CADI: Contextual Anomaly Detection using an Isolation Forest

Véronne Yepmo, Grégory Smits, Marie-Jeanne Lesot, Olivier Pivert

CADI: Contextual Anomaly Detection using an Isolation Forest

Véronne Yepmo
veronne.yepmo-tchaghe@irisa.fr
Université de Rennes - IRISA
Lannion, France

Grégory Smits
gregory.smits@imt-atlantique.fr
IMT Atlantique - Lab STICC
Brest, France

Marie-Jeanne Lesot
marie-jeanne.lesot@lip6.fr
Sorbonne Université - LIP6
Paris, France

Olivier Pivert
olivier.pivert@irisa.fr
Université de Rennes - IRISA
Lannion, France

ABSTRACT

Reconstructing the data inner structure and identifying abnormal points are two major tasks in many data analysis processes. A step beyond the decomposition of a data set as inliers and outliers, that then may be interpreted as anomalies, is to distinguish local from global outliers. This paper introduces a unified approach based on a revised version of an isolation forest that allows for both the reconstruction of dense regions of points, the identification of anomalies and the generation of contextual explanations about the abnormality of these points. To make the anomaly detection more informative and reliable, anomalies are compared to the reconstructed partition of the inliers so as to explain why they are considered abnormal and from which local generation mechanism they could originate from. Relying on a common data property, namely the isolation of anomalies from dense groups of regularities, eases the understanding of the data set structure and makes the provided explanations more informative than those provided by two independent mechanisms, one for clustering and one for detecting anomalies. Conducted experimentations show the relevance of the structural knowledge extracted from the proposed isolation forest and the effectiveness and robustness of the approach thanks to the unified isolation-based data model to analyse different facets of the data.

CCS CONCEPTS

• **Computing methodologies** → **Anomaly detection**; **Cluster analysis**; **Knowledge representation and reasoning**.

KEYWORDS

Anomaly detection, contextual/local anomaly, isolation forest, clustering, anomaly explanation, XAI

ACM Reference Format:
Véronne Yepmo, Grégory Smits, Marie-Jeanne Lesot, and Olivier Pivert. 2024. CADI: Contextual Anomaly Detection using an Isolation Forest. In *Proceedings of ACM SAC Conference (SAC'24).* ACM, New York, NY, USA, Article 4, 10 pages. https://doi.org/10.1145/3605098.3635969

SAC'24, April 8 –April 12, 2024, Avila, Spain

ACM ISBN 979-8-4007-0243-3/24/04…$15.00
https://doi.org/10.1145/3605098.3635969

1 INTRODUCTION

Detecting points that significantly deviate from the rest of the data set is a crucial issue in many applicative contexts as these outliers may correspond to anomalies, frauds, attacks or suspicious behaviors. Many machine learning techniques are dedicated to this task of separating outliers from the other points considered as regularities (see e.g. [6, 33]). Most of these anomaly detection approaches focus on calculating an anomaly score for each point to quantify the extent to which the point globally deviates from the rest of the data set. Other methods quantify how much the point deviates from its neighborhood. However, anomaly detectors are often embedded into decision-aid systems where their end users require explanations in addition to these scores to be convinced that an action has to be taken to deal with these anomalies. A way to increase the trust in the automatic mechanism that has built this separation of the data into two unbalanced classes (regular points *vs.* anomalies), is to provide end users with explanations about the reasons why a point is tagged as an anomaly [37].

Many anomaly detection algorithms acknowledge the existence of local anomalies, which deviate only from a subset of the data set (red circles on Fig. 1), as opposed to global anomalies which deviate from all the other instances in the data set (red square on Fig. 1). However, the distinction between local and global anomalies is often forgotten in the outputs of anomaly detectors, and during the explanation process. On the data set in Figure 1, even if the detector is able to flag all red instances as anomalies, no existing anomaly explanation method will intrinsically capture that x_1 is an anomaly for the blue cluster because of its value on A_1. This explanation can be produced only if some knowledge about the groups of regularities in the data set is available, and that knowledge is leveraged.

The COIN strategy introduced in [25] is an anomaly explanation method that makes a step towards the extraction of these so-called contextual explanations. The method identifies, for each outlier to explain, its nearest groups of inliers and the outlying attributes wrt. that neighborhood. However, COIN relies on an external anomaly detector and an external clustering algorithm. It combines different machine learning approaches to detect anomalies first, second to locally analyze the data distribution, and then to generate data structure aware explanations about each found anomaly. With such a pipeline, the accuracy of each component depends on specific

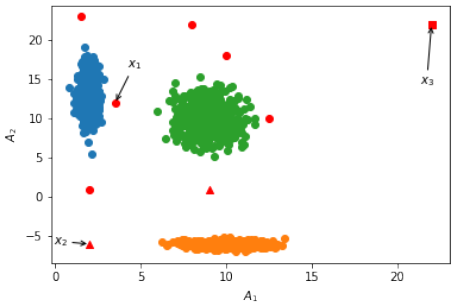


Figure 1: Illustrative example

hyper-parameters that have to be tuned and often on distances that have to be carefully selected as well. The work proposed in this paper suggests the encapsulation of these different steps in a unified approach.

The main contribution of this work is to introduce a unified approach leveraging an isolation-based data model both to identify and explain local anomalies. This approach called CADI which stands for Contextual Anomaly Detection using an Isolation forest, capitalizes on the advantages of the initial Isolation Forest (IF) method introduced in [23], namely to be accurate and interpretable with only few hyper-parameters to set and no distance measure to choose. To go beyond the detection of anomalies, a density constraint is applied on the randomly generated separation lines to ensure that they do not split dense regions whose identification is a prerequisite to the reconstruction of the data inner structure. With the identification of anomalies and the reconstruction of the data inner structure, each anomaly is then explained wrt. the identified clusters of regular data points, still using the knowledge embedded in the IF.

The rest of the paper is organized as follows. Section 2 explores the existing literature on the different components of CADI. Section 3 presents the approach whose relevance is then confirmed by experimentations conducted in Section 4.

2 RELATED WORK

CADI aims at associating each detected anomaly with informative descriptions about the strength of its abnormality and the reasons for its classification as an anomaly according to its neighborhood. This section thus positions CADI wrt. existing approaches in the fields of anomaly detection and, in particular, anomaly explanation.

2.1 Anomaly Detection

Many machine learning strategies have been proposed to address specifically the issue of identifying anomalies from a data set [6]. The unsupervised case is the most attractive one because of the unpredictability of anomalies and the difficulty of labeling data sets. Considering that regular observations are largely predominant in a data set, deep learning-based strategies identify as anomalies those points that do not fit the concise representation of the training data [38]. Local Outlier Factor (LOF) [4], One-Class Support Vector Machines [2] and IF [23] are also among the most popular unsupervised methods. They leverage the fact that anomalies are located in low density subspaces and easily separable from the other points to identify them. The Isolation Forest algorithm is very appealing for anomaly detection because it is fast, interpretable at the tree level and makes no assumption regarding the distribution of the data set. Plus, the efficiency and the robustness of the approach against the choice of the hyper-parameters have been confirmed throughout the years by different benchmarks [7, 16]. As CADI relies on a revisited version of an IF, a focus is now made on this particular technique.

Algorithm 1 Isolation Forest : *build_tree* [23]

Inputs: a sample $D \subset \mathcal{D}$, the depth d of the current node; $d = 0$ during the first call of the method
Output: a node in an isolation tree
if $|D| = 1$ or $d > h_{lim}$ **then**
  Return $node(null, null, D, d, null, null)$ ▷ Leaf (terminal node)
else
  $A \leftarrow random(\mathcal{A})$ ▷ Random attribute selection
  $v \leftarrow random([\min_{x \in D} x.A, \max_{x \in D} x.A])$ ▷ Random value selection
  $D_l \leftarrow \{x \in D / x.A < v\}$
  $D_r \leftarrow \{x \in D / x.A \geq v\}$
  Return $node(build\_tree(D_l, d+1),$ ▷ Internal node
    $build\_tree(D_r, d+1), D, d, A, v)$
end if

IF [23] identifies so called global anomalies corresponding to points that can be easily separated from the rest of the data. It is an ensemble-based algorithm as a forest is composed of t trees built on t randomly drawn subsets of the data set $\mathcal{D}$. As summarized in Algorithm 1, a tree stems from recursive splits of a data subset D on randomly chosen attributes and randomly chosen values v in the range of values observed in D for each chosen attribute A. The points with a value lower than v on attribute A are transferred to the left child of the current node, and the others to the right child. This separation process is repeated recursively until one of the following two conditions is met:

- the node is no longer separable (it contains a single point) ;
- the depth limit, a predefined hyper-parameter of the method, is reached.

The algorithm depends on three hyper-parameters: the number of trees in the forest t, the sample size Ψ and the depth limit of a tree h_{lim}. A node is formally defined by a sextuplet (LN, RN, D, d, A, v), where LN and RN are pointers to its left and right nodes respectively, $D \subseteq \mathcal{D}$, $d \in \mathbb{N}$ is its depth in the tree, $A \in \mathcal{A}$ and $v \in dom(A)$. $\mathcal{A}$ denotes the set of all attributes and for each $A \in \mathcal{A}$, $dom(A)$ denotes its domain. In Algorithm 1, the method $node(left\_child, right\_child, D, d, A, v)$ returns a new node.

Once the forest built, each point to evaluate is propagated to the leaves of each tree in the forest and an anomaly score, function of the average depth of the node containing the data point in each tree, is computed as follows [23]:

$$s(x) = 2^{-\frac{E(h(x))}{c(\Psi)}}, \quad (1)$$

where $E(h(x))$ is the average depth of the data point over the t trees. $c(\Psi)$ is a normalization factor corresponding to the average path length of unsuccessful searches in a binary tree with Ψ nodes [23].

Several variants of the IF method have been proposed in the literature. Some focus on the calculation of the anomaly score, without modifying the process of building the trees and the forest. This is the case of [28] where five new functions to compute anomaly scores are proposed. Others modify the construction of the trees but not the calculation of scores. In [22] and [17], oblique separations are used, but with different goals: detecting clusters of anomalies for the former and improving score consistency for the latter. In [9], the separations are not completely random and aim at minimizing the weighted standard deviation of the subtrees depth induced by each separation. More generally, the variants proposed in the literature focus on the outlier detection task [35].

The CADI approach proposed in this paper introduces a novel usage of an IF as a unified data model to identify anomalies, reconstruct the inner structure of the regular points and provide explanations about the found anomalies simultaneously.

2.2 Anomaly Explanation and Interpretation

In many applicative contexts, end users expect more than a decomposition of the data into regular points vs. anomalies. Explanations about the reasons for the abnormality of each anomaly can indeed increase the trust and usability of the automatically extracted knowledge. Because of the diverse nature of anomalies, anomaly explanation deserves special treatment even though it has benefited from XAI works dedicated to the explanation of classifiers in general. A general categorization of explanation methods in general and anomaly explanation methods in particular is the model-agnostic *vs.* model-specific one. A model-agnostic method provides an explanation to any anomaly detection algorithm while a model-specific method is tailored for a particular detector. Recently, more refined taxonomies of anomaly explanation methods have been proposed. In [37], four categories of explanations are identified: attribute importance explanation, attribute value explanation, point comparison explanation, and intrinsic data structure analysis explanation. In [31], the following categories of explanations are introduced: methods that rank anomalies, methods that reveal causal relationships between anomalies, and methods that identify attributes responsible for the abnormality of points or groups of points. In both cases, it is stated that techniques returning important attributes are the most common in the literature [15, 29]. However, these techniques often ignore the local context of an outlier during the explanation generation. A way to remedy this omission and to go a step further is to link anomalies with the intrinsic structure of the data set. A few methods thus consider the anomaly detection as a part of a clustering process [21, 26] or a complement of it [34].

CADI shares a same objective with the COIN (Contextual Outlier INterpretation) approach [25], namely: to help end users understand why a given point is flagged as an anomaly considering its context. The additional information associated with each anomaly in the COIN approach is a triplet containing i) an anomaly score, ii) a list of attributes supporting the decision of considering the point as abnormal and iii) a description of the local context of the anomaly. Whereas COIN has to be combined with an anomaly detector and a clustering method, the latter being applied on the neighborhood of the anomalous point to explain, CADI provides a unified and self-contained approach to reach the same goal. Another anomaly explanation method, Attention-guided Triplet deviation network for Outlier interpretatioN (ATON) [36] also includes some local information when producing the explanations. It learns the deviations between the outlier to explain and its regular neighbors in an embedded feature space. However, unlike COIN, it only outputs feature importance weights.

2.3 Outlier-Aware Clustering

Being able to reconstruct the data inner structure without being affected by the presence of outliers makes CADI a robust clustering approach. The clustering task aims at decomposing a data set into homogeneous, i.e. compact, and distinct, i.e. well separated, subgroups in the data. As such, it can be seen as summarizing the underlying data distribution and providing a legible overview of the data content. Yet most clustering algorithms suffer from the presence of outliers: the points that do not conform with the global structure of the data most often hinder the identification of regular clusters. The so-called robust clustering methods aim at addressing this issue, providing data partitions that are not perturbed by outliers: they aim at outputting the same results as would be obtained without outliers and without requiring to perform a preliminary step of outlier detection and removal. Robust clustering can be roughly categorized into two types of methods [3]: some of them proceed by automatically down-weighting atypical data points [12], using several approaches to define these weights, e.g. including noise clustering [11], possibilistic clustering [30], replacing the traditional normal distributions by multivariate t-distributions [27] or dedicated approaches [18]. Other methods propose to replace the classical squared Euclidean distance, which is known to be highly sensitive to outliers, by other distances [14]. Along the same lines, some approaches are explicitly based on robust M-estimators incorporated in the cost function [10], the possibilistic c-means [19] can be seen in this framework. These approaches define robustness as the ability to ignore the outliers, possibly grouping them in a specific cluster, as in the noise clustering approach for instance.

By aiming at providing a rich overview of the whole dataset, including the regular points inner structure and the existing anomalies, CADI is related to the approaches introduced in [20], [8] and [24] but leverages a common data structure (viz. an isolation forest) to extract these two types of knowledge.

3 THE CADI APPROACH

This section details the CADI approach focusing first on the modification made on the IF construction algorithm to avoid splitting dense subspaces that will then form clusters or parts of them. The anomaly scoring function is also revisited to take into account the impact of the density constraint on the isolation process. The same IF is then used to reconstruct a partition of the regular data points, and identify local anomalies to the found clusters. Finally, the identified anomalies are explained wrt. the clusters of regular data points.

Table 1: Notations used throughout the paper

Notation	Meaning
$\mathcal{D}$	Data set of N points
$\mathcal{A} = \{A_1, \ldots, A_m\}$	Descriptive attributes
$dom(A)$	Domain of attribute A
I^A	Set of tested intervals on feature $A \in \mathcal{A}$
$x \in \mathcal{D}$	Data point
$x.A$	Value for data point x on attribute A
t	Size of the forest $\mathcal{F} = \{T_1, \ldots, T_t\}$
Ψ	Size of the sample used to build a tree
h_{lim}	Depth limit
$\eta_i(x)$	Leaf containing x in the i-th tree
α	Margin width percentage
C	Partition of $\mathcal{D}$ in k clusters $\{C_1, \ldots, C_k\}$
$T(l)$	Tree containing the leaf l
$dca(\eta_i(x), l_i)$	Deepest common ancestor between the paths from the root of $T(l_i)$ to l_i and $\eta_i(x)$

3.1 Density-Aware Isolation Forest

As reminded in Sec. 2.1, in the classical IF approach, an isolation tree is built through recursive splits of a data subset D. A split is a couple (A, v), where A is a randomly chosen attribute $A \in \mathcal{A}$ and v a value from its observed domain $v \in [\min_{x \in D} x.A, \max_{x \in D} x.A]$. CADI revisits this completely random process to keep only the splits that fall in low density regions. Anomalies being by definition detached from regular phenomenon materialized by dense subspaces, the objective of this revisited isolation algorithm is to find separation lines in low density areas surrounding dense regions. To do so, a density-based constraint is added and determines if a split is maintained or discarded. The hypothesis is the following: if a significant number of points are found in the neighborhood of the split, it is potentially separating a cluster. In that case, the split is discarded and another one is generated. The goal is to surround the clusters of regular points by the separations, so that some leaves may contain a cluster, or a significant portion of a cluster. One hyper-parameter denoted α is introduced in addition to the IF hyper-parameters to control the size of the margin around the separation which represents its neighborhood. Alg. 2 details how density-aware isolation trees are constructed and Fig. 2 illustrates an example of split selection. The split (A_1, v_1) on Fig. 2 is discarded as many points are located in the margin surrounding the separation. It is not the case for the split (A_2, v_2) that is kept.

Compared to the initial IF algorithm, a cost overhead is undeniably induced by this split selection, as in the original IF approach the splits are randomly drawn. However, to learn from the discarded splits and to avoid generating separations in intervals that have already been discarded because they contain many data points, the set of tested intervals on each attribute $\mathcal{J} = \{I^{A_1}, \ldots, I^{A_m}\}$ (I^A being the set of tested intervals on attribute A) is stored and passed as a parameter through the recursive calls to the *build_tree* function (line 20 in Alg. 2). If the method was not able to find a valid separation in the whole interval of values of an attribute (line 10), this attribute is discarded (line 11). The discarded attributes are therefore also stored, in the variable C. If the method is unable to find a valid separation on any attribute (line 3), then a terminal

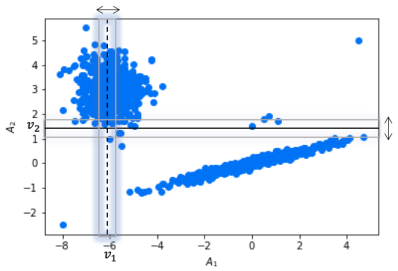


Figure 2: Example of a discarded separation line (A_1, v_1) falling in a dense area (dashed line) and a validated separation (A_2, v_2) (plain line)

Algorithm 2 CADI : *build_tree*

1: **Inputs:** data sample $D \subset \mathcal{D}$, depth d of the current node, margin width percentage α, set of tested intervals $\mathcal{J} = \{I^{A_1}, \ldots, I^{A_m}\}$, set of covered attributes C; $\mathcal{J}$ and C are empty when the method is first called, and $d = 0$
2: **Output:** a node in an isolation tree
3: **if** $C = \mathcal{A}$ or $|D| = 1$ or $d > h_{lim}$ **then**
4: Return $node(null, null, D, d, null, null)$ ▷ returns a leaf
5: **else**
6: $A \leftarrow random(\mathcal{A} \setminus C)$ ▷ random attribute selection
7: $v \leftarrow random([min_{x \in D} x.A, max_{x \in D} x.A] \setminus \cup_J \{J \in I^A\})$ ▷ random value selection
8: $marg \leftarrow \frac{1}{2}\alpha(\max_{x \in D} x.A - \min_{x \in D} x.A)$
9: $I^A \leftarrow I^A \cup [v - marg, v + marg]$
10: **if** $\exists J \in I^A \; st. \; [\min_{x \in D} x.A, \max_{x \in D} x.A] \subseteq J$ **then**
11: $C \leftarrow C \cup \{A\}$ ▷ A is entirely scanned
12: **end if**
13: $D_m \leftarrow \{x \in D / x.A \in [v - marg, v + marg]\}$ ▷ points in the margin
14: **if** $|D_m| \leq \alpha \times |D|$ **then**
15: $D_l \leftarrow \{x \in D / x.A < v\}$
16: $D_r \leftarrow \{x \in D / x.A \geq v\}$
17: Return $node(build\_tree(D_l, d + 1, \alpha, \emptyset, \emptyset),$
18: $build\_tree(D_r, d + 1, \alpha, \emptyset, \emptyset), D, d, A, v)$ ▷ return an internal node
19: **end if**
20: Return $build\_tree(D, d, \alpha, \mathcal{J}, C)$ ▷ select another split
21: **end if**

node is returned (line 4), the current set of points being considered as inseparable.

In a tree generated by CADI a leaf may be of three different types depending on the termination condition that yields it. An **Isolation Node** (IN) stores a data point that has been isolated from the rest of the dataset, it is generated when $|D| = 1$. A terminal node is called a **Dense Node** (DN) if it gathers a set of inseparable points, formally if $|D| > 1$ and $C = \mathcal{A}$ (l.3 in Alg. 2). Finally, a **Depth-Limit Node** (DLN) is such that $d = h_{lim}$ and $C \neq \mathcal{A}$. Whereas the classical IF algorithm yields only nodes of type IN and DLN, the nodes of type DN induced by the density constraint applied on

the randomly generated splits are particularly informative in the prospect of reconstructing the data inner structure (Sec. 3.2).

As compared to a classical IF, a CADI forest induces an additional cost related to the storage of the excluded intervals. As they are stored only for one node at a time, this overhead is a constant. The time complexity differs from that of a classical IF by the selection of the separations. This difference is, in the worst case, linear with respect to the number of attributes: $O(|\mathcal{A}|)$.

3.2 Clustering from an IF

A CADI forest has three types of terminal nodes: IN, DN, and DLN. IN leaves contain potential anomalies, as the data points were isolated. DLN leaves are those containing points which have not been separated after a certain number of splits, just like in IF. DN leaves contain points that cannot be separated no matter the attribute. They therefore gather dense group of points, corresponding to clusters or portions of clusters. As a result, in order to obtain a partition of the data set, these leaves need to be combined.

The combination strategy of CADI's DN leaves is inspired by grid-based clustering [1] where the feature space is first partitioned by a grid. Each cell of the grid is a combination of intervals on the attributes in $\mathcal{A}$, and contains some points. Then, contiguous *dense* cells are merged to form clusters. In practice, a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is built. Each vertex of the graph is a cell and there is an edge E between two vertices $V_1, V_2 \in \mathcal{V}$ if the corresponding cells are contiguous. The connected components of the graph are later extracted, and each connected component is a cluster. Like cells in grid-based clustering, DN leaves in CADI contain data points and delimitate dense regions of the data space. However, there are some major differences between the two units. First, DN leaves coming from different trees may have some points in common (because of the sampling), whereas grid cells are disjoint in terms of points. Second, the subspaces delimitated by DN leaves may overlap. As a result, instead of adding an edge between two vertices if the corresponding leaves are contiguous, an edge E is created if the two leaves are somewhat similar in terms of points. This similarity is measured by the *Jaccard index* between the two leaves and corresponds to the weight of E:

$$w_E = \frac{|V_1 \cap V_2|}{|V_1 \cup V_2|},$$

with $E = (V_1, V_2)$. V_1 and V_2 are the sets of points contained is the respective leaves. Each connected component is a cluster. This strategy allows to automatically discover the number of clusters in the data set, just like in grid-based clustering. The data points not assigned to a cluster, because they were not part of any sample used to build the forest, are propagated through each tree until they reach a terminal node. A majority vote is then performed among the clusters of the corresponding DN leaves. Before the extraction of the connected components, a preprocessing step is applied: the leaves included in other leaves are deleted from $\mathcal{V}$ to remove redundancies.

3.3 Anomaly Interpretation

In the COIN [25] approach, an outlier explanation is composed of:

(1) a quantification of the point abnormality,
(2) a local positioning wrt. its surrounding regularities, which is equivalent for the method to a set of the clustered neighbors of the point,
(3) a subset of attributes weighted by their relative contribution to the abnormality of the suspicious point.

Similarly to COIN, the CADI approach provides for each outlier:

(1) a quantification of the point abnormality,
(2) a quantification of its deviation wrt. each identified cluster,
(3) a subset of attributes weighted both by their relative contribution to bringing the suspicious point closer to each cluster and their relative contribution to the abnormality of the suspicious point wrt. each cluster.

Anomaly Score. Leveraging the property that it is more likely to isolate anomalies than regularities using random splits, the anomaly score of a given point is computed in the original IF approach as a function of its depth of isolation in the different trees of the forest. Using the CADI approach, and due to the density constraint imposed on the randomly generated splits, a dense region may be completely scanned without increasing the depth of the tree, thus leading to a leaf of type DN containing a high number of inseparable points located at low depth. To differentiate leaves of type IN from those of type DN, it thus makes more sense to define an anomaly score based on the cardinality of the set of points isolated in a same leaf instead of its depth. Equation 2 is used to calculate an anomaly score for a given point x that depends on the cardinality of the node it is isolated in:

$$s_i(x) = 1 - \frac{|\eta_i(x)| - 1}{\Psi}, \tag{2}$$

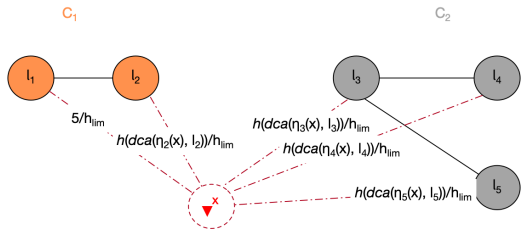
where $\eta_i(x)$ is the node containing x in the i-th tree. The score $s_i(x)$ varies in $]0, 1]$ taking its maximum value when x is isolated alone in an IN leaf and is close to 0 when the whole data subset ends in a same leaf. The latter situation occurs when no separation line can be validated on the whole universe: the dataset consists of a single indivisible cluster.

The global anomaly score is the average over the whole forest containing t trees:

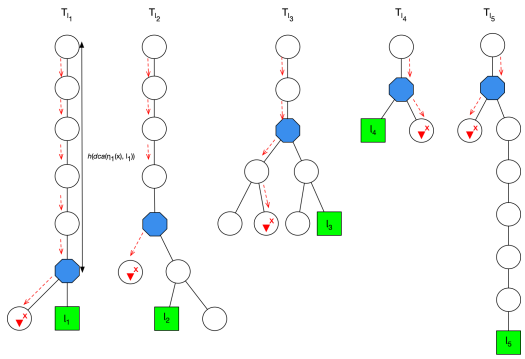
$$s(x) = \frac{1}{t} \sum_{i=1}^{t} s_i(x). \tag{3}$$

Local Structure-Aware Anomalies. In its original version, an IF detects points that may be easily separated from the rest of the dataset, leading to so-called global anomalies. Local anomalies may also be identified by an IF, but no distinction between local and global anomalies is made in the output of the method. Many recent works (see Sec. 2.2) focus on providing users with more informative descriptions of the data set, especially through a local contextualization of the found anomalies. It is now shown that a forest generated by CADI embeds all the necessary structural knowledge to identify possible links between anomalies and clusters.

Let x be a point whose anomaly score $s(x)$ is sufficiently high to be considered as an anomaly. The next step is to determine for each cluster $C \in \mathcal{C}$ whether x can be considered as an abnormal deviation of the regular phenomenon modelled by C. Let $\{l_1, \ldots, l_p\}$ be the set of DNs making up C, and $T(l_i), i = 1 \ldots p$ be the tree in which l_i is found. Still using the structural knowledge embedded in $T(l_i)$ only, viz. without having to choose an appropriate distance



(a) Clusters C_1 and C_2



(b) Deepest common ancestors (blue octogons) between x and the l_is (green squares). Partial trees are displayed.

Figure 3: Contextual/Local anomaly detection: leveraging CADI trees and DN leaves

measure, a contextual score denoted by $c(x, C)$ is computed as an aggregation of the comparisons between x and the l_is forming C. In a tree, e.g. $T(l_i)$, the path from the root to the leaf l_i implies different separations each narrowing the subspace originally enclosed by the root. As a result, if x and the points in l_i are found in the same node deep in the tree, they are more likely to be close from each other in the feature space. In that case, if x is separated from the points in l_i, it is more likely to be deviating from l_i. By applying this principle to all the l_is in a cluster C, a score corresponding to the local deviation of x wrt. C is computed. This contextual score depends on the depth of the deepest common ancestor (dca) between the node containing x in the i-th tree $\eta_i(x)$, and each l_i in the corresponding $T(l_i)$:

$$c(x, C) = \frac{1}{p} \sum_{i=1}^{p} \frac{h(dca(\eta_i(x), l_i))}{h_{lim}}, \tag{4}$$

where $dca(\eta_i(x), l_i)$ refers to the deepest common ancestor of $\eta_i(x)$ and leaf node l_i, and $h(dca(\eta_i(x), l_i))$ is its depth.

The process described above is illustrated on Figure 3. On Fig. 3a, two clusters C_1 and C_2 have been identified: $C_1 = \{l_1, l_2\}$ and $C_2 = \{l_3, l_4, l_5\}$. The contextual score of x with each cluster is computed using the depth of the deepest common ancestor between x and each l_i (Fig. 3b). These contextual scores are therefore given by: $c(x, C_1) = (5/h_{lim} + 4/h_{lim})/2 = 9/2h_{lim}$ and $c(x, C_2) = (2/h_{lim} + 1/h_{lim} + 1/h_{lim})/3 = 4/3h_{lim}$. As a conclusion, x is a local anomaly of C_1.

Common and Discriminating Attributes. In addition to the contextual scores computed between an anomaly x and each cluster $C \in \mathcal{C}$, it is possible and particularly interesting to determine which attributes make of x a contextual anomaly of a given cluster C. Each attribute A is thus associated with a couple of weights forming the vector $e(A, x, C)$. The first item of the couple indicates how much the value possessed by x on A ($x.A$) is shared by other members of C. The second item of the couple quantifies how much A makes x an anomaly of C. To compute these weights, the paths in each $T(l_i)$ from its root to $\eta_i(x)$ and l_i respectively are analyzed like in Fig. 3b. In a tree $T(l_i)$, the attribute involved in $dca(\eta_i(x), l_i)$, attribute denoted by $A(dca(\eta_i(x), l_i))$ in Equation 5, is considered as an explanation of the reason why x is an anomaly, whereas the other attributes involved in the path from the root of T_{l_i} to $dca(\eta_i(x), l_i)$ (dca excluded) describe the context shared by l_i and x. The couple represented by the vector $e_{l_i}(A, x)$ quantifies the contribution of attribute A to explain x as a local anomaly of the dense subset of points gathered in the leaf l_i.

$$e_{l_i}(A, x) = \begin{pmatrix} \omega(A) \\ 1 \text{ if } A = A(dca(\eta_i(x), l_i)) \text{ and } 0 \text{ otherwise} \end{pmatrix} \tag{5}$$

where $\omega(A)$ is the number of times attribute A is used as a separation attribute in the path from the root of $T(l_i)$ to $dca(\eta_i(x), l_i)$ (dca excluded).

At the forest and cluster levels, the weight vector attached to an attribute A to quantify its contribution to explain why A is an anomaly of C is computed as follows:

$$e(A, x, C) = \frac{1}{p} \sum_{i=1}^{p} e_{l_i}(A, x). \tag{6}$$

4 EXPERIMENTS

The objective of this section is to evaluate the proposed CADI approach. Answers to the following questions are sought: 1) Is CADI able to accurately detect anomalies in a data set? 2) Is CADI able to provide an accurate partition of the data set? 3) Is CADI able to identify contextual anomalies wrt. clusters of regular data points? 4) Is CADI able to provide accurate contextual explanations for the anomalies? Each component of CADI is therefore evaluated.

4.1 Data Sets and Experimental Setup

As anomaly detection has been extensively explored in the literature, several real-world data sets for the assessment of this task exist. Eleven real-world and two synthetic data sets [32], described in Table 2, are used to evaluate the anomaly detection step.

Unfortunately, there is no information neither on the presence of clusters in these data sets, the locality of anomalies, nor on the ground-truth explanations. In general, evaluating an explanation on real-world data sets is not an easy task, because of this absence of ground-truth. With some knowledge about the data, it is possible to have an insight on these ground-truth explanations. In the anomaly explanation literature, a common practice is to add controlled noise attributes [5, 25]. The hypothesis behind this practice is that true explanations should lie among the original (viz. not noise) attributes. We believe that an evaluation using this scheme is not faithful enough, since the true outlying attributes must be part of the original ones. This scheme therefore only evaluates the ability of a method to provide non-aberrant explanations. In [36], another technique to generate ground-truth explanations on real-world

Table 2: Real-world data sets

Data set $\mathcal{D}$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{D}\rvert$	# anomalies
Annthyroid	6	7200	534
Arrhythmia	271	420	57
Breast	9	683	239
Cover	10	286048	2747
HBK	4	75	14
HTTP	3	567498	2213
Ionosphere	32	351	126
Mammography	6	11183	260
Pima	8	768	268
Satellite	36	6435	2036
Shuttle	9	58000	3511
SMTP	3	95156	30
Wood	6	20	4

data sets is proposed. The outlying degree/score of *real* outliers in every possible subspace of the original feature space is computed. The ground-truth explanation is the subspace where the anomaly receives the highest score. Three different anomaly detectors are used, among which IF. Depending on the detector used, there are different ground-truth outlying attributes that are used separately during the evaluation. With this scheme, there is no information regarding the possible clusters of regular data points.

In contrast to real-world data sets, the true outlying attributes are known during the generation of synthetic data sets. Furthermore, since the generation process is known, data sets containing clusters and local anomalies can be produced. Since the best usage of CADI is this setting, only synthetic data sets are used for the evaluation of the subsequent components of the method. We follow a strategy similar to the one described in [25] for the generation of synthetic data sets. In the first synthetic data set, each anomaly is close to only one cluster. The outlying attributes wrt. the corresponding cluster constitue the ground-truth. In the second data set, some anomalies share some attributes values with more than one cluster. The third data set contains anomalies that deviate from all the clusters. The last cluster contains, in addition to local and global anomalies, a cluster of anomalies. To evaluate the ability of CADI to discover non-spherical clusters, the second and third data sets contain stretched clusters. In addition to that, the fourth data set is the moons data set generally used to evaluate clustering. It is composed of two interleaving half circles, to which we manually added outliers. Information about the synthetic data sets generated are summarized in Table 3.

To enhance the reproducibility of our experimental results, the code and the synthetic data sets along with the ground truths are publicly available [1].

The parameters of CADI are set to these default values: $t = 100$, $\Psi = 256$, $h_{lim} = 8$ and the margin size is $\alpha = 5\%$ of the attribute's initial range. The default values of t, Ψ and h_{lim} are the same as for IF. The intuition behind a fixed value of α is the following : if two points are separated by less than that $\alpha \times range(A)$ on an attribute A, they should remain together during the tree building process. However, the value of that parameter can be adjusted with some knowledge about the data. For example, if the user wants to keep together data points having a difference in values on a specific attribute A less than a quantity β, then the value of α for this attribute can be set to $\beta/range(A)$. This fixed value of α causes an update on l.14 in Alg. 2. If the data distribution is uniform, then $marg/(\max_{x\in D} x.A - \min_{x\in D} x.A) \times |D|$ points should be expected in the margin. Nevertheless, α is an upper bound of the quantity $marg/(\max_{x\in D} x.A - \min_{x\in D} x.A)$, as the quantity $(\max_{x\in D} x.A - \min_{x\in D} x.A)$ decreases with separation. l.14 is therefore not modified in the implementation and the experiments.

[1] https://gitlab.com/yveronne/cadi

4.2 Anomaly Detection

This part of the experiments aims at evaluating CADI in terms of anomaly detection. To this end, the real-world and synthetic data sets are used. For each data set, the ground-truth anomalies are known. The Area under the Receiver Operating Characteristic curve (AUC/AUROC) is computed. AUC is an appealing method for anomaly detection evaluation because it is independent of the outlying degree threshold. It represents the probability that an anomaly receives a higher score than a regular data point. CADI is compared to the classical IF method in terms of AUC. For comparison between IF and other anomaly detection algorithms, some benchmarks, like [16], are available. The means and standard deviations of the AUC after ten runs of CADI and IF are reported in Table 4.

In general, there is no significant difference between CADI and IF. However, CADI performs much better than IF on two data sets in particular: *mammography* and the synthetic data set *wood*. That performance increase is due to the fact that these data sets contain regular data points located in sparse regions of the feature space. As CADI combines the separability information and the local density information during the split selection, the method is able to make a better distinction than IF between regular data points located in sparse subspaces and isolated anomalies. IF in contrast uses only the separability information when computing the anomaly score, leading to a less precise distinction between regular data points located in sparse subspaces and anomalies. Another observation is that the standard deviations of the anomaly score with CADI are generally lower, implying that the results obtained are more stable. This stability is caused by the non-completely random selection of the separations in CADI.

4.3 Clustering Evaluation

To evaluate the clustering component of CADI, a forest is built for each synthetic data set. Anomalies are identified by choosing a score threshold of 0.95. Then, a partition of the regular data points is extracted by the procedure described in Sec. 3.2. For each synthetic data set, the true clustering labels of each data point are known. As a result, the Adjusted Rand Index (ARI) is used as evaluation metric. To have a ground-truth partition for computing the ARI, we assign all the known anomalies to a cluster, and the regular instances are assigned to their respective true cluster. CADI is compared with the density-based clustering algorithm DBSCAN [13] and the robust clustering algorithm $k-$means-- [8]. The choice of DBSCAN for comparison is motivated by three main reasons: 1) DBSCAN identifies not only the clusters of regular data points but

Table 3: Synthetic data sets

Data set $\mathcal{D}$	$\lvert\mathcal{A}\rvert$	# clusters	$\lvert\mathcal{D}\rvert$	# anomalies	Description
$\mathcal{D}_1$	2	2	900	25	Spherical clusters. Local anomalies only.
$\mathcal{D}_2$	2	3	1508	8	Two spherical and one stretched clusters. Local and global anomalies.
$\mathcal{D}_3$	3	4	408	8	Four stretched clusters. Each pair of clusters located in only two dimensions Local and global anomalies.
$\mathcal{D}_4$	2	2	517	17	Two moons. One anomaly cluster. Local and global anomalies.

Table 4: Anomaly detection performance: AUC

Data set	CADI	IF
Annthyroid	0.772 ± 0.014	**0.819 ± 0.013**
Arrhythmia	**0.812 ± 0.007**	0.775 ± 0.045
Breast	**0.994 ± 0.001**	0.981 ± 0.004
Cover	0.832 ± 0.027	**0.873 ± 0.022**
HBK	1.0 ± 0.0	1.0 ± 0.0
HTTP	0.998 ± 0.002	**0.999 ± 0.001**
Ionosphere	0.827 ± 0.009	**0.855 ± 0.005**
Mammography	**0.844 ± 0.011**	0.645 ± 0.037
Pima	**0.702 ± 0.007**	0.684 ± 0.010
Satellite	**0.700 ± 0.014**	0.699 ± 0.016
Shuttle	0.992 ± 0.002	**0.995 ± 0.001**
SMTP	0.883 ± 0.011	**0.889 ± 0.008**
Wood	**0.956 ± 0.061**	0.868 ± 0.069
$\mathcal{D}_1$	0.999 ± 0.001	0.999 ± 0.001
$\mathcal{D}_2$	0.965 ± 0.005	**0.979 ± 0.004**
$\mathcal{D}_3$	0.974 ± 0.005	**0.995 ± 0.002**
$\mathcal{D}_4$	**0.990 ± 0.001**	0.972 ± 0.005
Mean AUC	**0.8965**	0.8839

also the anomalies. 2) DBSCAN, as CADI, automatically discovers the number of clusters and therefore does not require it as an input parameter. 3) DBSCAN is able to discover non-elliptical clusters. $k-$means-- are a variant of the classic $k-$means clustering algorithm. The approach has two parameters: the number k of clusters to discover and the number l of outliers in the data set. The l farthest points are discarded during the cluster centers updates. We set the values of k and l to their true values for each dataset. The default parameters of DBSCAN are used on $\mathcal{D}_1$, $\mathcal{D}_2$ and $\mathcal{D}_3$. The most important parameter of DBSCAN is ϵ which controls the size of the neighborhood. On $\mathcal{D}_4$, the default value of ϵ does not produce good results. It was fine-tuned. On that same data sets, the weakest edges were removed by CADI to obtain two clusters. The best ARI obtained when using CADI, DBSCAN and $k-$means-- on the data sets are reported in Table 5.

$k-$means-- obtain the best ARI on $\mathcal{D}_1$ and $\mathcal{D}_2$. This behavior was expected, since the classic $k-$means algorithm is efficient for extracting spherical clusters. On $\mathcal{D}_3$ and $\mathcal{D}_4$, $k-$means-- struggles, just like the original $k-$means struggles on these data sets in the absence of anomalies. DBSCAN and CADI on the other hand do not struggle to discover non-elliptical clusters. CADI obtains the best ARI in average.

Table 5: Clustering performance: ARI

Data set	CADI	DBSCAN	$k-$means--
$\mathcal{D}_1$	0.986	0.914	**1.0**
$\mathcal{D}_2$	0.970	0.963	**0.994**
$\mathcal{D}_3$	0.936	**0.971**	0.333
$\mathcal{D}_4$	0.992	**0.999**	0.287
Mean	**0.971**	0.962	0.653

Table 6: Contextual anomaly detection performance

Data set	Precision	Recall
$\mathcal{D}_1$	1.0	1.0
$\mathcal{D}_2$	1.0	1.0
$\mathcal{D}_3$	0.875	0.875
$\mathcal{D}_4$	0.941	1.0

4.4 Contextual Anomaly Detection

In addition to identifying anomalies and groups of regular data points, CADI provides some insight about the cluster(s) from which each anomaly may be deviate. As, to the best of our knowledge, no method in the literature is able to do so, there is no baseline for comparison. However, since the true cluster assignments are known for each anomaly in the generated data sets, the performance of CADI regarding the contextual anomaly detection can be evaluated. To do so, for each outlier o, let $\mathcal{P}$ be the set of predicted clusters for o, i.e the set of clusters from which o may be deviating according to CADI. Let $\mathcal{T}$ be the set of ground-truth clusters for o, i.e the set of clusters from which o is deviating. The precision and recall are computed as $precision = |\mathcal{P} \cup \mathcal{T}|/|\mathcal{P}|$ and $recall = |\mathcal{P} \cup \mathcal{T}|/|\mathcal{T}|$. For each data set, the precisions and recalls are averaged over the outliers. The results are shown in Table 6. CADI performs well on all the data sets, with perfect precision and recall on $\mathcal{D}_1$ and $\mathcal{D}_2$. The method is more challenged on $\mathcal{D}_3$ as this data set contains several data points deviating from more than one cluster.

4.5 Contextual Explanations

The last part of CADI's assessment concerns the generated contextual explanations. CADI outputs for an outlier o a list of discriminating feature weights wrt. each identified cluster (second items of the pairs in Eq. 6). COIN [25] also outputs a list of feature weights, but with respect to the local context of o only, and not the set of clusters in the data set. ATON [36] on the other hand does not take the local

context of o into account, but still outputs a list of feature weights. Both COIN and ATON need as input the outlier o to explain. It is not the case for CADI which performs anomaly detection prior to the explanation. COIN and ATON will nonetheless serve as baseline for the evaluation. In addition to these two, CADI is compared to the ground-truth explanation extraction procedure introduced in [36] and detailed in Section 4.1. The IF is used as detector for this method called ATON-GT from here onwards. ATON-GT outputs for each *specified* outlier o, the subspace in which it receives the highest anomaly score. The implementations of COIN, ATON and ATON-GT were kindly made available by their respective authors.

Considering all the above-mentioned differences across CADI, COIN, ATON and ATON-GT, we propose the following evaluation procedure to provide a fair comparison:

- Since COIN, ATON and ATON-GT do not identify outliers *per se*, explanations for *true* outliers only are requested from all the four methods. This allows to also evaluate the ability of CADI to provide accurate explanations even for outliers the method was not able to identify as such.
- The ground-truth explanations are a list of discriminating attributes wrt. each cluster. As a result, for the methods outputting feature importance scores (CADI, COIN and ATON), the top k discriminating features are retrieved, with k being the length of the *true* explanatory subspace.
- For all four methods, the precision and recall of the explanations are computed in a similar manner as for contextual anomaly detection performance evaluation. This procedure is also used in [36]. For each data set, the precision and recall over all the outliers are computed.
- As CADI should not only produce the good cluster(s) but also the good explanatory subspaces wrt. these clusters, the two information should match. Consequently, during the precision and recall computation, the predicted cluster is compared to the ground-truth cluster first, before comparing the explanatory subspaces.
- For COIN, ATON and ATON-GT, the generated explanatory subspace is compared with the explanatory subspace of o wrt. the true cluster(s) from which it is deviating, as these methods do not indicate from which cluster o may be deviating.

The precision and recall for each data set and each method are shown in Table 7.

CADI has a high precision and recall on all the data sets, meaning that it is able not only to identify the cluster(s) from which an instance is deviating, but also to provide a faithful explanation wrt. these clusters in terms of discriminating attributes. COIN performs better than ATON on $\mathcal{D}_1$, $\mathcal{D}_2$ and $\mathcal{D}_4$. This may be because of the clustering step of COIN that allows to mitigate the influence of different groups of points on the attributes importance. ATON and ATON-GT perform better than COIN on $\mathcal{D}_3$. In this data set, clusters are located in different subspaces and local anomalies can also be identified in these subspaces. As ATON-GT explores different subspaces during the explanation generation process, it has a slight advantage. CADI is also able to discover clusters (and consequently outliers) in subspaces because of the split generation procedure. As a result, it does not fall far behind ATON-GT in terms of precision

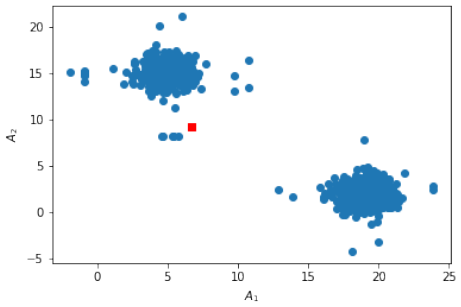


Figure 4: Data set $\mathcal{D}_1$. For the outlier represented by a red square, ATON-GT returns as explanatory subspace the full feature space.

on $\mathcal{D}_3$. In general, ATON-GT has a high recall, because it tends to overestimate the size of the explanatory subspace. For example, on data set $\mathcal{D}_1$, the explanatory subspace returned by ATON-GT for the red square outlier on Figure 4 is the full attribute space. Although feature A_1 is sufficient (regardless of the local context or not), that outlier is more easily isolated in the full feature space than in A_1.

5 CONCLUSION AND FUTURE WORK

This paper presented a unified model for contextual and interpretable anomaly detection. The proposed method, called CADI proposes a revised version of the IF as a basis to identify outliers, groups of regular data points and to provide explanations of both global and local anomalies wrt. clusters, without relying on external algorithms to perform the different tasks. Conducted experiments show that CADI is indeed able not only to identify the anomalies as well as classical IF on real-world data sets, but also to provide meaningful and accurate explanations regarding the abnormality of an instance wrt. a group of points in synthetic data sets containing clusters and local anomalies. The main limitation of this work is the absence of evaluation of the explanation component on real-world data and higher dimensional data sets. This will require some work to provide adequate ground-truths and is the main direction for further research.

ACKNOWLEDGMENTS

This research is part of the SEA DEFENDER project funded by the French DGA (Directorate General of Armaments).

REFERENCES

[1] Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulos, and Prabhakar Raghavan. 1998. Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications. In *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data*. Association for Computing Machinery, New York, NY, USA, 94–105. https://doi.org/10.1145/276304.276314

[2] Mennatallah Amer, Markus Goldstein, and Slim Abdennadher. 2013. Enhancing One-Class Support Vector Machines for Unsupervised Anomaly Detection. In *Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description*. Association for Computing Machinery, New York, NY, USA, 8–15. https://doi.org/10.1145/2500853.2500857

[3] Christian Borgelt, Christian Braune, Marie-Jeanne Lesot, and Rudolf Kruse. 2015. Handling Noise and Outliers in Fuzzy Clustering. In *Fifty Years of Fuzzy Logic*

Table 7: Outlier interpretation performance

Data set	Precision				Recall			
	CADI	COIN	ATON	ATON-GT	CADI	COIN	ATON	ATON-GT
$\mathcal{D}_1$	**1.0**	0.76	0.76	0.76	**1.0**	0.76	0.76	**1.0**
$\mathcal{D}_2$	**1.0**	0.81	0.60	0.81	**1.0**	0.81	0.69	**1.0**
$\mathcal{D}_3$	0.89	0.69	0.87	**1.0**	**0.89**	0.69	0.87	0.75
$\mathcal{D}_4$	**1.0**	**1.0**	0.94	0.67	**1.0**	**1.0**	0.94	0.91

and its Applications. Springer International Publishing, Cham, 315–335. https://doi.org/10.1007/978-3-319-19683-1_17

[4] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. 2000. LOF: Identifying Density-Based Local Outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*. Association for Computing Machinery, 93–104. https://doi.org/10.1145/342009.335388

[5] Mattia Carletti, Matteo Terzi, and Gian Antonio Susto. 2023. Interpretable Anomaly Detection with DIFFI: Depth-based feature importance of Isolation Forest. *Engineering Applications of Artificial Intelligence* 119 (2023). https://doi.org/10.1016/j.engappai.2022.105730

[6] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly Detection: A Survey. *ACM Comput. Surv.* 41, 3, Article 15 (jul 2009), 58 pages. https://doi.org/10.1145/1541880.1541882

[7] Chun-Hao Chang, Jinsung Yoon, Sercan Ö Arik, Madeleine Udell, and Tomas Pfister. 2023. Data-efficient and interpretable tabular anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, 190–201. https://doi.org/10.1145/3580305.3599294

[8] Sanjay Chawla and Aristides Gionis. 2013. k-means--: A unified approach to clustering and outlier detection. In *Proceedings of the 2013 SIAM International Conference on Data Mining (SDM)*. 189–197. https://doi.org/10.1137/1.9781611972832.21

[9] David Cortes. 2021. Revisiting randomized choices in isolation forests. *arXiv preprint arXiv:2110.13402* (2021).

[10] R.N. Dave and R. Krishnapuram. 1997. Robust clustering methods: a unified view. *IEEE Transactions on Fuzzy Systems* 5, 2 (1997), 270–293. https://doi.org/10.1109/91.580801

[11] Rajesh N Dave. 1991. Characterization and detection of noise in clustering. *Pattern Recognition Letters* 12, 11 (1991), 657–664. https://doi.org/10.1016/0167-8655(91)90002-4

[12] Francesco Dotto, Alessio Farcomeni, Luis Angel García-Escudero, and Agustín Mayo-Iscar. 2018. A reweighting approach to robust clustering. *Statistics and Computing* 28, 2 (2018), 477–493. https://doi.org/10.1007/s11222-017-9742-x

[13] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, Vol. 96. 226–231.

[14] Patrick J.F. Groenen and Krzysztof Jajuga. 2001. Fuzzy clustering with squared Minkowski distances. *Fuzzy Sets and Systems* 120, 2 (2001), 227–237. https://doi.org/10.1016/S0165-0114(98)00403-5

[15] Nikhil Gupta, Dhivya Eswaran, Neil Shah, Leman Akoglu, and Christos Faloutsos. 2019. Beyond Outlier Detection: LookOut for Pictorial Explanation. In *Machine Learning and Knowledge Discovery in Databases*. Springer International Publishing, 122–138. https://doi.org/10.1007/978-3-030-10925-7_8

[16] Songqiao Han, Xiyang Hu, Hailiang Huang, Minqi Jiang, and Yue Zhao. 2022. ADBench: Anomaly Detection Benchmark. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 32142–32159.

[17] Sahand Hariri, Matias Carrasco Kind, and Robert J. Brunner. 2021. Extended Isolation Forest. *IEEE Transactions on Knowledge and Data Engineering* 33, 4 (2021), 1479–1489. https://doi.org/10.1109/TKDE.2019.2947676

[18] Frank Klawonn and Frank Höppner. 2003. What Is Fuzzy about Fuzzy Clustering? Understanding and Improving the Concept of the Fuzzifier. In *Advances in Intelligent Data Analysis V*. Springer Berlin Heidelberg, Berlin, Heidelberg, 254–264.

[19] R. Krishnapuram and J.M. Keller. 1996. The possibilistic C-means algorithm: insights and recommendations. *IEEE Transactions on Fuzzy Systems* 4, 3 (1996), 385–393. https://doi.org/10.1109/91.531779

[20] M.-J. Lesot and B. Bouchon-Meunier. 2004. Descriptive concept extraction with exceptions by hybrid clustering. In *2004 IEEE International Conference on Fuzzy Systems*, Vol. 1. 389–394. https://doi.org/10.1109/FUZZY.2004.1375756

[21] Marie-Jeanne Lesot and Adrien Revault d'Allonnes. 2012. Credit-Card Fraud Profiling Using a Hybrid Incremental Clustering Methodology. In *Scalable Uncertainty Management*. Springer Berlin Heidelberg, 325–336.

[22] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2010. On Detecting Clustered Anomalies Using SCiForest. In *Machine Learning and Knowledge Discovery in Databases*. Springer Berlin Heidelberg, Berlin, Heidelberg, 274–290.

[23] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2012. Isolation-Based Anomaly Detection. *ACM Trans. Knowl. Discov. Data* 6 (mar 2012). https://doi.org/10.1145/2133360.2133363

[24] Hongfu Liu, Jun Li, Yue Wu, and Yun Fu. 2021. Clustering With Outlier Removal. *IEEE Transactions on Knowledge and Data Engineering* 33, 6 (2021), 2369–2379. https://doi.org/10.1109/TKDE.2019.2954317

[25] Ninghao Liu, Donghwa Shin, and Xia Hu. 2018. Contextual outlier interpretation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. AAAI Press, 2461–2467. https://doi.org/10.5555/3304889.3305002

[26] Meghanath Macha and Leman Akoglu. 2018. Explaining anomalies in groups with characterizing subspace rules. *Data Mining and Knowledge Discovery* 32, 5 (2018), 1444–1480. https://doi.org/10.1007/s10618-018-0585-7

[27] Geoffrey J. McLachlan and David Peel. 1998. Robust cluster analysis via mixtures of multivariate t-distributions. In *Advances in Pattern Recognition*. Springer Berlin Heidelberg, Berlin, Heidelberg, 658–666.

[28] Antonella Mensi and Manuele Bicego. 2021. Enhanced anomaly scores for isolation forests. *Pattern Recognition* 120 (2021), 108–115. https://doi.org/10.1016/j.patcog.2021.108115

[29] Tshepiso Mokoena, Turgay Celik, and Vukosi Marivate. 2022. Why is this an anomaly? Explaining anomalies using sequential explanations. *Pattern Recognition* 121 (2022). https://doi.org/10.1016/j.patcog.2021.108227

[30] N.R. Pal, K. Pal, J.M. Keller, and J.C. Bezdek. 2005. A possibilistic fuzzy c-means clustering algorithm. *IEEE Transactions on Fuzzy Systems* 13, 4 (2005), 517–530. https://doi.org/10.1109/TFUZZ.2004.840099

[31] Egawati Panjei, Le Gruenwald, Eleazar Leal, Christopher Nguyen, and Shejuti Silvia. 2022. A survey on outlier explanations. *The VLDB Journal* 31, 5 (2022), 977–1008. https://doi.org/10.1007/s00778-021-00721-1

[32] Shebuti Rayana. 2016. ODDS Library. http://odds.cs.stonybrook.edu

[33] Lukas Ruff, Jacob R. Kauffmann, Robert A. Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G. Dietterich, and Klaus-Robert Müller. 2021. A unifying review of deep and shallow anomaly detection. *Proc. IEEE* 109, 5 (2021), 756–795. https://doi.org/10.1109/JPROC.2021.3052449

[34] Amit K Shukla, Grégory Smits, Olivier Pivert, and Marie-Jeanne Lesot. 2020. Explaining Data Regularities and Anomalies. In *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 1–8. https://doi.org/10.1109/FUZZ48607.2020.9177689

[35] Haolong Xiang, Hongsheng Hu, and Xuyun Zhang. 2022. DeepiForest: A Deep Anomaly Detection Framework with Hashing Based Isolation Forest. In *2022 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1251–1256. https://doi.org/10.1109/ICDM54844.2022.00163

[36] Hongzuo Xu, Yijie Wang, Songlei Jian, Zhenyu Huang, Yongjun Wang, Ning Liu, and Fei Li. 2021. Beyond Outlier Detection: Outlier Interpretation by Attention-Guided Triplet Deviation Network. In *Proceedings of the Web Conference 2021*. Association for Computing Machinery, New York, NY, USA, 1328–1339. https://doi.org/10.1145/3442381.3449868

[37] Véronne Yepmo, Grégory Smits, and Olivier Pivert. 2022. Anomaly explanation: A review. *Data & Knowledge Engineering* 137 (2022). https://doi.org/10.1016/j.datak.2021.101946

[38] Chong Zhou and Randy C. Paffenroth. 2017. Anomaly Detection with Robust Deep Autoencoders. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, New York, NY, USA, 665–674. https://doi.org/10.1145/3097983.3098052