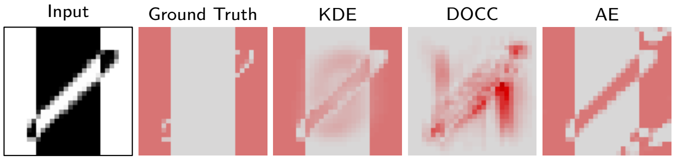

Fig. 9. Explaining anomaly prediction: Highlighting the input features that are most relevant for the prediction helps to understand the model's decision strategy, here on MNIST-C:Stripes.

Although all models predict accurately on the stripe data, the strategies are very different: The kernel density estimator highlights the anomaly, but also some regions of the digit itself. The deep one-class classifier strongly emphasizes vertical edges. The autoencoder produces a result similar to KDE but with decision artifacts in the corners of the image and on the digit itself.

From these observations, it is clear that each model, although predicting with 100% accuracy on the current data, will have different generalization properties and vulnerabilities when encountering subsequent anomalies. (In section VIII-B we will work through an example showing how explanations can help to diagnose and improve a detection model.)

To conclude, we emphasize that a standard quantitative evaluation can be imprecise or even misleading when the available data is not fully representative, and in that case, explanations can be produced to more comprehensively assess the quality of an anomaly detection model.

VIII. Worked-Through Examples

In this section, we work through two specific, real-world examples to exemplify the modeling and evaluation process and provide some best practices.

A. Example 1: Thyroid Disease Detection

In the first example our goal is to learn a model to detect thyroid gland dysfunctions such as hyperthyroidism. The Thyroid dataset[4] includes n = 3772 data instances and has D = 6 real-valued features. It contains a total of 93 ($\sim$2.5%) anomalies. For a quantitative evaluation, we consider a dataset split of 60:10:30 corresponding to the training, validation, and test sets respectively, while preserving the ratio of $\sim$2.5% anomalies in each of the sets.

We choose the OC-SVM [6] with standard RBF kernel $k(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \exp(-\gamma\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|^2)$ as a method for this task since the data is real-valued, low-dimensional, and the OC-SVM scales sufficiently well for this comparatively small dataset. In addition, the ν-parameter formulation (cf., Eq. (20)) enables us to use our prior knowledge and thus approximately control the false alarm rate α and with it implicitly also the miss rate, which leads to our first recommendation:

Assess the risks of false alarms and missed anomalies

Calibrating the false alarm rate and miss rate of a detection model can decide over life and death in a medical context such as disease detection. Though the consequences must not always be as dramatic as in a medical setting, it is important to carefully consider the risks and costs involved with type I and type II errors in advance. In our example, a false alarm would suggest a thyroid dysfunction although the patient is healthy. On the other hand, a missed alarm would occur if the model recognizes a patient with a dysfunction as healthy. Such asymmetric risks, with a greater expected loss for anomalies that go undetected, are very common in medical diagnosis [446]–[449]. Given only D = 6 measurements per data record, we therefore seek to learn a detector with a miss rate ideally close to zero, at the cost of an increased false alarm rate. Patients falsely ascribed with a dysfunction by such a detector could then undergo further, more elaborate clinical testing to verify the disease. Assuming our data is representative and $\sim$12%[5] of the population is at risk of thyroid dysfunction, we choose a slightly higher $\nu = 0.15$ to further increase the robustness against potential data contamination (here the training set contains $\sim$2.5% contamination in the form of unlabeled anomalies). We then train the model and choose the kernel scale γ according to the best AUC we observe on the small, labeled validation set which includes 9 labeled anomalies. We select γ from $\gamma \in \{(2^i D)^{-1} \,|\, i = -5, \ldots, 5\}$, that is from a $\log_2$ span that accounts for the dimensionality D.

Following the above, we observe a rather poor best validation set AUC of 83.9% at $\gamma = (2^{-5}D)^{-1}$, which is the largest value from the hyperparameter range. This is an indication that we forgot an important preprocessing step, namely:

Apply feature scaling to normalize value ranges

Any method, including kernel methods, that relies on computing distances requires the features to be scaled to similar ranges to prevent features with wider value ranges from dominating the computed distances. If this is not done, it can cause anomalies that deviate on smaller scale features to be undetected. Similar reasoning also holds for clustering and classification (see e.g. the discussion in [450]). Min-max normalization or standardization are common choices,

[4] Available from the ODDS Library [430] at http://odds.cs.stonybrook.edu/

[5] https://www.thyroid.org/

but since we assume there may be some contamination, we apply a robust feature scaling via the median and interquartile range. Remember that scaling parameters should be computed using only information from the training data and then applied to all of the data. After we have scaled the features, we observe a much improved best validation set AUC of 98.6% at $\gamma = (2^2 D)^{-1}$. The so-trained and selected model finally achieves a test set AUC of 99.2%, a false alarm rate of 14.8% (i.e., close to our a priori specified $\nu = 0.15$), and a miss rate of zero.

B. Example 2: MVTec Industrial Inspection

For our second example, we consider the task of detecting anomalies in wood images drawn from the MVTec-AD dataset. Unlike the first worked-through example, the MVTec data is high-dimensional and corresponds to arrays of pixel values. Hence, all input features are already on a similar scale (between -1 and $+1$) and therefore we do not need to apply feature rescaling.

Following the standard model training / validation procedure, we train a set of models on the training data, select their hyperparameters on hold out data (e.g., a few inliers and anomalies extracted from the test set), and then evaluate their performance on the remaining part of the test set. The AUC performance of the nine models in our benchmark is shown in Table VI.

TABLE VI
AUC DETECTION PERFORMANCE ON THE MVTEC-AD 'WOOD' CLASS.

Gauss	MVE	PCA	KDE	SVDD	kPCA	AGAN	DOCC	AE
54.0	80.1	90.4	**94.7**	94.1	90.6	74.5	91.6	88.5

We observe that the best performing model is the kernel density estimation (KDE). This is particularly surprising, because this model does not compute the kinds of higher-level image features that deep models, such as DOCC, learn and apply. Examination of the data set shows that the anomalies involve properties such as small perforations and stains that do not require high-level semantic information to be detected. But is that the only reason why the KDE performance is so high? In order to get insight into the strategy used by KDE to arrive at its prediction, we employ the neuralization/LRP approach presented in section VII-D.

Apply XAI to analyze model predictions

Fig. 10 shows an example of an image along with its ground-truth pixel-level anomaly as well as the computed pixel-wise explanation for KDE.

Ideally, we would like the model to make its decision based on the actual anomaly (here, the three drill holes), and therefore, we would expect the ground-truth annotation and the KDE explanation to coincide. However, it is clear from inspection of the explanation that KDE is *not* looking at the true cause of the anomaly and is looking instead at the vertical stripes present everywhere in the input image. This discrepancy between the explanation and the ground truth can be observed on other images of the 'wood' class. The high AUC score of KDE thus must be due to a spurious correlation in the test set between the reaction of the model to these stripes and the presence of anomalies. We call this a 'Clever Hans' effect [244], because just like the horse Clever Hans, the model appears to work because of a spurious correlation. Obviously the KDE model is unlikely to generalize well when the anomalies and the stripes become decoupled (e.g., as we observe more data or under some adversarial manipulation). This illustrates the importance of generating explanations to identify these kinds of failures. Once we have identified the problem, how can we change our anomaly detection strategy so that it is more robust and generalizes better?

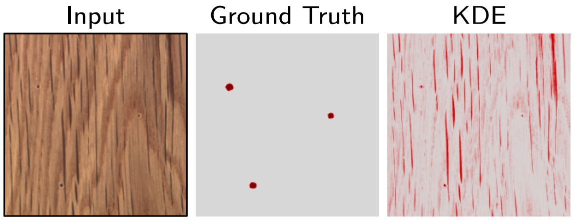

Fig. 10. Input image, ground-truth source of anomaly (here, three drill holes), and explanation of the KDE anomaly prediction. The KDE model assigns high relevance to the wood strains instead of the drill holes. This discrepancy between ground truth and model explanation reveals a 'Clever Hans' strategy used by the KDE model.

Improve the model based on explanations

In practice, there are various approaches to improve the model based on explanation feedback:

1) *Data extension:* We can extend the data with missing training cases, e.g., anomalous wood examples that lack stripes or normal wood examples that have stripes to break to spurious correlation between stripes and anomalies. When further data collection is not possible, synthetic data extension schemes such as blurring or sharpening can also be considered.
2) *Model extension:* If the first approach is not sufficient, or if the model is simply not capable of implementing the necessary prediction structure, the model itself can be changed (e.g., using a more flexible deep model). In other cases, the model may have enough representation power but is statistically inefficient (e.g., subject to the curse of dimensionality). In that case, adding structure (e.g., convolutions) or regularization can also help to learn a model with an appropriate prediction strategy.
3) *Ensembles:* If all considered models have their own strengths and weaknesses, ensemble approaches can be considered. Ensembles have a conceptual justification in the context of anomaly detection [322], and they have been shown to work well empirically [451], [452].

Once the model has been improved based on these strategies, explanations can be recomputed and examined to verify that the decision strategy has been corrected. If that is not the case, the process can be iterated until we reach a satisfactory model.

IX. Concluding Remarks, Open Challenges, and Future Research Paths

Anomaly detection is a blossoming field of broad theoretical and practical interest across the disciplines. In this work, we have given a review of the past and present state of anomaly detection research, established a systematic unifying view, and discussed many practical aspects. While we have included some of our own contributions, we hope that we have fulfilled our aim of providing a balanced and comprehensive snapshot of this exciting research field. Focus was given to a solid theoretical basis, which then allowed us put today's two main lines of development into perspective: the more classical kernel world and the more recent world of deep learning and representation learning for anomaly detection.

We will conclude our review by turning to what lies ahead. Below, we highlight some critical open challenges — of which there are many — and identify a number of potential avenues for future research that we hope will provide useful guidance.

A. Unexplored Combinations of Modeling Dimensions

As can be seen in Fig. 1 and Table II, there is a zoo of different anomaly detection algorithms that have historically been explored along various dimensions. This review has shown conceptual similarities between anomaly detection members from kernel methods and deep learning. Note, however, that the exploration of novel algorithms has been substantially different in both domains, which offers unique possibilities to explore new methodology: steps that have been pursued in kernel learning but not in deep anomaly detection could be transferred (or vice versa) and powerful new directions could emerge. In other words, ideas could be readily transferred from kernels to deep learning and back, and novel combinations in our unified view in Fig. 1 would emerge.

Let us now discuss some specific opportunities to clarify this point. Consider the problem of robustness to noise and contamination. For shallow methods, the problem is well studied, and we have many effective methods [5], [253], [316], [359], [361], [362]. In deep anomaly detection, very little work has addressed this problem. A second example is the application of Bayesian methods. Bayesian inference has been mostly considered for shallow methods [315], [350], owing to the prohibitive cost or intractability of exact Bayesian inference in deep neural networks. Recent progress in approximate Bayesian inference and Bayesian neural networks [403], [453]–[456] raise the possibility of developing methods that complement anomaly scores with uncertainty estimates or uncertainty estimates of their respective explanations [457]. In the area of semi-supervised anomaly detection, ideas have already been successfully transferred from kernel learning [183], [228] to deep methods [143] for one-class classification. But probabilistic and reconstruction methods that can make use of labeled anomalies are unexplored. For time series anomaly detection [169], [200]–[202], where forecasting (i.e., conditional density estimation) models are practical and widely deployed, semi-supervised extensions of such methods could lead to significant improvements in applications in which some labeled examples are available (e.g., learning from failure cases in monitoring tasks). Concepts from density ratio estimation [458] or noise contrastive estimation [459] could lead to novel semi-supervised methods in principled ways. Finally, active learning strategies for anomaly detection [334]–[337], which identify informative instances for labeling, have primarily only been explored for shallow detectors and could be extended to deep learning approaches.

This is a partial list of opportunities that we have noticed. Further analysis of our framework will likely expose additional directions for innovation.

B. Bridging Related Lines of Research on Robustness

Other recent lines of research on robust deep learning are closely related to anomaly detection or may even be interpreted as special instances of the problem. These include out-of-distribution detection, model calibration, uncertainty estimation, and adversarial examples or attacks. Bridging these lines of research by working out the nuances of the specific problem formulations can be insightful for connecting concepts and transferring ideas to jointly advance research.

A basic approach to creating robust classifiers is to endow them with the ability to reject input objects that are likely to be misclassified. This is known as the problem of *classification with a reject option*, and it has been studied extensively [460]–[466]. However, this work focuses on objects that fall near the decision boundary where the classifier is uncertain.

One approach to making the rejection decision is to calibrate the classification probabilities and then reject objects for which no class is predicted to have high probability following Chow's optimal rejection rule [461]. Consequently, many researchers have developed techniques for calibrating the probabilities of classifiers [454], [467]–[472] or for Bayesian uncertainty quantification [402], [403], [453], [455], [456], [473].

Recent work has begun to address other reasons for rejecting an input object. *Out-of-distribution (OOD) detection* considers cases where the object is drawn from a distribution different from the training distribution $\mathbb{P}^+$ [470], [472], [474]–[477]. From a formal standpoint, it is impossible to determine whether an input x is drawn from one of two distributions $\mathbb{P}_1$ and $\mathbb{P}_2$ if both distributions have support at x. Consequently, the OOD problem reduces to determining whether x lies outside regions of high density in $\mathbb{P}^+$, which is exactly the anomaly detection problem we have described in this review.

A second reason to reject an input object is because it belongs to a class that was not part of the training data. This is the problem of *open set recognition*. Such objects can also be regarded as being generated by a distribution $\mathbb{P}^-$, so this problem also fits within our framework and can be addressed with the algorithms described here. Nonetheless, researchers have developed a separate set of methods for open set recognition [238], [478]–[481], and an important goal for future research is to evaluate these methods from the anomaly detection perspective and to evaluate anomaly detection algorithms from the open set perspective.

In rejection, out-of-distribution, and open set recognition problems, there is an additional source of information that is not available in standard anomaly detection problems: the class labels of the objects. Hence, the learning task combines classification with anomaly detection. Formally, the goal is to train a classifier on labeled data $(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_n, y_n)$ with class labels $y \in \{1, \dots, k\}$ while also developing some measure to decide whether an unlabeled test point $\tilde{\boldsymbol{x}}$ should be rejected (for any of the reasons listed above). The class label information tells us about the structure of $\mathbb{P}^+$ and allows us to model it as a joint distribution $\mathbb{P}^+ \equiv \mathbb{P}_{X,Y}$. Methods for rejection, out-of-distribution, and open set recognition all take advantage of this additional structure. Note that the labels y are different from the labels that mark normal or anomalous points in supervised or semi-supervised anomaly detection (cf., section II-C).

Research on the unresolved and fundamental issue of adversarial examples and attacks [482]–[491] is related to anomaly detection as well. We may interpret adversarial attacks as extremely hard-to-detect out-of-distribution samples [454], as they are specifically crafted to target the decision boundary and confidence of a learned classifier. Standard adversarial attacks find a small perturbation δ for an input $\boldsymbol{x}$ so that $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \delta$ yields some class prediction desired by the attacker. For instance, a perturbed image of a dog may be indistinguishable from the original to the human's eye, yet the predicted label changes from 'dog' to 'cat'. Note that such an adversarial example $\tilde{\boldsymbol{x}}$ still likely is (and probably should) be normal under the data marginal $\mathbb{P}_X$ (an imperceptibly perturbed image of a dog shows a dog after all!) but the pair $(\tilde{\boldsymbol{x}}, \text{'cat'})$ should be anomalous under the joint $\mathbb{P}_{X,Y}$ [492]. Methods for OOD detection have been found to also increase adversarial robustness [154], [454], [477], [493], [494], some of which model the class conditional distributions for detection [476], [492], for the reason just described.

The above highlights the connection of these lines of research towards the general goal of robust deep models. Hence, we believe that connecting ideas and concepts in these lines (e.g., the use of spherical losses in both anomaly detection [136], [156] and OOD [493], [495]) may help them to advance together. Finally, the assessment of the robustness of neural networks and their fail-safe design and integration are topics of high practical relevance that have recently found their way in international standardization initiatives (e.g., ITU/WHO FG-AI4H, ISO/IEC CD TR 24029-1, or IEEE P7009). Beyond doubt, understanding the brittleness of deep networks (also in the context of their explanations [496]) will also be critical for their adoption in anomaly detection applications that involve malicious attackers such as fraudsters or network intruders.

C. Interpretability and Trustworthiness

Much of anomaly detection research has been devoted to developing new methods that improve detection accuracy. In most applications, however, accuracy alone is not sufficient [322], [497], and further criteria such as interpretability (e.g., [243], [498]) and trustworthiness [456], [499], [500] are equally critical as demonstrated in sections VII and VIII. For researchers and practitioners alike [501] it is vital to understand the underlying reasons for how a specific anomaly detection model reaches a particular prediction. Interpretable, explanatory feedback enhances model transparency, which is indispensable for accountable decision-making [502], for uncovering model failures such as Clever Hans behavior [244], [322], and for understanding model vulnerabilities that can be insightful for improving a model or system. This is especially relevant in safety-critical environments [503], [504]. Existing work on interpretable anomaly detection has considered finding subspaces of anomaly-discriminative features [442], [505]–[509], deducing sequential feature explanations [443], the use of feature-wise reconstruction errors [57], [189], utilizing fully convolutional architectures [326], integrated gradients [38], and explaining anomalies via LRP [322], [444]. In relation to the vast body of literature though, research on interpretability and trustworthiness in anomaly detection has seen comparatively little attention. The fact that anomalies may not share similar patterns (i.e., the heterogeneity of anomalies) poses a challenge for their explanation, which also distinguishes this setting from interpreting supervised classification models. Furthermore, anomalies might arise due to the presence of abnormal patterns, but conversely also due to a lack of normal patterns. While for the first case an explanation that highlights the abnormal features is satisfactory, how should an explanation for missing features be conceptualized? For example given the MNIST dataset of digits, what should an explanation of an anomalous all-black image be? The matters of interpretability and trustworthiness get more pressing as the task and data become more complex. Effective solutions of complex tasks will necessarily require more powerful methods, for which explanations become generally harder to interpret. We thus believe that future research in this direction will be imperative.

D. The Need for Challenging and Open Datasets

Challenging problems with clearly defined evaluation criteria on publicly available benchmark datasets are invaluable for measuring progress and moving a field forward. The significance of the ImageNet database [510], together with corresponding competitions and challenges [511], for progressing computer vision and supervised deep learning in the last decade give a prime example of this. Currently, the standard evaluation practices in deep anomaly detection [129], [134], [136], [140], [143], [148], [153]–[156], [325], [512] out-of-distribution detection [269], [470], [474]–[477], [513], [514], and open set recognition [238], [478]–[481] still extensively repurpose classification datasets by deeming some dataset classes to be anomalous or considering in-distribution vs. out-of-distribution dataset combinations (e.g., training a model on Fashion-MNIST clothing items and regarding MNIST digits to be anomalous). Although these synthetic protocols have some value, it has been questioned how well they reflect real progress on challenging anomaly detection tasks [199], [320]. Moreover, we think the tendency that only few methods seem to dominate most of the benchmark datasets in the work cited above is alarming, since it suggests a bias towards evaluating only the upsides of newly proposed methods, yet often critically leaving out an analysis of their downsides and limitations. This situation suggests a lack of diversity in the current evaluation practices and the benchmarks being used. In the spirit of *all models are wrong* [515], we stress that more research effort should go into studying when and how certain models are wrong and behave like Clever Hanses. We need to understand the trade-offs that different methods make. For example, some methods are likely making a trade-off between detecting low-level vs. high-level, semantic anomalies (cf., section II-B2 and [199]). The availability of more diverse and challenging datasets would be of great benefit in this regard. Recent datasets such as MVTec-AD [189] and competitions such as the Medical Out-of-Distribution Analysis Challenge [422] provide excellent examples, but the field needs many more challenging open datasets to foster progress.

E. Weak Supervision and Self-Supervised Learning

The bulk of anomaly detection research has been studying the problem in absence of any kind of supervision, that is, in an unsupervised setting (cf., section II-C2). Recent work suggests, however, that significant performance improvements on complex detection tasks seem achievable through various forms of weak supervision and self-supervised learning.

Weak supervision or *weakly supervised learning* describes learning from imperfectly or scarcely labeled data [516]–[518]. Labels might be inaccurate (e.g., due to labeling errors or uncertainty) or incomplete (e.g., covering only few normal modes or specific anomalies). Current work on semi-supervised anomaly detection indicates that including even only few labeled anomalies can already yield remarkable performance improvements on complex data [61], [143], [320], [324], [326], [519]. A key challenge here is to formulate and optimize such methods so that they generalize well to novel anomalies. Combining these semi-supervised methods with active learning techniques helps identifying informative candidates for labeling [334]–[337]. It is an effective strategy for designing anomaly detection systems that continuously improve via expert feedback loops [443], [520]. This approach has not yet been explored for deep detectors, though. Outlier exposure [325], that is, using massive amounts of data that is publicly available in some domains (e.g., stock photos for computer vision or the English Wikipedia for NLP) as auxiliary negative samples (cf., section IV-E), can also be viewed as a form of weak supervision (imperfectly labeled anomalies). Though such negative samples may not coincide with ground-truth anomalies, we believe such contrasting can be beneficial for learning characteristic representations of normal concepts in many domains (e.g., using auxiliary log data to well characterize the normal logs of a specific computer system [521]). So far, this has been little explored in applications. Transfer learning approaches to anomaly detection also follow the idea of distilling more domain knowledge into a model, for example, through using and possibly fine-tuning pre-trained (supervised) models [138], [141], [322], [410], [522]. Overall, weak forms of supervision or domain priors may be essential for achieving effective solutions in semantic anomaly detection tasks that involve high-dimensional data, as has also been found in other unsupervised learning tasks such as disentanglement [209], [523], [524]. Hence, we think that developing effective methods for weakly supervised anomaly detection will contribute to advancing the state of the art.

Self-supervised learning describes the learning of representations through solving auxiliary tasks, for example, next sentence and masked words prediction [111], future frame prediction in videos [525], or the prediction of transformations applied to images [526] such as colorization [527], cropping [528], [529], or rotation [530]. These auxiliary prediction tasks do not require (ground-truth) labels for learning and can thus be applied to unlabeled data, which makes self-supervised learning particularly appealing for anomaly detection. Self-supervised methods that have been introduced for visual anomaly detection train multi-class classification models based on pseudo labels that correspond to various geometric transformations (e.g., flips, translations, rotations, etc.) [153]–[155]. An anomaly score can then be derived from the softmax activation statistics of a so-trained classifier, assuming that a high prediction uncertainty (close to a uniform) indicates anomalies. These methods have shown significant performance improvements on the common k-classes-out image benchmarks (cf., Table III). Bergman and Hoshen [156] have recently proposed a generalization of this idea to non-image data, called GOAD, which is based on random affine transformations. We can identify GOAD and self-supervised methods based on geometric transformations (GT) as classification-based approaches within our unifying view (cf., Table II). In a broader context, the interesting question will be to what extent self-supervision can facilitate the learning of semantic representations. There is some evidence that self-supervised learning helps to improve the detection of semantic anomalies and thus exhibits inductive biases towards semantic representations [199]. On the other hand, there also exists evidence showing that self-supervision mainly improves learning of effective feature representations for low-level statistics [531]. Hence, this research question remains to be answered, but bears great potential for many domains where large amounts of unlabeled data are available.

F. Foundation and Theory

The recent progress in anomaly detection research has also raised more fundamental questions. These include open questions about the out-of-distribution generalization properties of various methods presented in this review, the definition of anomalies in high-dimensional spaces, and information-theoretic interpretations of the problem.

Nalisnick et al. [269] have recently observed that deep generative models (DGMs) such as normalizing flows, VAEs, or autoregressive models (cf., section III) can often assign higher likelihood to anomalies than to in-distribution samples. For example, models trained on Fashion-MNIST clothing items can systematically assign higher likelihood to MNIST digits [269]. This counter-intuitive finding, which has been replicated in subsequent work [149], [260], [325], [513], [514], [532], revealed that there is a critical lack of theoretical understanding of these models. Solidifying evidence [513], [514], [533], [534] indicates that one reason seems to be that the likelihood in current DGMs is still largely biased towards low-level background statistics. Consequently, simpler data points attain higher likelihood (e.g., MNIST digits under models trained on Fashion-MNIST, but not vice versa). Another critical remark in this context is that for (truly) high-dimensional data, the region with highest likelihood must not necessarily coincide with the region of highest probability mass (called the 'typical set'), that is, the region where data points most likely occur [532]. For instance, while the highest density of a D-dimensional standard Gaussian is given at the origin, points sampled from the distribution concentrate around an annulus with radius $\sqrt{D}$ for large D [535]. Therefore, points close to the origin have high density, but are very unlikely to occur. This mismatch questions the standard theoretical density (level set) problem formulation (cf., section II-B) and use of likelihood-based anomaly detectors for some settings. Hence, theoretical research aimed at understanding the above phenomenon and DGMs themselves presents an exciting research opportunity.

Similar observations suggest that reconstruction-based models can systematically well reconstruct simpler out-of-distribution points that sit within the convex hull of the data. For example, an anomalous all-black image can be well reconstructed by an autoencoder trained on MNIST digits [536]. An even simpler example is the perfect reconstruction of points the lie within the linear subspace spanned by the principal components of a PCA model, even in regions far away from the normal training data (e.g., along the principal component in Fig. 7). While such out-of-distribution generalization properties might be desirable for representation learning in general [537], such behavior critically can be undesirable for anomaly detection. Therefore, we stress that more theoretical research on understanding such out-of-distribution generalization properties or biases, especially for more complex models, will be necessary.

Finally, the push towards deep learning also presents new opportunities to interpret and analyze the anomaly detection problem from different theoretical angles. For example, from the perspective of information theory [538], autoencoders can be understood as adhering to the *Infomax principle* [539]–[541] by implicitly maximizing the mutual information between the input and the latent code — subject to structural constraints or regularization of the code (e.g., 'bottleneck', latent prior, sparsity, etc.) — via the reconstruction objective [378]. Similarly, information-theoretic perspectives of VAEs have been formulated showing that these models can be viewed as making a rate-distortion trade-off [542] when balancing the latent compression (negative rate) and reconstruction accuracy (distortion) [543], [544]. This view has recently been employed to draw a connection between VAEs and Deep SVDD, where the latter can be seen as a special case that only seeks to minimize the rate (maximize compression) [545]. Overall, anomaly detection has been studied comparatively less from an information-theoretic perspective [546], [547], yet we think this could be fertile ground for building a better theoretical understanding of representation learning for anomaly detection.

Concluding, we firmly believe that anomaly detection in all its exciting variants will also in the future remain an indispensable practical tool in the quest to obtain robust learning models that perform well on complex data.

Appendix A
Notation and Abbreviations

For reference, we provide the notation and abbreviations used in this work in Tables VII and VIII respectively.

TABLE VII
NOTATION CONVENTIONS

Symbol	Description
$\mathbb{N}$	The natural numbers
$\mathbb{R}$	The real numbers
D	The input data dimensionality $D \in \mathbb{N}$
$\mathcal{X}$	The input data space $\mathcal{X} \subseteq \mathbb{R}^D$
$\mathcal{Y}$	The labels $\mathcal{Y} = \{\pm 1\}$ $(+1$: normal; -1 : anomaly)
$\boldsymbol{x}$	A vector, e.g. a data point $\boldsymbol{x} \in \mathcal{X}$
$\mathcal{D}_n$	An unlabeled dataset $\mathcal{D}_n = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ of size n
$\mathbb{P}, p$	The data-generating distribution and pdf
$\mathbb{P}^+, p^+$	The normal data distribution and pdf
$\mathbb{P}^-, p^-$	The anomaly distribution and pdf
$\hat{p}$	An estimated pdf
ε	An error or noise distribution
$\operatorname{supp}(p)$	The support of a data distribution $\mathbb{P}$ with density p, i.e. $\{\boldsymbol{x} \in \mathcal{X} \mid p(\boldsymbol{x}) > 0\}$
$\mathcal{A}$	The set of anomalies
C_α	An α-density level set
$\hat{C}_\alpha$	An α-density level set estimator
τ_α	The threshold $\tau_\alpha \geq 0$ corresponding to C_α
$c_\alpha(\boldsymbol{x})$	The threshold anomaly detector corresponding to C_α
$s(\boldsymbol{x})$	An anomaly score function $s : \mathcal{X} \to \mathbb{R}$
$\mathbb{1}_A(\boldsymbol{x})$	The indicator function for some set A
$\ell(s, y)$	A loss function $\ell : \mathbb{R} \times \{\pm 1\} \to \mathbb{R}$
$f_\theta(\boldsymbol{x})$	A model $f_\theta : \mathcal{X} \to \mathbb{R}$ with parameters θ
$k(\boldsymbol{x}, \tilde{\boldsymbol{x}})$	A kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$
$\mathcal{F}_k$	The RKHS or feature space of kernel k
$\phi_k(\boldsymbol{x})$	The feature map $\phi_k : \mathcal{X} \to \mathcal{F}_k$ of kernel k
$\phi_\omega(\boldsymbol{x})$	A neural network $\boldsymbol{x} \mapsto \phi_\omega(\boldsymbol{x})$ with weights ω

Appendix B
Details of Training

For PCA, we compute the reconstruction error whilst maintaining 90% of variance of the training data. We do the same for kPCA, and additionally choose the kernel width such that 50% neighbors capture 50% of total similarity scores. For MVE, we use the fast minimum covariance determinant estimator [299] with a default support fraction of 0.9 and a contamination rate parameter of 0.01. To facilitate MVE computation on MVTec-AD, we first reduce the dimensionality via PCA retaining 90% of variance. For KDE, we choose the bandwidth parameter to maximize the likelihood of a small hold-out set from the training data. For SVDD, we consider $\nu \in \{0.01, 0.05, 0.1, 0.2\}$ and select the kernel scale using a small labeled hold-out set. The deep one-class classifier applies a whitening transform on the representations after the first fully-connected layer of a pre-trained VGG16 model (on MVTec-AD) or a CNN classifier trained on the EMNIST letter subset (on MNIST-C). For the AE on MNIST-C, we use a LeNet-type encoder that has two convolutional layers with max-pooling followed by two fully connected layers that map to an encoding of 64 dimensions, and construct the

TABLE VIII
LIST OF ABBREVIATIONS

Abbreviation	Description
AE	Autoencoder
AAE	Adversarial Autoencoder
AUC	Area Under the ROC curve
CAE	Contrastive Autoencoder
DAE	Denoising Autoencoder
DGM	Deep Generative Model
DSVDD	Deep Support Vector Data Description
DSAD	Deep Semi-supervised Anomaly Detection
EBM	Energy Based Model
ELBO	Evidence Lower Bound
GAN	Generative Adversarial Network
GMM	Gaussian Mixture Model
GT	Geometric Transformations
iForest	Isolation Forest
KDE	Kernel Density Estimation
k-NN	k-Nearest Neighbors
kPCA	Kernel Principal Component Analysis
LOF	Local Outlier Factor
LPUE	Learning from Positive and Unlabeled Examples
LSTM	Long short-term memory
MCMC	Markov chain Monte Carlo
MCD	Minimum Covariance Determinant
MVE	Minimum Volume Ellipsoid
OOD	Out-of-distribution
OE	Outlier Exposure
OC-NN	One-Class Neural Network
OC-SVM	One-Class Support Vector Machine
pPCA	Probabilistic Principal Component Analysis
PCA	Principal Component Analysis
pdf	Probability density function
PSD	Positive semidefinite
RBF	Radial basis function
RKHS	Reproducing Kernel Hilbert Space
rPCA	Robust Principal Component Analysis
SGD	Stochastic Gradient Descent
SGLD	Stochastic Gradient Langevin Dynamics
SSAD	Semi-Supervised Anomaly Detection
SVDD	Support Vector Data Description
VAE	Variational Autoencoder
VQ	Vector Quantization
XAI	Explainable AI

decoder symmetrically. On MVTec-AD, we use an encoder-decoder architecture as presented in [130] which maps to a bottleneck of 512 dimensions. Both, the encoder and decoder here consist of four blocks having two 3×3 convolutional layers followed by max-pooling or upsampling respectively. We train the AE such that the reconstruction error of a small training hold-out set is minimized. For AGAN, we use the AE encoder and decoder architecture for the discriminator and generator networks respectively, where we train the GAN until convergence to a stable equilibrium.

ACKNOWLEDGMENTS

LR acknowledges support by the German Federal Ministry of Education and Research (BMBF) in the project ALICE III (01IS18049B). RV, KRM, GM, and WS acknowledge support by the Berlin Institute for the Foundations of Learning and Data (BIFOLD) sponsored by the BMBF. MK acknowledges support by the German Research Foundation (DFG) award KL 2698/2-1 and by the BMBF awards 01IS18051A and 031B0770E. KRM, GM, and WS acknowledge financial support by the BMBF for the Berlin Center for Machine Learning (01IS18037A-I) and under the Grants 01IS14013A-E, 031L0207A-D. KRM also acknowledges financial support by BMBF Grants 01GQ1115 and 01GQ0850; DFG under Grant Math+, EXC 2046/1, Project ID 390685689 and partial support by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea Government (No. 2019-0-00079, Artificial Intelligence Graduate School Program, Korea University). TD acknowledges financial support from the US Defense Advanced Research Projects Agency (DARPA) under Contract Nos. HR001119C0112, FA8750-19-C-0092, and HR001120C0022. Correspondence to MK, TD and KRM.

REFERENCES

[1] K. Pearson, “On lines and planes of closest fit to systems of points in space,” *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[2] H. Hotelling, “Analysis of a complex of statistical variables into principal components.” *Journal of Educational Psychology*, vol. 24, no. 6, pp. 417–441, 1933.

[3] B. Schölkopf, A. Smola, and K.-R. Müller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998.

[4] H. Hoffmann, “Kernel PCA for novelty detection,” *Pattern Recognition*, vol. 40, no. 3, pp. 863–874, 2007.

[5] P. J. Huber and E. M. Ronchetti, *Robust Statistics*, 2nd ed. John Wiley & Sons, 2009.

[6] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, “Estimating the Support of a High-Dimensional Distribution,” *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[7] D. M. J. Tax and R. P. W. Duin, “Support vector data description,” *Machine Learning*, vol. 54, no. 1, pp. 45–66, 2004.

[8] E. M. Knorr, R. T. Ng, and V. Tucakov, “Distance-based outliers: algorithms and applications,” *The VLDB Journal*, vol. 8, no. 3, pp. 237–253, 2000.

[9] S. Ramaswamy, R. Rastogi, and K. Shim, “Efficient algorithms for mining outliers from large data sets,” in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2000, pp. 427–438.

[10] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, “LOF: Identifying density-based local outliers,” in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2000, pp. 93–104.

[11] M. Rosenblatt, “Remarks on some nonparametric estimates of a density function,” *The Annals of Mathematical Statistics*, vol. 27, no. 3, pp. 832–837, 1956.

[12] E. Parzen, “On estimation of a probability density function and mode,” *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[13] F. Y. Edgeworth, “On discordant observations,” *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 23, no. 5, pp. 364–375, 1887.

[14] T. S. Kuhn, *The Structure of Scientific Revolutions*. Chicago: University of Chicago Press, 1970.

[15] A. Patcha and J.-M. Park, “An overview of anomaly detection techniques: Existing solutions and latest technological trends,” *Computer Networks*, vol. 51, no. 12, pp. 3448–3470, 2007.

[16] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, and K.-Y. Tung, “Intrusion detection system: A comprehensive review,” *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 16–24, 2013.

[17] M. Ahmed, A. N. Mahmood, and J. Hu, “A survey of network anomaly detection techniques,” *Journal of Network and Computer Applications*, vol. 60, pp. 19–31, 2016.

[18] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, “A survey of deep learning-based network anomaly detection,” *Cluster Computing*, vol. 10, pp. 1–13, 2017.

[19] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, “Machine learning and deep learning methods for cybersecurity,” *IEEE Access*, vol. 6, pp. 35 365–35 381, 2018.

[20] R. K. Malaiya, D. Kwon, J. Kim, S. C. Suh, H. Kim, and I. Kim, "An empirical evaluation of deep learning for network anomaly detection," in *International Conference on Computing, Networking and Communications*, 2018, pp. 893–898.

[21] R. J. Bolton and D. J. Hand, "Statistical fraud detection: A review," *Statistical Science*, vol. 17, no. 3, pp. 235–255, 2002.

[22] S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C. Westland, "Data mining for credit card fraud: A comparative study," *Decision Support Systems*, vol. 50, no. 3, pp. 602–613, 2011.

[23] H. Joudaki, A. Rashidian, B. Minaei-Bidgoli, M. Mahmoodi, B. Geraili, M. Nasiri, and M. Arab, "Using data mining to detect health care fraud and abuse: A review of literature," *Global Journal of Health Science*, vol. 7, no. 1, pp. 194–202, 2015.

[24] M. Ahmed, A. N. Mahmood, and M. R. Islam, "A survey of anomaly detection techniques in financial domain," *Future Generation Computer Systems*, vol. 55, pp. 278–288, 2016.

[25] A. Abdallah, M. A. Maarof, and A. Zainal, "Fraud detection system: A survey," *Journal of Network and Computer Applications*, vol. 68, pp. 90–113, 2016.

[26] G. van Capelleveen, M. Poel, R. M. Mueller, D. Thornton, and J. van Hillegersberg, "Outlier detection in healthcare fraud: A case study in the medicaid dental domain," *International Journal of Accounting Information Aystems*, vol. 21, pp. 18–31, 2016.

[27] Y.-J. Zheng, X.-H. Zhou, W.-G. Sheng, Y. Xue, and S.-Y. Chen, "Generative adversarial network based telecom fraud detection at the receiving bank," *Neural Networks*, vol. 102, pp. 78–86, 2018.

[28] J. Rabatel, S. Bringay, and P. Poncelet, "Anomaly detection in monitoring sensor data for preventive maintenance," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7003–7015, 2011.

[29] J. Marzat, H. Piet-Lahanier, F. Damongeot, and E. Walter, "Model-based fault diagnosis for aerospace systems: a survey," in *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 226, no. 10, 2012, pp. 1329–1360.

[30] L. Martí, N. Sanchez-Pi, J. M. Molina, and A. C. B. Garcia, "Anomaly detection based on sensor data in petroleum industry applications," *Sensors*, vol. 15, no. 2, pp. 2774–2797, 2015.

[31] W. Yan and L. Yu, "On accurate and reliable anomaly detection for gas turbine combustors: A deep learning approach," in *Annual Conference of the Prognostics and Health Management Society*, vol. 6, no. 25, 2015.

[32] F. Lopez, M. Saez, Y. Shao, E. C. Balta, J. Moyne, Z. M. Mao, K. Barton, and D. Tilbury, "Categorization of anomalies in smart manufacturing systems to support the selection of detection mechanisms," *Robotics and Automation Letters*, vol. 2, no. 4, pp. 1885–1892, 2017.

[33] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, "Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding," in *International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 387–395.

[34] D. J. Atha and M. R. Jahanshahi, "Evaluation of deep learning approaches based on convolutional neural networks for corrosion detection," *Structural Health Monitoring*, vol. 17, no. 5, pp. 1110–1128, 2018.

[35] D. Ramotsoela, A. Abu-Mahfouz, and G. Hancke, "A survey of anomaly detection in industrial wireless sensor networks with critical water system infrastructure as a case study," *Sensors*, vol. 18, no. 8, p. 2491, 2018.

[36] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical Systems and Signal Processing*, vol. 115, pp. 213–237, 2019.

[37] A. Borghesi, A. Bartolini, M. Lombardi, M. Milano, and L. Benini, "Anomaly detection using autoencoders in high performance computing systems," in *AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 9428–9433.

[38] J. Sipple, "Interpretable , multidimensional , multimodal anomaly detection with negative sampling for detection of device failure," in *International Conference on Machine Learning*, 2020, pp. 4368–4377.

[39] K. Golmohammadi and O. R. Zaiane, "Time series contextual anomaly detection for detecting market manipulation in stock market," in *IEEE International Conference on Data Science and Advanced Analytics*, 2015, pp. 1–10.

[40] ——, "Sentiment analysis on twitter to improve time series contextual anomaly detection for detecting stock market manipulation," in *International Conference on Big Data Analytics and Knowledge Discovery*, 2017, pp. 327–342.

[41] A. Rabaoui, M. Davy, S. Rossignol, and N. Ellouze, "Using one-class svms and wavelets for audio surveillance," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 4, pp. 763–775, 2008.

[42] E. Marchi, F. Vesperini, F. Eyben, S. Squartini, and B. Schuller, "A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional lstm neural networks," in *International Conference on Acoustics, Speech, and Signal Processing*, 2015, pp. 1996–2000.

[43] H. Lim, J. Park, and Y. Han, "Rare sound event detection using 1d convolutional recurrent neural networks," in *Workshop on Detection and Classification of Acoustic Scenes and Events*, 2017, pp. 80–84.

[44] E. Principi, F. Vesperini, S. Squartini, and F. Piazza, "Acoustic novelty detection with adversarial autoencoders," in *International Joint Conference on Neural Networks*, 2017, pp. 3324–3330.

[45] Y. Koizumi, S. Saito, H. Uematsu, Y. Kawachi, and N. Harada, "Unsupervised detection of anomalous sound based on deep learning and the neyman–pearson lemma," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 1, pp. 212–224, 2018.

[46] L. Tarassenko, P. Hayton, N. Cerneaz, and M. Brady, "Novelty detection for the identification of masses in mammograms," in *International Conference on Artificial Neural Networks*, 1995, pp. 442–447.

[47] W.-K. Wong, A. W. Moore, G. F. Cooper, and M. M. Wagner, "Bayesian network anomaly pattern detection for disease outbreaks," in *International Conference on Machine Learning*, 2003, pp. 808–815.

[48] S. Chauhan and L. Vig, "Anomaly detection in ECG time signals via deep long short-term memory networks," in *IEEE International Conference on Data Science and Advanced Analytics*, 2015, pp. 1–7.

[49] C. Leibig, V. Allken, M. S. Ayhan, P. Berens, and S. Wahl, "Leveraging uncertainty information from deep neural networks for disease detection," *Scientific Reports*, vol. 7, no. 1, pp. 1–14, 2017.

[50] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.

[51] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *International Conference on Information Processing in Medical Imaging*, 2017, pp. 146–157.

[52] X. Chen and E. Konukoglu, "Unsupervised detection of lesions in brain MRI using constrained adversarial auto-encoders," in *Medical Imaging with Deep Learning*, 2018.

[53] D. K. Iakovidis, S. V. Georgakopoulos, M. Vasilakakis, A. Koulaouzidis, and V. P. Plagianakos, "Detecting and locating gastrointestinal anomalies using deep learning and iterative cluster unification," *IEEE Transactions on Medical Imaging*, vol. 37, no. 10, pp. 2196–2210, 2018.

[54] S. Latif, M. Usman, R. Rana, and J. Qadir, "Phonocardiographic sensing using deep learning for abnormal heartbeat detection," *IEEE Sensors Journal*, vol. 18, no. 22, pp. 9393–9400, 2018.

[55] N. Pawlowski, M. C. Lee, M. Rajchl, S. McDonagh, E. Ferrante, K. Kamnitsas, S. Cooke, S. Stevenson, A. Khetani, T. Newman, F. Zeiler, R. Digby, J. P. Coles, D. Rueckert, D. K. Menon, V. F. J. Newcombe, and B. Glocker, "Unsupervised lesion detection in brain CT using bayesian convolutional autoencoders," in *Medical Imaging with Deep Learning*, 2018.

[56] C. Baur, B. Wiestler, S. Albarqouni, and N. Navab, "Fusing unsupervised and supervised deep learning for white matter lesion segmentation," in *Medical Imaging with Deep Learning*, 2019, pp. 63–72.

[57] T. Schlegl, P. Seeböck, S. M. Waldstein, G. Langs, and U. Schmidt-Erfurth, "f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks," *Medical Image Analysis*, vol. 54, pp. 30–44, 2019.

[58] P. Seeböck, J. I. Orlando, T. Schlegl, S. M. Waldstein, H. Bogunović, S. Klimscha, G. Langs, and U. Schmidt-Erfurth, "Exploiting epistemic uncertainty of anatomy segmentation for anomaly detection in retinal OCT," *IEEE Transactions on Medical Imaging*, vol. 39, no. 1, pp. 87–98, 2019.

[59] P. Guo, Z. Xue, Z. Mtema, K. Yeates, O. Ginsburg, M. Demarco, L. R. Long, M. Schiffman, and S. Antani, "Ensemble deep learning for cervix image selection toward improving reliability in automated cervical precancer screening," *Diagnostics*, vol. 10, no. 7, p. 451, 2020.

[60] L. Naud and A. Lavin, "Manifolds for unsupervised visual anomaly detection," *arXiv preprint arXiv:2006.11364*, 2020.

[61] N. Tuluptceva, B. Bakker, I. Fedulova, H. Schulz, and D. V. Dylov, "Anomaly detection with deep perceptual autoencoders," *arXiv preprint arXiv:2006.13265*, 2020.

[62] W.-K. Wong, A. Moore, G. Cooper, and M. Wagner, "What's strange about recent events (wsare): An algorithm for the early detection of

disease outbreaks," *Journal of Machine Learning Research*, vol. 6, no. Dec, pp. 1961–1998, 2005.

[63] R. Blender, K. Fraedrich, and F. Lunkeit, "Identification of cyclone-track regimes in the north atlantic," *Quarterly Journal of the Royal Meteorological Society*, vol. 123, no. 539, pp. 727–741, 1997.

[64] J. Verbesselt, A. Zeileis, and M. Herold, "Near real-time disturbance detection using satellite image time series," *Remote Sensing of Environment*, vol. 123, pp. 98–108, 2012.

[65] W. D. Fisher, T. K. Camp, and V. V. Krzhizhanovskaya, "Anomaly detection in earth dam and levee passive seismic data using support vector machines and automatic feature selection," *Journal of Computational Science*, vol. 20, pp. 143–153, 2017.

[66] M. Flach, F. Gans, A. Brenning, J. Denzler, M. Reichstein, E. Rodner, S. Bathiany, P. Bodesheim, Y. Guanche, S. Sippel *et al.*, "Multivariate anomaly detection for earth observations: a comparison of algorithms and feature extraction techniques," *Earth System Dynamics*, vol. 8, no. 3, pp. 677–696, 2017.

[67] Y. Wu, Y. Lin, Z. Zhou, D. C. Bolton, J. Liu, and P. Johnson, "Deepdetect: A cascaded region-based densely connected network for seismic event detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 1, pp. 62–75, 2018.

[68] T. Jiang, Y. Li, W. Xie, and Q. Du, "Discriminative reconstruction constrained generative adversarial network for hyperspectral anomaly detection," *IEEE Transactions on Geoscience and Remote Sensing*, 2020.

[69] T. I. Oprea, "Chemical space navigation in lead discovery," *Current Opinion in Chemical Biology*, vol. 6, no. 3, pp. 384–389, 2002.

[70] P. S. Gromski, A. B. Henson, J. M. Granda, and L. Cronin, "How to explore chemical space using algorithms and automation," *Nature Reviews Chemistry*, vol. 3, no. 2, pp. 119–128, 2019.

[71] S. Min, B. Lee, and S. Yoon, "Deep learning in bioinformatics," *Briefings in Bioinformatics*, vol. 18, no. 5, pp. 851–869, 2017.

[72] S. A. Tomlins, D. R. Rhodes, S. Perner, S. M. Dhanasekaran, R. Mehra, X.-W. Sun, S. Varambally, X. Cao, J. Tchinda, R. Kuefer *et al.*, "Recurrent fusion of TMPRSS2 and ETS transcription factor genes in prostate cancer," *Science*, vol. 310, no. 5748, pp. 644–648, 2005.

[73] R. Tibshirani and T. Hastie, "Outlier sums for differential gene expression analysis," *Biostatistics*, vol. 8, no. 1, pp. 2–8, 2007.

[74] O. Cerri, T. Q. Nguyen, M. Pierini, M. Spiropulu, and J.-R. Vlimant, "Variational autoencoders for new physics mining at the large hadron collider," *Journal of High Energy Physics*, vol. 2019, no. 5, p. 36, 2019.

[75] Y. A. Kharkov, V. Sotskov, A. Karazeev, E. Kiktenko, and A. Fedorov, "Revealing quantum chaos with machine learning," *Physical Review B*, vol. 101, no. 6, p. 064406, 2020.

[76] P. Protopapas, J. Giammarco, L. Faccioli, M. Struble, R. Dave, and C. Alcock, "Finding outlier light curves in catalogues of periodic variable stars," *Monthly Notices of the Royal Astronomical Society*, vol. 369, no. 2, pp. 677–696, 2006.

[77] H. Dutta, C. Giannella, K. Borne, and H. Kargupta, "Distributed top-k outlier detection from astronomy catalogs using the DEMAC system," in *SIAM International Conference on Data Mining*, 2007, pp. 473–478.

[78] M. Henrion, D. J. Mortlock, D. J. Hand, and A. Gandy, "Classification and anomaly detection for astronomical survey data," in *Astrostatistical Challenges for the New Astronomy*. Springer, 2013, pp. 149–184.

[79] E. Reyes and P. A. Estévez, "Transformation based deep anomaly detection in astronomical images," *arXiv preprint arXiv:2005.07779*, 2020.

[80] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[81] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[82] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[83] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[84] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[85] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[86] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[87] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[88] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99.

[89] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2414–2423.

[90] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[91] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[92] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410.

[93] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, "Self-training with noisy student improves imagenet classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 687–10 698.

[94] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Advances in Neural Information Processing Systems*, 2009, pp. 1096–1104.

[95] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, 2011.

[96] A.-r. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 14–22, 2011.

[97] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[98] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *International Conference on Acoustics, Speech, and Signal Processing*, 2013, pp. 6645–6649.

[99] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[100] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen, J. Chen, J. Chen, Z. Chen, M. Chrzanowski, A. Coates, G. Diamos, K. Ding, N. Du, E. Elsen, J. Engel, W. Fang, L. Fan, C. Fougner, L. Gao, C. Gong, A. Hannun, T. Han, L. Johannes, B. Jiang, C. Ju, B. Jun, P. LeGresley, L. Lin, J. Liu, Y. Liu, W. Li, X. Li, D. Ma, S. Narang, A. Ng, S. Ozair, Y. Peng, R. Prenger, S. Qian, Z. Quan, J. Raiman, V. Rao, S. Satheesh, D. Seetapun, S. Sengupta, K. Srinet, A. Sriram, H. Tang, L. Tang, C. Wang, J. Wang, K. Wang, Y. Wang, Z. Wang, Z. Wang, S. Wu, L. Wei, B. Xiao, W. Xie, Y. Xie, D. Yogatama, B. Yuan, J. Zhan, and Z. Zhu, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *International Conference on Machine Learning*, vol. 48, 2016, pp. 173–182.

[101] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing*, 2016, pp. 4960–4964.

[102] J. Chorowski, R. J. Weiss, S. Bengio, and A. van den Oord, "Unsupervised speech representation learning using WaveNet autoencoders," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 2041–2053, 2019.

[103] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," in *Interspeech*, 2019, pp. 3465–3469.

[104] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research*, vol. 3, no. Feb, pp. 1137–1155, 2003.

[105] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.

[106] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Conference on Empirical Methods in Natural Language Processing*, 2014, pp. 1532–1543.

[107] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Conference on Empirical Methods in Natural Language Processing*, 2014, pp. 1724–1734.

[108] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[109] A. Joulin, É. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," in *Conference of the European Chapter of the Association for Computational Linguistics*, 2017, pp. 427–431.

[110] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *North American Chapter of the Association for Computational Linguistics*, 2018, pp. 2227–2237.

[111] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *North American Chapter of the Association for Computational Linguistics*, 2019, pp. 4171–4186.

[112] C.-S. Wu, A. Madotto, E. Hosseini-Asl, C. Xiong, R. Socher, and P. Fung, "Transferable multi-domain state generator for task-oriented dialogue systems," in *Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 808–819.

[113] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020.

[114] T. Lengauer, O. Sander, S. Sierra, A. Thielen, and R. Kaiser, "Bioinformatics prediction of HIV coreceptor usage," *Nature Biotechnology*, vol. 25, no. 12, p. 1407, 2007.

[115] P. Baldi, P. Sadowski, and D. Whiteson, "Searching for exotic particles in high-energy physics with deep learning," *Nature Communications*, vol. 5, p. 4308, 2014.

[116] K. T. Schütt, F. Arbabzadah, S. Chmiela, K.-R. Müller, and A. Tkatchenko, "Quantum-chemical insights from deep tensor neural networks," *Nature Communications*, vol. 8, no. 1, pp. 1–8, 2017.

[117] G. Carleo and M. Troyer, "Solving the quantum many-body problem with artificial neural networks," *Science*, vol. 355, no. 6325, pp. 602–606, 2017.

[118] K. Schütt, M. Gastegger, A. Tkatchenko, K.-R. Müller, and R. Maurer, "Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions," *Nature Communications*, vol. 10, p. 5024, 2019.

[119] P. Jurmeister, M. Bockmayr, P. Seegerer, T. Bockmayr, D. Treue, G. Montavon, C. Vollbrecht, A. Arnold, D. Teichmann, K. Bressem *et al.*, "Machine learning analysis of DNA methylation profiles distinguishes primary lung squamous cell carcinomas from head and neck metastases," *Science Translational Medicine*, vol. 11, no. 509, 2019.

[120] F. Klauschen, K.-R. Müller, A. Binder, M. Bockmayr, M. Hägele, P. Seegerer, S. Wienert, G. Pruneri, S. de Maria, S. Badve *et al.*, "Scoring of tumor-infiltrating lymphocytes: From visual estimation to machine learning," *Seminars in Cancer Biology*, vol. 52, no. 2, p. 151, 2018.

[121] F. Arcadu, F. Benmansour, A. Maunz, J. Willis, Z. Haskova, and M. Prunotto, "Deep learning algorithm predicts diabetic retinopathy progression in individual patients," *npj Digital Medicine*, vol. 2, no. 1, pp. 1–9, 2019.

[122] D. Ardila, A. P. Kiraly, S. Bharadwaj, B. Choi, J. J. Reicher, L. Peng, D. Tse, M. Etemadi, W. Ye, G. Corrado *et al.*, "End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography," *Nature Medicine*, vol. 25, no. 6, pp. 954–961, 2019.

[123] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, and J. Dean, "A guide to deep learning in healthcare," *Nature Medicine*, vol. 25, no. 1, pp. 24–29, 2019.

[124] R. Chalapathy, A. K. Menon, and S. Chawla, "Robust, deep and inductive anomaly detection," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2017, pp. 36–51.

[125] J. Chen, S. Sathe, C. C. Aggarwal, and D. S. Turaga, "Outlier Detection with Autoencoder Ensembles," in *SIAM International Conference on Data Mining*, 2017, pp. 90–98.

[126] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *International Conference on Knowledge Discovery & Data Mining*, 2017, pp. 665–674.

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

[128] C. Aytekin, X. Ni, F. Cricri, and E. Aksu, "Clustering and unsupervised anomaly detection with l_2 normalized deep auto-encoder representations," in *International Joint Conference on Neural Networks*, 2018, pp. 1–6.

[129] D. Abati, A. Porrello, S. Calderara, and R. Cucchiara, "Latent space autoregression for novelty detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 481–490.

[130] C. Huang, J. Cao, F. Ye, M. Li, Y. Zhang, and C. Lu, "Inverse-transform autoencoder for anomaly detection," *arXiv preprint arXiv:1911.10676*, 2019.

[131] D. Gong, L. Liu, V. Le, B. Saha, M. R. Mansour, S. Venkatesh, and A. v. d. Hengel, "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *International Conference on Computer Vision*, 2019, pp. 1705–1714.

[132] P. Oza and V. M. Patel, "C2AE: Class conditioned auto-encoder for open-set recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2307–2316.

[133] D. T. Nguyen, Z. Lou, M. Klar, and T. Brox, "Anomaly detection with multiple-hypotheses predictions," in *International Conference on Machine Learning*, vol. 97, 2019, pp. 4800–4809.

[134] K. H. Kim, S. Shim, Y. Lim, J. Jeon, J. Choi, B. Kim, and A. S. Yoon, "RaPP: Novelty detection with reconstruction along projection pathway," in *International Conference on Learning Representations*, 2020.

[135] S. M. Erfani, S. Rajasegarar, S. Karunasekera, and C. Leckie, "High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning," *Pattern Recognition*, vol. 58, pp. 121–134, 2016.

[136] L. Ruff, R. A. Vandermeulen, N. Görnitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," in *International Conference on Machine Learning*, vol. 80, 2018, pp. 4390–4399.

[137] M. Sabokrou, M. Khalooei, M. Fathy, and E. Adeli, "Adversarially learned one-class classifier for novelty detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3379–3388.

[138] P. Oza and V. M. Patel, "One-class convolutional neural network," *IEEE Signal Processing Letters*, vol. 26, no. 2, pp. 277–281, 2019.

[139] L. Ruff, Y. Zemlyanskiy, R. Vandermeulen, T. Schnake, and M. Kloft, "Self-attentive, multi-context one-class classification for unsupervised anomaly detection on text," in *Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 4061–4071.

[140] P. Perera, R. Nallapati, and B. Xiang, "OCGAN: One-class novelty detection using GANs with constrained latent representations," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2898–2906.

[141] P. Perera and V. M. Patel, "Learning deep features for one-class classification," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5450–5463, 2019.

[142] J. Wang and A. Cherian, "GODS: Generalized one-class discriminative subspaces for anomaly detection," in *International Conference on Computer Vision*, 2019, pp. 8201–8211.

[143] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, and M. Kloft, "Deep semi-supervised anomaly detection," in *International Conference on Learning Representations*, 2020.

[144] Z. Ghafoori and C. Leckie, "Deep multi-sphere support vector data description," in *SIAM International Conference on Data Mining*, 2020, pp. 109–117.

[145] S. Zhai, Y. Cheng, W. Lu, and Z. Zhang, "Deep structured energy based models for anomaly detection," in *International Conference on Machine Learning*, vol. 48, 2016, pp. 1100–1109.

[146] R. Chalapathy, E. Toth, and S. Chawla, "Group anomaly detection using deep generative models," in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2018, pp. 173–189.

[147] L. Deecke, R. A. Vandermeulen, L. Ruff, S. Mandt, and M. Kloft, "Image anomaly detection with generative adversarial networks," in

European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, 2018, pp. 3–17.

[148] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "Ganomaly: Semi-supervised anomaly detection via adversarial training," in *Computer Vision – ACCV 2018*, C. V. Jawahar, H. Li, G. Mori, and K. Schindler, Eds. Cham: Springer International Publishing, 2019, pp. 622–637.

[149] H. Choi, E. Jang, and A. A. Alemi, "WAIC, but why? generative ensembles for robust anomaly detection," *arXiv preprint arXiv:1810.01392*, 2018.

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

[151] H. Zenati, M. Romain, C.-S. Foo, B. Lecouat, and V. Chandrasekhar, "Adversarially learned anomaly detection," in *IEEE International Conference on Data Mining*, 2018, pp. 727–736.

[152] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar, "Efficient GAN-based anomaly detection," *arXiv preprint arXiv:1802.06222*, 2018.

[153] I. Golan and R. El-Yaniv, "Deep anomaly detection using geometric transformations," in *Advances in Neural Information Processing Systems*, 2018, pp. 9758–9769.

[154] D. Hendrycks, M. Mazeika, S. Kadavath, and D. Song, "Using self-supervised learning can improve model robustness and uncertainty," in *Advances in Neural Information Processing Systems*, 2019, pp. 15 637–15 648.

[155] S. Wang, Y. Zeng, X. Liu, E. Zhu, J. Yin, C. Xu, and M. Kloft, "Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network," in *Advances in Neural Information Processing Systems*, 2019, pp. 5960–5973.

[156] L. Bergman and Y. Hoshen, "Classification-based anomaly detection for general data," in *International Conference on Learning Representations*, 2020.

[157] M. Markou and S. Singh, "Novelty detection: a review—part 1: statistical approaches," *Signal Processing*, vol. 83, no. 12, pp. 2481–2497, 2003.

[158] ——, "Novelty detection: a review—part 2:: neural network based approaches," *Signal Processing*, vol. 83, no. 12, pp. 2499–2521, 2003.

[159] V. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artificial Intelligence Review*, vol. 22, no. 2, pp. 85–126, 2004.

[160] S. Walfish, "A review of statistical outlier methods," *Pharmaceutical Technology*, vol. 30, no. 11, pp. 1–5, 2006.

[161] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009.

[162] A. S. Hadi, R. Imon, and M. Werner, "Detection of outliers," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 1, no. 1, pp. 57–70, 2009.

[163] P. Gogoi, D. K. Bhattacharyya, B. Borah, and J. K. Kalita, "A survey of outlier detection methods in network anomaly identification," *Computer Journal*, vol. 54, no. 4, pp. 570–588, 2011.

[164] K. Singh and S. Upadhyaya, "Outlier detection: applications and techniques," *International Journal of Computer Science Issues*, vol. 9, no. 1, p. 307, 2012.

[165] A. Zimek, E. Schubert, and H.-P. Kriegel, "A survey on unsupervised outlier detection in high-dimensional numerical data," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 5, no. 5, pp. 363–387, 2012.

[166] H. Aguinis, R. K. Gottfredson, and H. Joo, "Best-practice recommendations for defining, identifying, and handling outliers," *Organizational Research Methods*, vol. 16, no. 2, pp. 270–301, 2013.

[167] J. Zhang, "Advancements of outlier detection: A survey," *ICST Transactions on Scalable Information Systems*, vol. 13, no. 1, pp. 1–26, 2013.

[168] M. A. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "A review of novelty detection," *Signal Processing*, vol. 99, pp. 215–249, 2014.

[169] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han, "Outlier detection for temporal data: A survey," *Transactions on Knowledge and Data Engineering*, vol. 26, no. 9, pp. 2250–2267, 2014.

[170] S. Agrawal and J. Agrawal, "Survey on anomaly detection using data mining techniques," *Procedia Computer Science*, vol. 60, pp. 708–713, 2015.

[171] L. Akoglu, H. Tong, and D. Koutra, "Graph based anomaly detection and description: A survey," *Data Mining and Knowledge Discovery*, vol. 29, no. 3, pp. 626–688, 2015.

[172] S. Ranshous, S. Shen, D. Koutra, S. Harenberg, C. Faloutsos, and N. F. Samatova, "Anomaly detection in dynamic networks: A survey," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 7, no. 3, pp. 223–247, 2015.

[173] J. Tamboli and M. Shukla, "A survey of outlier detection algorithms for data streams," in *Proceedings of the 3rd International Conference on Computing for Sustainable Global Development*, 2016, pp. 3535–3540.

[174] M. Goldstein and S. Uchida, "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data," *PLoS ONE*, vol. 11, no. 4, p. e0152173, 2016.

[175] X. Xu, H. Liu, and M. Yao, "Recent progress of anomaly detection," *Complexity*, vol. 2019, pp. 1–11, 2019.

[176] H. Wang, M. J. Bah, and M. Hammad, "Progress in outlier detection techniques: A survey," *IEEE Access*, vol. 7, pp. 107 964–108 000, 2019.

[177] V. Barnett and T. Lewis, *Outliers in Statistical Data*, 3rd ed. Wiley, 1994.

[178] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. John Wiley & Sons, 2005.

[179] C. C. Aggarwal, *Outlier Analysis*, 2nd ed. Springer, 2016.

[180] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," *arXiv preprint arXiv:1901.03407*, 2019.

[181] F. D. Mattia, P. Galeone, M. D. Simoni, and E. Ghelfi, "A survey on GANs for anomaly detection," *arXiv preprint arXiv:1906.11632*, 2019.

[182] G. Pang, C. Shen, L. Cao, and A. van den Hengel, "Deep learning for anomaly detection: A review," *arXiv preprint arXiv:2007.02500*, 2020.

[183] D. M. J. Tax, "One-class classification," Ph.D. dissertation, Delft University of Technology, 2001.

[184] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An introduction to kernel-based learning algorithms," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 181–201, 2001.

[185] D. Rumsfeld, *Known and Unknown: A Memoir*. Penguin, 2011.

[186] F. J. Anscombe, "Rejection of outliers," *Technometrics*, vol. 2, no. 2, pp. 123–146, 1960.

[187] F. E. Grubbs, "Procedures for detecting outlying observations in samples," *Technometrics*, vol. 11, no. 1, pp. 1–21, 1969.

[188] D. M. Hawkins, *Identification of Outliers*. Springer, 1980, vol. 11.

[189] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, "MVTec AD–a comprehensive real-world dataset for unsupervised anomaly detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9592–9600.

[190] X. Song, M. Wu, C. Jermaine, and S. Ranka, "Conditional anomaly detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 5, pp. 631–645, 2007.

[191] K. Smets, B. Verdonk, and E. M. Jordaan, "Discovering novelty in spatio/temporal data using one-class support vector machines," in *International Joint Conference on Neural Networks*, 2009, pp. 2956–2963.

[192] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection for discrete sequences: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 5, pp. 823–839, 2010.

[193] W. Lu, Y. Cheng, C. Xiao, S. Chang, S. Huang, B. Liang, and T. Huang, "Unsupervised sequential outlier detection with deep architectures," *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4321–4330, 2017.

[194] W. Samek, S. Nakajima, M. Kawanabe, and K.-R. Müller, "On robust parameter estimation in brain–computer interfacing," *Journal of Neural Engineering*, vol. 14, no. 6, p. 061001, 2017.

[195] L. Xiong, B. Póczos, and J. G. Schneider, "Group anomaly detection using flexible genre models," in *Advances in Neural Information Processing Systems*, 2011, pp. 1071–1079.

[196] K. Muandet and B. Schölkopf, "One-class support measure machines for group anomaly detection," in *Conference on Uncertainty in Artificial Intelligence*, 2013, pp. 449–458.

[197] R. Yu, X. He, and Y. Liu, "GLAD: Group anomaly detection in social media analysis," *ACM Transactions on Knowledge Discovery from Data*, vol. 10, no. 2, pp. 1–22, 2015.

[198] L. Bontemps, J. McDermott, N.-A. Le-Khac *et al.*, "Collective anomaly detection based on long short-term memory recurrent neural networks," in *International Conference on Future Data and Security Engineering*. Springer, 2016, pp. 141–152.

[199] F. Ahmed and A. Courville, "Detecting semantic anomalies," in *AAAI Conference on Artificial Intelligence*, 2020, pp. 3154–3162.

[200] A. J. Fox, "Outliers in time series," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 34, no. 3, pp. 350–363, 1972.

[201] R. S. Tsay, "Outliers, level shifts, and variance changes in time series," *Journal of Forecasting*, vol. 7, no. 1, pp. 1–20, 1988.

[202] R. S. Tsay, D. Peña, and A. E. Pankratz, "Outliers in multivariate time series," *Biometrika*, vol. 87, no. 4, pp. 789–804, 2000.

[203] A. Lavin and S. Ahmad, "Evaluating real-time anomaly detection algorithms–the numenta anomaly benchmark," in *International Conference on Machine Learning and Applications*. IEEE, 2015, pp. 38–44.

[204] S. Chawla and P. Sun, "SLOM: a new measure for local spatial outliers," *Knowledge and Information Systems*, vol. 9, no. 4, pp. 412–429, 2006.

[205] E. Schubert, A. Zimek, and H.-P. Kriegel, "Local outlier detection reconsidered: a generalized view on locality with applications to spatial, video, and network outlier detection," *Data Mining and Knowledge Discovery*, vol. 28, no. 1, pp. 190–237, 2014.

[206] C. C. Noble and D. J. Cook, "Graph-based anomaly detection," in *International Conference on Knowledge Discovery & Data Mining*, 2003, pp. 631–636.

[207] J. Höner, S. Nakajima, A. Bauer, K.-R. Müller, and N. Görnitz, "Minimizing trust leaks for robust sybil detection," in *International Conference on Machine Learning*, vol. 70, 2017, pp. 1520–1528.

[208] M. Ahmed, "Collective anomaly detection techniques for network traffic analysis," *Annals of Data Science*, vol. 5, no. 4, pp. 497–512, 2018.

[209] F. Locatello, S. Bauer, M. Lucic, G. Raetsch, S. Gelly, B. Schölkopf, and O. Bachem, "Challenging common assumptions in the unsupervised learning of disentangled representations," in *International Conference on Machine Learning*, vol. 97, 2019, pp. 4114–4124.

[210] B. Schölkopf and A. J. Smola, *Learning with Kernels*. MIT press, 2002.

[211] I. Steinwart, D. Hush, and C. Scovel, "A classification framework for anomaly detection," *Journal of Machine Learning Research*, vol. 6, no. Feb, pp. 211–232, 2005.

[212] O. Chapelle, B. Schölkopf, and A. Zien, "Semi-supervised learning," *IEEE Transactions on Neural Networks*, vol. 20, no. 3, pp. 542–542, 2009.

[213] W. Polonik, "Measuring mass concentrations and estimating density contour clusters-an excess mass approach," *The Annals of Statistics*, vol. 23, no. 3, pp. 855–881, 1995.

[214] A. B. Tsybakov, "On nonparametric estimation of density level sets," *The Annals of Statistics*, vol. 25, no. 3, pp. 948–969, 1997.

[215] S. Ben-David and M. Lindenbaum, "Learning distributions by their density levels: A paradigm for learning without a teacher," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 171–182, 1997.

[216] P. Rigollet, R. Vert *et al.*, "Optimal rates for plug-in estimators of density level sets," *Bernoulli*, vol. 15, no. 4, pp. 1154–1178, 2009.

[217] W. Polonik, "Minimum volume sets and generalized quantile processes," *Stochastic Processes and Their Applications*, vol. 69, no. 1, pp. 1–24, 1997.

[218] J. N. Garcia, Z. Kutalik, K.-H. Cho, and O. Wolkenhauer, "Level sets and minimum volume sets of probability density functions," *International Journal of Approximate Reasoning*, vol. 34, no. 1, pp. 25–47, 2003.

[219] C. D. Scott and R. D. Nowak, "Learning minimum volume sets," *Journal of Machine Learning Research*, vol. 7, no. Apr, pp. 665–704, 2006.

[220] L. E. Ghaoui, M. I. Jordan, and G. R. Lanckriet, "Robust novelty detection with single-class MPM," in *Advances in Neural Information Processing Systems*, 2003, pp. 929–936.

[221] A. K. Menon and R. C. Williamson, "A loss framework for calibrated anomaly detection," in *Advances in Neural Information Processing Systems*, 2018, pp. 1494–1504.

[222] D. M. J. Tax and R. P. W. Duin, "Support vector domain description," *Pattern Recognition Letters*, vol. 20, no. 11, pp. 1191–1199, 1999.

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

[224] S. Clémençon and J. Jakubowicz, "Scoring anomalies: a m-estimation formulation," in *International Conference on Artificial Intelligence and Statistics*, 2013, pp. 659–667.

[225] N. Goix, A. Sabourin, and S. Clémençon, "On anomaly ranking and excess-mass curves," in *International Conference on Artificial Intelligence and Statistics*, 2015, pp. 287–295.

[226] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust Statistics: The Approach Based on Influence Functions*. John Wiley & Sons, 2005.

[227] Y. Liu and Y. F. Zheng, "Minimum enclosing and maximum excluding machine for pattern description and discrimination," in *International Conference on Pattern Recognition*, 2006, pp. 129–132.

[228] N. Görnitz, M. Kloft, K. Rieck, and U. Brefeld, "Toward supervised anomaly detection," *Journal of Artificial Intelligence Research*, vol. 46, pp. 235–262, 2013.

[229] E. Min, J. Long, Q. Liu, J. Cui, Z. Cai, and J. Ma, "SU-IDS: A semi-supervised and unsupervised framework for network intrusion detection," in *International Conference on Cloud Computing and Security*, 2018, pp. 322–334.

[230] B. R. Kiran, D. M. Thomas, and R. Parakkal, "An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos," *Journal of Imaging*, vol. 4, no. 2, p. 36, 2018.

[231] A. Siddiqui, A. Fern, T. G. Dietterich, R. Wright, A. Theriault, and D. W. Archer, "Feedback-guided anomaly discovery via online optimization," in *International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2200–2209.

[232] F. Denis, "PAC learning from positive statistical queries," in *International Conference on Algorithmic Learning Theory*, 1998, pp. 112–126.

[233] B. Zhang and W. Zuo, "Learning from positive and unlabeled examples: A survey," in *Proceedings of the IEEE International Symposium on Information Processing*, 2008, pp. 650–654.

[234] M. C. Du Plessis, G. Niu, and M. Sugiyama, "Analysis of learning from positive and unlabeled data," in *Advances in Neural Information Processing Systems*, 2014, pp. 703–711.

[235] J. Muñoz-Marí, F. Bovolo, L. Gómez-Chova, L. Bruzzone, and G. Camp-Valls, "Semi-Supervised One-Class Support Vector Machines for Classification of Remote Sensing Sata," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 8, pp. 3188–3197, 2010.

[236] G. Blanchard, G. Lee, and C. Scott, "Semi-supervised novelty detection," *Journal of Machine Learning Research*, vol. 11, no. Nov, pp. 2973–3009, 2010.

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

[238] S. Liu, R. Garrepalli, T. Dietterich, A. Fern, and D. Hendrycks, "Open category detection with pac guarantees," in *International Conference on Machine Learning*, vol. 80, 2018, pp. 3169–3178.

[239] M. Sugiyama, M. Krauledat, and K.-R. Müller, "Covariate shift adaptation by importance weighted cross validation," *Journal of Machine Learning Research*, vol. 8, no. May, pp. 985–1005, 2007.

[240] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset Shift in Machine Learning*. The MIT Press, 2009.

[241] M. Sugiyama and M. Kawanabe, *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. MIT press, 2012.

[242] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. Müller, "How to explain individual classification decisions," *Journal of Machine Learning Research*, vol. 11, no. Jun, pp. 1803–1831, 2010.

[243] G. Montavon, W. Samek, and K.-R. Müller, "Methods for interpreting and understanding deep neural networks," *Digital Signal Processing*, vol. 73, pp. 1–15, 2018.

[244] S. Lapuschkin, S. Wäldchen, A. Binder, G. Montavon, W. Samek, and K.-R. Müller, "Unmasking Clever Hans predictors and assessing what machines really learn," *Nature Communications*, vol. 10, p. 1096, 2019.

[245] W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, and K.-R. Müller, Eds., *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, ser. Lecture Notes in Computer Science. Springer, 2019, vol. 11700.

[246] W. Härdle, *Applied Nonparametric Regression*. Cambridge university press, 1990, no. 19.

[247] J. Laurikkala, M. Juhola, E. Kentala, N. Lavrac, S. Miksch, and B. Kavsek, "Informal identification of outliers in medical data," in *5th International Workshop on Intelligent Data Analysis in Medicine and Pharmacology*, vol. 1, 2000, pp. 20–24.

[248] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. USA: Prentice-Hall, Inc., 1988.

[249] S. Roberts and L. Tarassenko, "A probabilistic resource allocating network for novelty detection," *Neural Computation*, vol. 6, no. 2, pp. 270–284, 1994.

[250] C. M. Bishop, "Novelty detection and neural network validation," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 141, no. 4, pp. 217–222, 1994.

[251] L. Devroye and L. Györfi, *Nonparametric Density Estimation: The L1 View*. New York; Chichester: John Wiley & Sons, 1985.

[252] S. Frühwirth-Schnatter, *Finite Mixture and Markov Switching Models*. Springer Science & Business Media, 2006.

[253] J. Kim and C. D. Scott, "Robust kernel density estimation," *Journal of Machine Learning Research*, vol. 13, no. 82, pp. 2529–2565, 2012.

[254] R. Vandermeulen and C. Scott, "Consistency of robust kernel density estimators," in *Conference on Learning Theory*, 2013, pp. 568–591.

[255] S. E. Fahlman, G. E. Hinton, and T. J. Sejnowski, "Massively parallel architectures for AI: NETL, Thistle, and Boltzmann machines," in *AAAI Conference on Artificial Intelligence*, 1983, pp. 109–113.

[256] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 79, no. 8, pp. 2554–2558, 1982.

[257] Y. Lecun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang, *A Tutorial on Energy-Based Learning*. MIT Press, 2006.

[258] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

[259] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient langevin dynamics," in *International Conference on Machine Learning*, 2011, pp. 681–688.

[260] W. Grathwohl, K.-C. Wang, J.-H. Jacobsen, D. Duvenaud, M. Norouzi, and K. Swersky, "Your classifier is secretly an energy based model and you should treat it like one," in *International Conference on Learning Representations*, 2020.

[261] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[262] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machines," in *International Conference on Artificial Intelligence and Statistics*, 2009, pp. 448–455.

[263] J. Ngiam, Z. Chen, P. W. Koh, and A. Ng, "Learning deep energy models," in *International Conference on Machine Learning*, 2011, pp. 1105–1112.

[264] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *International Conference on Learning Representations*, 2014.

[265] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *International Conference on Machine Learning*, vol. 32, 2014, pp. 1278–1286.

[266] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.

[267] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[268] H. Xu, W. Chen, N. Zhao, Z. Li, J. Bu, Z. Li, Y. Liu, Y. Zhao, D. Pei, Y. Feng *et al.*, "Unsupervised anomaly detection via variational autoencoder for seasonal KPIs in web applications," in *World Wide Web Conference*, 2018, pp. 187–196.

[269] E. Nalisnick, A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan, "Do deep generative models know what they don't know?" in *International Conference on Learning Representations*, 2019.

[270] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," *Special Lecture on IE*, vol. 2, pp. 1–18, 2015.

[271] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, 2016, pp. 2234–2242.

[272] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," *arXiv preprint arXiv:1701.07875*, 2017.

[273] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in Neural Information Processing Systems*, 2017, pp. 5767–5777.

[274] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," *arXiv preprint arXiv:1812.04948*, 2018.

[275] L. Dinh, D. Krueger, and Y. Bengio, "NICE: non-linear independent components estimation," in *International Conference on Learning Representations*, 2015.

[276] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference," *arXiv preprint arXiv:1912.02762*, 2019.

[277] I. Kobyzev, S. Prince, and M. Brubaker, "Normalizing flows: An introduction and review of current methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[278] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real NVP," in *International Conference on Learning Representations*, 2017.

[279] C.-W. Huang, D. Krueger, A. Lacoste, and A. Courville, "Neural autoregressive flows," in *International Conference on Machine Learning*, vol. 80, 2018, pp. 2078–2087.

[280] F. Noé, S. Olsson, J. Köhler, and H. Wu, "Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning," *Science*, vol. 365, no. 6457, p. eaaw1147, 2019.

[281] B. Nachman and D. Shih, "Anomaly detection with density estimation," *Physical Review D*, vol. 101, p. 075042, Apr 2020.

[282] L. Wellhausen, R. Ranftl, and M. Hutter, "Safe robot navigation via multi-modal anomaly detection," *Robotics and Automation Letters*, vol. 5, no. 2, pp. 1326–1333, 2020.

[283] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[284] S. Suh, D. H. Chae, H. Kang, and S. Choi, "Echo-state conditional variational autoencoder for anomaly detection," in *International Joint Conference on Neural Networks*, 2016, pp. 1015–1022.

[285] A. Abdelhamed, M. A. Brubaker, and M. S. Brown, "Noise flow: Noise modeling with conditional normalizing flows," in *International Conference on Computer Vision*, 2019, pp. 3165–3173.

[286] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S.-K. Ng, "MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks," in *International Conference on Artificial Neural Networks*, 2019, pp. 703–716.

[287] S. R. Bowman, L. Vilnis, O. Vinyals, A. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," in *The SIGNLL Conference on Computational Natural Language Learning*, Aug. 2016, pp. 10–21.

[288] L. Chen, S. Dai, C. Tao, H. Zhang, Z. Gan, D. Shen, Y. Zhang, G. Wang, R. Zhang, and L. Carin, "Adversarial text generation via feature-mover's distance," in *Advances in Neural Information Processing Systems*, 2018, pp. 4666–4677.

[289] W. Jin, R. Barzilay, and T. Jaakkola, "Junction tree variational autoencoder for molecular graph generation," in *International Conference on Machine Learning*, vol. 80, 2018, pp. 2323–2332.

[290] A. Bojchevski, O. Shchur, D. Zügner, and S. Günnemann, "NetGAN: Generating graphs via random walks," in *International Conference on Machine Learning*, vol. 80, 2018, pp. 610–619.

[291] R. Liao, Y. Li, Y. Song, S. Wang, W. Hamilton, D. K. Duvenaud, R. Urtasun, and R. Zemel, "Efficient graph generation with graph recurrent attention networks," in *Advances in Neural Information Processing Systems*, 2019, pp. 4255–4265.

[292] V. Vapnik, *Statistical Learning Theory*. Wiley, 1998.

[293] M. M. Moya, M. W. Koch, and L. D. Hostetler, "One-class classifier networks for target recognition applications," in *Proceedings World Congress on Neural Networks*, 1993, pp. 797–801.

[294] M. M. Moya and D. R. Hush, "Network constraints and multi-objective optimization for one-class classification," *Neural Networks*, vol. 9, no. 3, pp. 463–474, 1996.

[295] S. S. Khan and M. G. Madden, "One-class classification: taxonomy of study and review of techniques," *The Knowledge Engineering Review*, vol. 29, no. 3, pp. 345–374, 2014.

[296] T. Minter, "Single-class classification," in *LARS Symposia*, 1975, p. 54.

[297] R. El-Yaniv and M. Nisenson, "Optimal single-class classification strategies," in *Advances in Neural Information Processing Systems*, 2007, pp. 377–384.

[298] P. J. Rousseeuw, "Multivariate estimation with high breakdown point," *Mathematical Statistics and Applications*, vol. 8, pp. 283–297, 1985.

[299] P. J. Rousseeuw and K. V. Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999.

[300] A. Muñoz and J. M. Moguerza, "Estimation of high-density regions using one-class neighbor machines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 3, pp. 476–480, 2006.

[301] B. Schölkopf, S. Mika, C. J. Burges, P. Knirsch, K.-R. Müller, G. Rätsch, and A. J. Smola, "Input space versus feature space in kernel-based methods," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1000–1017, 1999.

[302] L. M. Manevitz and M. Yousef, "One-class SVMs for document classification," *Journal of Machine Learning Research*, vol. 2, no. Dec, pp. 139–154, 2001.

[303] R. Vert and J.-P. Vert, "Consistency and convergence rates of one-class SVMs and related algorithms," *Journal of Machine Learning Research*, vol. 7, no. May, pp. 817–854, 2006.

[304] G. Lee and C. D. Scott, "The one class support vector machine solution path," in *International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, 2007, pp. 521–524.

[305] K. Sjöstrand and R. Larsen, "The entire regularization path for the support vector domain description," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2006, pp. 241–248.

[306] G. Lee and C. Scott, "Nested support vector machines," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1648–1660, 2009.

[307] A. Glazer, M. Lindenbaum, and S. Markovitch, "q-OCSVM: A q-quantile estimator for high-dimensional distributions," in *Advances in Neural Information Processing Systems*, 2013, pp. 503–511.

[308] N. Görnitz, L. A. Lima, K.-R. Müller, M. Kloft, and S. Nakajima, "Support vector data descriptions and k-means clustering: One class?" *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 9, pp. 3994–4006, 2017.

[309] S. Das, B. L. Matthews, A. N. Srivastava, and N. C. Oza, "Multiple kernel learning for heterogeneous anomaly detection: Algorithm and aviation safety case study," in *International Conference on Knowledge Discovery & Data Mining*, 2010, pp. 47–56.

[310] C. Gautam, R. Balaji, K. Sudharsan, A. Tiwari, and K. Ahuja, "Localized multiple kernel learning for anomaly detection: One-class classification," *Knowledge-Based Systems*, vol. 165, pp. 241–252, 2019.

[311] G. Rätsch, S. Mika, B. Schölkopf, and K.-R. Müller, "Constructing boosting algorithms from SVMs: An application to one-class classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1184–1199, 2002.

[312] V. Roth, "Outlier detection with one-class kernel fisher discriminants," in *Advances in Neural Information Processing Systems*, 2005, pp. 1169–1176.

[313] ——, "Kernel fisher discriminants for outlier detection," *Neural Computation*, vol. 18, no. 4, pp. 942–960, 2006.

[314] F. Dufrenois, "A one-class kernel fisher criterion for outlier detection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 5, pp. 982–994, 2014.

[315] A. Ghasemi, H. R. Rabiee, M. T. Manzuri, and M. H. Rohban, "A bayesian approach to the data description problem," in *AAAI Conference on Artificial Intelligence*, 2012, pp. 907–913.

[316] W. Liu, G. Hua, and J. R. Smith, "Unsupervised one-class learning for automatic outlier removal," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3826–3833.

[317] P. Wu, J. Liu, and F. Shen, "A deep one-class neural network for anomalous event detection in complex scenes," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 7, pp. 2609–2622, 2020.

[318] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *International Conference on Learning Representations*, 2015.

[319] G. Goh, "Why momentum really works," *Distill*, 2017.

[320] L. Ruff, R. A. Vandermeulen, B. J. Franks, K.-R. Müller, and M. Kloft, "Rethinking assumptions in deep anomaly detection," *arXiv preprint arXiv:2006.00339*, 2020.

[321] S. Goyal, A. Raghunathan, M. Jain, H. V. Simhadri, and P. Jain, "DROCC: Deep robust one-class classification," in *International Conference on Machine Learning*, 2020, pp. 11 335–11 345.

[322] J. Kauffmann, L. Ruff, G. Montavon, and K.-R. Müller, "The Clever Hans effect in anomaly detection," *arXiv preprint arXiv:2006.10609*, 2020.

[323] P. Chong, L. Ruff, M. Kloft, and A. Binder, "Simple and effective prevention of mode collapse in deep one-class classification," in *International Joint Conference on Neural Networks*, 2020.

[324] G. Pang, C. Shen, and A. van den Hengel, "Deep anomaly detection with deviation networks," in *International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 353–362.

[325] D. Hendrycks, M. Mazeika, and T. G. Dietterich, "Deep anomaly detection with outlier exposure," in *International Conference on Learning Representations*, 2019.

[326] P. Liznerski, L. Ruff, R. A. Vandermeulen, B. J. Franks, M. Kloft, and K.-R. Müller, "Explainable deep one-class classification," *arXiv preprint arXiv:2007.01760*, 2020.

[327] M. Sabokrou, M. Fathy, G. Zhao, and E. Adeli, "Deep end-to-end one-class classifier," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–10, 2020.

[328] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York, NY: Springer, 2009.

[329] G. Steinbuss and K. Böhm, "Generating artificial outliers in the absence of genuine ones – a survey," *arXiv preprint arXiv:2006.03646*, 2020.

[330] J. P. Theiler and D. M. Cai, "Resampling approach for anomaly detection in multispectral images," in *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery IX*, vol. 5093. International Society for Optics and Photonics, 2003, pp. 230–240.

[331] M. A. Davenport, R. G. Baraniuk, and C. D. Scott, "Learning minimum volume sets with support vector machines," in *IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, 2006, pp. 301–306.

[332] W. Fan, M. Miller, S. Stolfo, W. Lee, and P. Chan, "Using artificial anomalies to detect unknown and known network intrusions," *Knowledge and Information Systems*, vol. 6, no. 5, pp. 507–527, 2004.

[333] P. Cheema, N. L. D. Khoa, M. Makki Alamdari, W. Liu, Y. Wang, F. Chen, and P. Runcie, "On structural health monitoring using tensor analysis and support vector machine with artificial negative data," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 1813–1822.

[334] N. Abe, B. Zadrozny, and J. Langford, "Outlier detection by active learning," in *International Conference on Knowledge Discovery & Data Mining*, 2006, pp. 504–509.

[335] J. W. Stokes, J. C. Platt, J. Kravis, and M. Shilman, "ALADIN: Active learning of anomalies to detect intrusions," Microsoft Research, Technical Report MSR-TR-2008-24, 2008.

[336] N. Görnitz, M. Kloft, and U. Brefeld, "Active and semi-supervised data domain description," in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2009, pp. 407–422.

[337] D. Pelleg and A. W. Moore, "Active learning for anomaly and rare-category detection," in *Advances in Neural Information Processing Systems*, 2005, pp. 1073–1080.

[338] N. Japkowicz, C. Myers, and M. Gluck, "A novelty detection approach to classification," in *International Joint Conferences on Artificial Intelligence*, vol. 1, 1995, pp. 518–523.

[339] S. Hawkins, H. He, G. Williams, and R. Baxter, "Outlier detection using replicator neural networks," in *International Conference on Data Warehousing and Knowledge Discovery*, vol. 2454, 2002, pp. 170–180.

[340] J. A. Lee and M. Verleysen, *Nonlinear Dimensionality Reduction*. Springer New York, 2007.

[341] R. Pless and R. Souvenir, "A survey of manifold learning for images," *IPSJ Transactions on Computer Vision and Applications*, vol. 1, pp. 83–94, 2009.

[342] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[343] ——, *Self-Organizing Maps*, 3rd ed. Springer, 2001.

[344] L. van der Maaten, E. Postma, and J. van den Herik, "Dimensionality reduction: A comparative review," Tilburg University, Technical Report TiCC-TR 2009-005, 2009.

[345] J. Schmidhuber, "Learning factorial codes by predictability minimization," *Neural Computation*, vol. 4, no. 6, pp. 863–879, 1992.

[346] M. Tschannen, O. Bachem, and M. Lucic, "Recent advances in autoencoder-based representation learning," in *3rd Workshop on Bayesian Deep Learning (NeurIPS 2018)*, 2018.

[347] Y. Linde, A. Buzo, and R. Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communications*, vol. 28, no. 1, pp. 84–95, 1980.

[348] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Springer Science & Business Media New York, 2012, vol. 159.

[349] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, vol. 61, no. 3, pp. 611–622, 1999.

[350] C. M. Bishop, "Bayesian PCA," in *Advances in Neural Information Processing Systems*, 1999, pp. 382–388.

[351] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed., ser. Springer Series in Statistics. New York, NY: Springer, 2002.

[352] D. M. Hawkins, "The detection of errors in multivariate data using principal components," *Journal of the American Statistical Association*, vol. 69, no. 346, pp. 340–344, 1974.

[353] J. E. Jackson and G. S. Mudholkar, "Control procedures for residuals associated with principal component analysis," *Technometrics*, vol. 21, no. 3, pp. 341–349, 1979.

[354] L. Parra, G. Deco, and S. Miesbach, "Statistical independence and novelty detection with information preserving nonlinear maps," *Neural Computation*, vol. 8, no. 2, pp. 260–269, 1996.

[355] M.-L. Shyu, S.-C. Chen, K. Sarinnapakorn, and L. Chang, "A novel anomaly detection scheme based on principal component classifier," in *IEEE International Conference on Data Mining*, 2003, pp. 353–365.

[356] L. Huang, X. Nguyen, M. Garofalakis, M. I. Jordan, A. Joseph, and N. Taft, "In-network PCA and anomaly detection," in *Advances in Neural Information Processing Systems*, 2007, pp. 617–624.

[357] V. Sharan, P. Gopalan, and U. Wieder, "Efficient anomaly detection via matrix sketching," in *Advances in Neural Information Processing Systems*, 2018, pp. 8069–8080.

[358] J. Ham, D. D. Lee, S. Mika, and B. Schölkopf, "A kernel view of the dimensionality reduction of manifolds," in *International Conference on Machine Learning*, 2004, p. 47.

[359] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 9, pp. 1672–1680, 2008.

[360] M. H. Nguyen and F. Torre, "Robust kernel principal component analysis," in *Advances in Neural Information Processing Systems*, 2009, pp. 1185–1192.

[361] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, pp. 1–37, 2011.

[362] Y. Xiao, H. Wang, W. Xu, and J. Zhou, "L1 norm based KPCA for novelty detection," *Pattern Recognition*, vol. 46, no. 1, pp. 389–396, 2013.

[363] E. Oja, "A simplified neuron model as a principal component analyzer," *Journal of Mathematical Biology*, vol. 15, no. 3, pp. 267–273, 1982.

[364] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing – Explorations in the Microstructure of Cognition*. MIT Press, 1986, ch. 8, pp. 318–362.

[365] D. H. Ballard, "Modular learning in neural networks," in *AAAI Conference on Artificial Intelligence*, 1987, pp. 279–284.

[366] G. E. Hinton, "Connectionist learning procedures," *Artificial Intelligence*, vol. 40, no. 1, pp. 185–234, 1989.

[367] M. A. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE journal*, vol. 37, no. 2, pp. 233–243, 1991.

[368] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[369] P. Baldi and K. Hornik, "Neural networks and principal component analysis: Learning from examples without local minima," *Neural Networks*, vol. 2, no. 1, pp. 53–58, 1989.

[370] E. Oja, "Principal components, minor components, and linear neural networks," *Neural Networks*, vol. 5, no. 6, pp. 927–935, 1992.

[371] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.

[372] ——, "Sparse coding with an overcomplete basis set: A strategy employed by V1?" *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[373] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Computation*, vol. 12, no. 2, pp. 337–365, 2000.

[374] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Advances in Neural Information Processing Systems*, 2007, pp. 801–808.

[375] A. Makhzani and B. Frey, "k-sparse autoencoders," in *International Conference on Learning Representations*, 2014.

[376] N. Zeng, H. Zhang, B. Song, W. Liu, Y. Li, and A. M. Dobaie, "Facial expression recognition via learning deep sparse autoencoders," *Neurocomputing*, vol. 273, pp. 643–649, 2018.

[377] D. Arpit, Y. Zhou, H. Ngo, and V. Govindaraju, "Why regularized autoencoders learn sparse representation?" in *International Conference on Machine Learning*, vol. 48, 2016, pp. 136–144.

[378] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *International Conference on Machine Learning*, 2008, pp. 1096–1103.

[379] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[380] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *International Conference on Machine Learning*, 2011, pp. 833–840.

[381] S. You, K. C. Tezcan, X. Chen, and E. Konukoglu, "Unsupervised lesion detection via image restoration with a normative prior," in *International Conference on Medical Imaging with Deep Learning*, 2019, pp. 540–556.

[382] D. Park, Y. Hoshi, and C. C. Kemp, "A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder," *Robotics and Automation Letters*, vol. 3, no. 3, pp. 1544–1551, 2018.

[383] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," *arXiv preprint arXiv:1511.05644*, 2015.

[384] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "LSTM-based encoder-decoder for multi-sensor anomaly detection," *arXiv preprint arXiv:1607.00148*, 2016.

[385] T. Kieu, B. Yang, C. Guo, and C. S. Jensen, "Outlier detection for time series with recurrent autoencoder ensembles," in *International Joint Conferences on Artificial Intelligence*, 2019, pp. 2725–2732.

[386] C. D. Hofer, R. Kwitt, M. Dixit, and M. Niethammer, "Connectivity-optimized representation learning via persistent homology," in *International Conference on Machine Learning*, vol. 97, 2019, pp. 2751–2760.

[387] T. Amarbayasgalan, B. Jargalsaikhan, and K. H. Ryu, "Unsupervised novelty detection using deep autoencoders with density based clustering," *Applied Sciences*, vol. 8, no. 9, p. 1468, 2018.

[388] N. Sarafijanovic-Djukic and J. Davis, "Fast distance-based anomaly detection in images using an inception-like autoencoder," in *International Conference on Discovery Science*. Springer, 2019, pp. 493–508.

[389] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.

[390] G. Voronoi, "Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Premier mémoire. Sur quelques propriétés des formes quadratiques positives parfaites." *Journal für die Reine und Angewandte Mathematik*, vol. 1908, no. 133, pp. 97–178, 1908.

[391] ——, "Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Deuxième mémoire. Recherches sur les parallélloèdres primitifs." *Journal für die Reine und Angewandte Mathematik*, vol. 1908, no. 134, pp. 198–287, 1908.

[392] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means, spectral clustering and normalized cuts," in *International Conference on Knowledge Discovery & Data Mining*, 2004, pp. 551–556.

[393] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *International Conference on Machine Learning*, vol. 48, 2016, pp. 478–487.

[394] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 6306–6315.

[395] A. Razavi, A. van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with VQ-VAE-2," in *Advances in Neural Information Processing Systems*, 2019, pp. 14 866–14 876.

[396] M. Kampffmeyer, S. Løkse, F. M. Bianchi, L. Livi, A.-B. Salberg, and R. Jenssen, "Deep divergence-based approach to clustering," *Neural Networks*, vol. 113, pp. 91–101, 2019.

[397] B. Yang, X. Fu, N. D. Sidiropoulos, and M. Hong, "Towards k-means-friendly spaces: Simultaneous deep learning and clustering," in *International Conference on Machine Learning*, vol. 70, 2017, pp. 3861–3870.

[398] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *European Conference on Computer Vision*, 2018, pp. 132–149.

[399] P. Bojanowski and A. Joulin, "Unsupervised learning by predicting noise," in *International Conference on Machine Learning*, vol. 70, 2017, pp. 517–526.

[400] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[401] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT press, 2012.

[402] D. J. C. MacKay, "A practical bayesian framework for backpropagation networks," *Neural Computation*, vol. 4, no. 3, pp. 448–472, 1992.

[403] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural networks," in *International Conference on Machine Learning*, vol. 37, 2015, pp. 1613–1622.

[404] S. Harmeling, G. Dornhege, D. M. J. Tax, F. Meinecke, and K.-R. Müller, "From outliers to prototypes: ordering data," *Neurocomputing*, vol. 69, no. 13-15, pp. 1608–1618, 2006.

[405] M. Zhao and V. Saligrama, "Anomaly detection with score functions based on nearest neighbor graphs," in *Advances in Neural Information Processing Systems*, 2009, pp. 2250–2258.

[406] X. Gu, L. Akoglu, and A. Rinaldo, "Statistical analysis of nearest neighbor methods for anomaly detection," in *Advances in Neural Information Processing Systems*, 2019, pp. 10 923–10 933.

[407] P. Juszczak, D. M. J. Tax, E. Pe, R. P. W. Duin *et al.*, "Minimum spanning tree based one-class classifier," *Neurocomputing*, vol. 72, no. 7, pp. 1859–1869, 2009.

[408] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *IEEE International Conference on Data Mining*, 2008, pp. 413–422.

[409] S. Guha, N. Mishra, G. Roy, and O. Schrijvers, "Robust random cut forest based anomaly detection on streams," in *International Conference on Machine Learning*, vol. 48, 2016, pp. 2712–2721.

[410] L. Bergman, N. Cohen, and Y. Hoshen, "Deep nearest neighbor anomaly detection," *arXiv preprint arXiv:2002.10445*, 2020.

[411] J. Glasser and B. Lindauer, “Bridging the gap: A pragmatic approach to generating insider threat data,” in *IEEE Symposium on Security and Privacy Workshops*, 2013, pp. 98–104.

[412] N. Mu and J. Gilmer, “MNIST-C: A robustness benchmark for computer vision,” *arXiv preprint arXiv:1906.02337*, 2019.

[413] D. Hendrycks and T. Dietterich, “Benchmarking neural network robustness to common corruptions and perturbations,” in *International Conference on Learning Representations*, 2019.

[414] P. Laskov, C. Schäfer, I. Kotenko, and K.-R. Müller, “Intrusion detection in unlabeled data with quarter-sphere support vector machines,” *PIK - Praxis der Informationsverarbeitung und Kommunikation*, vol. 27, no. 4, pp. 228–236, 2004.

[415] M. Kloft and P. Laskov, “Security analysis of online centroid anomaly detection,” *Journal of Machine Learning Research*, vol. 13, no. 118, pp. 3681–3724, 2012.

[416] A. F. Emmott, S. Das, T. Dietterich, A. Fern, and W.-K. Wong, “Systematic construction of anomaly detection benchmarks from real data,” in *KDD 2013 Workshop on Outlier Detection and Description*, 2013, pp. 16–21.

[417] A. Emmott, S. Das, T. Dietterich, A. Fern, and W.-K. Wong, “A Meta-Analysis of the Anomaly Detection Problem,” *arXiv*, vol. 1503.01158, no. v2, pp. 1–35, 2016.

[418] D. Hendrycks, K. Zhao, S. Basart, J. Steinhardt, and D. Song, “Natural adversarial examples,” *arXiv preprint arXiv:1907.07174*, 2019.

[419] W. Huang and P. Wei, “A PCB dataset for defects detection and classification,” *arXiv preprint arXiv:1901.08204*, 2019.

[420] B. E. Bejnordi, M. Veta, P. J. Van Diest, B. Van Ginneken, N. Karssemeijer, G. Litjens, J. A. Van Der Laak, M. Hermsen, Q. F. Manson, M. Balkenhol *et al.*, “Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer,” *Journal of the American Medical Association*, vol. 318, no. 22, pp. 2199–2210, 2017.

[421] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, “ChestX-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2097–2106.

[422] D. Zimmerer, J. Petersen, G. Köhler, P. Jäger, P. Full, T. Roß, T. Adler, A. Reinke, L. Maier-Hein, and K. Maier-Hein, “Medical out-of-distribution analysis challenge,” Mar. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3784230

[423] K. Faust, Q. Xie, D. Han, K. Goyle, Z. Volynskaya, U. Djuric, and P. Diamandis, “Visualizing histopathologic deep learning classification and anomaly detection using nonlinear feature space dimensionality reduction,” *BMC Bioinformatics*, vol. 19, no. 1, p. 173, 2018.

[424] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, “Credit card fraud detection: A realistic modeling and a novel learning strategy,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3784–3797, 2018.

[425] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Identifying suspicious URLs: An application of large-scale online learning,” in *International Conference on Machine Learning*, 2009, pp. 681–688.

[426] N. Moustafa and J. Slay, “UNSW-NB15: a comprehensive data set for network intrusion detection systems,” in *Military Communications and Information Systems Conference*, 2015, pp. 1–6.

[427] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, “Unsupervised real-time anomaly detection for streaming data,” *Neurocomputing*, vol. 262, pp. 134–147, 2017.

[428] N. Laptev, S. Amizadeh, and I. Flint, “Generic and scalable framework for automated time-series anomaly detection,” in *International Conference on Knowledge Discovery & Data Mining*, 2015, pp. 1939–1947.

[429] G. O. Campos, A. Zimek, J. Sander, R. J. Campello, B. Micenková, E. Schubert, I. Assent, and M. E. Houle, “On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study,” *Data Mining and Knowledge Discovery*, vol. 30, no. 4, pp. 891–927, 2016.

[430] S. Rayana, “ODDS library,” 2016. [Online]. Available: http://odds.cs.stonybrook.edu

[431] R. Domingues, M. Filippone, P. Michiardi, and J. Zouaoui, “A comparative evaluation of outlier detection algorithms: Experiments and analyses,” *Pattern Recognition*, vol. 74, pp. 406–421, 2018.

[432] D. Dua and C. Graff, “UCI machine learning repository,” 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[433] A. P. Bradley, “The use of the area under the ROC curve in the evaluation of machine learning algorithms,” *Pattern Recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.

[434] J. Davis and M. Goadrich, “The relationship between precision-recall and roc curves,” in *International Conference on Machine Learning*, 2006, pp. 233–240.

[435] T. Fawcett, “An introduction to ROC analysis,” *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[436] X. Ding, Y. Li, A. Belatreche, and L. P. Maguire, “An experimental evaluation of novelty detection methods,” *Neurocomputing*, vol. 135, pp. 313–327, 2014.

[437] M. T. Ribeiro, S. Singh, and C. Guestrin, “”Why should I trust you?”: Explaining the predictions of any classifier,” in *International Conference on Knowledge Discovery & Data Mining*, 2016, pp. 1135–1144.

[438] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in *International Conference on Computer Vision*. IEEE Computer Society, 2017, pp. 618–626.

[439] M. Sundararajan, A. Taly, and Q. Yan, “Axiomatic attribution for deep networks,” in *International Conference on Machine Learning*, vol. 70, 2017, pp. 3319–3328.

[440] Z. Qi, S. Khorram, and F. Li, “Visualizing deep networks by optimizing with integrated gradients,” in *CVPR 2019 Workshops*, vol. 2, 2019.

[441] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, “On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation,” *PLoS ONE*, vol. 10, no. 7, p. e0130140, 07 2015.

[442] B. Micenková, R. T. Ng, X.-H. Dang, and I. Assent, “Explaining outliers by subspace separability,” in *IEEE International Conference on Data Mining*, 2013, pp. 518–527.

[443] M. D. Siddiqui, A. Fern, T. G. Dietterich, and W. K. Wong, “Sequential feature explanations for anomaly detection,” *ACM Transactions on Knowledge Discovery from Data*, vol. 13, no. 1, pp. 1–22, 2019.

[444] J. Kauffmann, K.-R. Müller, and G. Montavon, “Towards explaining anomalies: A deep taylor decomposition of one-class models,” *Pattern Recognition*, vol. 101, p. 107198, 2020.

[445] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, “A survey on deep learning in medical image analysis,” *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.

[446] P. T. Huynh, A. M. Jarolimek, and S. Daye, “The false-negative mammogram.” *Radiographics*, vol. 18, no. 5, pp. 1137–1154, 1998.

[447] M. Petticrew, A. Sowden, D. Lister-Sharp, and K. Wright, “False-negative results in screening programmes: systematic review of impact and implications,” *Health Technology Assess.*, vol. 4, no. 5, pp. 1–120, 2000.

[448] M. S. Pepe, *The Statistical Evaluation of Medical Tests for Classification and Prediction*. Oxford University Press, 2003.

[449] X.-H. Zhou, N. A. Obuchowski, and D. K. McClish, *Statistical Methods in Diagnostic Medicine*, 2nd ed. John Wiley & Sons, 2011.

[450] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. John Willey & Sons, 1973.

[451] A. Lazarevic and V. Kumar, “Feature bagging for outlier detection,” in *International Conference on Knowledge Discovery & Data Mining*, 2005, pp. 157–166.

[452] H. V. Nguyen, H. H. Ang, and V. Gopalkrishnan, “Mining outliers with ensemble of heterogeneous detectors on random subspaces,” in *DASFAA (1)*, ser. Lecture Notes in Computer Science, vol. 5981. Springer, 2010, pp. 368–383.

[453] Y. Gal and Z. Ghahramani, “Dropout as a bayesian approximation: Representing model uncertainty in deep learning,” in *International Conference on Machine Learning*, vol. 48, 2016, pp. 1050–1059.

[454] B. Lakshminarayanan, A. Pritzel, and C. Blundell, “Simple and scalable predictive uncertainty estimation using deep ensembles,” in *Advances in Neural Information Processing Systems*, 2017, pp. 6402–6413.

[455] A. Kendall and Y. Gal, “What uncertainties do we need in bayesian deep learning for computer vision?” in *Advances in Neural Information Processing Systems*, 2017, pp. 5574–5584.

[456] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, and J. Snoek, “Can you trust your model’s uncertainty? evaluating predictive uncertainty under dataset shift,” in *Advances in Neural Information Processing Systems*, 2019, pp. 13 991–14 002.

[457] K. Bykov, M. M.-C. Höhne, K.-R. Müller, S. Nakajima, and M. Kloft, “How much can i trust you?–quantifying uncertainties in explaining neural networks,” *arXiv preprint arXiv:2006.09000*, 2020.

[458] S. Hido, Y. Tsuboi, H. Kashima, M. Sugiyama, and T. Kanamori, “Statistical outlier detection using direct density ratio estimation,” *Knowledge and Information Systems*, vol. 26, no. 2, pp. 309–336, 2011.

[459] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *International Conference on Artificial Intelligence and Statistics*, 2010, pp. 297–304.

[460] C. K. Chow, "An optimum character recognition system using decision functions," *IRE Transactions on Electronic Computers*, no. 4, pp. 247–254, 1957.

[461] ——, "On optimum recognition error and reject tradeoff," *IEEE Transactions on Information Theory*, vol. 16, no. 1, pp. 41–46, 1970.

[462] P. L. Bartlett and M. H. Wegkamp, "Classification with a reject option using a hinge loss," *Journal of Machine Learning Research*, vol. 9, no. Aug, pp. 1823–1840, 2008.

[463] D. M. J. Tax and R. P. W. Duin, "Growing a multi-class classifier with a reject option," *Pattern Recognition Letters*, vol. 29, no. 10, pp. 1565–1570, 2008.

[464] Y. Grandvalet, A. Rakotomamonjy, J. Keshet, and S. Canu, "Support vector machines with a reject option," in *Advances in Neural Information Processing Systems*, 2009, pp. 537–544.

[465] C. Cortes, G. DeSalvo, and M. Mohri, "Learning with rejection," in *International Conference on Algorithmic Learning Theory*, 2016, pp. 67–82.

[466] Y. Geifman and R. El-Yaniv, "Selective classification for deep neural networks," in *Advances in Neural Information Processing Systems*, 2017, pp. 4878–4887.

[467] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in Large Margin Classifiers*, vol. 10, no. 3, pp. 61–74, 1999.

[468] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *International Conference on Machine Learning*, vol. 70, 2017, pp. 1321–1330.

[469] T. DeVries and G. W. Taylor, "Learning confidence for out-of-distribution detection in neural networks," *arXiv preprint arXiv:1802.04865*, 2018.

[470] K. Lee, H. Lee, K. Lee, and J. Shin, "Training confidence-calibrated classifiers for detecting out-of-distribution samples," in *International Conference on Learning Representations*, 2018.

[471] C. Ni, N. Charoenphakdee, J. Honda, and M. Sugiyama, "On the calibration of multiclass classification with rejection," in *Advances in Neural Information Processing Systems*, 2019, pp. 2586–2596.

[472] A. Meinke and M. Hein, "Towards neural networks that provably know when they don't know," in *International Conference on Learning Representations*, 2020.

[473] D. J. MacKay and M. N. Gibbs, "Density networks," in *Statistics and Neural Networks: Advances at the Interface*, 1999.

[474] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in *International Conference on Learning Representations*, 2017.

[475] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," in *International Conference on Learning Representations*, 2018.

[476] K. Lee, K. Lee, H. Lee, and J. Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in *Advances in Neural Information Processing Systems*, 2018, pp. 7167–7177.

[477] S. Choi and S.-Y. Chung, "Novelty detection via blurring," in *International Conference on Learning Representations*, 2020.

[478] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, "Toward open set recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 7, pp. 1757–1772, 2012.

[479] W. J. Scheirer, L. P. Jain, and T. E. Boult, "Probability models for open set recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2317–2324, 2014.

[480] A. Bendale and T. E. Boult, "Towards open set deep networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1563–1572.

[481] L. Shu, H. Xu, and B. Liu, "DOC: Deep open classification of text documents," in *Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2911–2916.

[482] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *International Conference on Learning Representations*, 2014.

[483] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *International Conference on Learning Representations*, 2015.

[484] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *IEEE Symposium on Security and Privacy*. IEEE, 2017, pp. 39–57.

[485] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble adversarial training: Attacks and defenses," in *International Conference on Learning Representations*, 2018.

[486] B. Biggio and F. Roli, "Wild patterns: Ten years after the rise of adversarial machine learning," *Pattern Recognition*, vol. 84, pp. 317–331, 2018.

[487] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *International Conference on Machine Learning*, vol. 80, 2018, pp. 274–283.

[488] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *International Conference on Learning Representations*, 2018.

[489] N. Carlini, A. Athalye, N. Papernot, W. Brendel, J. Rauber, D. Tsipras, I. Goodfellow, A. Madry, and A. Kurakin, "On evaluating adversarial robustness," *arXiv preprint arXiv:1902.06705*, 2019.

[490] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan, "Theoretically principled trade-off between robustness and accuracy," in *International Conference on Machine Learning*, vol. 97, 2019, pp. 7472–7482.

[491] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial examples are not bugs, they are features," in *Advances in Neural Information Processing Systems*, 2019, pp. 125–136.

[492] T. Che, X. Liu, S. Li, Y. Ge, R. Zhang, C. Xiong, and Y. Bengio, "Deep verifier networks: Verification of deep discriminative models with deep generative models," *arXiv preprint arXiv:1911.07421*, 2019.

[493] G. Shalev, Y. Adi, and J. Keshet, "Out-of-distribution detection using multiple semantic label representations," in *Advances in Neural Information Processing Systems*, 2018, pp. 7375–7385.

[494] D. Hendrycks, K. Lee, and M. Mazeika, "Using pre-training can improve model robustness and uncertainty," in *International Conference on Machine Learning*, vol. 97, 2019, pp. 2712–2721.

[495] A. R. Dhamija, M. Günther, and T. Boult, "Reducing network agnostophobia," in *Advances in Neural Information Processing Systems*, 2018, pp. 9157–9168.

[496] A.-K. Dombrowski, M. Alber, C. Anders, M. Ackermann, K.-R. Müller, and P. Kessel, "Explanations can be manipulated and geometry is to blame," in *Advances in Neural Information Processing Systems*, 2019, pp. 13 589–13 600.

[497] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission," in *International Conference on Knowledge Discovery & Data Mining*, 2015, pp. 1721–1730.

[498] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K.-R. Müller, "Toward interpretable machine learning: Transparent deep neural networks and beyond," *arXiv preprint arXiv:2003.07631*, 2020.

[499] J. D. Lee and K. A. See, "Trust in automation: Designing for appropriate reliance," *Human Factors*, vol. 46, no. 1, pp. 50–80, 2004.

[500] H. Jiang, B. Kim, M. Guan, and M. Gupta, "To trust or not to trust a classifier," in *Advances in Neural Information Processing Systems*, 2018, pp. 5541–5552.

[501] Z. C. Lipton, "The doctor just won't accept that!" in *NIPS 2017 Interpretable ML Symposium*, 2017.

[502] B. Goodman and S. Flaxman, "European union regulations on algorithmic decision-making and a "right to explanation"," *AI Magazine*, vol. 38, no. 3, pp. 50–57, 2017.

[503] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, "Concrete problems in ai safety," *arXiv preprint arXiv:1606.06565*, 2016.

[504] C. Richter and N. Roy, "Safe visual navigation via deep learning and novelty detection," in *Robotics: Science and Systems XIII*, 2017.

[505] X. H. Dang, B. Micenková, I. Assent, and R. T. Ng, "Local outlier detection with interpretation," in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2013, pp. 304–320.

[506] X. H. Dang, I. Assent, R. T. Ng, A. Zimek, and E. Schubert, "Discriminative features for identifying and interpreting outliers," in *International Conference on Data Engineering*. IEEE, 2014, pp. 88–99.

[507] L. Duan, G. Tang, J. Pei, J. Bailey, A. Campbell, and C. Tang, "Mining outlying aspects on numeric data," *Data Mining and Knowledge Discovery*, vol. 29, no. 5, pp. 1116–1151, 2015.

[508] N. X. Vinh, J. Chan, S. Romano, J. Bailey, C. Leckie, K. Ramamohanarao, and J. Pei, "Discovering outlying aspects in large datasets," *Data Mining and Knowledge Discovery*, vol. 30, no. 6, pp. 1520–1555, 2016.

[509] M. Macha and L. Akoglu, "Explaining anomalies in groups with characterizing subspace rules," *Data Mining and Knowledge Discovery*, vol. 32, no. 5, pp. 1444–1480, 2018.

[510] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[511] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[512] J. Wang, S. Sun, and Y. Yu, "Multivariate triangular quantile maps for novelty detection," in *Advances in Neural Information Processing Systems*, 2019, pp. 5061–5072.

[513] J. Ren, P. J. Liu, E. Fertig, J. Snoek, R. Poplin, M. Depristo, J. Dillon, and B. Lakshminarayanan, "Likelihood ratios for out-of-distribution detection," in *Advances in Neural Information Processing Systems*, 2019, pp. 14 680–14 691.

[514] J. Serrà, D. Álvarez, V. Gómez, O. Slizovskaia, J. F. Núñez, and J. Luque, "Input complexity and out-of-distribution detection with likelihood-based generative models," in *International Conference on Learning Representations*, 2020.

[515] G. E. Box, "Science and statistics," *Journal of the American Statistical Association*, vol. 71, no. 356, pp. 791–799, 1976.

[516] A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, and C. Ré, "Snorkel: Rapid training data creation with weak supervision," in *Proceedings of the VLDB Endowment*, vol. 11, no. 3, 2017, pp. 269–282.

[517] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *National Science Review*, vol. 5, no. 1, pp. 44–53, 2018.

[518] Y. Roh, G. Heo, and S. E. Whang, "A survey on data collection for machine learning: A big data - ai integration perspective," *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[519] T. Daniel, T. Kurutach, and A. Tamar, "Deep variational semi-supervised novelty detection," *arXiv preprint arXiv:1911.04971*, 2019.

[520] S. Das, W.-K. Wong, T. Dietterich, A. Fern, and A. Emmott, "Discovering anomalies by incorporating feedback from an expert," *Transactions on Knowledge Discovery from Data*, vol. 14, no. 4, pp. 1–32, 2020.

[521] S. Nedelkoski, J. Bogatinovski, A. Acker, J. Cardoso, and O. Kao, "Self-attentive classification-based anomaly detection in unstructured logs," *arXiv preprint arXiv:2008.09340*, 2020.

[522] K. Ouardini, H. Yang, B. Unnikrishnan, M. Romain, C. Garcin, H. Zenati, J. P. Campbell, M. F. Chiang, J. Kalpathy-Cramer, V. Chandrasekhar *et al.*, "Towards practical unsupervised anomaly detection on retinal images," in *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data*. Springer, 2019, pp. 225–234.

[523] R. Shu, Y. Chen, A. Kumar, S. Ermon, and B. Poole, "Weakly supervised disentanglement with guarantees," in *International Conference on Learning Representations*, 2020.

[524] F. Locatello, B. Poole, G. Rätsch, B. Schölkopf, O. Bachem, and M. Tschannen, "Weakly-supervised disentanglement without compromises," in *International Conference on Machine Learning*, 2020, pp. 7753–7764.

[525] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," in *International Conference on Learning Representations*, 2016.

[526] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International Conference on Machine Learning*, 2020, pp. 10 709–10 719.

[527] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European Conference on Computer Vision*, 2016, pp. 649–666.

[528] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *International Conference on Computer Vision*, 2015, pp. 1422–1430.

[529] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *European Conference on Computer Vision*, 2016, pp. 69–84.

[530] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *International Conference on Learning Representations*, 2018.

[531] A. YM., R. C., and V. A., "A critical analysis of self-supervision, or what we can learn from a single image," in *International Conference on Learning Representations*, 2020.

[532] E. Nalisnick, A. Matsukawa, Y. W. Teh, and B. Lakshminarayanan, "Detecting out-of-distribution inputs to deep generative models using typicality," in *4th Workshop on Bayesian Deep Learning (NeurIPS 2019)*, 2019.

[533] R. T. Schirrmeister, Y. Zhou, T. Ball, and D. Zhang, "Understanding anomaly detection with deep invertible networks through hierarchies of distributions and features," *arXiv preprint arXiv:2006.10848*, 2020.

[534] P. Kirichenko, P. Izmailov, and A. G. Wilson, "Why normalizing flows fail to detect out-of-distribution data," *arXiv preprint arXiv:2006.08545*, 2020.

[535] R. Vershynin, *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press, 2018, vol. 47.

[536] A. Tong, R. Yousefzadeh, G. Wolf, and S. Krishnaswamy, "Fixing bias in reconstruction-based anomaly detection with lipschitz discriminators," *arXiv preprint arXiv:1905.10710*, 2019.

[537] D. Krueger, E. Caballero, J.-H. Jacobsen, A. Zhang, J. Binas, R. L. Priol, and A. Courville, "Out-of-distribution generalization via risk extrapolation," *arXiv preprint arXiv:2003.00688*, 2020.

[538] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[539] R. Linsker, "Self-organization in a perceptual network," *Computer*, vol. 21, no. 3, pp. 105–117, 1988.

[540] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[541] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," in *International Conference on Learning Representations*, 2019.

[542] T. Berger, "Rate-distortion theory," *Wiley Encyclopedia of Telecommunications*, 2003.

[543] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "β-VAE: Learning basic visual concepts with a constrained variational framework," in *International Conference on Learning Representations*, 2017.

[544] A. Alemi, B. Poole, I. Fischer, J. Dillon, R. A. Saurous, and K. Murphy, "Fixing a broken ELBO," in *International Conference on Machine Learning*, vol. 80, 2018, pp. 159–168.

[545] S. Park, G. Adosoglou, and P. M. Pardalos, "Interpreting rate-distortion of variational autoencoder and using model uncertainty for anomaly detection," *arXiv preprint arXiv:2005.01889*, 2020.

[546] W. Lee and D. Xiang, "Information-theoretic measures for anomaly detection," in *IEEE Symposium on Security and Privacy*. IEEE, 2001, pp. 130–143.

[547] A. Høst-Madsen, E. Sabeti, and C. Walton, "Data discovery and anomaly detection using atypicality: Theory," *Transactions on Information Theory*, vol. 65, no. 9, pp. 5302–5322, 2019.