Analyzing Advances in Anomaly and Outliers Detection Based Algorithms

Florida Atlantic University ∗ Shaun A Pritchard ∗ Email:spritchard2021@fau.edu

Abstract - This research compares a baseline study with 40 novel and state-of-the-art unsupervised machine learning algorithms and techniques. Which examines a number of factors that contribute to the emergence of outliers and advances in state-of-the-art anomaly detection. Outlier and anomaly detection are increasingly popular with the advent of newly emerging technologies and diverse applications. Outliers (e.g. Anomalies) are patterns of different data within given data, whereas Outliers would be merely extreme data points within data. In various fields, detecting outliers and anomalies is crucial. Statistical analysis leads can be grossly distorted by outliers, which can indicate faults, intrusions, exploitations, faulty systems, in critical devices or in numerous fields including health, industry, security, finance, and many others. Through the detection of these anomalies, models can be improved as well as critical issues detected. This research will assess baseline study and more modern advanced benchmarked techniques in unsupervised machine learning algorithms. while also evaluating problems with current outlier detection systems. The evaluation will take an in-depth look at promising Outlier and Anomaly Detection Techniques and algorithms for multiple data types and scenarios (e.g. general, mixed type, high-dimensional, and so on) Data as well as discuss well-defined analysis distribution from the perspective of several research fields and applications including Intrusion Detection, Fraud Detection, Medical Health, Industrial Damage Detection, Sensor Networks, Textual Anomaly Detection, autonomous vehicles, Image Processing. The datasets used for individual benchmark comparisons include both synthetic and novel datasets with a mix of ordinal and categorical features. The results of this study include the creation of comparison and comprehensive resources of outlier and anomalies detection algorithms, including their extensive features of state-of-the-art detection methods.

Index Terms – Anomaly detection, Data Mining, State-of-the-art detection, Unsupervised outlier detection.

1. Introduction

Outlier detection is a vital component of data mining and has high practical value in a number of fields. Outliers in a dataset can be detected and removed by fundamental preprocessing methods without which the data analysis can be misleading and error prone. In addition, anomalies in the data can adversely affect machine learning algorithms[1]. Outliers in data can occur due to the variability in measurements, experimental errors, or noise [1], and the existence of outliers in data makes the analysis of data misleading and degrades the performance of machine learning algorithms. In contrast, observations have also shown that in some cases removing outliers and anomalies negatively impacts data analysis. Data mining is Inherently subjective and involves an understanding of the generative behavior of data to define an objective function or model. The assumptions underlying such generative processes are very subjective, and a specific algorithm used will only be able to describe a limited aspect of that process. The model can provide effective results for some parts of the data, whereas it may not provide effective results for other parts of the data. Likewise, a given model may sometimes work on one set of data but may not work on another [2]. The goal of this research is to provide a comprehensive overview of the various implementations of outlier and anomaly detection techniques using baseline study and other comparative research to define assumptions about data and efficient algorithms of outlier and anomaly detection, as well as an overview of the contrasts between the various approaches and their respective use-cases. Based on the underlying approach adopted by each technique, I have divided existing techniques into different categories based on types, approaches, methods, and models. The techniques use key assumptions to distinguish between normal and atypical behavior within each category. Whenever a given technique is applied to a specific domain, these assumptions can be used as guidelines to assess its effectiveness. It will present a basic technique for locating outliers and anomalies in each category, followed by a discussion of their benefits and drawbacks. It is my goal to assess each of the techniques in each category in a concise and easy-to-understand manner. In addition, descriptions of each category, historical context to address the issues in the field of research, as well as advantages and disadvantages of the techniques in each category are provided. This study evaluates a number of novel, baseline[1], and state-of-the-art detection techniques. These unsupervised implementations diverge according to the variety of experiments and datasets to detect outliers and anomalies. For each initial experiment, datasets could range from novel, synthetic, and real-time analyses data. The proposed techniques are based on proven statistical methods and ongoing published research considering data variants, dimensionality, and other properties. According to most studies, state-of-the-art techniques are efficient in terms of performance, ease of implementation, and computational complexity. A comprehensive review of 40 algorithms as shown below in Table 1 will be evaluated and compared in this study

Table 1. Outlier anomaly detection algorithms evaluated and compared.

Detection Type	Model Name	ID
Probabilistic-based detection	Gaussian Mixture Model	GMM
Probabilistic-based detection	Dirichlet Process Mixture Model	DPMM
Probabilistic-based detection	Kernel Density Estimators	KDE
Probabilistic-based detection	Robust Kernel Density Estimator	RKDE
Probabilistic-based detection	Probabilistic Principal Component Analysis	PPCA
Probabilistic-based detection	Least-Squares Anomaly Detection	LSA
Probabilistic-based detection	Support Vector Data Description	SVDD
Probabilistic-based detection	One-Class SVM	OCSVM
Probabilistic-based detection	Kullback-Leibler	KL
Probabilistic-based detection	Mahalanobis Distance	MAHA
Probabilistic-based detection	Robust Kernel Estimation	RKDE
Probabilistic-based detection	Local Outlier Probabilities	LoOP
Neighbor-based detection	KNN Angle-Based Outlier Detection	ABOD
Neighbor-based detection	Local Outlier Factor	LOF
Neighbor-based detection	Outlier Detection Using Indegree Number	ODIN
Neighbor-based detection	Density Based Outlier Detection	DBOD
Neighbor-based detection	Random Cut Forest	RCF
Subspace-based detection	Subspace Outlier Detection	SOD
Subspace-based detection	Rarity Based Outlier Detection	RODS
Subspace-based detection	Outrank	OR
Ensemble-based detection:	Ensemble Gaussian Mixture Models	EGMM
Ensemble-based detection:	Randnet Model-Auto Encoders	RAE
Ensemble-based detection:	Data-Centered Ensembles	DCE
Ensemble-based detection:	Feature Bagging	FB
Ensemble-based detection:	Isolation-Forest	IFOR
Ensemble-based detection:	Grow When Required	GWR
Ensemble-based detection:	Lightweight Online Detector of Anomalies	LODA
Ensemble-based detection:	Extreme Gradient Boosting	XGBOD
Mixed-based detection	Local Subspace Based Outlier Detection	LSBOD
Mixed-based detection	Link-Based Outlier and Anomaly Detection	LOADED
Mixed-based detection	Pattern-Based Outlier Detection	PBOD
Mixed-based detection	Micro cluster-Based Detector of Anomalies in Edge Streams	MIDAS
Mixed-based detection	Anomaly Detection in Sound	ADS

The remainder of this paper is structured as follows. In Section 2, I will discuss related work. Section 3 discusses our proposed issues for outlier and anomaly detection. Section 4 discusses the techniques/approaches used with outlier and anomaly detection. Section 5 will discuss the critical evaluation of current approaches, while the summary is presented in section 6 and the conclusions are presented in Section 7.

2. Methodologies

Anomaly detection refers to the process of eliminating anomalies in data. In different application domains, anomalies are sometimes called outliers, discordant observations, exceptions, aberrations, surprises, peculiarities, or contaminants[3]. Outliers and anomalies are frequently used interchangeably. An anomaly detection system finds extensive use in many applications, including credit card fraud detection, fraud detection in insurance and health care, cyber-security intrusion detection, fault detection in safety critical systems (e.g. autonomous vehicles, medical), Image processing, military surveillance and so on[4]. Detecting anomalies in data is important because anomalies often result in significant (and sometimes critical) insights that can be used to drive action. According to [4]An abnormal traffic pattern in a computer network could indicate that a hacked computer is sending sensitive data to an unauthorized location. Temperature anomalies that are detected by remote sensing could indicate a heat wave or cold snap, or even faulty equipment. An anomalous MRI image may indicate the presence of malignant tumors or early signs of Alzheimer's disease. An anomalous credit card transaction could indicate credit card or identity theft, or odd readings from a spacecraft sensor could indicate a malfunction. The characteristic descriptions of baseline evaluations with some standard methodologies are as follows.

2.1 Outlier detection Types

In this paper, the terms that will be used to best describe Outlier detection types is of two types of outliers: ***univariate*** and ***multivariate***. Univariate outliers analyze values in a single *1 x N* feature space. Outliers of multivariate data are found in n-dimensional spaces (with n features).

2.2 Anomaly detection schema

Anomaly detection schema can be categorized into two groups. In this regard, there are two general categories: ***parametric*** (also known as statistical) where the data distribution is well known and unlikely to change, and ***non-parametric*** models used in dynamic environments where the statistical distribution is unknown and unfamiliar.

2.3 Outlier anomalies and novelty assumptions

The aim is to detect any new observations that can be fit into a data set containing only good data, known as 'novelty detection'. Detecting outliers, e.g. anomalies, is concerned with identifying outliers in data where $X_i, \dots X_{n,}\ each\ X_i \in \mathbb{R}^d$. According to[5] we assume working with data that are in the form of feature vectors in d dimensional *real* space. With regard to our data, we have a combination of what is known as nominal points, which are "non-anomalies", as well as anomaly points, where an anomaly is not just a statistical outlier in every case. If the key is that these anomaly points are generated by a different process from the one generating the nominal points, then it may not be a statistical outlier. A variety of methods and algorithms are used for the detection of "nominal points," i.e. outliers and anomalies; those that are sparse and adaptable to reconcile well-defined assumptions about anomalies in the data. In this case, we must ask if the outliers are drawn from known probability distributions, such as repeated examples of unknown machine failures in a sensor network compared to some common transaction fees in financial applications. A question should also consider the possibility of novel causes, changes in time, or adversarial situations (fraud, threats, security, etc.) in addition to the known assumption. As a result, we can obtain different types of data with a variety of parameters, and the algorithm we will need will depend on these assumptions viability. [6]States that three types of data have been used to analyze and evaluate anomaly detection algorithms. The first type of data is based on specific application problems. Secondly, there are synthetic datasets (with known nominal points). Lastly, there are datasets created by treating one or more classes as anomalous in an existing supervised classification problem. Such datasets can be quite helpful, they can help evaluate and understand the algorithmic refinements needed in a particular application to achieve high performance. Despite this, many datasets are not publicly accessible because of privacy or security reasons hence the data trust and scarcity issue[7].

2.4 **Considerations in Data**

Several types of data need to be considered in unsupervised outlier detection problems, including general, high-dimensional, high-density, mixed type, and real-time data. One of the most important factors in detecting outliers and anomalies is the data set explaining variability[8]. When the data conveyance associated with the problem is understood, the data assumption can be made. Thus, I thought it prudent to define the data types and categories outlined for this study which are preemptively responsible for these assumptions and applications as follows [9].

2.5 Variability of data

General data: There are two types of general data, univariate and multivariate, which contain nominal or categorical data, which contribute to data problems of lower feature dimensionality and are assumed to be normally distributed.

High-dimensional data: Datasets with a greater number of feature characteristics p than observations N. high-dimensional data implies many dimensions/variables/features/columns.

High-density data: Often refers to massive amounts of data from multiple sources, multiple data sets, or crowded datasets with multiple data points per feature making the data dense.

Mixed type data: Refers to data that are a combination of realizations from both continuous and categorical from random variables and is typically handled with clustering methods.

Real-Time data: **R**efers to asynchronous information that is delivered immediately after collection and persistence. There is no delay in the timeliness of the information provided. Real-time data is often used for navigation or tracking.

2.6 Processing data for outlierness

Normal data: These are instances nominal or mixed data types are dependent on the normal distribution.

Time series data: There are two types of outliers in a time series: point outliers (which are deviations from expectations at a given time) and shape outliers (which are points in a contiguous window which are anomalous). Outliers can occur within a time series as particular elements (or time points) or subsequences, referred to as point outliers and shape outliers, respectively. Dynamic or real-time data can also be time series data.

Evolving Data: In a real-time or adaptive data stream, data points may change, with feature values changing, and feature space may change, with newly emerging features over time. Row-streams, on the other hand, deliver points with fixed features one by one[10].

Graph & Network data: These are data and techniques defined as structured graph data have been of focus recently. As objects in graphs have long-range correlations, a suite of novel technology has been developed for anomaly detection in graph data[11].

2.7 Unsupervised outlier methods for anomaly detection

Statistics has explored the problem of detecting outliers extensively[9]. Data points are usually modeled by a probability distribution, followed by a hypothetical model to determine whether a data point is an outlier. The construction of an outlier detector is typically the first step in data mining and machine learning [12]. Following are some of the most popular methods and models for outlier analysis.

Probability-based methods: Statistical models rely on the assumption that the majority of the data follows a statistical distribution, and the degree to which an item is an outlier is determined by evaluating the likelihood that the item is generated by the same distribution. In general, the smaller the likelihood, the more unlikely the object is to be from the same distribution, and the more likely it is to be an outlier. Univariate tails are defined as extreme regions that have a probability density below a certain threshold after a model distribution has been chosen[13].

Clustering-based methods: In clustering, data points are grouped together based on their occurrence together. As a result of clustering, models based on clustering locate data points that are isolated from clusters and determine outliers. These outliers form small clusters of their own. [14]States, let $X1, \ldots, Xn$ be a set of clusters of a dataset Let $X1, \ldots, Xm$ be a set of clusters of a dataset T generated by a clustering algorithm, listed in the order of $|X1| \geq |X2| \geq \cdots \geq |Xn\,|$. Given parameters, α, and β, clustering-based outliers are those clusters in Xm through Xn such that the sum of $|X1|+|X2|+\cdots+|Xm-1|\ \ |T\ |*\alpha,\ |X1|+|X2|+\cdots+|Xm-2| \leq |T\ |*\alpha, and|Xm-1|/|Xm| > \beta.$ A data point must either belong to a cluster or be considered an outlier. By examining the relationships between objects and clusters, cluster-based approaches detect outliers.

Distance-based methods: [14] states, Distance-based models use k-nearest neighbors to determine whether a data point is an outlier based on its Euclidean distance. Data points that are outliers have an average distance to their k-closest neighbors that is much greater than the distance to their k-th nearest neighbors. $DB(p, D)$ is used to indicate that an object O is a distance-based outlier in a dataset T. Nearest neighbor techniques calculate the distance or similarity measure between an observation and its neighbors.

Density-based methods: Based on an outlier score, the density of the data point determines the outlier score of a density-based model. An object's outlier score is usually computed by comparing its local density to the average of its k-nearest neighbors' local density. This model is most commonly used when cluster density and shape significantly vary with data location[1].

Projection-based Methods: Using data-driven estimates of partial distances in the projection models, it is possible to establish distance-based thresholds for outlier detection. These thresholds are based on a sequential method of outlier detection. Outliers are observed in projection models as occurrences that are less frequent but differ in terms of their values[1].

Ranking-based methods: A ranked-based method examines whether a target is 'central' among its nearest neighbors based on the density of its nearest neighbors. This eliminates the problem of the calculation of density in the neighborhood of the ranking values to data points or evaluations located in a cluster where the cumulative sum of the ranks is relatively low[14]. These are just a few of the many methods or combinations of methods which can be used to detect unsupervised outliers. I have developed a simple nomenclature for describing the methods that are accessible through the following algorithmic models.

2.8 Algorithm Detection Models

Generally, the existing anomaly detection models can be further categorized into groups of model categories based on the proposed methods as follows: statistically based, neighbor-based, subspace-based, and ensemble-based and mixed-based detection methods. From these techniques, we can implement typical base methods for algorithms that can be used for outlier and anomaly detection.

Probabilistic-based detection: Detects anomalies in statistical distributions using statistical measures. Outlier anomaly detection can rely on rudimentary based algorithm techniques like z-score, modified z-score, IQR, boxplot, and histogram as basic filters. In learning problems, the data tend to follow a statistical distribution, and there can be different levels of complexity for unsupervised learning problems.

Neighbor-based detection: *according to* [15]Identifying anomalies by using neighborhood information. Typical. The concept of nearest neighbor is used in several anomaly detection techniques, whereby normal data instances are located within dense neighborhoods, whereas anomalous data instances are located far away examples include [16]–[18]etc.

Subspace-based detection: Anomalies can be detected through inference of different feature subsets. These include [19], [20].

Ensemble-based detection: Using Ensemble techniques, integrating multiple anomaly detection events to achieve a consensus. Such algorithms are [21]–[25].

Mixed-based detection: Separating data types or integrating different data types into a unified model. Typical examples included: [3], [26]. Also, deep learning-based detection methods for mixed models are among the most advanced and state of the art.

2.9 Applications of outlier detection

The goal of outlier detection is to discover unusual patterns that exist in a data[27]. To detect outliers, assumptions are made about outliers versus the rest of the data. According to the assumptions made, unsupervised outlier detection methods usually fall into one of the above main categories[6]. A method for detecting outliers is highly dependent on the modeling of normal objects and outliers. Nevertheless, since it is often difficult to know in advance all possible normal behaviors in an application, and there is usually no fine line between data normality and abnormality, building a comprehensive model that captures both normality and abnormality is a very challenging task. This study evaluates and compares the advances and applications of 40 different outlier algorithms and presents the state-of-the-art outlier methods for general, synthetic, high-dimensional, and time-series data. One is Intrusion Detection, two is Fraud Detection, three is Medical Health, four is Industrial Damage Detection, five is Sensor Networks, six is Detecting Anomaly in Text, seven is

Image Processing, and eight is Sound Processing. According to Table 2, the following algorithms have been tested and proven through various research to be robust and effective for as according to general specifics of applications as follows.

Table 2. General Detection of Outlier Anomalies in Unsupervised Applications

Detection Type			**Model Name**			**ID**			**Applications**		
			Intrusion Detection,	**Fraud Detection**	**Medical Health**	**Industrial Damage**	**Autonomous Vehicles**	**Sensor Networks**	**Textual Anomaly**	**Image Processing**	**Sound Processing**
Probabilistic-based detection	Gaussian Mixture Model	GMM	X	X	X				X		
Probabilistic-based detection	Dirichlet Process Mixture Model	DPMM	X	X	X	X	X	X		X	
Probabilistic-based detection	Kernel density estimators	KDE	X	X	X	X	X	X	X	X	
Probabilistic-based detection	Robust Kernel Density Estimator	RKDE	X	X	X	X	X	X	X	X	X
Probabilistic-based detection	Probabilistic Principal component analysis	PPCA	X	X	X	X	X	X	X	X	
Probabilistic-based detection	Least-squares anomaly detection	LSA	X	X	X	X	X	X	X	X	
Probabilistic-based detection	Support vector data description	SVDD	X	X	X	X	X	X		X	
Probabilistic-based detection	One-class SVM	OCSVM	X	X	X		X	X	X	X	
Probabilistic-based detection	Kullback-Leibler	KL	X	X	X	X	X	X	X	X	
Probabilistic-based detection	Mahalanobis distance	MAHA	X	X	X				X		
Probabilistic-based detection	Robust kernel the estimation	RKDE	X	X	X	X	X	X	X	X	
Probabilistic-based detection	local outlier probabilities	LoOP	X	X	X	X	X	X	X	X	X
Neighbor-based detection	KNN	KNN	X	X	X	X	X	X	X	X	

Neighbor-based detection	KNN Angle-Based Outlier Detection	ABOD	X	X	X	X	X	X	X	X	
Neighbor-based detection	Cluster Based Local Outlier	CBLOF	X	X	X	X	X	X	X	X	
Neighbor-based detection	Local outlier factor	LOF	X	X	X	X	X	X	X	X	
Neighbor-based detection	Outlier detection using indegree number	ODIN	X	X	X		X	X	X	X	
Neighbor-based detection		DBOD	X	X	X	X	X	X	X	X	
Neighbor-based detection	Cluster Based Local Outlier Factor	CBLOF	X	X		X	X	X	X	X	
Neighbor-based detection	Random cut forest	RCF	X	X	X	X	X	X	X	X	
Subspace-based detection	Subspace outlier detection	SOD	X	X	X	X	X	X	X	X	
Subspace-based detection	Rarity based Outlier Detection	RODS	X	X	X	X	X	X	X	X	
Subspace-based detection	Outrank	OR	X	X	X	X	X		X	X	
Ensemble-based detection:	Ensemble Gaussian Mixture Models	EGMM	X	X		X	X	X	X	X	
Ensemble-based detection:	RandNET Model-auto encoders	RAE	X	X	X	X	X	X	X		
Ensemble-based detection:	High Contrast Subspaces for Density-Based Outlier Ranking	HiCS	X	X	X	X	X	X	X	X	
Ensemble-based detection:	Feature Bagging	FB	X	X	X	X	X		X	X	
Ensemble-based detection:	Isolation Forest	IFOR	X	X	X	X	X	X	X	X	
Ensemble-based detection:	Grow When Required	GWR		X	X	X	X		X	X	
Ensemble-based detection:	Boosting Autoencoder	BAE	X	X	X	X	X	X	X	X	
Ensemble-based detection:	Lightweight online detector of anomalies	LODA	X	X	X	X	X	X	X	X	
Ensemble-based detection:	Extreme gradient boosting	XGBOD	X	X	X			X	X	X	
Mixed-based detection	Local subspace-based outlier detection	LSBOD	X	X	X	X	X	X	X	X	

2.10 Issues with outlier and anomaly detection

As new technologies emerge, the size and dimensionality of data collected from real-world scenarios grows. The data objects are nearly equidistant from each other because of their high dimensionality and complexity. Consequently, unsupervised outlier detection techniques are facing many theoretical and practical challenges. Is it implied that as data dimensions increase, any distance between objects becomes meaningless? Traditional methods and human analytical abilities alone are insufficient to detect outliers in data that is extremely rich in information. Therefore, the first concern would be identifying outliers and modeling normal objects. Methods for detecting outliers depend heavily on their ability to model normal objects and outliers. There is a question regarding how to define an outlier since applications of outlier analysis are diverse and include fault detection, intrusion detection, financial fraud, web log analytics, sensor systems, and medical applications. There is also the issue of handling noise in outlier detection. Because noise tends to be invariable in a wide range of applications domains, data sets collected across a wide range of applications have poor quality. The selection of the appropriate similarity or distance measure, along with the relationship model for describing data objects, depends on the outlier detection application, because every application has its own set of requirements. In addition, comprehension needs to be considered. Outlier detection is not the only purpose of outlier detection; users may also wish to know why the detected objects are outliers. To satisfy this requirement, an outlier detection method should be able to justify its detection in some way. What are the best methods for identifying outliers in data mining? Outliers are data points that differ greatly from the majority; therefore, to determine an outlier score, it is often necessary to model the normal patterns. However. Outliers are data points that do not follow the normal pattern. With advances in algorithms and models, it has become easier to detect outliers.

2.11 General Issues with the data and algorithms

In order to choose the right outlier and anomaly detection technique and models, one needs to understand many commonplace issues concerning data and models.

- Anomalies may degrade the final model if the training algorithm lacks robustness.
- If anomalies overlap in nominal clusters, it can be hard to detect them, and these clusters must be dense enough for a reliable model to be developed.
- Using a dataset contaminated by outliers is a major concern, as it results in inaccurate results.
- When outliers are not detected correctly, system reliability can suffer in safety critical environments, where the presence of outliers may imply abnormal activity, such as fraud.

of being extended to more advanced deep learning models as shown in[28].

- Distorted data can blur the distinction between normal objects and outliers.
- Noise and missing data may hide outliers and reduce the effectiveness of outlier detection.
- For production, testing novel data is insufficient for real-world training models in most cases.
- Based on benchmarks and complexity, which algorithms implement the best performance depends on the data type.
- Another main issue is that in real world cases the underlying distribution is usually unknown and cannot be estimated from data without outliers affecting the estimate, thus creating the chicken and the egg problem of which came first.

Unsupervised algorithms may target a different use case than these issues, which are present in many real-world datasets. There is therefore a need for a comprehensive survey analysis that brings together distinct model techniques on various datasets for various applications.

3. Evaluation of current approaches:

Anomaly detection aims to detect abnormal patterns deviating from the rest of the data, called anomalies or outliers. Anomaly detection is an important technique for recognizing fraud activities, suspicious activities, network intrusion, and other abnormal events that may have great significance but are difficult to detect [25 Srikanth]. We provide an evaluation of comparative outlier detection algorithms based on the baseline research [TOG], as well as a comparison with other research and state-of-the-art detection algorithms [1,2,3,4,5,6 add all the references for algorithms here...]. As in Table2 above, the methods below are categorized by methodology and model type to provide clarity and brevity.

3.1 Probabilistic-based detection

Gaussian Mixture Model: A Gaussian mixture model (GMM) is a model of finite mixture probability distributions. The model assumes that all data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. The algorithm is trained using the Expectation Maximization (EM) algorithm[28], which maximizes a bound on the lower bound of the likelihood iteratively. It has been applied successfully to many fields, including health cyber security and has the basis of being extended to more advanced deep learning models as shown in[29]. mixture of a finite number of Gaussian distributions with unknown parameters. The algorithm is trained using the Expectation Maximization (EM) algorithm [28], which maximizes a bound on the lower bound of the likelihood iteratively. It has been applied successfully to many fields, including health cyber security and has the basis

Dirichlet Process Mixture Model: explained in [base] Are a nonparametric Bayesian algorithm which optimizes the model parameters and tests for convergence by monitoring a nondecreasing lower bound on the log-marginal likelihood. The result is a mixture model where each component is a product of exponential-family distributions. As explained by [30]Outliers are detected by forming a maximum a posteriori (MAP) estimate of the data partition. Observations that comprise small or singleton clusters in the estimated partition are considered outliers.

Kernel Density Estimators: also called Parzen windows estimators, approximate the density function of a dataset by assigning a kernel function to each data point then summing the local contributions of the kernels. A bandwidth parameter acts as a smoothing parameter on the density shape and can be estimated by methods such as Least-Squares Cross-Validation (LSCV). As shown in [1]KDE methods are efficient when applied to novelty detection problems. However, these approaches are sensitive to outliers and struggle in finding a good estimate of the nominal data density in datasets contaminated by outliers. This issue is shown by [1] . The KDE live algorithm has been around for some time but still holds value for many future applications. Applications utilizing KDE for streaming real-time outlier detection have advanced as shown in [31].

Robust Kernel Density Estimator: According to [1], RKDE Overcomes the limitations of plain KDE by using robust loss functions, which use M-estimation methods. Kernel density estimator (KDE) is a well-known nonparametric estimator of univariate or multivariate densities, and numerous articles have been written on its properties, applications, and extensions as shown in [32].

Probabilistic Principal Component Analysis: According to [33], PPCA finds the directions of highest variance in data by decomposing the covariance matrix into orthogonal eigenvectors. When the eigenvalue of an eigenvector is higher, the variance in its direction is higher. Often, the principal eigenvectors of a lower dimensional hyperplane are the k eigenvectors with the highest eigenvalues. A score of outliers can be derived by summing up the distances between each data point and the principal eigenvectors.

Least-squares anomaly detection: According to [1], Recent work by[34]. has extended multiclass probabilistic classifiers to a one-class problem using least-squares anomaly detection (LSA) [34]. It is a probabilistic nonparametric method for anomaly detection, based on a squared-loss objective function.

Support Vector Data Description: According to [35]As with the Support Vector Classifier, it can be made flexible by employing other kernel functions to form a spherical boundary around the dataset. In addition to being robust against outliers in the training set, the method has the capability of tightening the description by utilizing negative examples. In many real-world applications, SVDD is able to improve outlier detection performance and reduce loss caused by outliers. A SVDD model, however, is built only by using the normal data, resulting in overfitting when the normal data contain noise or uncertainty due to a limited number of outliers.

***One-class SVM*:** According to [36], In contrast to the general SVM algorithm , SVMs are trained based only on normal, non-outlier observations. As a result, OCSVM is able to learn the boundaries of a distribution and classify any data points that do not fall within these boundaries as outliers. The choice of the kernel hyperparameter plays a crucial role in any SVM, and the RBF kernel is generally the most popular.

Kullback-Leibler Divergence: According to [1], the method involves training a Gaussian mixture model on a training set, then estimating the information content of new data points by measuring the kl divergence between the estimated density and the density on the training set and the new point.

Mahalanobis Distance: according to [1], can be used to detect anomalies in multivariate datasets composed of a single Gaussian-shaped cluster. It is similar to a one-component GMM with a full covariance matrix in that the model parameters are the mean and inverse covariance of the data.

Local Outlier Probabilities: According to [37], Local Outlier Probability (LoOP) is a more moderate advanced model developed by Kriegel, Kruger, Schubert, and Zimek that provides outlier scores that are directly interpreted as percentages of outliers in a sample. The LoOP model combines the idea of local, density-based outlier scores like LOF[15], its variants, and LOCI [15] with probabilistic concepts to model the "outlierness" of a point. It uses a probabilistic approach to offer a natural tolerance to noise effects in the data. Where, $PLOF\lambda, S(o) := pdist(\lambda, o, S(o))\ Es \in S(o)[pdist(\lambda, s, S(s))] - 1.$ and where$LoOPS(o) := \ max\,\{0, erf\ (PLOF\lambda, S(o)\ /\ nPLOF\ \cdot \surd\,2\ )\}$.

3.2 Neighbor-based detection

KNN: According to [14], The k-Nearest Neighbors (k-NN) algorithm measures the distance between a data point and its k nearest neighbors as a proxy for the density of data in that area. If the k nearest neighbors of an observation have all been determined, then the maximum, the median, or the median of k calculated distances, one distance for each neighbor, can all be considered measures of outlierness. As seen in the book [book], k-nearest neighbor has made great advances in outlier detection as shown in fast distance-based outlier detection by using diverse hierarchical clustering algorithms and An Effective Boundary Point Detection Algorithm Via k-Nearest Neighbor-Based Centroid.

KNN Angle-Based Outlier Detection: According to [38]The ABOD algorithm uses angles instead of distances to determine outlier observations. A variance in the angles between the difference vectors of an observation and other points is calculated for each observation. Consequently, an outlier is a datapoint located in a similar direction to the majority of another datapoints, meaning a lower variance of angles, which correlates with points at the border of a cluster. A low variance of angles, on the other hand, indicates that there are a lot of data points lying in varying directions, corresponding to the outer points of the cluster. Since the ABOD algorithm suffers less from the "curse of dimensionality", it is more suitable and computationally less expensive for high-dimensional data than distance-based approaches.

Local Outlier Factor: In the LOF algorithm [1], a datapoint's locality is determined by its vicinity and its density is estimated by its distance from its neighbors. Datapoints that have a significantly lower density than their neighbors can be considered outliers by comparing their local densities to those of their neighbors. LOF works best when the data density is not uniform throughout the dataset.

Cluster Based Local Outlier Factor: As described in [39], the CBLOF methodology uses clustering to identify high density areas in the dataset. The data is first divided into clusters using k-means, followed by a heuristic approach to further classify the clusters into small and large categories. By combining the distance of each observation from its cluster's centroid with the number of observations belonging to the cluster, the outlier confidence score can be calculated. If an observation is part of a small cluster, then the distance to the nearest large cluster will be used instead. Because CBLOF relies on k-means for clustering, its performance is highly dependent on the initial selection of the hyperparameter k.

Outlier detection using indegree number: According to [40], (ODIN) algorithm utilizes k-nearest neighbor graphs. Given a kNN graph for G dataset S , an outlier is a vertex, whose indegree is less or equal to T. An algorithm calculates each data point's indegree. In-degree refers to the number of nearest neighbors to which this point belongs. The higher this value, the more likely this point belongs to a dense area in space. Alternatively, a lesser value of this would indicate that it is not part of very many nearest neighbor sets and that it is somewhat isolated in the space. This is more or less the reverse of KNN. While studies [40]show that ODIN is outperformed when compared to other methods on synthetic data it has proven to outperform on real-world data sets and is found to be robust.

Robust Random Cut Forest: According to [41],This algorithm detects outliers in streaming data by using an ensemble method known as Robust Random Cut Forest (RRCF). Many competing algorithms do not offer the features that RRCF offers. This algorithm specializes in handling dynamic streaming data but has shown to be robust in both synthetic and real-world anomaly detection. A robust random cut forest (RRCF) is a collection of independent RRCTs. Where A robust random cut tree (RRCT) on point set S is generated as follows a random dimension proportional where, $(li\ )/(\sum\_j\ \ lj) is\ chosen\ where, li = \ maxx\ \in S\ xi\ - \ minx\ \in\ xi, Xi\ \sim\ Uniform[minx\ \in S\ xi, maxx\ \in S\ xi]\ where,\ S1\ =\ \{x|x\ \in\ S, xi\ \le\ Xi\}\ and\ S2\ =\ S \setminus\ S1\ and\ recurses\ on\ S1\ and\ S2$

3.3 Subspace-based detection

Subspace Outlier Detection: according to [1], For each point p, the algorithm finds the set of m neighbors shared between p and its k-nearest neighbors. A given subspace, which is composed of a subset of dimensions, then has an outlier score of p from its mean. Subspaces are created by selecting attributes that have a small variance over m points.

Rarity Based Outlier Detection: according to [42], An outlier definition for high-dimensional data has been proposed based on a new definition of outliers. In a dictionary of atoms learned via sparse coding, the outlierness of a data point is determined by two things: the frequency of each atom in the reconstruction of all data points (or its negative log activity ratio, NLAR), and the strength with which it is used to reconstruct the current point. This is a Rarity based Outlier Detection algorithm within a Sparse Coding Framework (RODS) that includes NLAR learning and outlier scoring. Using sparse coding and reconstruction, RODS is a fast algorithm for detecting outliers. A linear model for the data is assumed whereby each data point $\sim x$ 2 Rm is represented as the linear combination of a dictionary D ¼ ½$d\ \sim 1$; …; $d\ \sim k$ 2 RMK of non-orthogonal bases (or atoms). That is, $\sim x$ ¼ $P\ j\ d\ \sim jgj\ where\ gj$ 2 R is the coefficient corresponding to d ~j. The absolute value of gj signifies the strength of $d\ \sim j$ in representing $\sim x$. Informally, we define a data point as an outlier if it consists of one or more atoms with significant strength that rarely occur in the other observed data points.

Outrank: According to [43], this approach is capable of handling heterogeneous high-dimensional data. Researchers introduced novel scoring functions to assess the deviation of objects from the rest of the data as determined by subspace clustering analysis. Where OUTRANK extends a recent subspace clustering model [1] to heterogeneous data in a consistent manner for both types of attributes. Preliminary experiments show that our algorithm outperforms LOADED [44], a link-based approach for heterogeneous data.

Minimum Covariance Determinant: According to [15], MCD estimates the mean and covariance matrix in a way that minimizes the effect of outliers. A subset of the data should not contain outliers, the subset with the most tightly distributed data, so that these parameters can be estimated from it [45]developed the FAST-MCD algorithm to compute it efficiently. In 1929, a method was developed for identifying plants. In a one-cluster setting, the MCD is used to identify outliers, and it is subsequently extended to the multiple cluster case, leading to a robust outlier detection method [32], MCD is most useful for Gaussian-distributed data, however it could be applicable to data drawn from a unimodal or symmetric distribution; however, it should not be used with multimodal data.

3.4 Ensemble-based detection

Ensemble Gaussian Mixture Models: according to [46], this is an ensemble learning approach for density estimation using Gaussian mixture models (GMM). The ensemble GMM combines a set of individual GMM complementing each other. Research showed [46], the performance of the method in a classification task and also in estimating non-Gaussian distributions. The ensemble GMM model proved to be more accurate than competing approaches in classification. Also, the study proved that the ensemble GMM outperforms comparable complex single GMM in terms of robustness.

Randnet Model-Auto Encoders: In [47], an ensemble of autoencoders is implemented, which is a multi-layer neural network that can perform hierarchical and nonlinear dimensionality reduction. The purpose of an autoencoder is to train the output to reconstruct the input as accurately as possible. Since the nodes in the middle layers are fewer, the only way of reconstructing the input is to adjust the weights so that the intermediate outputs of the middle layers are reduced representations. Autoencoders create reduced representations of the data, making them a natural tool for discovering outliers. Outliers (or normal points) are much harder to represent accurately in this form than inliers (or normal points). As a result, the error in reconstructing an outlier will be high. In this way, the data point will be scored for each autoencoder in the ensemble.

Boosting-based Autoencoder Ensemble: According to [48] **,** BAE is an unsupervised ensemble method that builds an adaptive cascade of autoencoders using both autoencoders and ensemble boosting approaches to improve and produce robust results. As BAE trains the autoencoder components sequentially, it performs a weighted sampling of the data, in order to reduce the number of outliers while training, as well as to inject diversity into the ensemble.

High Contrast Subspaces for Density-Based Outlier Ranking: according to [49], high contrast subspaces are selected for outlier ranking using HiCS, a novel subspace search method. The method is used as a preprocessing step prior to outlier ranking. Searches for subspaces with a high contrast and a high degree of conditional dependence among the subspace dimensions. Data distributions within local subspace regions are compared with marginal distributions. Relationships between attributes highlight the contrast between subspaces. Our contrast measure is derived from these statistical tests and the detected dependence between attributes. It allows for the optimal ranking of outliers in a range of subspaces with high contrast.

Feature Bagging: The feature bagging outlier detector [50] is a meta-detector comprised of several base detectors, such as the KNN, LOF, and ABOD outlier detection methods. Using a variety of random sub-samples of the dataset, as well as averaging, these base detectors are fitted to produce a final robust detector with increased predictive power and decreased overfitting potential. Despite the same size of the random sub-sample, the sampled features differ for each trained base detector. This reduces their correlation and increases their diversity.

Isolation Forest: According to[1], [6], [15] , The Isolation Forest algorithm takes advantage of two main characteristics of outliers: Firstly, they are by definition the minority within a dataset, accounted for by a small number of instances, and Secondly, they display characteristics that deviate from the mainstream distribution. After recursively partitioning the feature space, a random feature is selected from the feature space and a split value is selected between the maximum and minimum value for the feature. As a result of these two properties, such a partitioning will most likely result in outlier observations at the root, resulting in a shorter average path length and fewer splits. Samples can be taken in isolation forests, which is one of their major benefits. New baseline methods and mixed models, which I will discuss later, have improved the performance of the IFOR algorithm in terms of consumption.

Grow When Required: According to [1], Graphs of adaptive topology lying in the input space are modeled using Kohonen networks, also called Self Organizing maps (SOM)[51]. When training the network, nodes and edges are added or removed to best fit the data. The objective is to place nodes in all dense data areas while edges propagate the displacement of neighboring nodes.

Lightweight Online Detector of Anomalies: according to [52], LoDA approximates the joint probability by using a collection of one-dimensional histograms, where every one-dimensional histogram is constructed in an input space projected onto a randomly generated vector. The rationale behind the use of one-dimensional histograms is that they can be efficiently constructed in one pass over data and the query operation needed during classification is simple. Consequently, Loda's complexity is linear with respect to the number of training samples n and the dimension of the input space d. LoDA therefore achieves a very good accuracy to complexity ratio and therefore it is well suited for processing large data.

Extreme Gradient Boosting Outlier: according to [53], XGBOD (Extreme Gradient Boosting Outlier Detection) is described and demonstrated in various practical datasets for the detection of outliers from normal observations. By creating a hybrid approach that exploits each of their individual strengths in the detection of outliers, the proposed framework combines the strengths of both supervised and unsupervised machine learning. XGBOD extracts useful representations from the underlying data using multiple unsupervised outlier mining algorithms that enhance the predictability of an embedded supervised classifier on an improved feature space using multiple outlier mining algorithms.

3.5 Mixed-based detection

Global Local Subspace Based Outlier Detection: According to [19], The GLOSS algorithm provides a general solution to the issue of Generic Local Subspace Outliers in Global Neighborhoods. In order to detect outliers in mixed data of high dimensions, GLOSS selects neighborhoods from the global data space. GLOSS outperforms state-of-the-art algorithms at finding local subspace outliers. Furthermore, the experiments show that not only can GLOSS detect outliers in local subspaces, but also it performs on par with the state-of-the-art in the regular outlier detection task.

Link-Based Outlier and Anomaly Detection: According to [44], LOADED is an algorithm for detecting outliers in data sets containing both continuous and categorical attributes. LOADED is a tunable algorithm in which one can trade off computation for accuracy so as to achieve domain-specific response times. LOADED shows excellent detection and false positive rates, which are several times better than those of the existing distance-based scheme. Links are used by the algorithm to capture dependencies. Two data points pi and pj are considered linked if they are considerably similar. Additionally, each link is associated with a link strength, which captures the degree of linkage, and is based on a similarity metric. In a domain-independent way, Loaded captures dependencies between continuous and categorical attributes.

Frequent Pattern-Based Outlier Detection: According to [54], Frequent Pattern-based Outlier Detection (FPOD), which considers interactions between different types of attributes without modifying their attributes (discretizing or recoding). As well as describing most data, patterns capture interactions between different types of attributes. Then, a new outlier factor is developed for mixed attribute data based on the notation of pattern. This means that the more an object deviates from these patterns, the higher its outlier factor will be. In mixed attribute datasets, POD uses logistic regression to acquire patterns and then formulate outlier factors.

Micro cluster-Based Detector Of Anomalies In Edge Streams: According to [55], MIDAS focuses on detecting micro cluster anomalies, or suddenly arriving groups of suspiciously similar edges, such as lockstep behavior, including denial of service attacks in network traffic data. MIDAS has the following properties: (1) it detects micro cluster anomalies while providing theoretical guarantees about its false positive probability; (2) it is online, thus processing each edge in constant time and constant memory, and also processes the data $162-644$ times faster than state-of-the-art approaches; (3) it provides 42%-48% higher accuracy (in terms of AUC) than state-of-the-art approaches.

Anomaly Detection in Sound: According to 1[56], Implementation of an optimization principle for unsupervised anomaly detection in sound (ADS) using an autoencoder (AE). Detecting unknown anomalous sounds without any training data is the objective of unsupervised-ADS. ADS is viewed as a statistical hypothesis test based on the Neyman-Pearson-Lemma[57]. In the proposed objective function, the AE[29] is trained to maximize the TPR under any low FPR condition. In order to calculate the TPR in the objective function, they consider that the set of anomalous sounds is the complementary set of normal sounds and simulate anomalous sounds by means of a rejection sampling algorithm.

4. Challenges to current approaches

Consequently, many of the proposed algorithms have limitations, while others have proven versatile and robust, which have been overcome or replaced by modern advances in technology through advances in statistical techniques for artificial intelligence. In many cases, this research shows astounding performance, yet a lack of benchmarks yet is still apropos for future work through the use of mixed models and ensemble methods. In the context of this study, it appears that outlier detection and corresponding techniques have vast potential for advancement in a number of different fields. In the context of outlier detection, the baseline paper [1], implemented sound benchmark tests on the average precision, robustness, noise-resistance, computation time and memory usage of 14 algorithms on synthetic and real datasets.

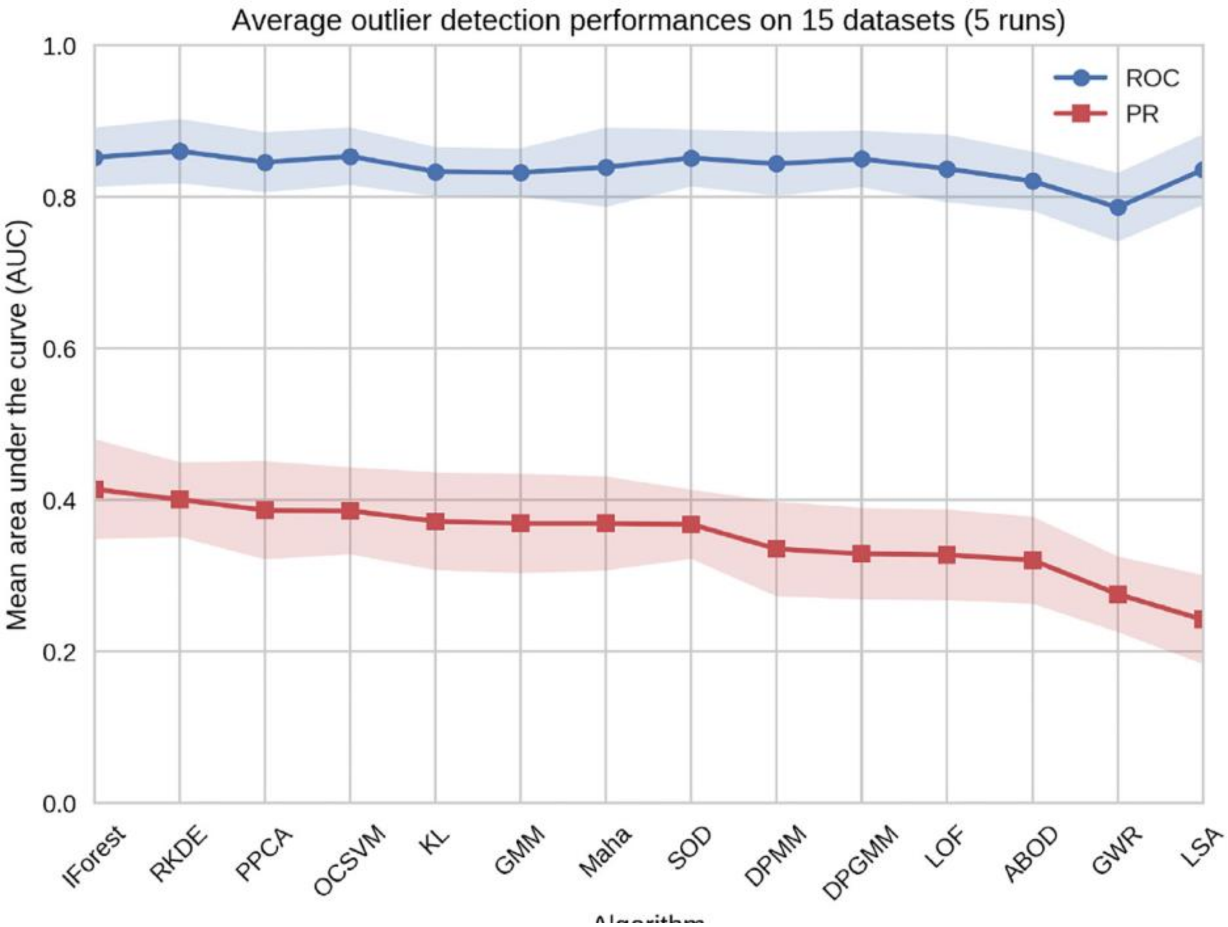


Fig. 1. Mean area under the ROC and PR curve per algorithm (descending PR).

IFOREST[1] has been shown in this study to be an excellent method for identifying outliers in large datasets while also showing excellent scalability on large datasets and a reasonable memory usage for datasets up to one million samples. According to the results, this algorithm is better suited to a production environment than RKDE, as the latter is significantly more memory intensive and computationally expensive. The study [6], [58], [59], showed that IFOREST outperformed on multiple arrays of real-time weather sensors outperformed LODA, ABOD, LOF, SVDD, and OCSVM when implemented with PAC analyst feedback[60] which is an advancement concept of getting analyst feedback on the anomalies where the most anomalous instances are presented to the analyst along with their explanations to obtain their labels. PAC learning theory[60] of anomaly detection was developed by Allen fern and then some work by Imran Siddiqui on incorporating feedback during the anomaly detection. The analyst investigates one or more items and labels each as either a true positive or false positive. This feedback is incorporated using a recent work [60] that shows that feedback can improve the anomaly detection performance significantly as shown in Fig.2.

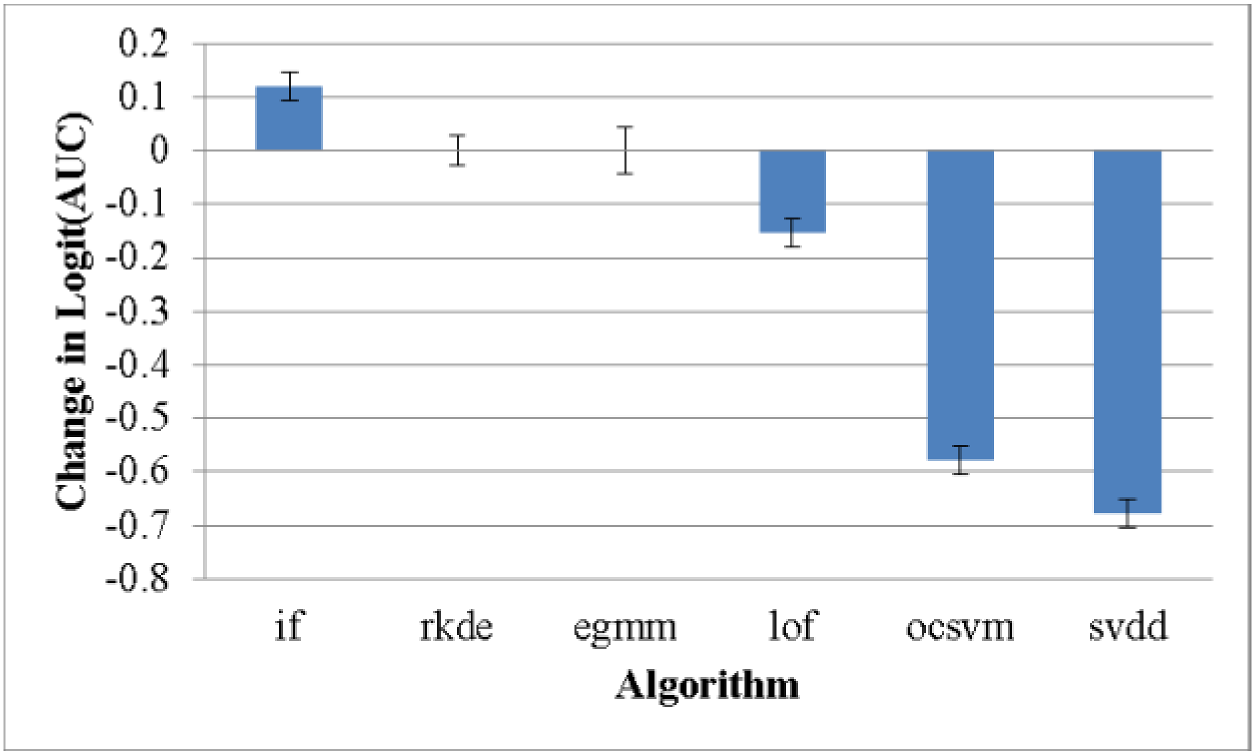
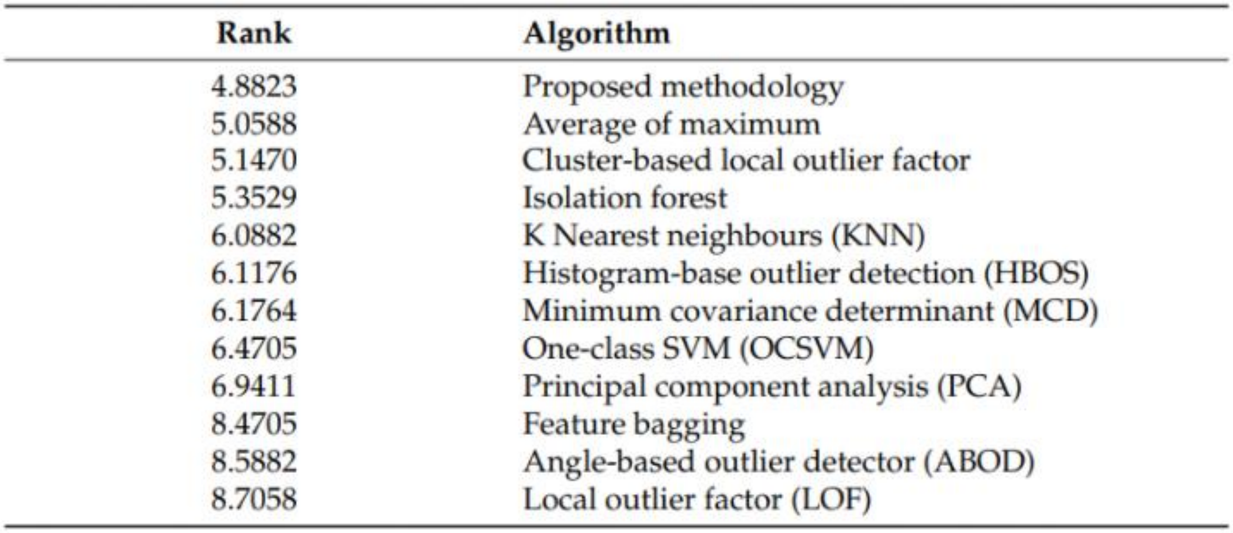


Fig.2 Performance evaluation of advanced anomaly detection algorithms

The baseline study[1] showed that KL, PPCA, and the Mahalanobis Distance were effective. However, due to their sparse performance, they did not scale due to dense clouds of outliers, which made methods such as GMM with one component more viable. However, using categorical distributions in DPMM resulted in reduced computation time and improved outlier detection. The lowest performance was achieved by LOF, ABOD, GWR, KL AND LSA, while the three first methods were also unable to scale. Whereas other studies showed in comparison that AM, KNN,HBOS,MCD, OCSVM, PCA, ABOD, IFOREST, CBLOF, AND LOF Where outperformed by ensemble meta learning algorithms when measured with Freidman comparison ROC performance metrics which implemented 17 diverse datasets of different meta-data including brain tumor diagnosis, telecommunications fraud identification, credit card fraud detection, component failure prediction, network intrusions, as shown in [61] in Fig 3.

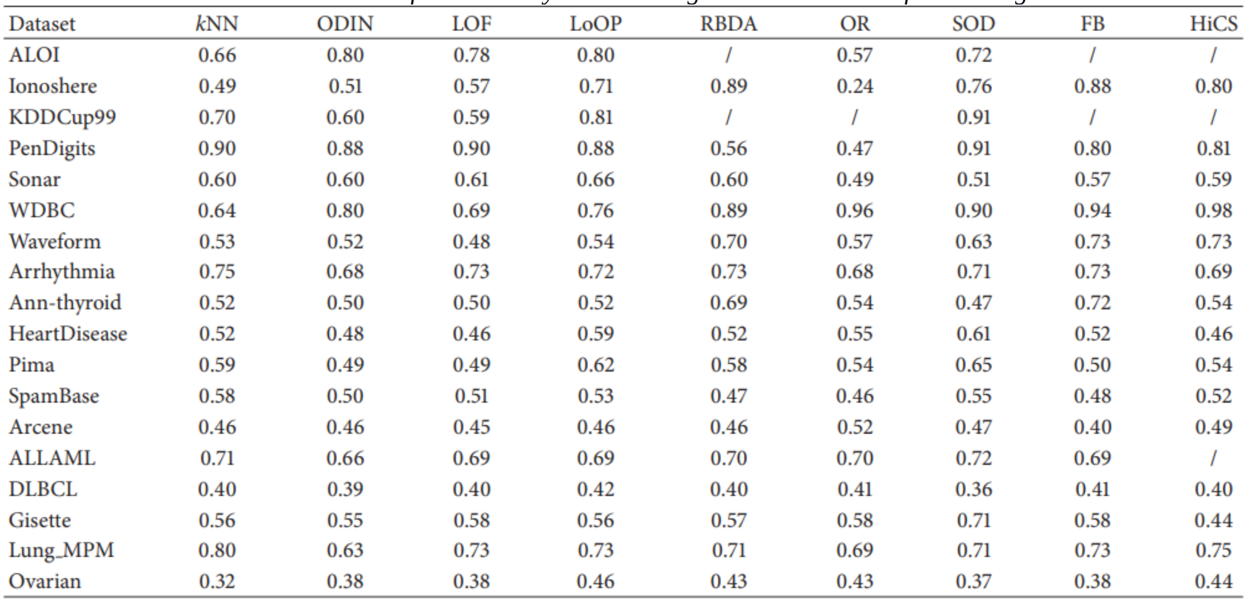


Rank	Algorithm
4.8823	Proposed methodology
5.0588	Average of maximum
5.1470	Cluster-based local outlier factor
5.3529	Isolation forest
6.0882	K Nearest neighbours (KNN)
6.1176	Histogram-base outlier detection (HBOS)
6.1764	Minimum covariance determinant (MCD)
6.4705	One-class SVM (OCSVM)
6.9411	Principal component analysis (PCA)
8.4705	Feature bagging
8.5882	Angle-based outlier detector (ABOD)
8.7058	Local outlier factor (LOF)

Fig.3 Evaluation of anomaly detection algorithms on complex data with high dimensionalities

Additional research from [15], using 18 real-world datasets gave evidence that conventional anomaly detection methods could not compare in working effectively and efficiently as dimensionality increases and data objects became more sparse, making it more challenging to detect anomalies with high-dimensional and mixed-type data. The study[15] conducted extensive experiments on publicly available datasets to evaluate the typical and popular anomaly detection methods, KNN,ODIN, LOF, LOOP, RBDA, OR, SOD,FB, and HiCS[49]. The study found that the subspace-based and ensemble-based methods have relatively good performance if the diversity of the subspaces or base learners is large. R-precision and AUC were adopted to evaluate the detection algorithms with k-fold cross validation incrementing 5,7,10,50 k-folds. 10 folds showed algorithms to be most efficient as follows.

Table 4: AUC of the anomaly detection algorithms with k=10 for the neighbors.



Dataset	*k*NN	ODIN	LOF	LoOP	RBDA	OR	SOD	FB	HiCS
ALOI	0.66	0.80	0.78	0.80	/	0.57	0.72	/	/
Ionoshere	0.49	0.51	0.57	0.71	0.89	0.24	0.76	0.88	0.80
KDDCup99	0.70	0.60	0.59	0.81	/	/	0.91	/	/
PenDigits	0.90	0.88	0.90	0.88	0.56	0.47	0.91	0.80	0.81
Sonar	0.60	0.60	0.61	0.66	0.60	0.49	0.51	0.57	0.59
WDBC	0.64	0.80	0.69	0.76	0.89	0.96	0.90	0.94	0.98
Waveform	0.53	0.52	0.48	0.54	0.70	0.57	0.63	0.73	0.73
Arrhythmia	0.75	0.68	0.73	0.72	0.73	0.68	0.71	0.73	0.69
Ann-thyroid	0.52	0.50	0.50	0.52	0.69	0.54	0.47	0.72	0.54
HeartDisease	0.52	0.48	0.46	0.59	0.52	0.55	0.61	0.52	0.46
Pima	0.59	0.49	0.49	0.62	0.58	0.54	0.65	0.50	0.54
SpamBase	0.58	0.50	0.51	0.53	0.47	0.46	0.55	0.48	0.52
Arcene	0.46	0.46	0.45	0.46	0.46	0.52	0.47	0.40	0.49
ALLAML	0.71	0.66	0.69	0.69	0.70	0.70	0.72	0.69	/
DLBCL	0.40	0.39	0.40	0.42	0.40	0.41	0.36	0.41	0.40
Gisette	0.56	0.55	0.58	0.56	0.57	0.58	0.71	0.58	0.44
Lung_MPM	0.80	0.63	0.73	0.73	0.71	0.69	0.71	0.73	0.75
Ovarian	0.32	0.38	0.38	0.46	0.43	0.43	0.37	0.38	0.44

Further research shows promise in the field of deep learning ensemble-based and mixed-based detection [], These studies show promising research with new state of the art models for detecting outliers. Several algorithms of key interest I found were RandNET[47], BAE[48], MIDAS[55], [62], XGBOD[53], and LOADED[44]. RandNET Randomized Neural Network for Outlier Detection is an ensemble of autoencoders used in order to perform anomaly detection. The autoencoder based approach for outlier detection implements a multi-layer symmetric neural network whose goal is to reconstruct the data provided in input. To achieve this goal, an autoencoder learns a new reduced representation of the input data that minimizes the reconstruction error. When using an autoencoder for outlier detection, the reconstruction error indicates the level of outlierness of the corresponding input. While studies [47] show this technique very robust and efficient it also showed its technique improved significantly over the earlier neural network methods for anomaly detection.

BAE Which is a more advanced ensemble based autoencoder method which essentially is a mixed type deep learning model. This study[] compared four outlier detection algorithms, namely LOF [32], a single autoencoder with nine layers (in short, SAE9), one-class SVM [31], and Hawkins [14]; and two ensemble techniques, i.e. HiCS [11] and RandNET [15]. And showed superior results in diversity, complexity, and performance. The study showed [29], [63] the progressive reduction of outliers enables the autoencoders to learn better representations of the inliers, which also results in accurate outlier scores. In addition, each autoencoder is exposed to a different set of outliers, thus promoting diversity among them.

Also, I found according to [64],that using ensemble techniques to improve classification is proves to be more robust based on a performance metrics and sound theory in which many empirical studies[65]–[71] have been conducted using ensemble techniques in the unsupervised clustering area In addition, the idea of combining a number of clustering results is relevant not only to ensemble clustering itself but also to related techniques such as multi-view clustering, subspace clustering, and alternative clustering [64]Combining evaluation measures using the ensemble concept has also been used [72].

4.1 Critiques

The techniques and methods for outlier detection are very rich. In some cases, there is an overwhelming amount of circumambient information-based studies that have taken baseline approaches to the next level. In addition, the baseline paper [1] I found this study focused more on novel benchmarks. Though it used some real-world datasets, the focus was based on benchmark performance more so than not. In real world applications, assumptions about real data need to be considered which I felt this study lacked. I would like to see more research in the use of feature selection and extraction techniques implemented with baseline methods. I feel this would be valuable research. Transference of application data was lacking also robustness of mixed data types was lacking. As mentioned before, the main issue is that in real world cases the underlying distribution is usually unknown and cannot be estimated from data without outliers affecting the estimate. I also felt that other metrics could have been used to gain more insight into the effectiveness of other performance metrics such as *Rank Power (RP)* where $m(m+1)2\sum i=1mRi$or *Average precision (AP)* where, $AP=1|O|\sum r=1|o|P@r$. I would have also liked to see more emphasis on testing for robust applications as well. The only algorithm that used feature selection in the baseline study was implemented with SOD, which as shown here [1], feature selection in outlier detection maximizes efficiency and overall performance. A more robust algorithm selection method should have been considered in consideration for extending the referenced baseline paper in the research. Emphasis should also be placed on considering well-defined assumptions about the data as discussed in this research. Though I do give them credit for measuring complexity, in my opinion, more research should incorporate those methods of testing computational and time complexity with higher performance metrics. It is my belief that ensemble and mixed model approaches are the most prevalent and hold the greatest advancements as algorithms which are by far the best. Performance, computation, and inference benchmarks outperform those algorithms for big data and stat-of-the-art applications as compared to those that have been around for decades. In spite of that, I do not want to minimize the old algorithms for detecting outliers, many of which depend on the application and use-case and are still very effective. Ensemble methods have shown to be highly efficient, but mixed methods that incorporate many different aspects of many different outlier algorithms yield the best results in the end according to this research. Also, other prominent techniques which have robust applications such as IFOREST, MIDAS, XGBOD are cutting edge techniques which show promise in outlier/anomaly detection. Especially with new application techniques such as analyst feedback which give historical context to solving real world applications. It is these methods of thinking outside the box to combine the best metrics which I feel will prevail in the long run

5. Summary

In this study 40 algorithms ranging from specific categories probabilistic, neighborhood, subspace, ensemble, and mixed- type methods and algorithms where evaluated. Important question to ask about any application in anomaly detection where addressed being whether we can assume that the anomalies are coming from a defined probability distribution and well-defined assumptions of data in consideration. In order to achieve the best performance and outcome, assumptions about the data must be made. Data set and relative frequency both explain variability in outlier and anomaly detection, while algorithms play a vital role in outlier and anomaly detection. Additionally, it is very risky to make assumptions about the distribution of anomaly detection. Particularly when it comes to adversarial situations such as fraud detection, insider threats, or cybersecurity. In this case, the adversaries will adapt to what the learning algorithm has discovered, so the distribution will shift. A second issue is that in the real world where there are a variety of causes of anomalies or processes, it is very risky to assume that we have a representative sample. The assumption of device failures, for example, could be misleading in sensor networks due to false positive faults and potentially disastrous when harmful faults are overlooked by models inadvertently. Furthermore, High Dimensionality poses a challenge for anomaly detection since, when the number of attributes or features increases, the amount of data required to generalize accurately increases, resulting in data sparsity, where data points are scattered and isolated. A lack of variables or a high level of noise from irrelevant attributes cause the data to be sparse, concealing the true anomalies.

6. Future Works

Future works would include extending this study with more in depth analysis on real-world production and novel datasets to test and evaluate of top outlier detection techniques and identifying key advanced algorithms for specific types defined in this study such as: Intrusion Detection, Fraud Detection, Medical Health, Industrial Damage Detection, Sensor Networks, Textual Anomaly Detection, Autonomous vehicle, Image Processing, and Sound Processing. In the future works, I would be interested in testing outlier detection models using more advanced feature selection, extraction, and performance methods for new benchmarks. Additionally, evaluate state-of-the-art and advanced operations, such as PAC analysis feedback[60] with baseline and more modern advanced ensemble and deep learning algorithms, or new techniques such as outlier interpretation (ATON)[72]–[74]. As opposed to searching subspaces, ATON directly learns an embedding space and identifies the contribution of each dimension to the outlierness of the query outlier (i.e., track the contribution of each dimension). The ATON algorithm consists of a feature embedding module and a self-attention module that are optimized using triplet deviation-based loss functions. I am particularly interested in machine learning and outlier detection applied to micro-sensor networks. This is something I would certainly like to contribute to if more research and study were devoted to this aspect of the application. Nonetheless, benchmark data sets need to be created and evaluated for real-world use. It is a pertinent issue that practitioners need access to real production data sets, which creates both a scarcity of data and a lack of trust[75], [76] in the data. There are many proactive contributions to solving these issues that I would certainly like to be a part of in the future as well.

7. References

[1] 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, Dec. 2018, doi: 10.1016/j.patcog.2017.09.037.

[2] A. C. Atkinson and D. M. Hawkins, “Identification of Outliers.,” *Biometrics*, vol. 37, p. 860, Dec. 1981, doi: 10.2307/2530182.

[3] K. Zhang and H. Jin, “An Effective Pattern Based Outlier Detection Approach for Mixed Attribute Data.” unknown, Dec. 2010. [Online]. Available: https://www.researchgate.net/publication/225104319_An_Effective_Pattern_Based_Outlier_Detection_Approach_for_Mixed_Attribute_Data

[4] K. D. D. 2015 Organisers, “Outlier and Anomaly Detection.” 2016. [Online]. Available: https://www.kdd.org/kdd2016/topics/view/outlier-and-anomaly-detection

[5] T. Zemicheal and T. G. Dietterich, “Anomaly detection in the presence of missing values for weather data quality control,” *Proceedings of the Conference on Computing & Sustainable Societies* , vol. 1, 2019, doi: 10.1145/3314344.3332490.

[6] A. Emmott, S. Das, T. Dietterich, A. Fern, and W.-K. Wong, “A Meta-Analysis of the Anomaly Detection Problem,” Mar. 2015, [Online]. Available: http://arxiv.org/abs/1503.01158

[7] E. Ted *et al.*, *Detecting insider threats in a real corporate database of computer usage activity*. 2013.

[8] A. Emmott, T. Dietterich, and A. Fern, “AD a Meta-Analysis of the Anomaly Detection Problem,” 2016. [Online]. Available: https://arxiv.org/pdf/1503.01158.pdf

[9] S. Thudumu, P. Branch, J. Jin, and J. (Jack) Singh, “A comprehensive survey of anomaly detection techniques for high dimensional big data,” *Journal of Big Data*, vol. 7, no. 1, 2020, doi: 10.1186/s40537-020-00320-x.

[10] Z. Zhao, Y. Zhang, X. Zhu, and J. Zuo, “Research on Time Series Anomaly Detection Algorithm and Application,” *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, vol. 4, no. 1, 2019, doi: 10.1109/iaeac47372.2019.8997819.

[11] 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, 2014, doi: 10.1007/s10618-014-0365-y.

[12] A. Emmott, T. Dietterich, and A. Fern, “AD A Meta-Analysis of the Anomaly Detection Problem,” 2016. [Online]. Available: https://arxiv.org/pdf/1503.01158.pdf

[13] A. Emmott, S. Das, T. Dietterich, A. Fern, and W.-K. Wong, “Systematic Construction of Anomaly Detection Benchmarks from Real Data.” 2019. [Online]. Available: http://web.engr.oregonstate.edu/~tgd/publications/emmott-das-dietterich-fern-wong-systematic-construction-of-anomaly-detection-benchmarks-from-real-data-odd13.pdf

[14] Charu C. Aggarwal, “Outlier Analysis,” *Springer*, vol. 1, no. 1, pp. 1–455, 2020, doi: 10.1007/978.

[15] X. Xu, H. Liu, and M. Yao, “Recent Progress of Anomaly Detection,” *Complexity*, vol. 2019. 2019. doi: 10.1155/2019/2686378.

[16] J. Ren, X. Liu, Q. Wang, H. He, and X. Zhao, “An Multi-Level Intrusion Detection Method Based on KNN Outlier Detection and Random Forests,” *Jisuanji Yanjiu yu Fazhan/Computer Research and Development*, vol. 56, no. 3, 2019, doi: 10.7544/issn1000-1239.2019.20180063.

[17] Z. Zheng, H.-Y. Jeong, T. Huang, and J. Shu, “KDE based outlier detection on distributed data streams in multimedia network,” *Multimedia Tools and Applications*, vol. 76, no. 17, pp. 18027–18045, 2016, doi: 10.1007/s11042-016-3681-y.

[18] V. Papastefanopoulos, P. Linardatos, and S. Kotsiantis, “Unsupervised Outlier Detection: A Meta-Learning Algorithm Based on Feature Selection,” *Electronics*, vol. 10, no. 18, p. 2236, 2021, doi: 10.3390/electronics10182236.

[19] B. van Stein, M. van Leeuwen, and T. Bäck, “Local Subspace-Based Outlier Detection using Global Neighbourhoods,” 2018. [Online]. Available: https://arxiv.org/pdf/1611.00183.pdf

[20] A. Agrawal, “Local Subspace Based Outlier Detection,” *Communications in Computer and*

Information Science, vol. 1, no. 1, pp. 149–157, 2009, doi: 10.1007/978-3-642-03547-0_15.

[21] A. Chiang and Y.-R. Yeh, “Anomaly Detection Ensembles: In Defense of the Average,” *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, vol. 1, no. 1, 2015, doi: 10.1109/wi-iat.2015.260.

[22] N. Jayanthi, D. Burra, V. Babu, and S. Rao, “An Ensemble Framework Based Outlier Detection System in High Dimensional Data,” *European Journal of Molecular & Clinical Medicine*, vol. 7, p. 2020, 2020, [Online]. Available: https://ejmcm.com/article_1813_a697785bac6d1dd6213559539a2d066b.pdf

[23] J. Chen, S. Sathe, C. Aggarwal, and D. Turaga, “Outlier Detection with Autoencoder Ensembles.” 2020. [Online]. Available: https://saketsathe.net/downloads/autoencode.pdf

[24] H. Sarvari, C. Domeniconi, B. Prenkaj, and G. Stilo, “Unsupervised Boosting-based Autoencoder Ensembles for Outlier Detection.” 2019.

[25] A. Zimek, R. Campello, and J. Sander, “Ensembles for Unsupervised Outlier Detection: Challenges and Research Questions [Position Paper].” 2015. [Online]. Available: https://www.kdd.org/exploration_files/V15-01-02-Zimek.pdf

[26] A. Ghoting, M. E. Otey, and S. Parthasarathy, “LOADED: Link-Based Outlier and Anomaly Detection in Evolving Data Sets,” *Fourth IEEE International Conference on Data Mining (ICDM’04)*, vol. 1, 2021, doi: 10.1109/icdm.2004.10011.

[27] Association for Computing Machinery. Special Interest Group on Management of Data and Association for Computing Machinery. Special Interest Group on Knowledge Discovery & Data Mining, *KDD ’13 : the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining : August 11-14, 2013, Chicago, Illinois, USA*.

[28] Guorong Xuan, Wei Zhang, and Peiqi Chai, “EM algorithms of Gaussian mixture model and hidden Markov model,” *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*, vol. 1, 2021, doi: 10.1109/icip.2001.958974.

[29] Z. Bo, S. Qi, C. Daeki, and H. Chen, “Deep Autoencoding Gaussian Mixture Model For Unsupervised Anomaly Detection,” *ICLR*, vol. 1, pp. 1–19, 2018.

[30] M. S. Shotwell and E. H. Slate, “Bayesian Outlier Detection with Dirichlet Process Mixtures,” *Bayesian Analysis*, vol. 6, Dec. 2011, doi: 10.1214/11-ba625.

[31] Z. Zheng, H.-Y. Jeong, T. Huang, and J. Shu, “KDE based outlier detection on distributed data streams in multimedia network,” *Multimedia Tools and Applications*, vol. 76, pp. 18027–18045, Dec. 2016, doi: 10.1007/s11042-016-3681-y.

[32] L. J. Latecki, A. Lazarevic, and D. Pokrajac, “Outlier Detection with Kernel Density Functions”.

[33] Z. Ma, C. B. Yun, H. P. Wan, Y. Shen, F. Yu, and Y. Luo, “Probabilistic principal component analysis-based anomaly detection for structures with missing data,” *Structural Control and Health Monitoring*, vol. 28, no. 5, p. e2698, May 2021, doi: 10.1002/STC.2698.

[34] J. A. Quinn and M. Sugiyama, “A least-squares approach to anomaly detection in static and sequential data”, Accessed: Dec. 11, 2021. [Online]. Available: http://cit.mak.ac.ug/

[35] E. J. Pauwels and O. Ambekar, “One Class Classification for Anomaly Detection: Support Vector Data Description Revisited,” *Advances in Data Mining. Applications and Theoretical Aspects*, vol. 1, no. 1, pp. 25–39, 2011, doi: 10.1007/978-3-642-23184-1_3.

[36] K. Yang, S. Kpotufe, and N. Feamster, “An Efficient One-Class SVM for Anomaly Detection in the Internet of Things,” Apr. 2021, Accessed: Dec. 11, 2021. [Online]. Available: https://arxiv.org/abs/2104.11146v1

[37] H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek, “LoOP,” *Proceeding of the 18th ACM conference on Information and knowledge management - CIKM ’09*, vol. 1, no. 1, 2009, doi: 10.1145/1645953.1646195.

[38] H.-P. Kriegel, M. Schubert, and A. Zimek, “Angle-Based Outlier Detection in High-dimensional Data,” *Ludwig-Maximilians-Universität München*, vol. 1, no. 1, pp. 1–9, 2018.

[39] Zengyou He, Xiafei Xu, and Shengehun Deng, “Discovering Cluster Based Local Outliers,” 1, 2018. Accessed: Dec. 11, 2021. [Online]. Available:

https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.20.4242&rep=rep1&type=pdf

[40] V. Hautamäki, I. Kärkkäinen, and P. Fränti, "Outlier detection using k-nearest neighbour graph," *Proceedings - International Conference on Pattern Recognition*, vol. 3, pp. 430–433, 2004, doi: 10.1109/ICPR.2004.1334558.

[41] S. Guha, N. Mishra, G. Roy, and O. Schrijvers, "Robust Random Cut Forest Based Anomaly Detection On Streams," 2020.

[42] J. K. Dutta, B. Banerjee, and C. K. Reddy, "RODS: Rarity based Outlier Detection in a Sparse Coding Framework," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 2, pp. 483–495, 2016, doi: 10.1109/tkde.2015.2475748.

[43] E. Muller, I. Assent, U. Steinhausen, and T. Seidl, "OutRank: ranking outliers in high dimensional data," *2008 IEEE 24th International Conference on Data Engineering Workshop*, vol. 24, no. 1, 2008, doi: 10.1109/icdew.2008.4498387.

[44] A. Ghoting, M. E. Otey, and S. Parthasarathy, "LOADED: Link-Based Outlier and Anomaly Detection in Evolving Data Sets," *Fourth IEEE International Conference on Data Mining (ICDM'04)*, vol. 1, no. 1, 2021, doi: 10.1109/icdm.2004.10011.

[45] P. J. Rousseeuw and K. van Driessen, "A Fast Algorithm for the Minimum Covariance Determinant Estimator," *Technometrics*, vol. 41, no. 3, p. 212, Aug. 1999, doi: 10.2307/1270566.

[46] M. Glodek, M. Schels, and F. Schwenker, "Ensemble Gaussian mixture models for probability density estimation," *Computational Statistics*, vol. 28, no. 1, pp. 127–138, 2012, doi: 10.1007/s00180-012-0374-5.

[47] T. Chang, B. Tolooshams, and D. Ba, "RandNet: deep learning with compressed measurements of images." 2019. [Online]. Available: https://arxiv.org/abs/1908.09258

[48] H. Sarvari, C. Domeniconi, B. Prenkaj, and G. Stilo, "Unsupervised Boosting-based Autoencoder Ensembles for Outlier Detection," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12712 LNAI, pp. 91–103, Oct. 2019, doi: 10.1007/978-3-030-75762-5_8.

[49] F. Keller, E. Muller, and K. Bohm, "HiCS: High Contrast Subspaces for Density-Based Outlier Ranking," *2012 IEEE 28th International Conference on Data Engineering*, vol. 28, no. 1, 2012, doi: 10.1109/icde.2012.88.

[50] A. Lazarevic and V. Kumar, "Feature bagging for outlier detection," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 157–166, 2005, doi: 10.1145/1081870.1081891.

[51] T. Kohonen, "The Self-Organizing Map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990, doi: 10.1109/5.58325.

[52] T. Pevný, "Loda: Lightweight on-line detector of anomalies," *Machine Learning*, vol. 102, no. 2, pp. 275–304, 2015, doi: 10.1007/s10994-015-5521-0.

[53] Y. Zhao and M. Hryniewicki, "XGBOD: Improving Supervised Outlier Detection with Unsupervised Representation Learning," 2018. [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1912/1912.00290.pdf

[54] Z. He, X. Xu, J. Z. Huang, and S. Deng, "FP-Outlier: Frequent Pattern Based Outlier Detection".

[55] S. Bhatia, B. Hooi, M. Yoon, K. Shin, and C. Faloutsos, "Midas: Microcluster-Based Detector of Anomalies in Edge Streams," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 3242–3249, 2020, doi: 10.1609/aaai.v34i04.5724.

[56] E. Carvalho Nunes, "Anomalous Sound Detection with Machine Learning: A Systematic Review," 2021. [Online]. Available: https://arxiv.org/pdf/2102.07820.pdf

[57] "Minimax Tests and the Neyman-Pearson Lemma for Capacities on JSTOR." https://www.jstor.org/stable/2958011 (accessed Dec. 11, 2021).

[58] T. Zemicheal and T. G. Dietterich, "Anomaly detection in the presence of missing values for weather data quality control," *Proceedings of the Conference on Computing & Sustainable Societies*, vol. 1, no. 1, 2019, doi: 10.1145/3314344.3332490.

[59] A. Emmott, T. Dietterich, and A. Fern, "AD a Meta-Analysis of the Anomaly Detection Problem," 2016. [Online]. Available: https://arxiv.org/pdf/1503.01158.pdf

[60] A. Siddiqui *et al.*, "Detecting Cyber Attacks Using Anomaly Detection with Explanations and Expert Feedback." 2020. [Online].

Available: https://www.microsoft.com/en-us/research/uploads/prod/2019/06/ADwithGraderFeedback.pdf

[61] V. Papastefanopoulos, P. Linardatos, and S. Kotsiantis, "Unsupervised Outlier Detection: A Meta-Learning Algorithm Based on Feature Selection," *Electronics*, vol. 10, no. 18, p. 2236, 2021, doi: 10.3390/electronics10182236.

[62] L. Ambalina, "Introducing MIDAS: A New Baseline for Anomaly Detection in Graphs - KDnuggets." 2020. [Online]. Available: https://www.kdnuggets.com/2020/04/midas-new-baseline-anomaly-detection-graphs.html

[63] A. Emmott, S. Das, T. Dietterich, A. Fern, and W.-K. Wong, "Systematic Construction of Anomaly Detection Benchmarks from Real Data." 2018. [Online]. Available: http://web.engr.oregonstate.edu/~tgd/publications/emmott-das-dietterich-fern-wong-systematic-construction-of-anomaly-detection-benchmarks-from-real-data-odd13.pdf

[64] A. Zimek, R. Campello, and J. Sander, "Ensembles for Unsupervised Outlier Detection: Challenges and Research Questions [Position Paper]," 2019. [Online]. Available: https://www.kdd.org/exploration_files/V15-01-02-Zimek.pdf

[65] A. Chiang and Y.-R. Yeh, "Anomaly Detection Ensembles: In Defense of the Average," *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, vol. 1, no. 1, 2015, doi: 10.1109/wi-iat.2015.260.

[66] N. Iftikhar, T. Baattrup-Andersen, F. E. Nordbjerg, and K. Jeppesen, "Outlier Detection in Sensor Data using Ensemble Learning," *Procedia Computer Science*, vol. 176, no. 1, pp. 1160–1169, 2020, doi: 10.1016/j.procs.2020.09.112.

[67] N. Iftikhar, T. Baattrup-Andersen, F. E. Nordbjerg, and K. Jeppesen, "Outlier Detection in Sensor Data using Ensemble Learning," *Procedia Computer Science*, vol. 176, no. 1, pp. 1160–1169, 2020, doi: 10.1016/j.procs.2020.09.112.

[68] N. Jayanthi, D. Burra, V. Babu, and S. Rao, "An Ensemble Framework Based Outlier Detection System in High Dimensional Data," *European Journal of Molecular & Clinical Medicine*, vol. 7, no. 4, p. 2020, 2020, [Online]. Available: https://ejmcm.com/article_1813_a697785bac6d1dd6213559539a2d066b.pdf

[69] M. Glodek, M. Schels, and F. Schwenker, "Ensemble Gaussian mixture models for probability density estimation," *Computational Statistics*, vol. 28, pp. 127–138, Dec. 2012, doi: 10.1007/s00180-012-0374-5.

[70] J. Chen, S. Sathe, C. Aggarwal, and D. Turaga, "Outlier Detection with Autoencoder Ensembles," *Proceedings of the 2017 SIAM International Conference on Data Mining*, vol. 174, no. 1, pp. 90–98, 2017, doi: 10.1137/1.9781611974973.11.

[71] H. Sarvari, C. Domeniconi, B. Prenkaj, and G. Stilo, "Unsupervised Boosting-based Autoencoder Ensembles for Outlier Detection," 2019.

[72] G. Pang, C. Shen, L. Cao, and A. van den Hengel, "Deep Learning for Anomaly Detection: A Review," *ACM Computing Surveys*, vol. 54, no. 2, Apr. 2021, doi: 10.1145/3439950.

[73] G. Pang, C. Shen, L. Cao, and A. van den Hengel, "Deep Learning for Anomaly Detection," *ACM Computing Surveys*, vol. 54, no. 2, pp. 1–38, 2021, doi: 10.1145/3439950.

[74] Y. Karadayı, M. N. Aydin, and A. S. Öğrenci, "A Hybrid Deep Learning Framework for Unsupervised Anomaly Detection in Multivariate Spatio-Temporal Data," *Applied Sciences*, vol. 10, no. 15, p. 5191, 2020, doi: 10.3390/app10155191.

[75] "How to solve data scarcity for AI." https://blog.bitext.com/how-to-solve-data-scarcity-for-ai (accessed Dec. 11, 2021).

[76] "Overcoming Data Scarcity and Privacy Challenges with Synthetic Data." https://www.infoq.com/articles/overcoming-privacy-challenges-synthetic-data/ (accessed Dec. 11, 2021).